Class 05

Kali Linux on RADISH

Hard Disk Drives

SSDs

Agenda

Setting up Kali Linux on RADISH

Hard Disks

SSDs

Volumes

Partitions

Questions

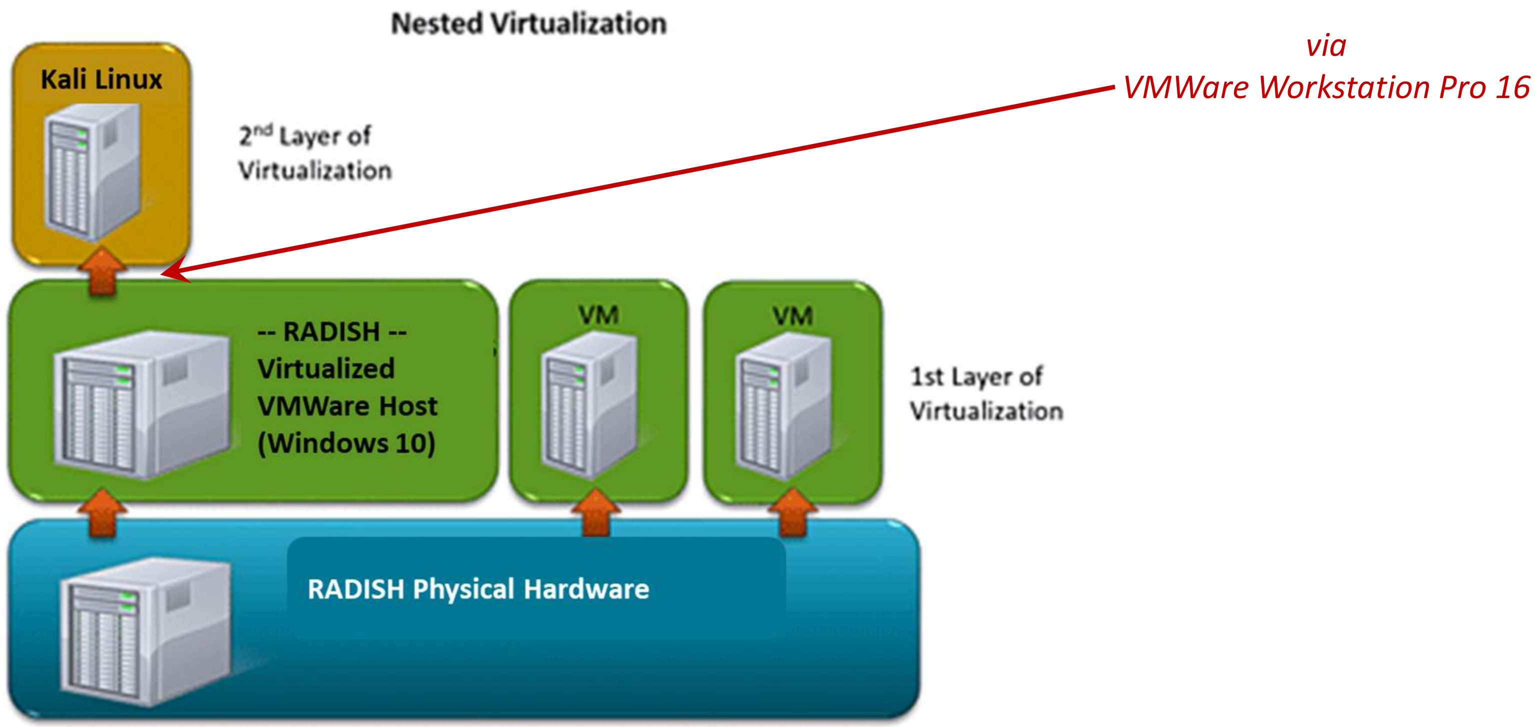
Setting Up Kali Linux

Using VMWare Workstation Pro on RADISH

Setting up Kali Linux

We will be setting up Kali Linux on your RADISH desktop.

You will be running a VM on top of a VM



Setting up Kali Linux

Basic Outline

- ❑ Bring up RADISH
- ❑ Start up VMWare Workstation Pro
- ❑ Verify VMWare Workstation Pro settings
- ❑ Open up Kali Linux appliance file on E: drive
- ❑ Edit Kali VM (e.g., add more RAM)
- ❑ Start up Kali VM
- ❑ Log in
- ❑ Update Kali Linux
- ❑ Tour of VMWare Workstation Pro
- ❑ Tour of Kali Linux
- ❑ Take Kali VM snapshot
- ❑ Shut down Kali VM
- ❑ Exit VMWare Workstation Pro

Hard Disks

Nelson, Chapter 5, pp 195 – 201 top

Carrier, Chapters 2 & 3

Course Plan

Up to now in this course we've used the Nelson text.

Starting today, we will shift emphasis to Carrier's book.

But still use Nelson some

We'll remain in this mode for awhile, and then use Nelson more again.

Chapter 2 of Carrier
Data Organization *section, pp 17-27*

The ***Data Organization*** section contains the following:

- *Binary, Decimal and Hex*
- *Data Sizes*
- *Strings & Character Encoding*
- *Data Structures*
- *Flag Values*

This class assumes that everyone in the class knows and is able to use the information contained in this section

If not:

- *Please study and understand the concepts and the integrated examples*
- ***Be able to use them***

Chapter 2 of Carrier
***Booting Process** section, pp 27-29*

Carrier treats the ***Booting Process*** section lightly and does not include the most recent technology

We will discuss booting in depth in the near future

Chapter 5 of Nelson

Understanding the Boot Sequence section, pp196-197

Nelson also treats this topic lightly.

We will discuss it more.

Microprocessor Architecture Terminology

Hopefully clears up some confusion in terminology.

This is not in any of the texts.

What's <u>IA32</u> ?

By the late 1970s ***Intel*** was a leader in microprocessors

Intel's 16-bit 8086 architecture was being used extensively

By the early 1980s, Intel had become the world's prominent microprocessor manufacturer

The 8086 architecture and instruction set had become a de facto standard

In 1985 Intel introduced the 80386

It was the 1st 32-bit microprocessor of significance

It had an instruction set that extended Intel's 8086 instruction set and architecture from 16 to 32 bits

During the rest of the 20th century, this architecture became known as ***IA32*** or ***IA-32*** *(<u>I</u>ntel <u>A</u>rchitecture 32)*

Sometimes also referred to as ***x86-32*** *or* ***x86***

What's <u>IA32</u> ?

By about the year 2003; Intel, AMD and others were moving to 64-bit architectures

- *These 64-bit architectures retained compatibility for IA32*

So *IA32* and *IA-32* have become de facto terms that

- *Describe all architectures that support x86 32-bit computing*
- *Includes most of the 64-bit computers today*
- *Other terms: x86-64, AMD64*

<u>Not</u> IA-64

- ***IA-64** refers to the instruction set and architecture of the "Itanium" family of 64-bit Intel microprocessors*

Note: It's my understanding that none of these terms are official.

Hard Disk Construction

Overview

Hard Disk Construction

Physical (low level) formatting

Not logical or high-level formatting

Disk Organization, Addressing, Interfaces

PATA ***SATA*** *SCSI*

This lecture applies to storage devices compatible with ***IA32*** architectures.

Operating system independent

Initially we'll focus on rotating disk drives

Later in this class session we'll discuss solid state drive (SSD) mass storage

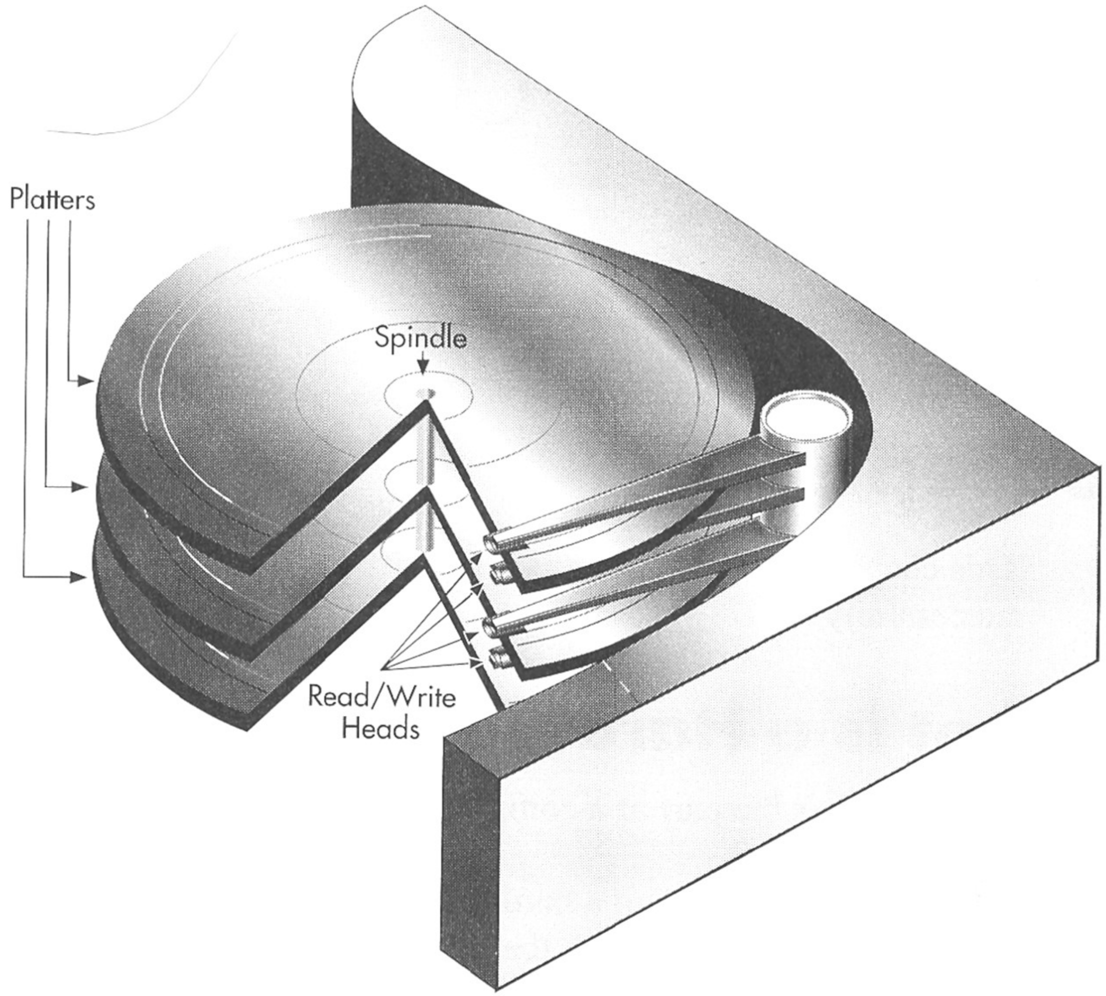
What's a Hard Disk

Part of a computer used for permanent data storage

Basic parts

- ***Platters***: *Rigid disks*
- ***Spindle** on which the several platters are mounted and rotated*
- *A number of movable combination **read/write heads***
  - At least one head for each side of each platter
- ***Electronics** is used to move the heads and to read or write information onto one or more platters*

Hard Disk
Basic Components



Platters

Platters

Usually made of metal or more recently composite layers of material

Both sides are thinly and very uniformly coated

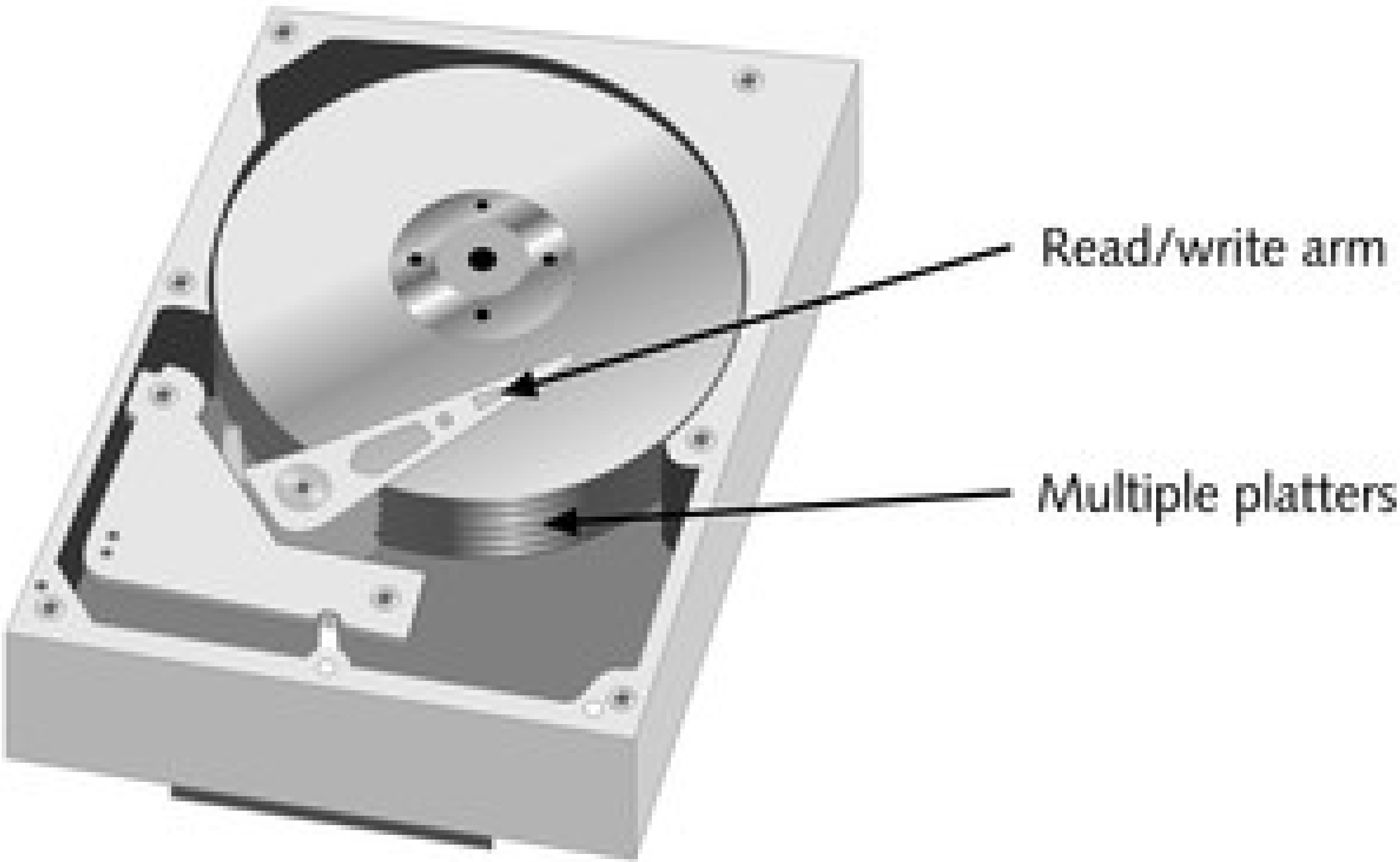
- Primarily with a magnetic iron oxide compound
- Compound often purposely include other trace elements

Magnetic Storage

Consider small local areas of uniform magnetic coating on a platter surface

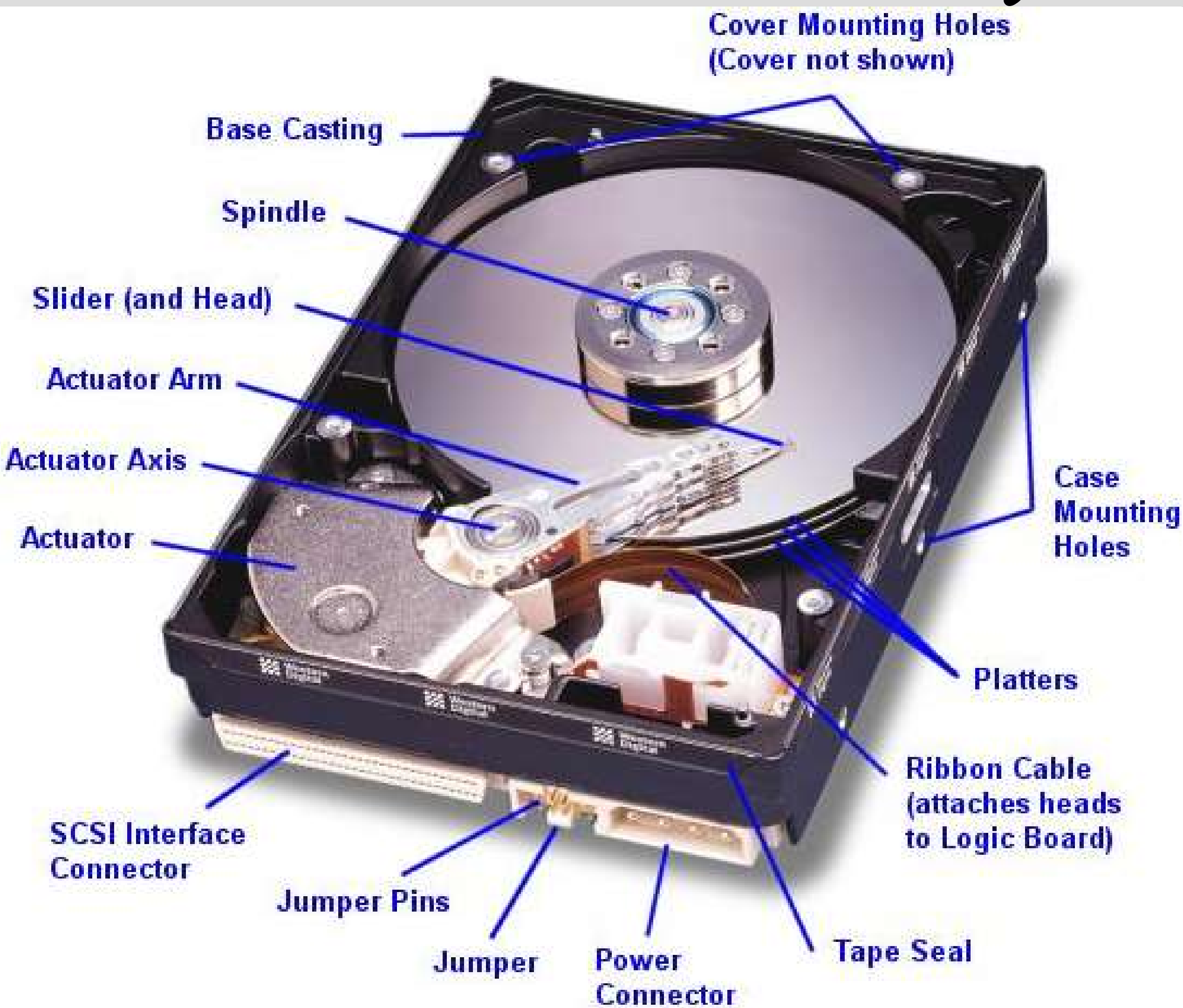
- *Capable of being magnetized in one of two magnetic directions or states that represent a logical 0 or 1*
  - Independent of magnetization of adjacent areas
- *Will hold the magnetic state indefinitely in the absence of external magnetic influences*
- *Will generate a signal when a read head is passed over them*
- *Nature of the signal when the head passes over the area depends upon the magnetic state of the area*
  - Signal indicates a ***0*** or a ***1*** state for that area

Hard Disk
Cutaway View



Hard Disk

Another Cutaway View



Platters are attached to the spindle

Spindle rotates the platters at a constant angular velocity (rpm)

Above and below each platter is at least one **arm** with a combination **read/write head** attached

Each arm extends over its platter and is moved precisely back and forth radially between the center and edge of the platter

Thus a read/write head can be positioned over any location on the platter within one platter revolution

Writing & Reading Data

Data is stored within the disk coating as a series of bits on the surface of the platter

In writing, bits of data are sent to the disk

The head magnetizes specific areas on one or more platters as determined by the disk drivers and related software & firmware

These magnetized areas represent the 0s and 1s.

In reading, the heads "fly" (at ~ 2 nm altitude) over the platters to the locations of the previously recorded bits

The head determines the magnetic state of each area thus determining whether it is a 0 or 1.

Writing & Reading Data

Hard disks are capable of what some call "random access"

It's not!

Really, it's two-dimension access on a platter surface

The arm can move the head over the disk approximately radially

Simultaneously the disk is rotating circumferentially

This animated graphic illustrates it well:

https://animagraffs.com/hard-disk-drive/

Disk Organization & Addressing

Disk Organization Introduction

Disks can store up to trillions (10^{12}) of bytes (terabytes) or more

- *Equivalent to close to 10^{13} bits*
- *Largest single disk drive advertised today is 22 TB*
- *(30 TB might be available later this year)*

There must be a way of organizing a disk to make it possible to

- *Read a specific sequence of bits on the disk*
- *Locate a part of the disk that is not being already used for storing information in order that it can be used to write newly arrived information*

This organization is done by **formatting** and **partitioning**

Formatting and partitioning organize the disk for reading and writing

Disk Organization Introduction

There are three things that must be done to a disk in the order shown

1st: Physical formatting

- Often called "**low level formatting**"
- Done on entire disk

2nd: Partitioning

- Conceptually separates the disk into disjoint non-overlapping parts

3rd. Logical formatting

- Often called "**high level formatting**"
- Done on each partition separately

Disk Organization Introduction

We will discuss *physical (low level)* formatting now

We'll discuss *partitioning* and *logical* (*high level*) formatting later

Physical Formatting

Low Level Formatting

Physical Formatting

Usually done by the disk manufacturer

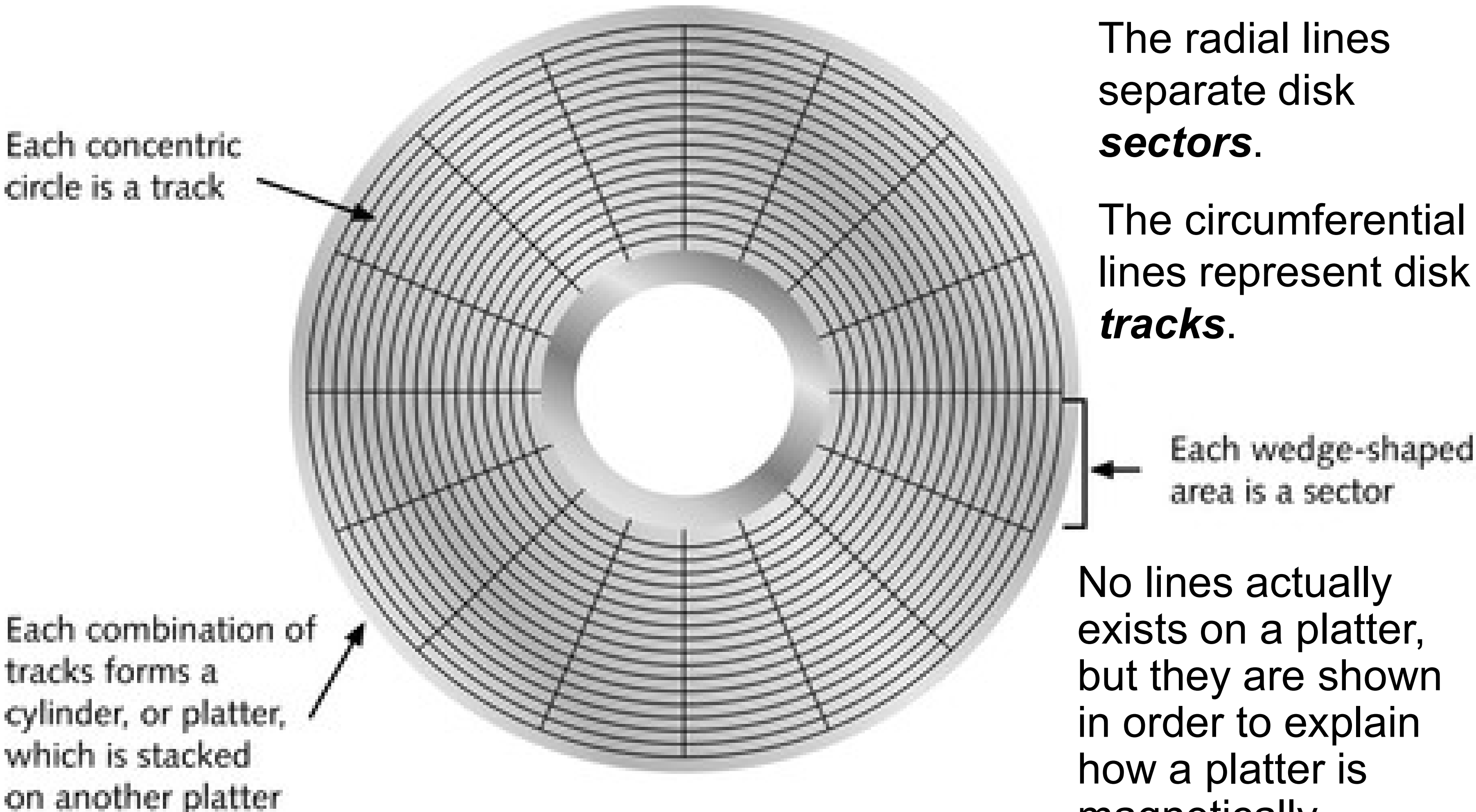
Divides (conceptually) a platter surface into circular **tracks**

Divides each track (conceptually) into arcs called **sectors**

Combines corresponding tracks on multiple platters into (conceptual) **cylinders**

These three elements define the physical locations in which data is physically recorded and from which it is read

Platter Surface Physical Format
Tracks and Sectors



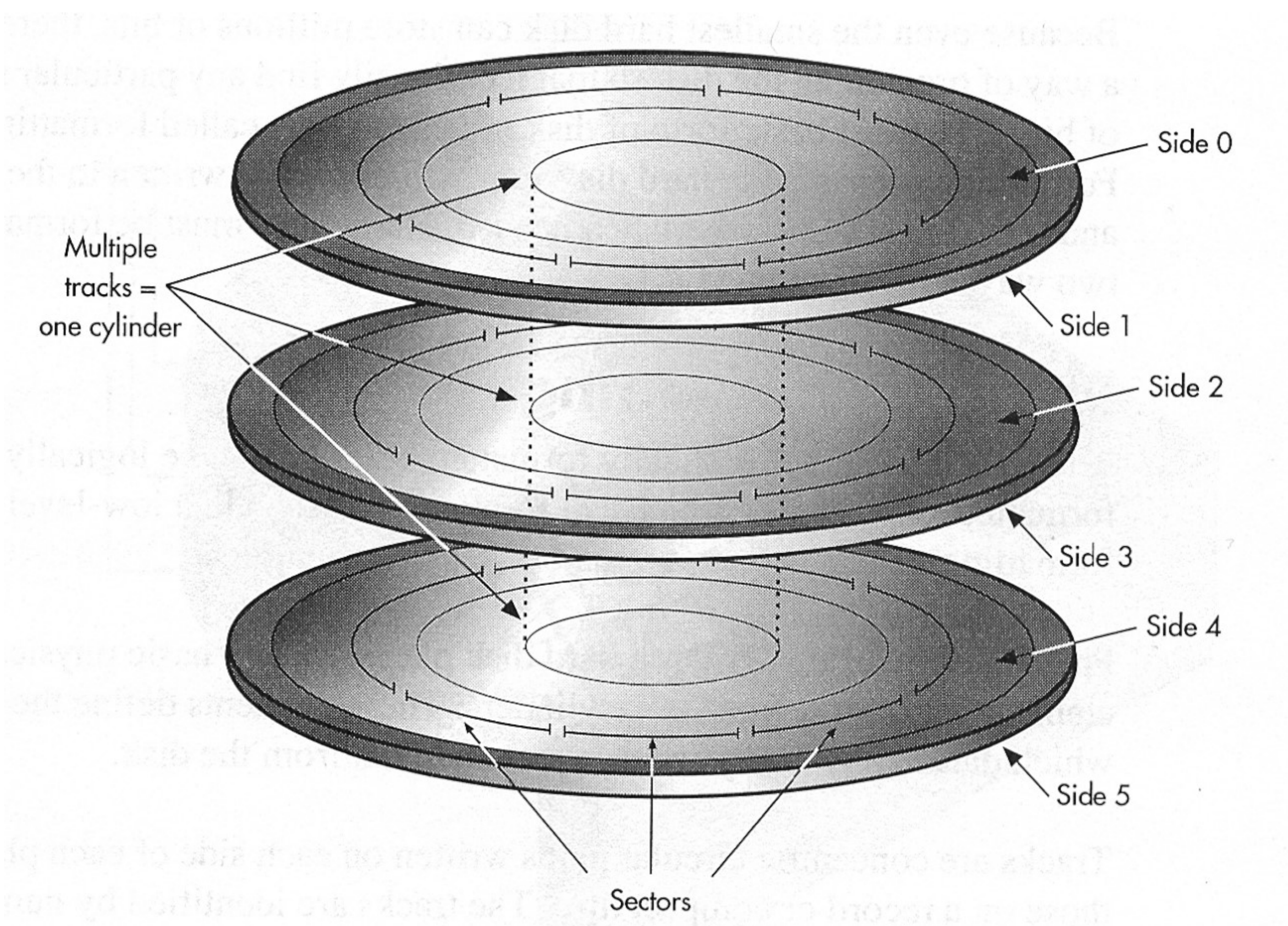
* Nelson p 198

The radial lines separate disk ***sectors***.

The circumferential lines represent disk ***tracks***.

No lines actually exists on a platter, but they are shown in order to explain how a platter is magnetically partitioned.

Disk Physical Format
Tracks, Sectors and Cylinders



Some Details

Physical addressing

We thus have a 3-dimensional structure

Position is defined by three coordinates

- Cylinder
- Head (one head per surface)
- Sector

At the physical level, this is how information is addressed

The atomic unit of information is the ***sector***

Sectors are the smallest unit that is written or read

Some Details

By convention, track numbers start with number **0** at the outer edge of a platter surface

Sectors usually consist of 512 8-bit bytes for data

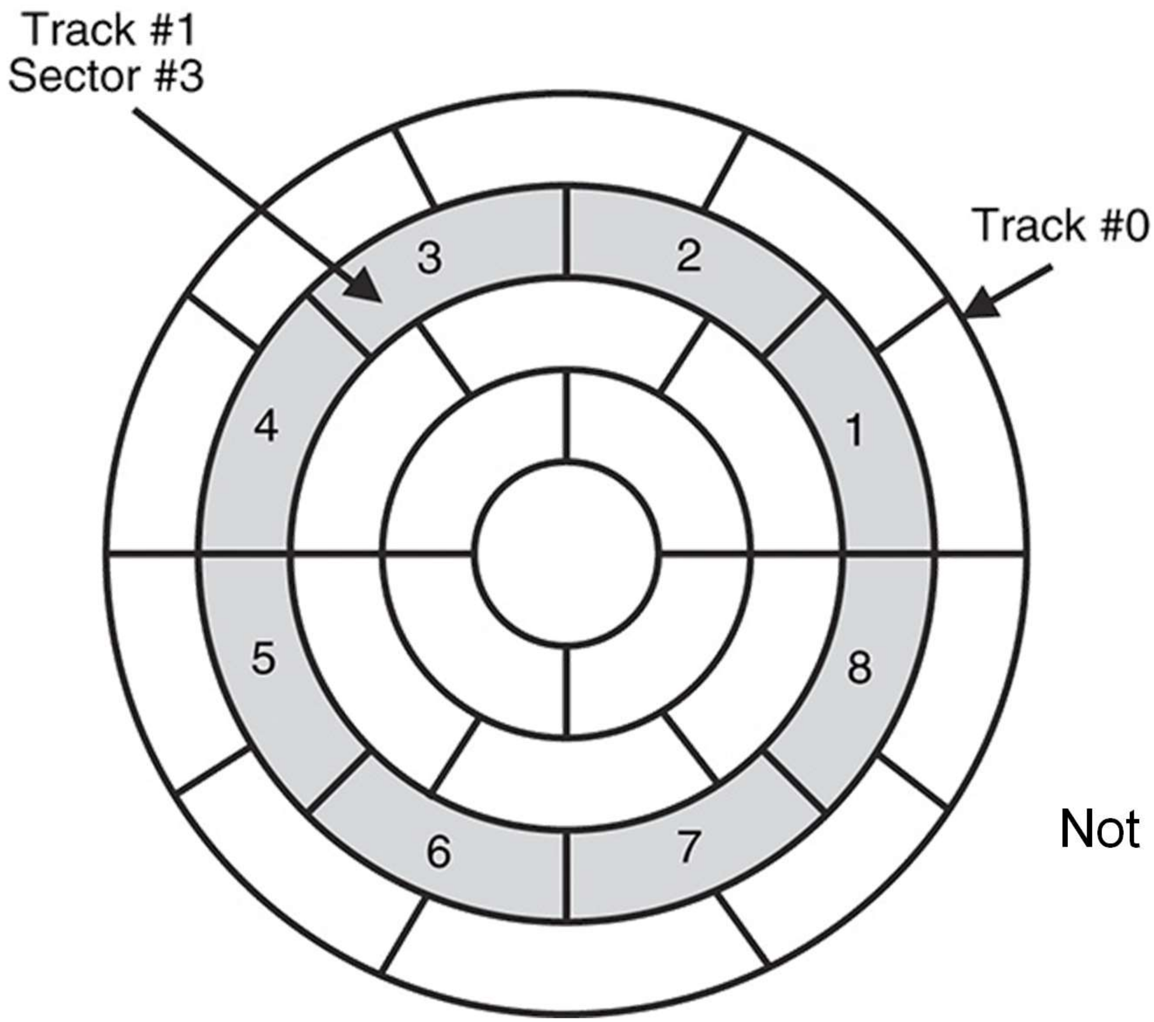
- *Bits are serially strung out along a track over a sector's arc*
- *There are a few more bytes used for a CRC for the sector*
  - OSs, file systems and users never see these

We will deal only with the data bytes

Disk capacity in bytes determined by

- *Cylinders* x *Heads* x *Sectors* x *BytesPerSector*
- *CHS = #cyl* x *#surfaces* x *#sectorsPerTrack* x *512*

Platter Surface Physical Format
Disk Geometry of One Platter



Not drawn to scale

** from Carrier, Chapter 2*

Disk Capacity Example*

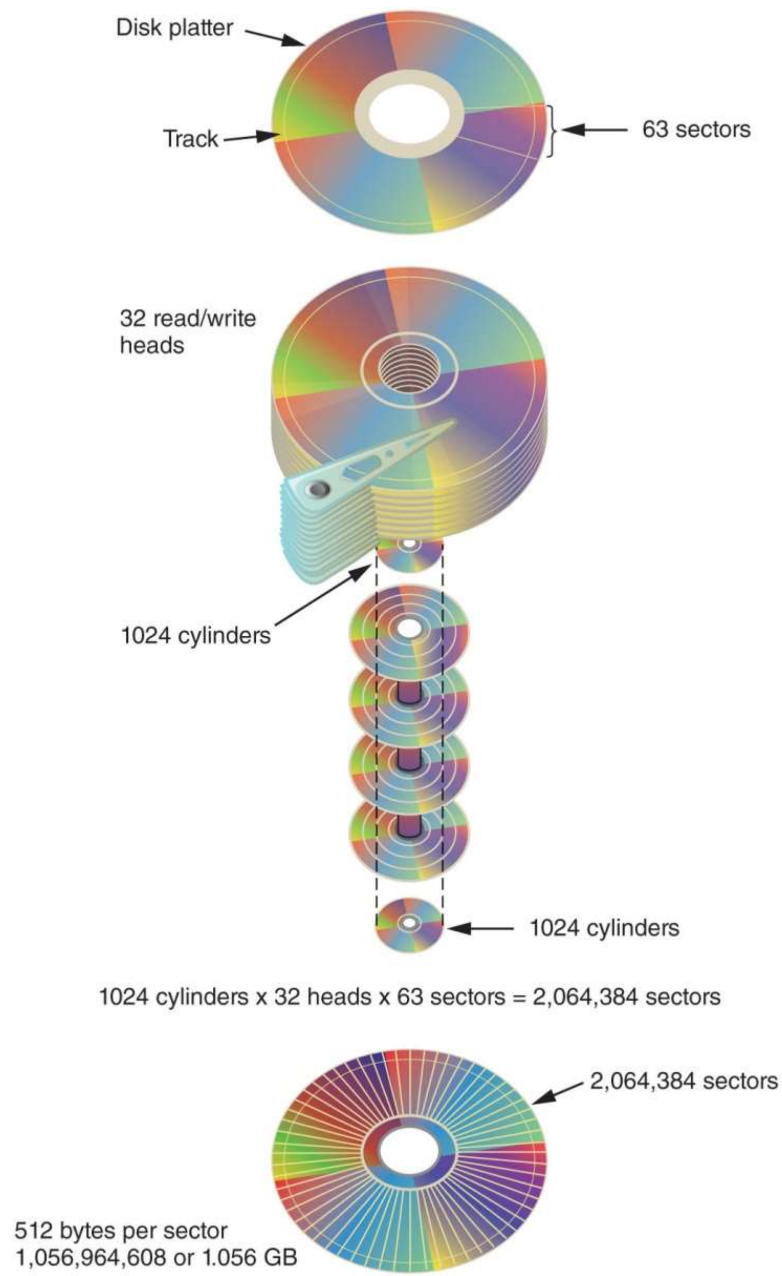


Figure 5-3 CHS calculation

* Nelson p 199

Bad Sectors

Magnetic properties of the surface coating may deteriorate either gradually or due to damage

- *Small deteriorated parts of a sector may get to the point where they cannot be read after writing*

Or, for some reason, the recorded CRC doesn't agree with the CRC taken when the sector is read

The entire sector is then labeled as "bad"

Modern disks don't have many bad sectors

- *Usually none*

File managers can determine when sectors are bad and label them as not useable

ATA Disk Organization

ATA Overview and Standards

ATA means ***A**dvanced **T**echnology **A**ttachment*

It is a standard interface to an ATA storage device

INCITS T13 committee

InterNational Committee for Information Technology Standards

ATA initial standard date: May 1992

ATA/ATAPI Command Set - 3 (ACS-3): early 2014

SATA revision 3.5: July 2020

ATA disks are the most popular disks today

ATA and IDE

ATA means ***A***dvanced ***T***echnology ***A***ttachment

ATA drives have electronics integrated into the device

IDE means ***I***ntegrated ***D***evice ***E***lectronics

Merely means that a storage device has electronics integrated into the device itself

So ATA is the better term to use for the interface

Unfortunately, the terms ATA and IDE are often used interchangeably or misused

ATA Hardware

Computers talk to the IDE on an ATA storage device

- *Through an ATA controller usually on the computer's motherboard*
- *A cable connects the ATA controller to the IDE on the storage device*

There are three types of ATA hard disk interfaces today

- *Parallel ATA (PATA)*
  - Obsolete today, but some are still around
- *External ATA (eSATA)*
  - Primarily used in enterprise applications
- *Serial ATA (SATA)*
  - The most used today

More recently, USB disks have gained acceptance

- *USB/Thunderbolt has replaced eSATA over the years*

ATA Hardware

Parallel ATA (PATA) uses a 40-pin cable

- *Today PATA is often called IDE*
  - This is incorrect, but we must live with it

Serial ATA (SATA) uses a 7-pin cable

- *This interface has almost totally taken over from PATA*
- *SATA is not referred to as IDE*

PATA Hardware

A PATA cable can daisy chain up to two PATA disks

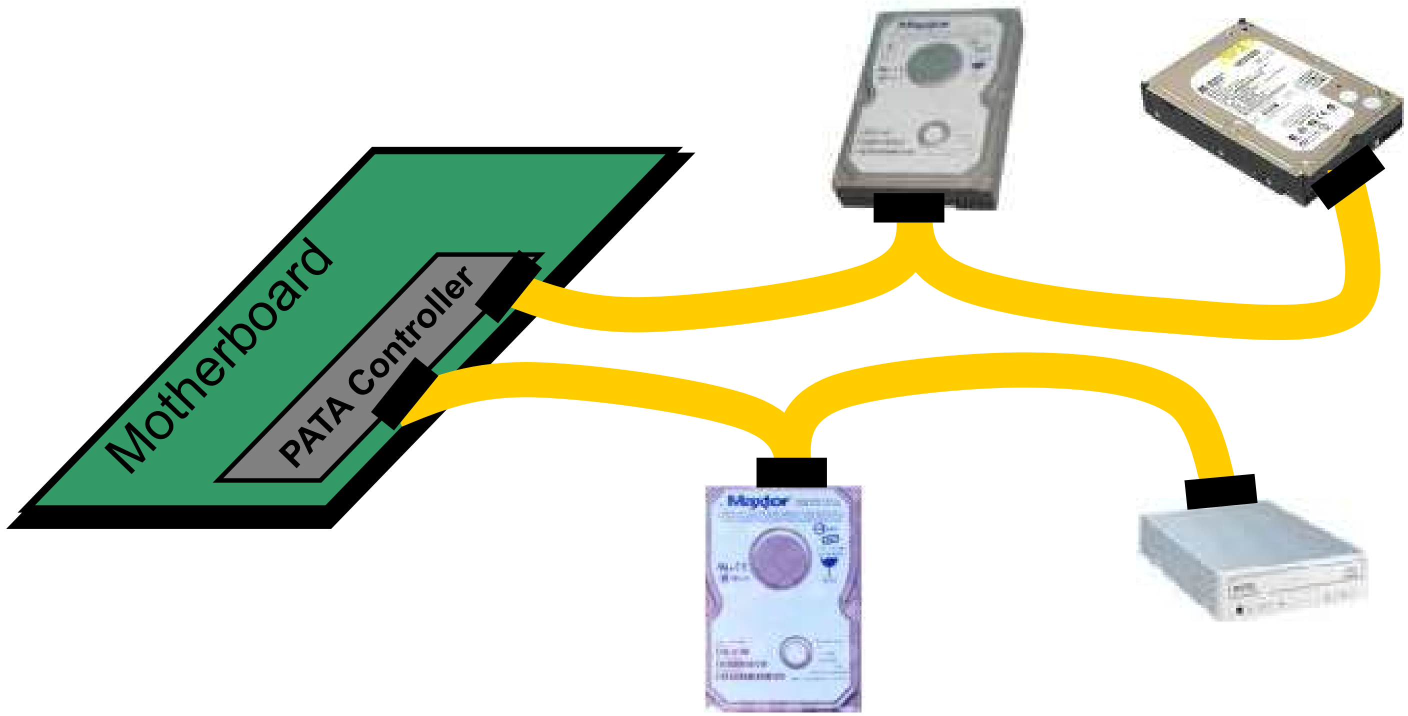
- *A **Master** and a **Slave***
- *Where neither is really a master nor a slave*
- *It is simply a way of distinguishing between the two disks*

PATA disks are configured as a master or a slave

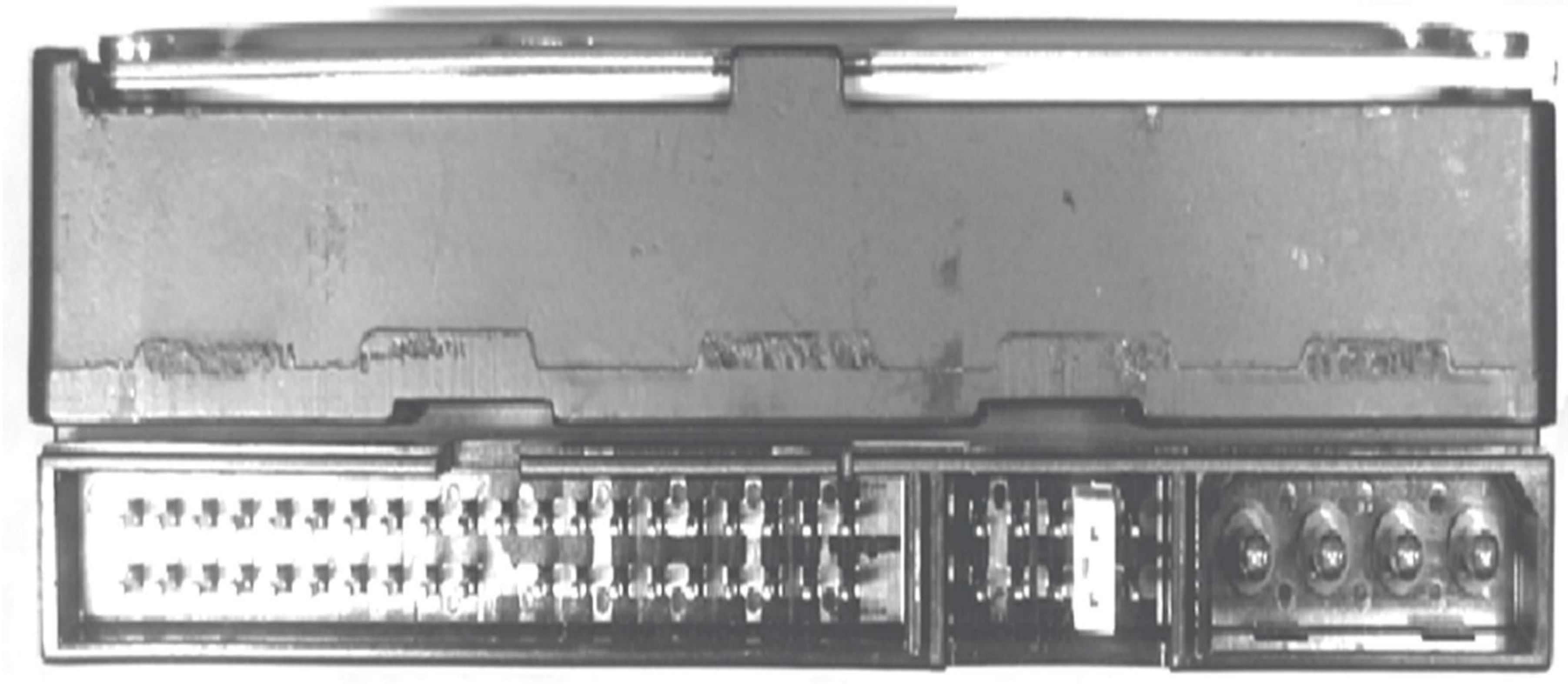
- *Using jumper pins to make it a Master or Slave or*
- *Put a jumper on the "Cable Select" pins*
  - Disk becomes Master or Slave depending upon which connector they use on the ATA cable

Mosty computers today that have a PATA controller support two cables or up to four PATA storage devices

PATA Hardware



Back of an PATA Disk



Ribbon Cable Connector
Used for control and data

Jumpers
To configure

Power

Serial ATA (SATA)

Motivation for Serial ATA

- PATA disks have some disadvantages
  - *40-pin connectors and 40-wire cables somewhat costly*
    - 16 bits of data (bidirectional) + 19 control
  - *Ribbon cables (flat and wide)*
    - Take up space, requiring large cases
    - Interfere with air flow & cooling
    - Flexible only in one dimension, making them hard to route
  - *Round cables are available but expensive*
- Hard disks were shrinking in size from 3.5" to 2.5" & smaller
  - *PATA connectors are bigger than the 1.5" & 2.0" disks*
- Notebook computers

Serial ATA

SATA cables much smaller than PATA cables

Cables have 7 wires & connectors have 7 pins

Much less of a cooling problem in towers

The SATA cables have a small cross section

Don't interfere with air flow

Towers can be smaller

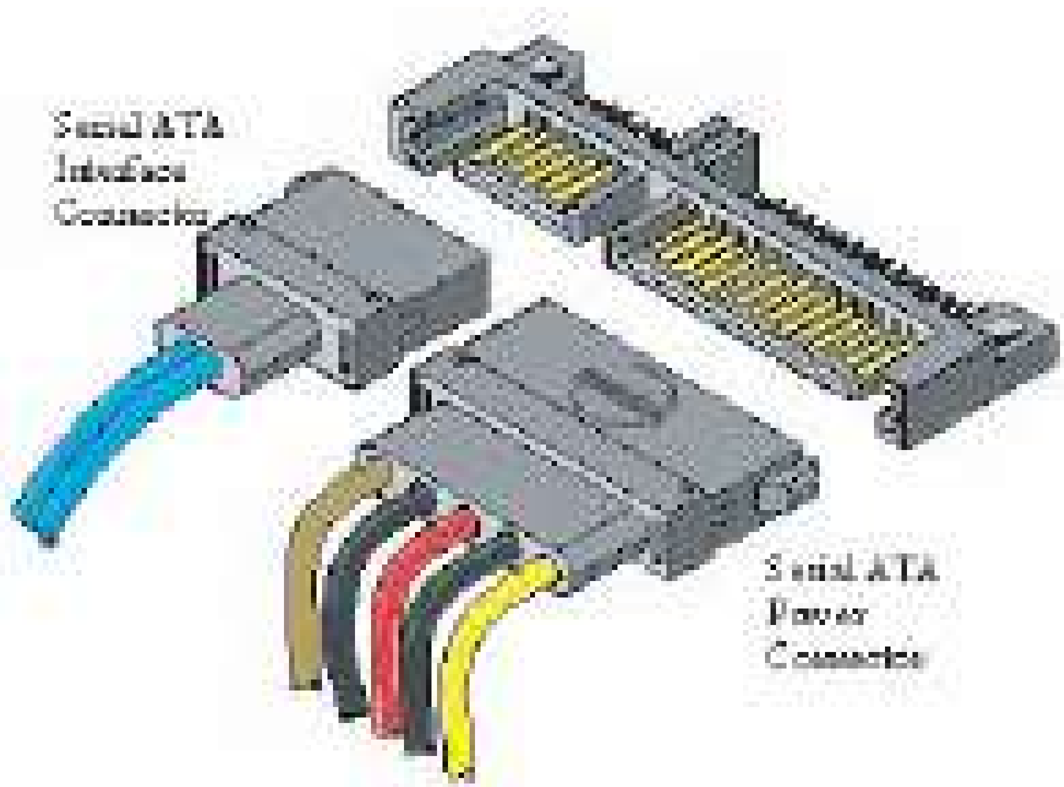
Connector size is compatible with 2.5" and smaller disks

Size is compatible with notebook computers

No master/slave

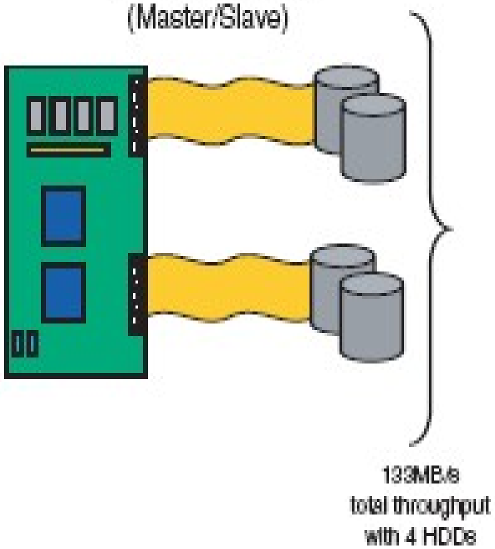
Host sees each disk as a master disk on its own channel

SATA Cables & Connectors



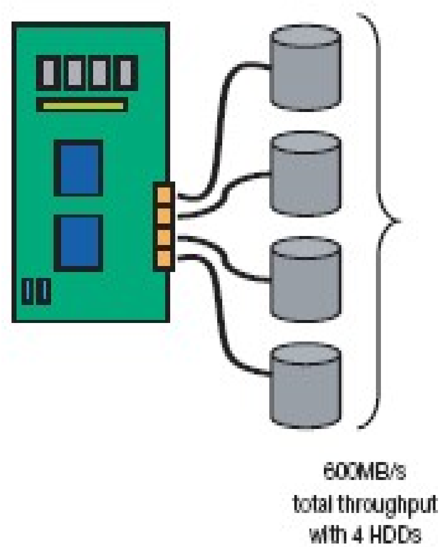
PATA / SATA Comparison

Bus Architecture
(Master/Slave)



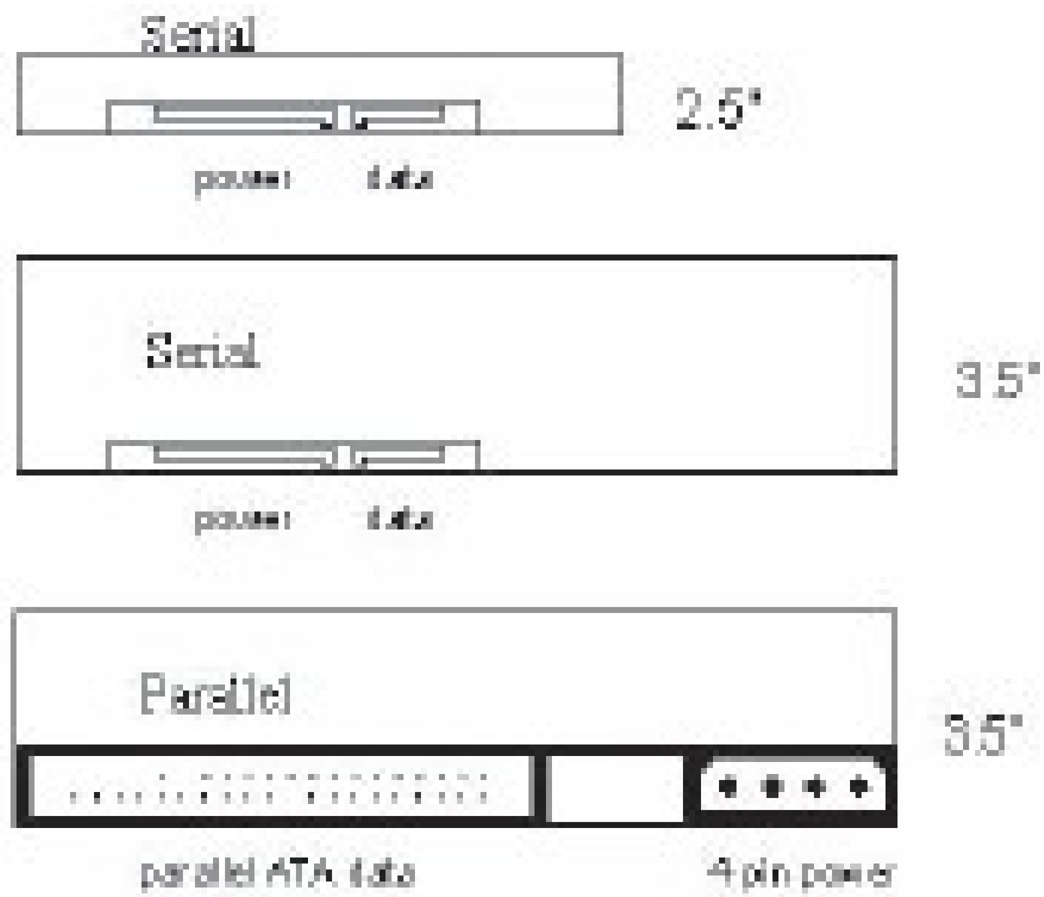
- Supports faster burst rates of 133 MB/sec
- Up to 2 devices per bus (PCs typically have 2 and often 4 ATA buses)
- ATA 66/100/133 require relatively short (18 inches) ribbon cabling, precluding IDE devices being external to the computer

Point-to-Point Connections



- Serial ATA is a point-to-point interface used to connect storage devices such as hard disks, DVD, and CD-RW drives to the PC motherboard
- Serial ATA delivers a scalable interface solution supporting several speed doublings to address the needs of future storage devices
- Directly connected to the host via a dedicated link – entire interface bandwidth dedicated to it with no interaction between devices.
- Eliminates overhead associated with coordinating accesses between the master and slave device sharing the same cable

PATA / SATA Comparison
Connectors



ATA Disk Organization & Addressing

Applies to both PATA and SATA

Disk Capacity Example Again

Today 1 GB is not a very large disk

And actually, even this size is not attainable using standard CHS addressing

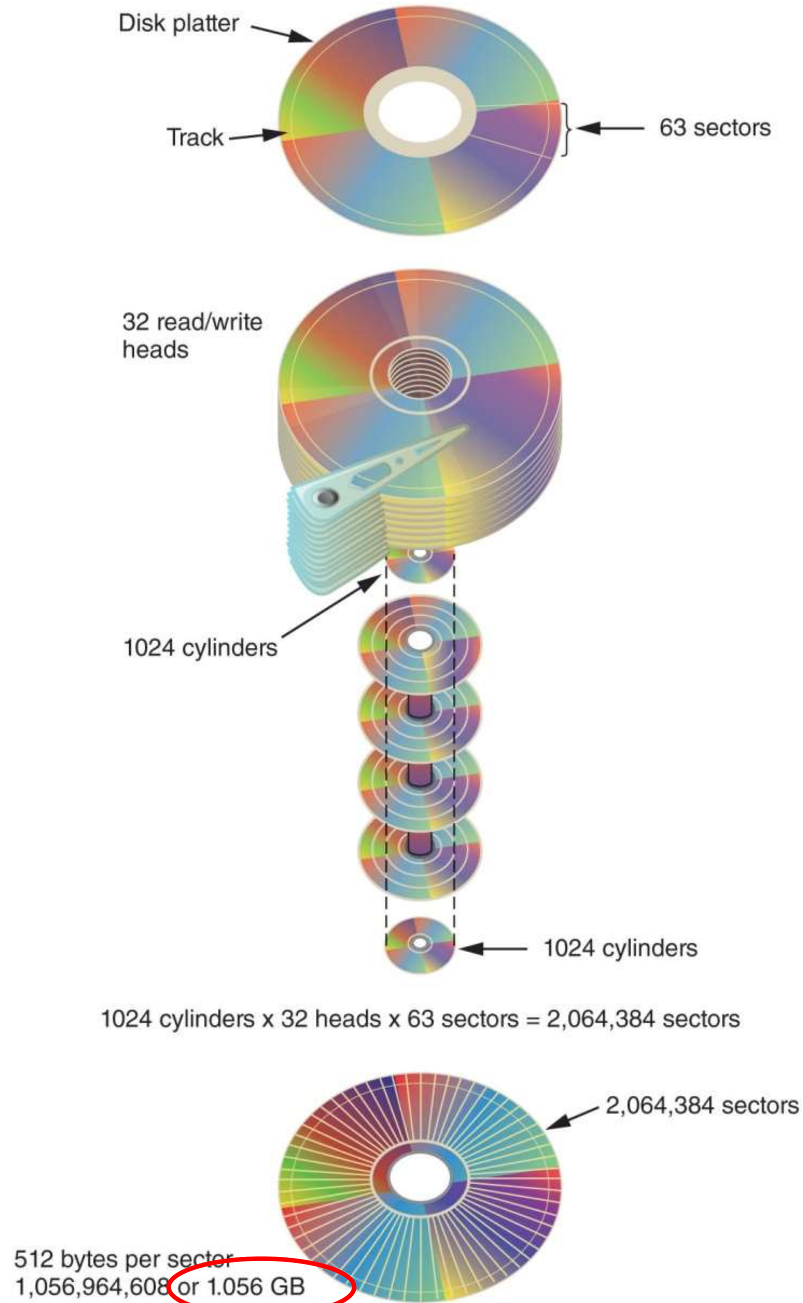


Figure 5-3 CHS calculation

Standard CHS Addressing

Limit is ~500 MB

Why?

Because older BIOS software (INT 13H) combined with the ATA standard had space allocated as follows:

	BIOS	ATA Std	
Cyl	**10** bits	16 bits	(maximum of 1024 cylinders)
Heads	8 bits	**4** bits	(maximum of 16 heads)
Sectors	**6** bits	8 bits	(maximum of 64 sectors of which one is the boot sector)

$1024 \times 16 \times 63 \times 512 = 528 \times 10^6$ Bytes (504 MB)

This was a hard limit

What to Do?

Extended CHS Addressing

Sometimes called *Large Mode* CHS addressing

The idea is as follows:

- *No ATA hard disks were likely to have more than 8 platters with 16 heads (4 "head" bits in the BIOS)*
  - So a BIOS space of say 256 heads (8 bits) is too large
- *No hard disks were likely to have more than 64 sectors per cylinder*
  - So the ATA Std of 256 sectors (8 bits) is too large

The BIOS should use the logical geometry that the hard disk specifies according to the ATA standard

Translates it into an equivalent geometry that will "fit" into the space allowed by the BIOS Int 13h standard

Extended CHS Addressing
Example

Suppose we have a hard disk that has, per ATA Std:

3.1 GB *6,136 cylinders*

16 heads *63 sectors*

This exceeds the BIOS limit of 1024 cylinders

A modified BIOS does minor bit-shift calculations

$6136 \div 8 = 767$ *"effective" cylinders (shift right 3 bits)*

$16 \times 8 = 128$ *"effective" heads (shift left 3 bits)*

The capacity is the same 3.1 GB

$767 \times 128 \times 63 \cong 3.1GB$

Traded cylinders to get heads

Extended CHS Addressing*

Standard	Maximum Cylinders	Maximum Heads	Maximum Sectors	Maximum Capacity
IDE/ATA	65,536	16	256	128 GiB
BIOS Int 13h	1,024	256	63	7.88 GiB
Combination (Smaller of Each)	1,024	16	63	504 MiB

	Cylinders	Heads	Sectors	Capacity
IDE/ATA Limits	65,536	16	256	128 GiB
Hard Disk Logical Geometry	6,136	16	63	2.95 GiB
BIOS Translation Factor	divide by 8	multiply by 8	--	--
BIOS Translated Geometry	767	128	63	2.95 GiB
BIOS Int 13h Limits	1,024	256	63	7.88 GiB

Extended CHS Addressing Not Enough

Large Mode or Extended CHS addressing allowed larger disks than standard CHS

But this rapidly was outstripped by disk technology

So Extended CHS Addressing was not the answer

What to do?

- *Throw out CHS addressing completely*
- *So, another totally different scheme was standardized and used*
  - Called **Logical Block Addressing** (**LBA**)

Logical Block Addressing (LBA)

Assign each sector a number in a linear number space beginning with zero

- *0, 1, 2, … to (n-1) for a disk with n sectors over all the platter surfaces*
- *Geometry is irrelevant*
- *Only size-relevant parameter is the number of sectors,* ***n***

Today all disks do LBA

- *CHS was dropped from the ATA-6 standard*

All new ATA / ATAPI disks accept LBA

ATA Commands

SATA, ATA and ATAPI* have the same basic ATA command-set

** ATA Packet Interface (used for removable media)*

ATA Passwords
Both PATA and SATA

ATA-3 (1997) standard specified two different optional disk passwords

- *User password*
- *Master password*
- *This has been extended to SATA*

If passwords are used, then there are two security modes

- *High*
  - Both passwords can unlock the disk
- *Master*
  - Only the Master password can unlock the disk

This is not disk encryption

ATA Passwords
Both PATA and SATA

After several unsuccessful attempts to unlock, the disk hangs

- *Must then reboot to try more passwords*

Programs exist that will allow you to unlock the disk if you have the password

- *atapwd (PATA only)*
- *Hdunlock (SATA & PATA)*
- *MHDD (SATA & PATA)*

Unlocking a Hard disk Drive

Boot from a bootable Linux CD, DVD or flash drive that has a hard disk unlock program on it

e.g., Hiren's Boot CD

Run the unlock program

General schemes

1. Delete the password, which then allows you to access the drive OR

2. Replace the password

You can do this from Linux because Linux allows you to send ATA commands directly to the drive, bypassing both the OS and the file system

ATA and ATAPI

ATA is the acronym for **AT Attachment**

ATA standards apply to hard disks

Specifies hardware, electronic signals, commands and operations regarding how a hard disk attaches to a computer

ATAPI is the acronym for **ATA Packet Interface**

ATAPI standards apply to removable media

ATAPI and **ATA** use the same cables and controllers but different drivers

SATA/PATA Commands

SATA 3.5 – The latest officially released standard (2020)

Commands by which the motherboard controller and the disk communicate

The IDE *(Integrated Device Electronics)* on the disk has several registers e.g.,

Command register *Sector register*

Size register *Data register*

Controller sets the registers with the command and data

Controller then tells IDE to start

Two Hidden Disk Areas

Two Hidden Disk Areas

Host Protected Area (HPA)

Sometimes referred to as ***Hidden Protected Area***

Device Configuration Overlay (DCO)

Host Protected Area (HPA)

An HPA part of disk not visible to normal users

Protected from users

Usually located at beginning or end of a disk

Used by computer vendors to store info that users cannot see or manipulate

e.g., Recovery software and data from last time machine was shut down

Dell uses it to hold Dell Media Direct software

Can also contain malware such as worms and rootkits

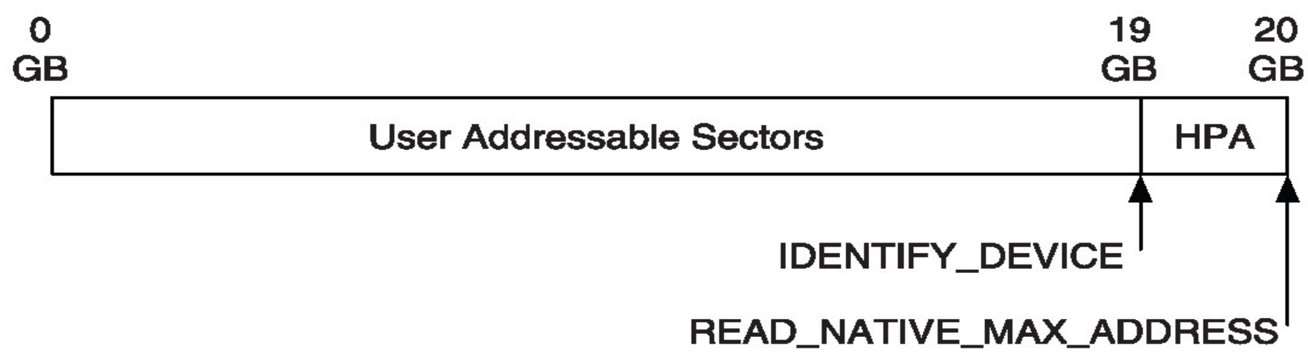
Host Protected Area (HPA)

Host Protected Area (HPA)

Does your disk have an HPA?

There are two ATA commands that return values for the maximum addressable sectors

- *If an HPA exists, the values returned will be different*
- **READ_NATIVE_MAX_ADDRESS** *returns the number of the last sector at the end of the disk (but not always)*
- **IDENTIFY_DEVICE** *returns the last sector that users can access*



Host Protected Area (HPA)

To create an HPA

Use **SET_MAX_ADDRESS** *to the maximum numbered sector that user should access*

To remove an HPA

Use **SET_MAX_ADDRESS** *to value obtained by first using* **READ_NATIVE_MAX_ADDRESS**

Host Protected Area (HPA)

- Volatility Bit
  - **SET_MAX_ADDRESS** *has a parameter that is used to make the HPA permanent*
  - *If set, the HPA remains over disk resets and power cycles*
- ATA lock commands
  - *Prevents modifications to the maximum user address until the next disk reset*
  - *Also a password can be applied to HPA*
- These commands are used by the BIOS to set info into the HPA when booting and keep it unchanged

Device Configuration Overlay (DCO)

In addition to the HPA, a disk can be configured to have another hidden area at the high-end sectors of the disk

Called a ***Device Configuration Overlay (DCO)***

Makes disk appear smaller that it really is

Device Configuration Overlay (DCO)

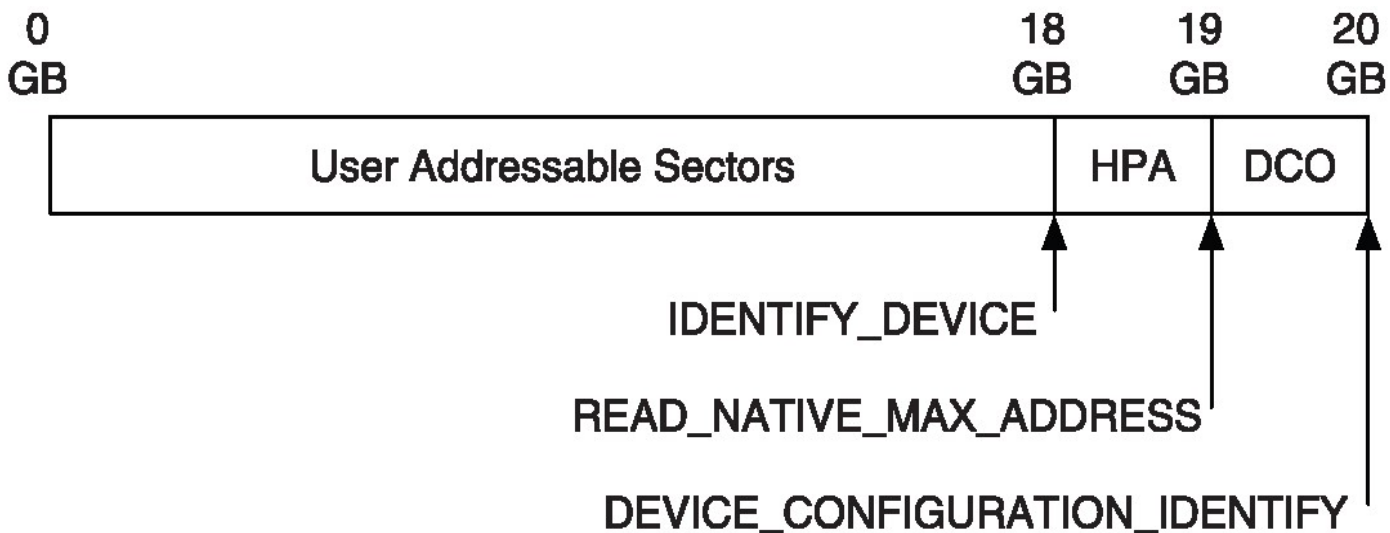
To detect a DCO, use the ATA commands:

IDENTIFY_DEVICE

READ_NATIVE_MAX_ADDRESS

DEVICE_CONFIGURATION_IDENTIFY

Give the true physical information about the disk



Device Configuration Overlay (DCO)

To add or change a DCO use

DEVICE_CONFIGURATION_SET

To remove a DCO use

DEVICE_CONFIGURATION_RESET

DCOs are permanent across resets and power cycles

Uses ?

Computer vendors can make different size disks from different manufacturers appear to be the same size

Also used to store info that users cannot see or erase

e.g., Image of operating system to reinstall or malware

Disk Access

Overview

We can access PATA & SATA disks in two ways

- *Via the BIOS or GPT*
- *Directly*

BIOS Disk Access

BIOS does a lot of things when you boot your computer

Two BIOS activities relevant to disks

- *Finds out what disks exist and their configurations*
- *Loads the Interrupt Table*

Disk drivers in the OS then communicate with the BIOS to read and write disks

Interrupt 0x13 aka ***INT13h*** contains the starting address of the BIOS code that interacts with the disk controller hardware

- *Includes functions that read & write the disk, format tracks on the disk, and query the disk for information*

BIOS Disk Access

Original ***INT13h*** functions for reading and writing used CHS addressing

Initial ***Extended INT13h*** functions

- *Kept the original capabilities*
- *Added BIOS code that handles 32-bit LBA addressing*
  - 4.3×10^{9} sectors (~ 2.2 TeraByte disks)

Latest ***Extended INT13h*** functions

- *Kept the original capabilities*
- *Modified BIOS code to handle 64-bit LBA addressing*
  - 1.8×10^{19} sectors (~ giga TeraByte disks)

Direct Disk Access

Bypass the BIOS

Must be able to communicate directly with the hard disk controller

e.g., The ATA command codes, what sectors to read and write and query disks for its configuration details

Special software exists for doing this

Or you can write assembly code

Software is very hardware-specific

Small Computer System Interface (SCSI)

SCSI Overview

SCSI disks are not as common as ATA disks

But they are used on many non-PC based systems and many large servers

*While the word "Small" in the SCSI name **was** appropriate,*

Today SCSI is used in the very largest server systems

SCSI standards are the responsibility of the INCITS T10 committee

INCITS → International Committee for Information Technology Standards

T10 → Responsible for SCSI Storage Devices

Currently the "umbrella" parallel SCSI standards are

SCSI-1 through SCSI-5

Parallel Interfaces

Parallel SCSI Interfaces

- *50, 68, 80 pin connectors and cables*
- *Three different electrical signaling schemes*

Other non-parallel interfaces (some but not all)

- *SAS (Serial Attached SCSI)*
- *FCP (Fibre Channel Protocol)*
- *UAS (USB Attached SCSI)*
- ...

SCSI Overview

SCSI doesn't need a controller on the motherboard

SCSI is really a bus with up to 16 devices on it

Addressed with 4 address bit wires

The motherboard is just another SCSI device, usually with address=0

Each SCSI device has its own integrated electronics

Makes SCSI disks more expensive than ATA

SCSI always used 32-bit or 64-bit LBA sector addressing

Thus, no size issues as with earlier ATA

Many SCSI disks have a jumper that makes the disk read-only

Don't need a write blocker

SCSI Overview

SCSI cables can be much longer than PATA or SATA cables

Can have as many as 15 storage devices plus the computer (not the controller)

- *All on one cable*

A bit harder to install devices for some

- *Must terminate the end of cable*
- *Must set the 4-bit SCSI address*

The Many Types of Parallel SCSI

Type	Frequency	8-bit Transfer Rate	16-bit (wide) Transfer Rate
SCSI (normal)	5 MHz	5MB/s	10MB/s
Fast SCSI	10 MHz	10MB/s	20MB/s
Ultra SCSI	20 MHz	20MB/s	40MB/s
Ultra2 SCSI	40 MHz	40MB/s	80MB/s
Ultra3 SCSI	80 MHz	N/A	160MB/s
Ultra160 SCSI	80 MHz	N/A	160MB/s
Ultra320 SCSI	160 MHz	N/A	320MB/s
Ultra640 SCSI	320MHz	N/A	640MB/s

Signals on the Wire

There are three different types of electronic signals that are possible on SCSI devices

SE: *Single ended*

HVD: *High voltage differential*

LVD: *Low voltage differential*

Differential types of signals are needed for the faster types of SCSI busses on longer cables

The most common signaling for new disks is LVD

A Rule: <u>Never</u> mix two types of signaling

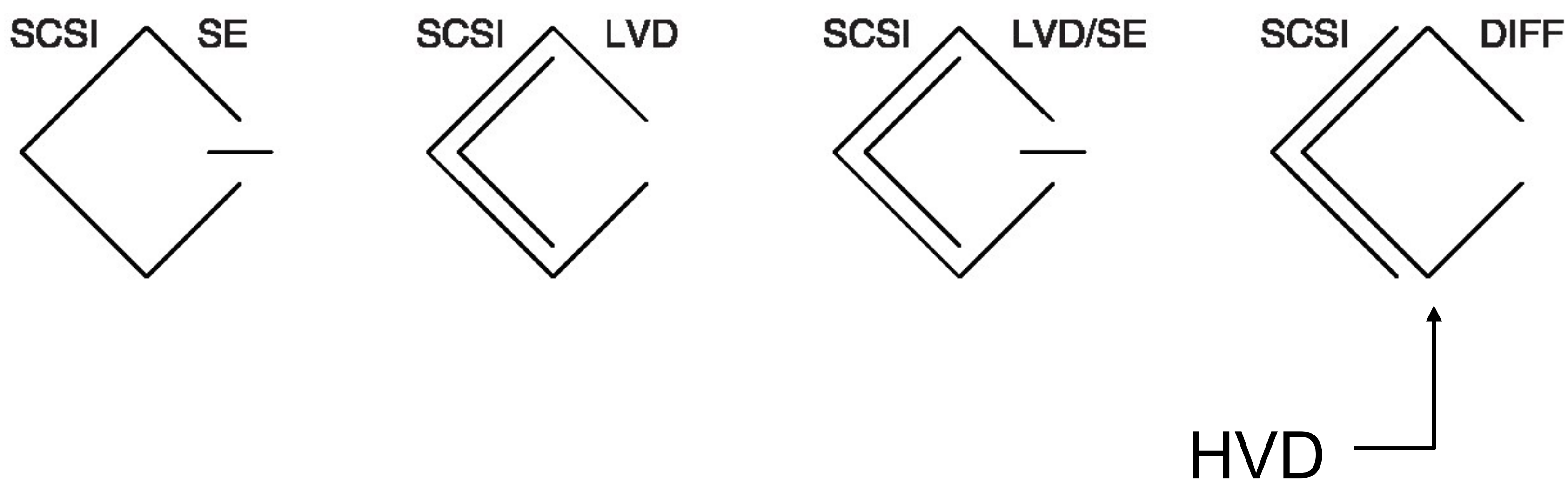
You'll damage the device and other devices on the bus

SCSI Types and Signaling

Signal Type	SCSI Types
SE	SCSI, Fast SCSI, Ultra SCSI
HVD	SCSI, Fast SCSI, Ultra SCSI, Ultra2 SCSI
LVD	Ultra2 SCSI, Ultra3 SCSI, Ultra160 SCSI, Ultra 320 SCSI**, Ultra 640 SCSI**

Electronic Signaling Symbols

Here are symbols that are often used on SCSI devices



Some Other Storage Device Interfaces

SAS: Serial Attached SCSI

Eliminates the large flat SCSI cables

FC: Fibre Channel

Uses optical fiber cables

We won't consider these further in this course

Except for the comparison on the next page

Storage Device Interface Comparison

	ATA	SATA	SCSI	SAS	FC
Interface Type	Parallel	Serial	Parallel	Serial	Serial
Addressing	2	1 or 16 w/SATA II	16	128	16 million
Distance (m)	.5	1	12	10	10km
Connection	40 -pin	7-pin	68-pin internal/ext	7-pin	copper/optical internal/external
Dual Port	no	no	no	yes	yes
Topology	bus	pt. to pt.	bus	pt. to pt. with expanders	loop, fabric
Duplex	half	half	half	full	full
Max. Devices	2	1 or 15 with port multiplier	16	4096 with expanders	127-loop 2^24 fabric
Cable Length	0.4m	1m	12m	10m	30m (copper) 300m (optical)
Applications	Internal Storage	ATA RAID, Server & high-end work-station storage	Mid-range & Enterprise servers	Mid-range and Enterprise servers	SAN and Enterprise servers