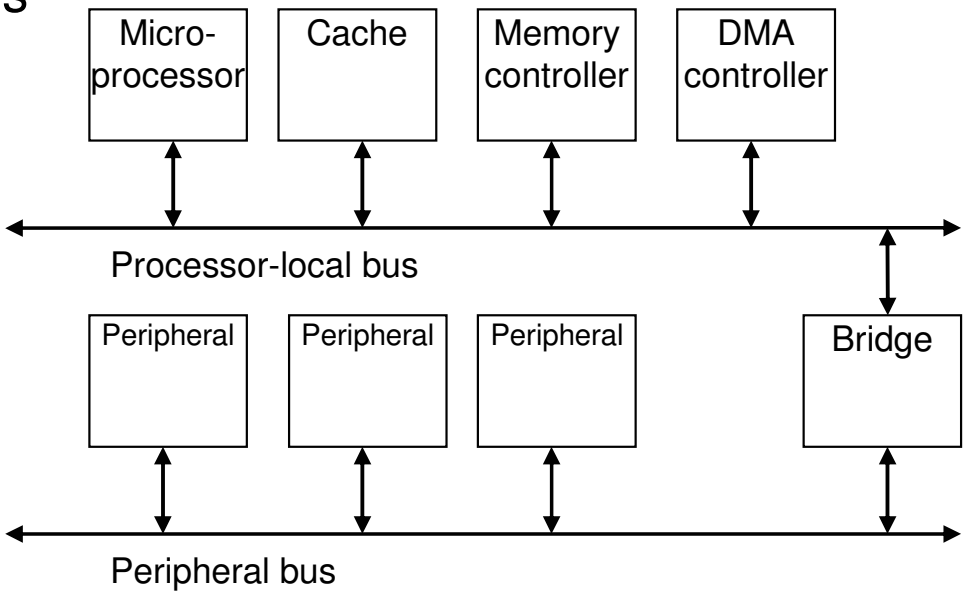
Multilevel bus architectures

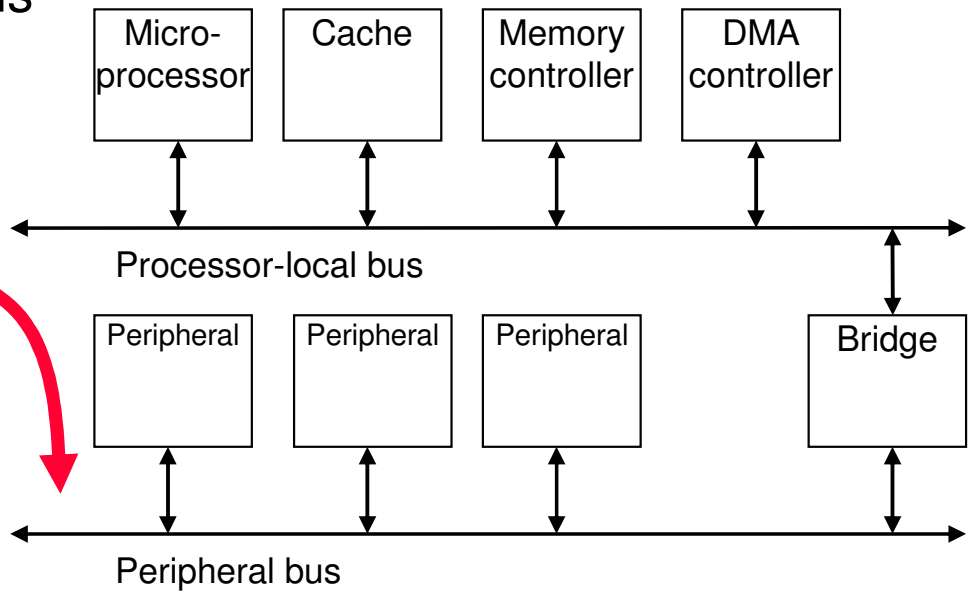
- Don't want one bus for all communication
  - Peripherals would need high-speed, processor-specific bus interface
    - excess gates, power consumption, and cost; less portable
  - Too many peripherals slows down bus

- Processor-local bus
  - High speed, wide, most frequent communication
  - Connects microprocessor, cache, memory controllers, etc.



Multilevel bus architectures

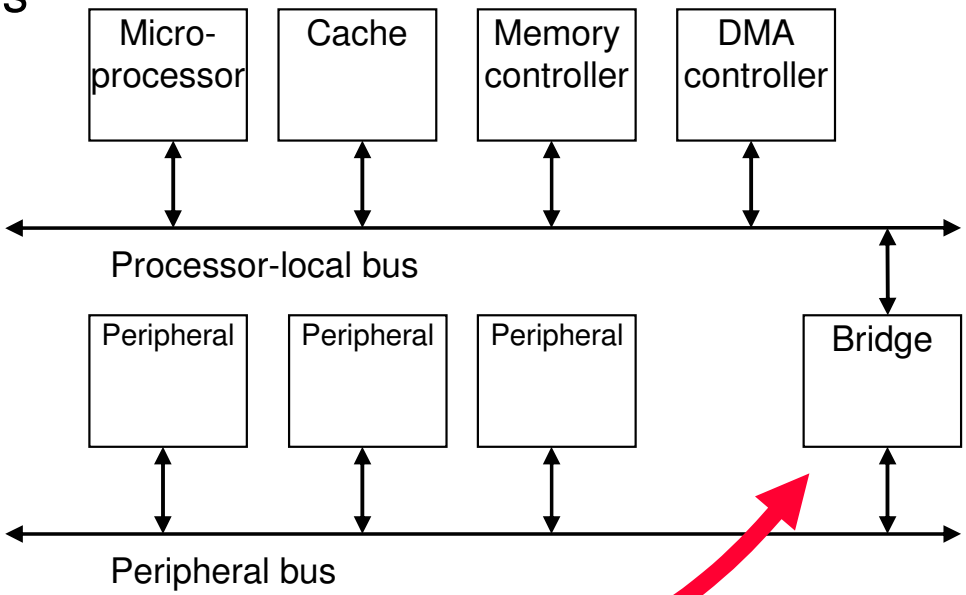
- Don't want one bus for all communication
  - Peripherals would need high-speed, processor-specific bus interface
    - excess gates, power consumption, and cost; less portable
  - Too many peripherals slows down bus



- Peripheral bus
  - Lower speed, narrower, less frequent communication
  - Typically industry standard bus (ISA, PCI) for portability

Multilevel bus architectures

- Don't want one bus for all communication
  - Peripherals would need high-speed, processor-specific bus interface
    - excess gates, power consumption, and cost; less portable
  - Too many peripherals slows down bus



- Bridge
  - Single-purpose processor converts communication between busses

Advanced communication principles

- Layering
  - Break complexity of communication protocol into pieces easier to design and understand
  - Lower levels provide services to higher level
    - Lower level might work with bits while higher level might work with packets of data
  - Physical layer
    - Lowest level in hierarchy

Advanced communication principles

- Open Systems Interconnection--Reference Model (OSI--RM)
  - http://www.its.bldrdoc.gov/fs-1037/dir-025/_3680.htm
  - http://homepages.uel.ac.uk/u0306091/OSI.htm
- It defines seven layers:
  - Physical *example: How many volts represent 1, how many for 0*
  - Data Link *example: parity and/or CRC checking*
  - Network *example: How is data routed to recipient*
  - Transport *example: split up information into data segments*
  - Session *example: log on, password*
  - Presentation *example: how are characters represented*
  - Application (upper most layer) *example: send an e-mail*

Advanced communication principles

- Parallel communication
  - Physical layer capable of transporting multiple bits of data
- Serial communication
  - Physical layer transports one bit of data at a time
- Wireless communication
  - No physical connection needed for transport at physical layer

Advanced communication principles

- Parallel communication
  - Physical layer capable of transporting multiple bits of data
- Serial communication
  - Physical layer transports one bit of data at a time
- Wireless communication
  - No physical connection needed for transport at physical layer

Advanced communication principles

- Quick Quiz:
  - A Megabyte is:
    - 1048576 bytes
    - 1024000 bytes
    - 1000000 bytes
  - MBs is commonly used for
    - Megabits per second
    - Megabytes per second
    - Megabits X seconds

Advanced communication principles

- Due to the use and misuse of megabytes, kilobytes, gigabytes etc, etc the following has been proposed by NIST:

one **kibibit** 1 Kibit = 2^{10} bit = **1024 bit**

one **kilobit** 1 kbit = 10^{3} bit = **1000 bit**

one **mebibyte** 1 MiB = 2^{20} B = **1 048 576 B**

one **megabyte** 1 MB = 10^{6} B = **1 000 000 B**

one **gibibyte** 1 GiB = 2^{30} B = **1 073 741 824 B**

one **gigabyte** 1 GB = 10^{9} B = **1 000 000 000 B**

and so on… http://physics.nist.gov/cuu/Units/binary.html

Advanced communication principles

- Keep in mind:
  - Mbs = Mega bits per Second 1,000,000 bits every second
  - MBs = Mega bytes per second 8,000,000 bits every second
  - Mbaud = Often misused, typically signal rate, which may be more than actual *data* rate.
    - If you are toggling a serial line 100 times a second, but are using 9 bit parity, one out of every 9 bits is not data, therefore, your data rate is 100 * 8/9 = 88.9 bits/second and your baud = 100 Hz.

Parallel communication

- Multiple data, control, and possibly power wires
  - One bit per wire
- High data throughput with short distances
- Typically used when connecting devices on same IC or same circuit board
  - Bus must be kept short
    - With a lot of wires switching at the same frequency, they may create noise that will effect nearby wires
    - Data misalignment between wires increases as length increases
- Higher cost, bulky

Serial communication

- Words transmitted one bit at a time
- Higher data throughput with long distances
- Cheaper, less bulky
- More complex interfacing logic and communication protocol
  - Sender needs to decompose word into bits
  - Receiver needs to recompose bits into word
  - Control signals often sent on same wire as data increasing protocol complexity

Serial communication

- Frequently use more complex electrical connections than just a wire:
  - Fiber-Optic
    - Uses light to communicate
  - Low Voltage Differential Signal (LVDS)
    - Consists of two signals, one inverted from the other

Wireless communication

- Infrared (IR)
  - Electronic wave frequencies just below visible light spectrum
  - Diode emits infrared light to generate signal
  - Infrared transistor detects signal, conducts when exposed to infrared light
  - Cheap to build
  - Need line of sight, limited range
- Radio frequency (RF)
  - Electromagnetic wave frequencies in radio spectrum
  - Analog circuitry and antenna needed on both sides of transmission
  - Line of sight not needed, transmitter power determines range

Error detection and correction

- Often part of bus protocol
- Error detection: ability of receiver to detect errors during transmission
- Error correction: ability of receiver and transmitter to cooperate to correct problem
  - Typically done by acknowledgement/retransmission protocol
- Bit error: single bit is inverted
- Burst of bit error: consecutive bits received incorrectly

Serial protocols: I^2C

- I^2C (Inter-IC)
  - Two-wire serial bus protocol developed by Philips Semiconductors nearly 20 years ago
  - Enables peripheral ICs to communicate using simple communication hardware
  - Data transfer rates up to 100 kbits/s and 7-bit addressing possible in normal mode
  - 3.4 Mbits/s and 10-bit addressing in fast-mode
  - Common devices capable of interfacing to I^2C bus:
    - EPROMS, Flash, and some RAM memory, real-time clocks, watchdog timers, and microcontrollers

Serial protocols: I²C

- Every component hooked up to the bus has its own unique address whether it is a CPU, LCD driver, memory, or complex function chip. Each of these chips can act as a receiver and/or transmitter depending on it's functionality. Obviously an LCD driver is only a receiver, while a memory or I/O chip can both be transmitter and receiver. Furthermore there may be one or more BUS MASTER's.
- The BUS MASTER is the chip issuing the commands on the BUS. In the I2C protocol specification it is stated that the IC that initiates a data transfer on the bus is considered the BUS MASTER. At that time all the others are regarded to as the BUS SLAVEs.
- The IC bus is a Multi-MASTER BUS. This means that more than one IC capable of initiating data transfer can be connected to it.

(from I2C FAQ V1.3)

Serial protocols: CAN

- CAN (Controller area network)
  - Protocol for real-time applications
  - Developed by Robert Bosch GmbH
  - Originally for communication among components of cars
  - Applications now using CAN include:
    - elevator controllers, copiers, telescopes, production-line control systems, and medical instruments
  - The CAN bus is used where high transmission reliability is needed, like motor control

- LIN (Local Interconnect Bus)
  - Slower and cheaper than CAN.
  - Frequently supplements CAN

Serial protocols: FireWire

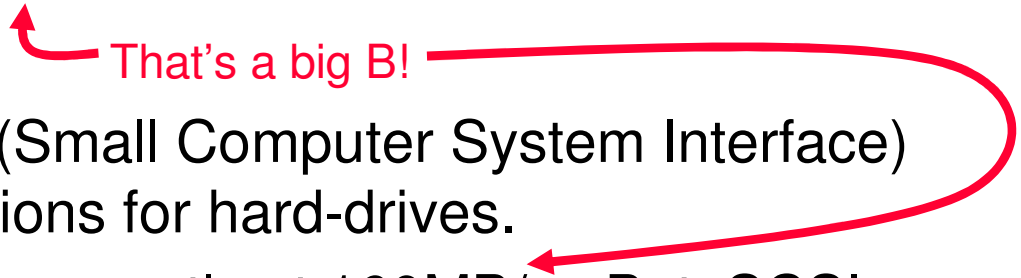
- FireWire (a.k.a. I-Link, Lynx, IEEE 1394)
  - High-performance serial bus developed by Apple Computer Inc.
  - Designed for interfacing independent electronic components
    - e.g., Desktop, scanner
  - Data transfer rates of 400 Mbits/s (now up to 800Mbs, soon 3.2Gbs!)
  - Plug-and-play capabilities
  - Applications using FireWire include:
    - disk drives, printers, scanners, video cameras
  - Capable of supporting a LAN similar to Ethernet

Serial protocols: USB

- USB (Universal Serial Bus)
  - Easier connection between PC and monitors, printers, digital speakers, modems, scanners, digital cameras, joysticks, multimedia game equipment
  - 2 data rates:
    - 12 Mbps for increased bandwidth devices
    - 1.5 Mbps for lower-speed devices (joysticks, game pads)
    - USB2.0 now goes up to 480Mbs!
  - Tiered star topology can be used
    - One USB device (hub) connected to PC
    - Multiple USB devices can be connected to hub
    - Up to 127 devices can be connected like this
    - Does not support peer-to-peer, but USB-On-The-Go promises to work around that.

Serial protocols: SATA

- SATA (Serial ATA) Where ATA stands for AT Attachment where AT stands for who knows what? It came from the IBM AT computer.
- This is a replacement for the standard hard-drive connection Ultra-ATA which is a parallel connection (16 data bits).
- Streams data at a whopping 150MB/s!



- SATA is hoping to displace SCSI (Small Computer System Interface) as the king of high-speed connections for hard-drives.
- Ultra3 SCSI is 16-bit parallel; and presently at 160MB/s. But, SCSI cables are *at least* five times as expensive as SATA cables

Parallel protocols: PCI Bus

- PCI Bus (Peripheral Component Interconnect)
  - High performance bus originated at Intel in the early 1990's
  - Standard adopted by industry and administered by PCISIG (PCI Special Interest Group)
  - Interconnects chips, expansion boards, processor memory subsystems
  - Synchronous bus architecture
  - Multiplexed data/address lines
  - Soon to be supplanted by PCI-X and some day, maybe replaced by PCI-Express (serial, up to 2.5Gbs faster than AGP 8X)

Parallel protocols: ARM Bus

- ARM Bus
  - Designed and used internally by ARM Corporation
  - Interfaces with ARM line of processors
  - Many IC design companies have incorporated this protocol
  - Data transfer rate is a function of clock speed
    - If clock speed of bus is X, transfer rate = 16 x X bits/s
  - 32-bit addressing

Parallel protocols: other…

- Wishbone
  - An open protocol from opencores.org
  - Can be implemented in 8 to 64 bit widths
- SCSI (Small Computer System Interface)
  - Has been around for quite a while and has grown from SCSI-1 (5MB/s) narrow (8 bit wide) to Ultra3 (160MB/s) and Ultra4 is under development (320MB/s)

Wireless protocols: IrDA

- IrDA
  - Protocol suite that supports short-range point-to-point infrared data transmission
  - Created and promoted by the Infrared Data Association (IrDA)
  - Data transfer rate of 9.6 kbps and 4 Mbps
  - IrDA hardware deployed in notebook computers, PDAs, digital cameras, public phones, cell phones
  - Lack of suitable drivers has slowed use by applications

Wireless protocols: Bluetooth

- Bluetooth
  - New, global standard for wireless connectivity
  - Based on low-cost, short-range radio link
  - Connection established when within 10 meters of each other
  - No line-of-sight required
    - e.g., Connect to printer in another room
  - Quickly becoming popular in cell-phones and PDA's

Wireless Protocols: IEEE 802.11

- IEEE 802.11
  - Proposed standard for wireless LANs
  - Specifies parameters for PHY and MAC layers of network
    - PHY layer
      - physical layer
      - handles transmission of data between nodes
      - provisions for data transfer rates up to 54Mbs
    - MAC layer
      - medium access control layer
      - protocol responsible for maintaining order in shared medium
      - collision avoidance/detection

Wireless Protocols: New Ones…

- ZigBee (lower speed—Home automation, slower, low power, cost effective)
- 802.16 (WiMax—high speed wireless communication)

Chapter Summary

- Basic protocol concepts
  - Actors, direction, time multiplexing, control methods
- General-purpose processors
  - Port-based or bus-based I/O
  - I/O addressing: Memory mapped I/O or Standard I/O
  - Interrupt handling: fixed or vectored
  - Direct memory access
- Arbitration
  - Priority arbiter (fixed/rotating) or daisy chain
- Bus hierarchy
- Advanced communication
  - Parallel vs. serial, wires vs. wireless, error detection/correction, layering
  - Serial protocols: I^2C, CAN, FireWire, and USB; Parallel: PCI and ARM.
  - Serial wireless protocols: IrDA, Bluetooth, and IEEE 802.11.