Computer Hardware

2017 Session 1

Lecture: Buses and Decoding
"All aboard, Ding! Ding!"

Buses and Addresses

- A bus is a circuit in a system that transfers data from one component to another component.
  - An *internal bus*
- A bus may transfer data between systems.
  - An *external bus*
- A bus may contain multiple parallel conductors that pass data on all conductors at once.
  - A *parallel bus*
- A bus may contain only one data conductor, and all data is passed one baud at a time.
  - A *serial bus*

Bus Examples

Parallel, Internal:

- Microprocessor Address or Data bus
- PCI
- ISA
- ATA

Parallel, External:

- SCSI
- Centronics
  - Parallel Printer Port

Bus Examples

Serial, Internal:

- HyperTransport/QuickPath Interconnect
  - Microprocessor Address and Data bus
- PCIe
- SATA
- SAS
- I2C
- SPI
- SMbus

Bus Examples

Serial, External:

- Infiniband
- FibreChannel
- SATA/SAS
- Ethernet
- IEEE 1394
- Thunderbolt
- USB

Address Spaces

- CPUs have to be able to specify what they are accessing.
- We distinguish access according to address spaces.
  - Memory Address Space
  - I/O Address Space
  - Configuration Address Space
    - PCIe
- Different access methods for each address space
- We need to be able to decode an address request to decide which device is being addressed

Address Decoding

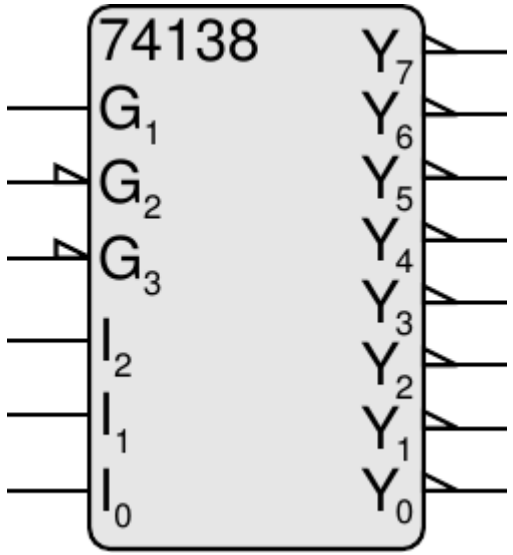
- Map part of an address space to a single device
  - May map more address space than required
  - May map from different address spaces to the one device
    - Memory address for bulk data transfers
    - I/O address for configuration information
- PIC chips use memory address space mapping to control peripheral units.
  - All mappings are in the register (data) address space
  - No mappings are in the program (code) address space
  - Each mapping is 1 byte at a time

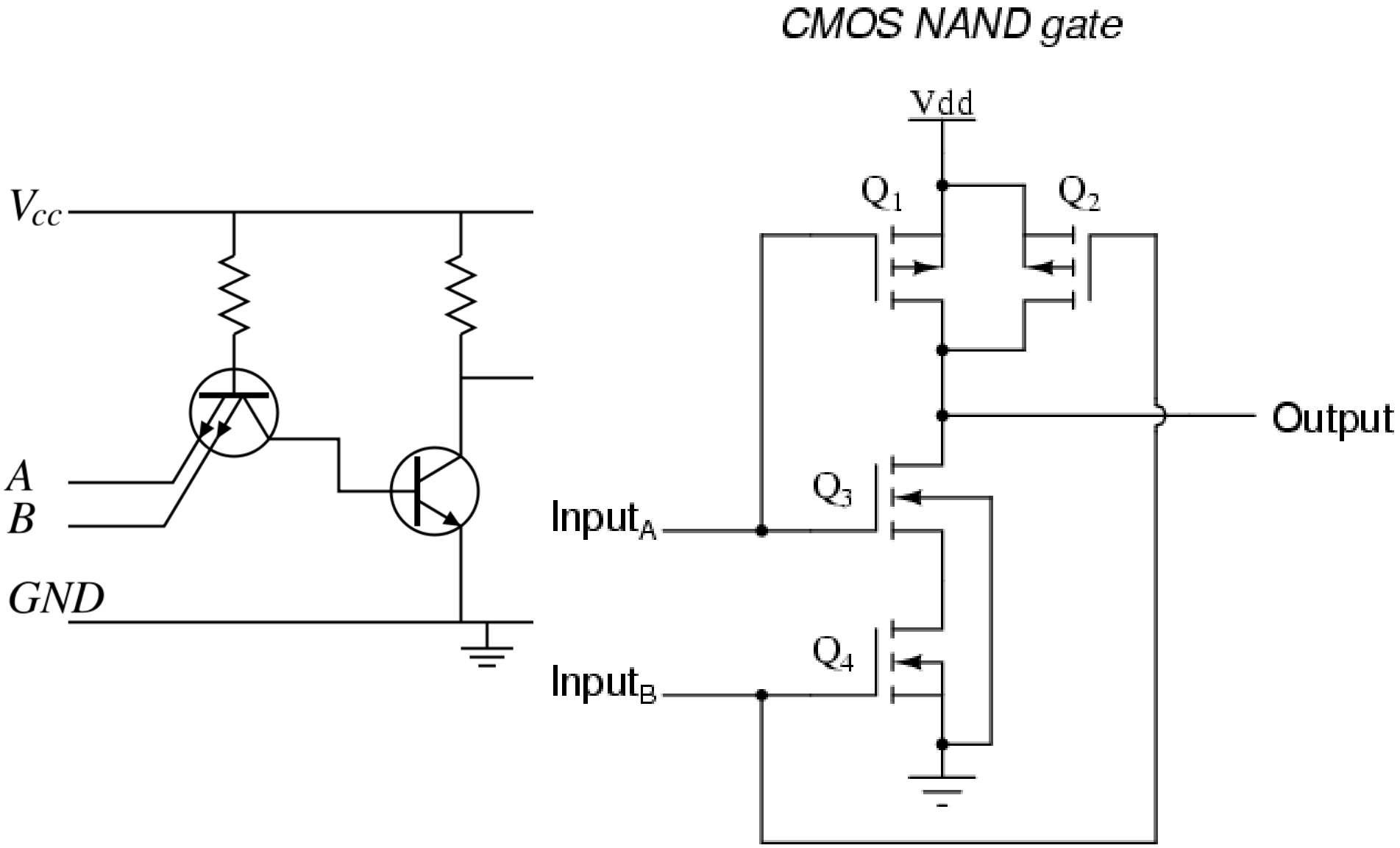
Address Decoding

- Can map using an external addressing mechanism
  - External address bus
    - Parallel (Microprocessor address spaces)
  - Decoders for Parallel addressing look at address specified on the bus and present *chip selects* or *chip enables* to the external peripheral units that are being accessed.
  - Examples are 74x138, 74x139, 74x154 chips which decode some number of parallel address lines into 1 of *n* chip select pins.
  - Pins are active LOW for historical TTL reasons

Address Decoding

- Can map using an address line as the chip select itself
  - Only works when using top address lines.
  - Some devices presented multiple positive and negative chip selects (74x138)
  - Three I inputs select 1 of 8 active low outputs
  - The three G inputs allow both positive and negative chip selects, as well as selecting 00, 01 and 10 from addresses
  - Can easily be cascaded




CMOS NAND gate
V_{cc}
A
B
GND
Vdd
Q_1
Q_2
Q_3
Q_4
$Input_A$
$Input_B$
Output

PCI and PCIe Buses

- High speed internal buses.
- PCIe has replaced PCI for most purposes
  - PCI still retained for backward compatibility
- PCI is parallel internal bus
- PCIe is serial internal bus
- PCI and PCIe buses have two distinct address spaces:
  - configuration address space
  - memory address space

PCI Speeds

- PCI is 32 or 64 bits in parallel
- PCI is clocked at 33 or 66 MHz
  - 128MB/s to 528MB/s
- PCI-X changed speed to 133MHz, (>1GB/s)

- PCIe is a one or more serial data link
  - Each PCIe data link (lane) transfers 2.5Gbits/sec, but
    - uses 8b/10b encoding to merge data and clock
    - Each lane can transfer 250MB/s
    - 32 bonded lanes can transfer 8GB/s (Theoretical)

PCI Configuration Address Space

- Controlled by Bus Number, Device ID and Function ID
  - 8bits/5bits/3bits => 16 bit number
  - Each PCI bus has a distinct bus number.
    - Bus 0 is the one connected to the CPU
    - Subsequent bus numbers are assigned by the PCI to PCI bridges
    - Each PCI bus can hold 32 devices
    - Each device can present 8 functions
    - Must always have function 0
    - Must always have device 0

PCI Configuration Address Space

- Devices are originally accessed using their bus/device/function address.
  - We scan all 0/device/0 entries first.
  - Any bridges/switches found allow us to enumerate additional buses.
  - We can then scan those buses.
    - Scan is usually performed depth first using a recursive function.
  - After scan we know where each device is in configuration address space.

PCI Configuration Address Space

- Each PCI/PCIe device is self identifying
  - Has a 16 bit vendor ID and 16 bit device ID

00:00.0 Host bridge [0600]: Intel Corporation 3rd Gen Core processor DRAM Controller [8086:0154] (rev 09)
00:02.0 VGA compatible controller [0300]: Intel Corporation 3rd Gen Core processor Graphics Controller [8086:0166] (rev 09)
00:14.0 USB controller [0c03]: Intel Corporation 7 Series/C210 Series Chipset Family USB xHCI Host Controller [8086:1e31] (rev 04)
00:16.0 Communication controller [0780]: Intel Corporation 7 Series/C210 Series Chipset Family MEI Controller #1 [8086:1e3a] (rev 04)
00:1a.0 USB controller [0c03]: Intel Corporation 7 Series/C210 Series Chipset Family USB Enhanced Host Controller #2 [8086:1e2d] (rev 04)
00:1b.0 Audio device [0403]: Intel Corporation 7 Series/C210 Series Chipset Family High Definition Audio Controller [8086:1e20] (rev 04)
00:1c.0 PCI bridge [0604]: Intel Corporation 7 Series/C210 Series Chipset Family PCI Express Root Port 1 [8086:1e10] (rev c4)
00:1c.1 PCI bridge [0604]: Intel Corporation 7 Series/C210 Series Chipset Family PCI Express Root Port 2 [8086:1e12] (rev c4)
00:1c.2 PCI bridge [0604]: Intel Corporation 7 Series/C210 Series Chipset Family PCI Express Root Port 3 [8086:1e14] (rev c4)
00:1c.3 PCI bridge [0604]: Intel Corporation 7 Series/C210 Series Chipset Family PCI Express Root Port 4 [8086:1e16] (rev c4)
00:1d.0 USB controller [0c03]: Intel Corporation 7 Series/C210 Series Chipset Family USB Enhanced Host Controller #1 [8086:1e26] (rev 04)
00:1f.0 ISA bridge [0601]: Intel Corporation HM76 Express Chipset LPC Controller [8086:1e59] (rev 04)
00:1f.2 SATA controller [0106]: Intel Corporation 7 Series Chipset Family 6-port SATA Controller [AHCI mode] [8086:1e03] (rev 04)
00:1f.3 SMBus [0c05]: Intel Corporation 7 Series/C210 Series Chipset Family SMBus Controller [8086:1e22] (rev 04)
01:00.0 Unassigned class [ff00]: Realtek Semiconductor Co., Ltd. RTS5229 PCI Express Card Reader [10ec:5229] (rev 01)
06:00.0 Network controller [0280]: Intel Corporation Centrino Advanced-N 6205 [Taylor Peak] [8086:0085] (rev 34)
0c:00.0 Ethernet controller [0200]: Realtek Semiconductor Co., Ltd. RTL8111/8168/8411 PCI Express Gigabit Ethernet Controller [10ec:8168] (rev 07)

- Vendor ID is centrally controlled.
- Device ID is issued by the vendor.
- VID/DID pair uniquely identified chipset
- Variants on usage or card with chipset is identified by Subsystem Vendor ID and Subsystem ID.
- Operating system uses these four numbers to identify which driver to load for this device.
- Each device can also return a class ID, a class of deviuces that have a standard interface (OHCI, AHCI, XHCI)
- Each PCI device advises how much memory address space it requires as part of the configuration information present.

PCI Memory Address Space

- After identification the memory address space has to be allocated.
- This is done by assigning the address base register to the device.
- Any address access from the base address for the range is accepted by this device.
- Bridges/Switches can have multiple address ranges, although we try to keep them adjacent for ease of configuration.