

CS10110: Hardware Fundamentals

Michael Clarke
mfc5@aber.ac.uk

Processors

- ▶ Understand *machine code* instructions:
 - ▶ Just 1s and 0s (i.e. numbers)
 - ▶ Interpreted through logic circuits
 - ▶ **Not the same as assembler**
- ▶ Clock speed:
 - ▶ How many instructions per second
 - ▶ Clock speed isn't sufficient to give a full overview of performance

Processors

- ▶ Processor performance relies on:
 - ▶ Clock speed
 - ▶ Registers available (to an extent)
 - ▶ Caches
 - ▶ Pipeline length
 - ▶ Architecture: RISC vs CISC, order of execution, number of cores, hyper-threading, etc
- ▶ Benchmark websites can give us some indication of performance:
 - ▶ <http://www.cpubenchmark.net/>

Processors

Some processor designers and manufacturers



MOTOROLA

FUJITSU



AMD



ORACLE



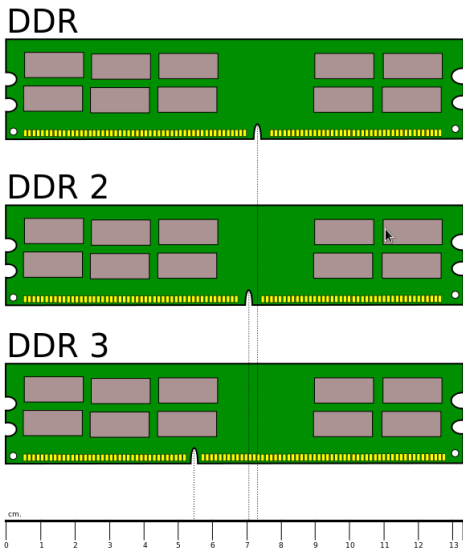
Memory

- ▶ When we say “memory” we are *usually* referring to Random Access Memory (RAM)
- ▶ We already know RAM stores running programs with their associated data, threads, etc
- ▶ Processors can access any address in memory; how much memory can they address?
- ▶ Most PCs have a minimum of 1GB; the price is falling according to **Moore's Law**

DDR (1, 2 and 3)

- ▶ Previously we used SDR SDRAM (Synchronous Dynamic RAM)
- ▶ Now we use DDR (at the moment DDR2 is commonplace, DDR3 more so for graphics cards)
- ▶ DDR stands for Double Data Rate
- ▶ SDR means Single Data Rate

DDR (1, 2 and 3)



DDR (1, 2, and 3)

- ▶ SDR SDRAM: Memory cells are at the BUS clock rate
- ▶ DDR SDRAM: “Double pumping” - use the rising and falling edge of the BUS clock
- ▶ DDR2 SDRAM: “Double pumping” and the cells are at $\frac{1}{4}$ the rate of the BUS clock. Rates are achieved through techniques such as prefetch buffers
- ▶ DDR3 SDRAM: As with DDR2 prefetch buffers are used to increase the rate to $\frac{1}{8}$ of the BUS clock

Calculating Data Rates

$$\frac{b_{max} \cdot d \cdot k_{\alpha}}{k_{\beta}}$$

b_{max} = maximum bus clock rate

d = BUS multiplier (1 for SDR, 2 for DDR, 4 for DDR2, etc)

k_{α} = constant, number of bits transferred (64)

k_{β} = constant, number of bits per byte (8)

Calculating Data Rates

- ▶ Assume a 100MHz bus:
 - ▶ SDR: $\frac{100 \cdot 1 \cdot 64}{8} = 800$ bytes (PC-100)
 - ▶ DDR: $\frac{100 \cdot 2 \cdot 64}{8} = 1600$ bytes (DDR-200 or PC-1600)
 - ▶ DDR2: $\frac{100 \cdot 4 \cdot 64}{8} = 3200$ bytes (DDR2-400 or PC-3200)
 - ▶ DDR3: $\frac{100 \cdot 8 \cdot 64}{8} = 6400$ bytes (DDR3-800 or PC-6400)

Memory Types

- ▶ RAM - Random Access Memory
Really read-write memory. *Volatile* - “forgets” when power turned off
- ▶ ROM - Read Only Memory
Non-volatile - program or data can't be changed. Written in the factory
- ▶ PROM - Programmable ROM
Can be programmed once only, by user

Memory Types

- ▶ EPROM - Erasable PROM
Can be erased and re-written (e.g. using UV light)
- ▶ EEPROM - Electrically Erasable PROM
Can be erased electronically, in circuit, using a high voltage, and re-programmed
- ▶ Flash
An improved EEPROM that is cheaper, faster and has a higher density (i.e. can store more)
- ▶ NVRAM - Non-volatile RAM, includes Flash
<http://en.wikipedia.org/wiki/NVRAM>

Buses

- ▶ A collection of wires which carry data between the CPU, memory and I/O devices.
- ▶ Three main parts:
 - ▶ *Address Bus*
Carries memory address values from the CPU to the memory and I/O
 - ▶ *Data Bus*
Carries all data, including machine codes, in all directions
 - ▶ *Control Bus*
A small number of wires carrying the read/write and enable signals, and a few other “control” signals

Important Bus Parameters

- ▶ *Width of the Address Bus*
Determines the size of the address space; i.e. how many locations can be addressed
- ▶ *Width of the Data Bus*
Determines how much data can be transferred in a single data transfer cycle (thus affecting speed). Usually the same as the machine (and memory) word size
- ▶ *Bus Speed*
The number of transfers possible per second

Bus Width

- ▶ An address or data bus that contains n wires, or lines, is said to be n lines wide.
- ▶ Such a bus can carry a n -bit number in a single operation
- ▶ This does *not* apply to the control bus, as it only needs to carry simple control signals

Address Space

- ▶ A n bit address bus can carry any address that fits into a n bit binary number
- ▶ Therefore, it can carry 2^n possible addresses
- ▶ These will be 0 to $2^n - 1$
- ▶ An address bus with 16 lines can address 2^{16} (=65536) locations; a 32 line address bus can address 2^{32} (=4,294,967,296) locations, etc

Bus Cycles

- ▶ *Bus Cycle*

A single data transfer operation over the bus. The speed of this operation is governed by the bus speed, and affects the speed of the computer

- ▶ *Read Cycle*

A bus cycle in which data is read by the CPU from memory or an I/O device

- ▶ *Write Cycle*

A bus cycle in which data is written by the CPU to memory or an I/O device

Data Throughput

- ▶ The data throughput rate is the number of bytes (or bits) that can be transferred over the bus in one second
- ▶ Therefore, it is the product of the bus speed (cycles per second) and the data bus width (bits per cycle)
- ▶ A 16-bit data bus running at 10 million cycles per second (10 MHz) has a data throughput of 20 million bytes (or 160 million bits) per second
- ▶ Data throughput rate is a major limitation on the speed of a computer

The CPU Controls the BUS...

- ▶ ... Usually. However, there are technologies such as DMAC (Direct Memory Access Controller) that the CPU can “ask” to do large transfers
- ▶ For each bus cycle, the CPU must:
 - ▶ specify the address
 - ▶ select read or write
 - ▶ supply the data, if it's a write

Addressing I/O Devices

- ▶ All data traffic from the CPU has to go down the data bus
- ▶ All addressing had to go down the address bus, but main memory is already using this address space
- ▶ Two possible solutions:
 1. Give I/O devices a separate address space
 2. Allow I/O devices to share the same address space as memory

Separate Address Space

- ▶ Use the control bus to specify whether we're addressing memory or I/O
- ▶ Memory responds only if 'Memory Select' is active; I/O only if I/O select is active
- ▶ This method requires the CPU to have separate machine codes for memory and I/O read and writes - Why?

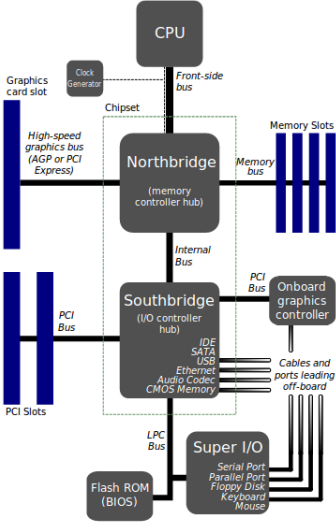
Same Address Space

- ▶ Known as Memory-Mapped I/O
- ▶ I/O devices are given specific addresses in the memory address space
- ▶ Reads and writes to these addresses access I/O devices instead of memory
- ▶ This does not require separate machine codes, the value of the address is sufficient to distinguish between I/O and memory

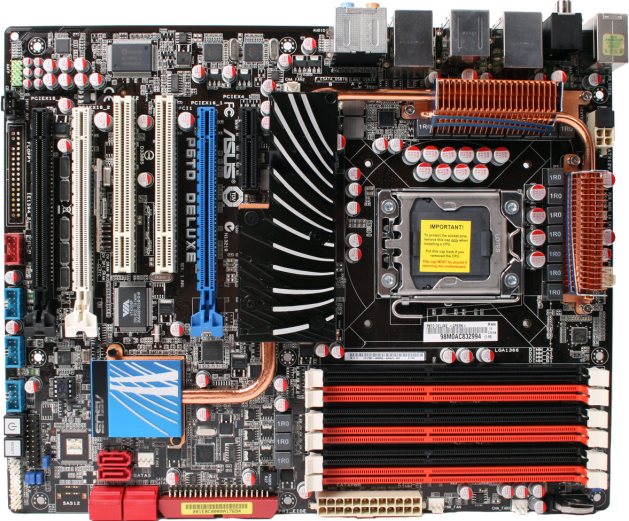
PC Hardware is Inefficient

- ▶ When Intel released the specification, they intentionally made it so that different people could make different components
- ▶ Manufacturers had to keep things backwards compatible
- ▶ As things got more complicated, this became a problem
- ▶ Apple Macs never suffered from this. Now they use Intel CPUs and the x86 architecture they are vulnerable

Chipset



Motherboard Specifications



Hard Discs

- ▶ Long term storage, cheap!
- ▶ Magnetic media
- ▶ Being replaced by Flash memory
- ▶ 80GB is small these days
- ▶ *Much* slower than RAM
- ▶ Removable discs - floppies are gone, Zip had its day, even CDs are being replaced with flash drives
- ▶ (note on spelling: both disc or disk are acceptable; disc is perhaps more UK English...)

Flash Memory

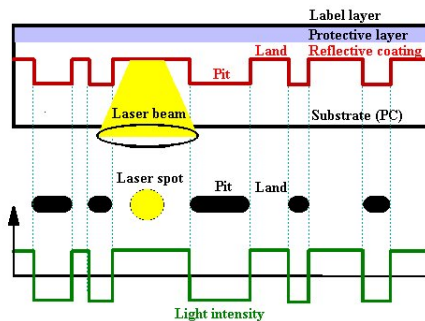
- ▶ Will it replace the hard disc?
- ▶ Quicker and has no moving parts
- ▶ Still considered not so reliable, but it is improving!
- ▶ Still more expensive, but for how long?

Graphics Cards

- ▶ Includes special purpose processors (chipsets)
- ▶ Very fast!
- ▶ Understand special languages (eg. Microsoft DIRECTX, SGI, Open GL, etc)
- ▶ A number of video chipsets are available: ATI, NVidia, Intel, etc
- ▶ Includes a Random Access Memory Digital-to-Analog Converter (RAMDAC) - to convert digital signals to analogue display
- ▶ Perform complex calculations for 3D
- ▶ Hardware support for texture maps, fogging, etc

CD-ROM

- ▶ 120 nanometers wide, 600 nanometers deep
- ▶ 160 nanometers between tracks



CD-ROM

- ▶ Physical constraints mean 17 transitions are used to store 8 bits of data
- ▶ A frame is 24 17-bit symbols plus some other data (eg. ECC)
- ▶ Blocks are build from frames
 - ▶ 2,352 bytes of user data (2,048 for CDROM)
- ▶ A 1x CD plays back 75 blocks per second (0.15 MB/s)

CD-R

- ▶ CD-Rs have an organic dye substrate layer
- ▶ Laser has 2 power levels
- ▶ Higher power heats up the dye and destroys it and so a non-reversible reflective 'pit' is formed.
- ▶ Note that high speed write needs a fast acting dye only found on quality CD-Rs

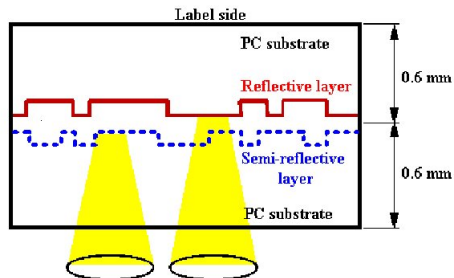
DVD

- ▶ Introduced around 1997
- ▶ What does DVD actually stand for?
- ▶ Whole bunch of formats:
 - ▶ Physical: e.g. DVD-ROM, DVD+R, DVD-R
 - ▶ Application: e.g. DVD-Video and DVD-Audio
 - ▶ <http://www.dvddemystified.com/dvdfaq.html>
- ▶ Denser bits than CD by refining the laser assembly
- ▶ Different frequency laser
- ▶ Better modulation code (less wasted bits) and 10x more powerful error correction code than CD

DVD

- ▶ DVD is a bonded disc, made of two 0.6mm substrates
- ▶ Several configurations of data layers

DVD-9: Single Side, Double Layer Disk (8.54 GB)



Blu-ray disc

- ▶ Higher frequency (so lower wavelength) laser
 - ▶ Blue lasers have been difficult to manufacture and expensive - progress is bringing the price down!
- ▶ Allows more data to be written to the same size disc
- ▶ <http://blu-ray.com/faq>