CSE 405
(Computer Interfacing)
Disk and Tape Storage

(Computer Peripherals by Cook & White)

Outline

Introduction

Stores which holds programs and data has been divided into: Main

- storage:
  - Typically semiconductor memory
  - Fast but expensive
  - Retains information while power is connected
  - RAM
- Backing storage:
  - Slower
  - Much larger
  - Retains information even when power is disconnected
  - Either magnetic or optical HARD DRIVE, CDs..

Magnetic Drum

- First type of **backing storage.**
- Drum has a **magnetic surface** (iron oxide)
- Small **electromagnets** are used known as **read/write head**
- Magnetization can be done in one of two directions: **north or south.** Area around the circumference of the drum is known as track Each
- head covers one track
- Can also use just one read/write head
-
    - Common on modern disk drives
    - Moves up and down the drum's length
    - Reduces cost of unit, as fewer heads are required but introduces time delay

Magnetic Drum

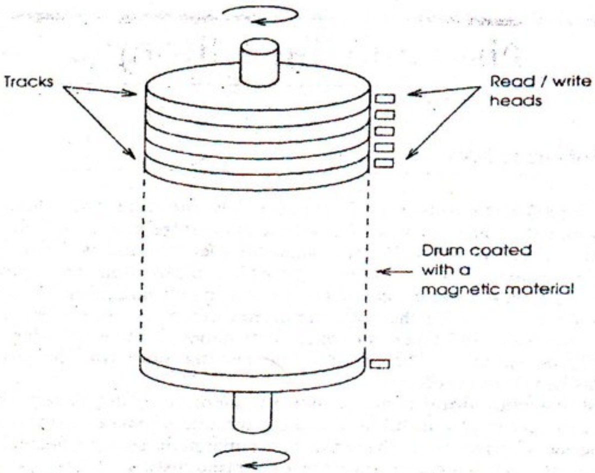
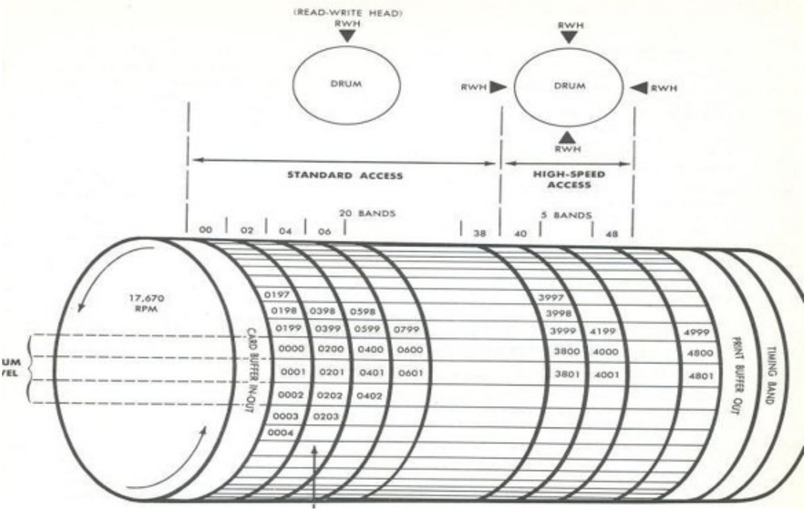


Figure 5.1 *Old style magnetic drum storage*

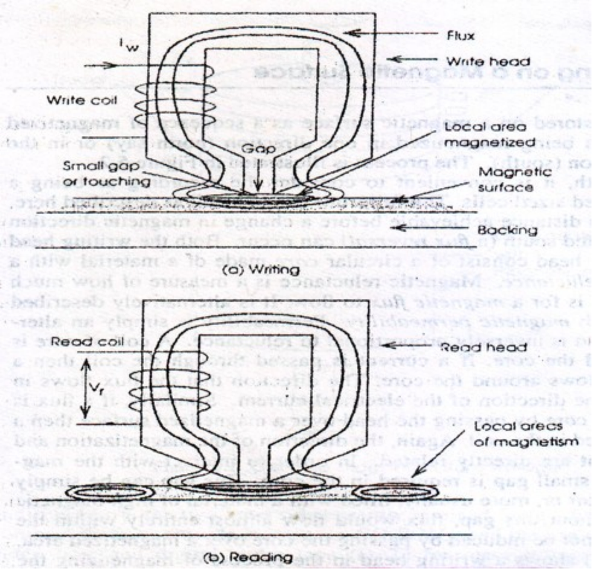
Magnetic Drum



Recording on a Magnetic surface

- Information is stored on a magnetic surface as a sequence of magnetized cells
    - Each is magnetized in one direction
- Writing and reading head consists of a **circular core** made of a **material with low magnetic reluctance**
- Coil of wire is wrapped around the core
- When current pass through the coil- **magnetic flux flows** around the core, the direction of which depends upon the direction of the current.

Magnetic Drum R/W



Magnetic Drum (WRITE)

- Magnetic flux is produced by passing current through the coil Causes
- the flux to spread out and flow through the surface
  - It magnetizes a small area of the surface(cell)
- The surface moves and sequence of cells are written **Coils**
- **should have few turns**
  - In order to **reduce inductance** to permit fast current changes.
  - Spacing of cells is critical

Magnetic Drum (READ)

- As the surface moves , **a flux change in the core is detected by the induced voltage in the coil**
- This voltage **is proportional to the rate of flux change**
- Coils should have **a large number of turns to maximize the detection of flux changes**

- Cells are magnetized with north and south poles lying across the surface.
- This process of recording is known as *Horizontal recording*

Recording Codes

Coding schemes to write data on magnetic surfaces--

- NRZ (Non Return to Zero)
- NRZI (Non Return to Zero invert)
- PE (Phase Encoding)
- FM (Frequency Modulation)
- MFM (Modified Frequency Modulation)
- RLL (Run Length Limit)

Recording Codes

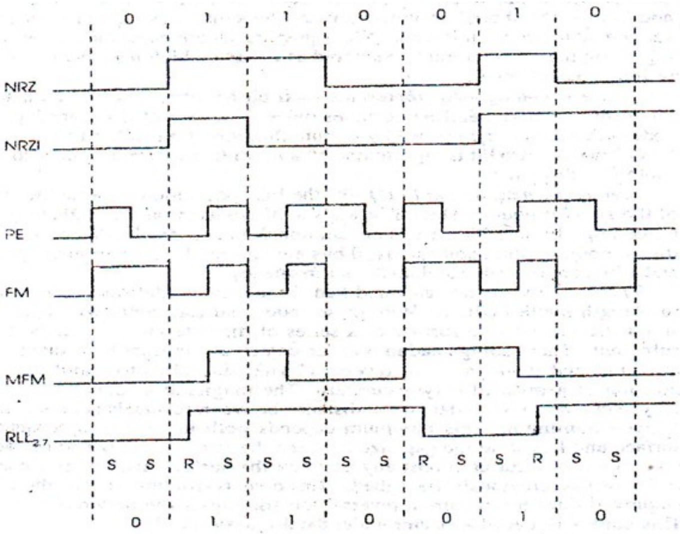


Figure 5.7 *Recording codes*

Run- Length- Limited (RLL)

- Specify a minimum and maximum distance between flux reversal
- A binary code in which a 1 is inserted after a certain number of 0's, in order to avoid long strings of 0's, which would require very accurate clocking in order to ensure that a bit was not lost.
- Abbreviated as RLL code.
- The word **"run" here refers to a sequence of spaces** in the output data stream **without flux reversals**
- The two parameters that define RLL are the ***run length*** and the
- ***run limit***.
- **Run length :** the minimum spacing between flux reversals
- **Run limit :** the maximum spacing between flux reversals
- The particular variety of RLL used on a drive is expressed as "RLL (X,Y)" or "X,Y RLL" where X is the run length and Y is the run limit. For example : $RLL_{2,7}$ $RLL_{1,7}$

Run- Length- Limited (RLL)

Group coded recording

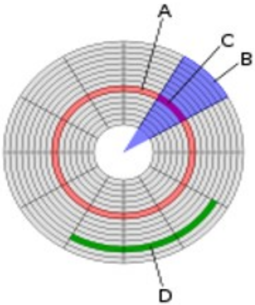
4-bit data	5-bit code
0000	11001
0001	11011
0010	10010
0011	10011
0100	11101
0101	10101
0110	10110
0111	10111
1000	11010
1001	01001
1010	01010
1011	01011
1100	11110
1101	01101
1110	01110
1111	01111

RLL(0,2)

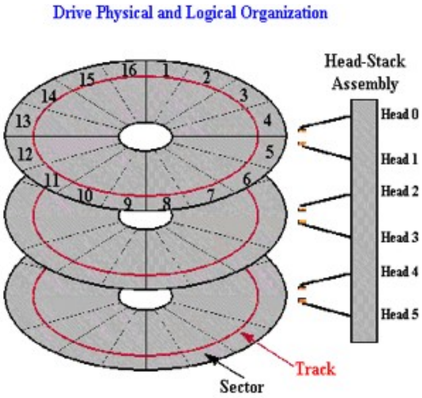
1: Reversal
0: Space

0: at least 0 zeros between consecutive 1s
2: at most 2 zeros between consecutive 1s.

Magnetic Disk



(A) Track
(B) *Geometrical* **sector**
(C) Track sector
(D) Cluster



Magnetic Disc Formats

- **Surface**:
  - There are two surfaces of a circular magnetic disc
  - Information is stored on both surface
- **Track:** Information is stored on tracks
- **Sector:**
  - Sector is the subdivision of track
  - Each track is divided into equal sized lengths by sectoring
- Tracks **near the edge are longer than those near the centre**:
  - Each track stores same amount of information
  - Information is packed more densely in inner tracks
  - All other tracks are less densely and thus space is wasted
- Floppy disk contains **100 tracks, and 8 to 18 sectors**
- In case of hard disk, 1000 tracks and many more sectors
- Each unit of information on the surface has an **address consisting** of two components: **track number & sector number**

Magnetic Disc Formats

Block: Minimum sized unit

- **One sector of one track is called a Block**
- Large block size increases effective efficiency of the surface
- Space will be wasted when smaller size is actually required
- On average, t**he last block will be half used**.
- So, if a surface stores, 1000 files, the equivalent of 500 blocks will be unused.
- A compromise is made; the amount of control information is much the same regardless of the block size, so this is in favor of large block sizes.

Cylinder:

- **Collectively all the tracks – one per surface-** that are addressed simultaneously are termed a cylinder
- If a single file is stored in a cylinder , then the track number is same, **it will decrease the seek time**.

Magnetic Disc Formats

- In order to transfer information, the head has to be **positioned over the required track**. This is called seeking, and the time it takes is called **seek time.**
- The disc **unit waits for the required sector of the track** to appear under the head. The time lag it creates is called **Latency**.

Disc access time = seek time + Latency + transfer time

Magnetic Disc Formats

[redacted]

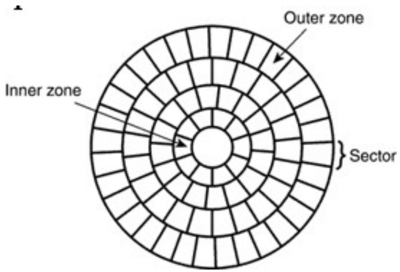
A disk has:

- 4 platters
- 32 sectors/track
- 2048 tracks/surface
- 65,536 blocks/surface
- 512 bytes/block
- Usually 95% of data can be stored per disk.

How many surfaces, sectors, tracks, cylinders, blocks, and bytes can be stored, actual size?

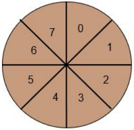
Zonin

- Outer tracks are loaded at much lower density than they are capable of.
- Here the surface is divided into different zones.
- The number of sectors is constant within a zone.
- Outer zones contain more sectors for each track

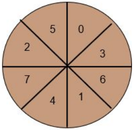


Interleaving

- If necessary block access are from same track (No time is wasted in seeking)
- If not in the same track but also in consecutive sectors on that track (Latency is also eliminated)
- Some disk controllers are not capable of reading **successive** blocks. (Delay in processing one sector results in next one having passed under the head)
- Sectors can be interleaved
- The choice of interleave factor depends upon the usual delay between reading block



Interleave factor of 1



Interleave factor of 3

Recording Codes Type

Horizontal recording: Cells are magnetized with **north and south poles** lying across the surface.

- Both surface can be recorded independently.
- simpler and cheaper.

Vertical recording: Cells are magnetized with north poles lying on one surface and south poles lying on another surface.

- Both surface can't be recorded independently.
- complex and expensive.
- not commonly used.

Magnetic Disk Organization and Head Positioning

- In order to identify **tracks and sectors**
    - **Positional control information** is recorded in addition of user data
    - Writing of this information is called **formatting disc** (low level formatting). Normally performed once.
- Later, a high level of formatting is performed by OS to permit the blocks to be organized in a file store.
- Reading/writing information from disc consists of 3 parts:
    - Seeking
    - Latency
    - Data read/write

Control Information

- **Positional information and error checking** information is stored in addition to user data in low level formatting.
    - Identifies track and sector,
    - ensure the integrity of data.
- Positional information can be written
    - **As a series of servo tracks on a dedicated surface**
    - Embedded within the data on each surface – more common
- **In case of servo track**, info are written on one surface. Each servo track contains the identity of a track and the **starting of each sector.**

Control Information

Index marker	Sector 0 header	Sector 0 data	Sector 0 trailer	Sector 1 header	....

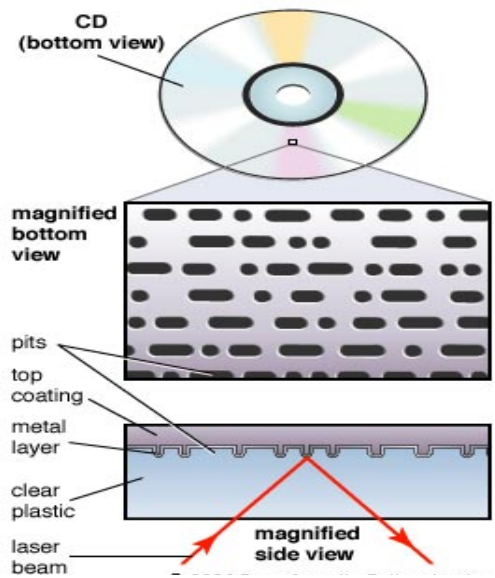
- In case of **embedded technique,** the info are stored alongside the data.
- Index marker:
  - Establishes **starting position** of the track
- Each sector is surrounded by header and trailer
  - Header identifies **track and sector** (block number)
  - **Trailer contains error checking information** (such as: CRC)

Optical Storage

Optical Storage

- Retains its information when power is removed
- Codes data as variations on an optical surface
- Variations affect how light is reflected
- Surface perfectly reflects when variations (bumps and pits) are not present
- Amount of light directly reflected from bumps and pits are less than that reflected from smooth surface (Gives means of binary storage)
- **Main difference with magnetic disk is: Optical recording requires much larger degree of error detection and correction.**
- Advantage:
  - High Capacity
- Disadvantage:
  - Not re-writable
  - Data transfer rate is slower

Optical Storage



Forms of Optical Disc storage

Common forms:

- **Compact Disc Read-Only Memory (CD-ROM):** Information is recorded at the time of manufacture.

  Advantage:

  1. Can hold large amount data like over 600MB.
  2. Useful for data distribution and large read-only databases.

  Disadvantage:

  1. Can't alter information.

- **Write Once Read Many (WORM):** Permits writing once and reading.

  Disadvantage:

  1. Once write can't be altered.
  2. In case of altering, the surface should be replaced.

- **Magneto- Optical disc (MO):** Permits re-writing and reading but used in the same manner of **magnetic disk**.

Optical Disk: READ

- A laser can be focused on to a very small area with a great accuracy.
- Laser is focused by a lens into cylindrical beam which passes through a beam-splitter (half silvered mirror).
- It lets half of the light passed and other half to be reflected.
- After the splitter, the beam is focused to a small area on the surface using lens.
- The amount of light reflected by the surface depends on its state.
- Then again the beam splitter reflects half of the light to sensor after a lens.
- Using half silvered mirror is inefficient because much light is wasted.

Optical Reading Mechanism

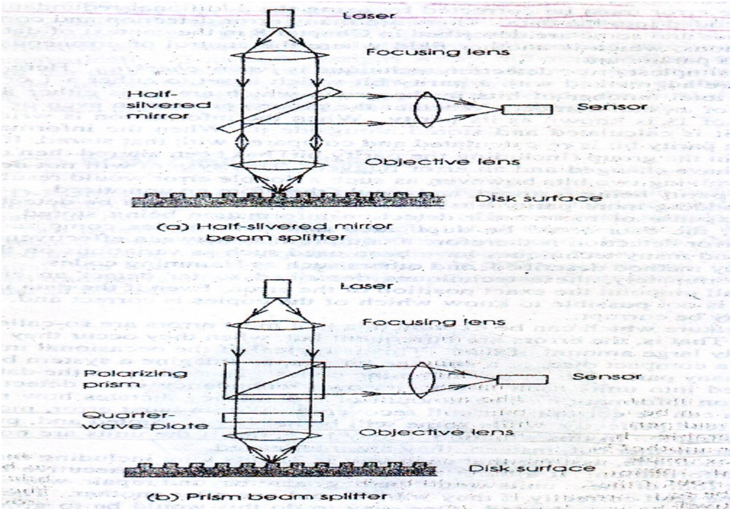
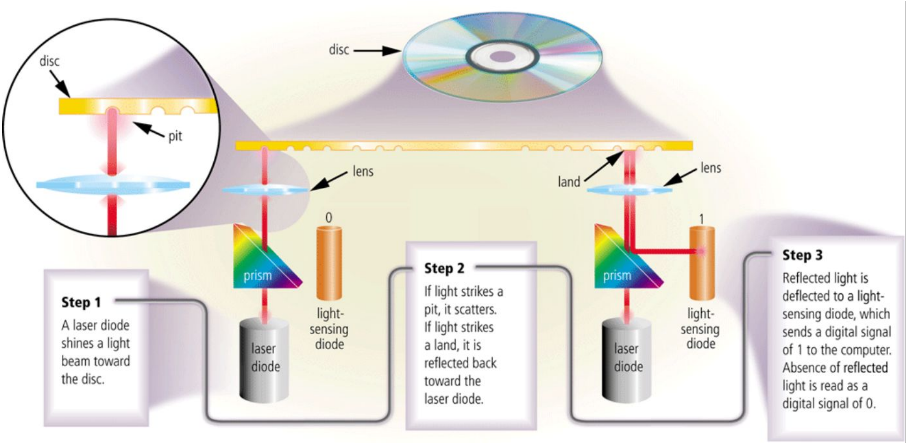


Figure 5.27 *Reading data with a laser beam*

Optical Storage



Optical Disk Error Detection technique: Parity Checking

- Groups of bits are augmented with a parity bit which is set to either ‘0’ or ‘1’ – so that total number of ‘1’ s are either an even number or an odd number
- When information is written – parity bit is calculated and stored alongside it
- When information is read – parity bit is re-calculated and compared with that stored
- **Problem:** cannot detect error involving **two bits**

Optical Disk Error Detection technique: Duplicated data

Data could be duplicated
Both copies compared when read

Compromise between effectiveness and space

Cannot pinpoint the exact position of the error

- Not possible to know which copy is correct
- Both may be corrupted

CD-ROM

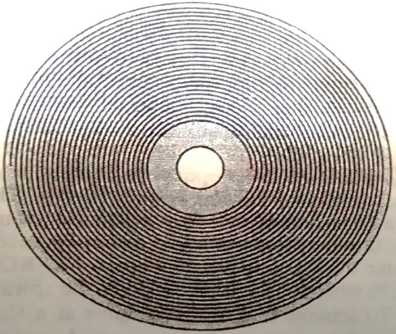


Figure 5.28 *Spiral track on a CD-ROM*

CD-ROM

- CD-ROM codes its data along tracks, like magnetic disk but a CD-ROM has **a single spiral track**.
- **Magnetic disk spin** at a Constant Angular Velocity (CAV).
- Surface moves under the head at a **faster rate on outer tracks** than on inner tracks.
- **CD-ROM uses Constant Linear Velocity**(CLV).
- Total length of the spiral track is scanned at the same rate.
- Disk spins at **slower rate when outer sections** of the track is being scanned.
- And, **speeds up proportionally as the laser beam moves towards center**.
- Time taken to adjust speeds when moving from one section to another, increases latency.
-

Magnetic Disk vs CD-ROM

CD-ROM:

- has single spiral track
- uses Constant Linear Velocity (CLV)
- can store more data
- more time needed to adjust the speed for each section.

Magnetic Disk:

- has many circular concentric tracks
- uses Constant Angular Velocity (CAV).
- can store less data.
- No extra time required to adjust the speed.

CD-ROM

- Inner recording surface is coated with aluminum-highly reflective
- A pit is only 0.1 μm.
- Aluminum surface is protected by a thin layer of lacquer- one type of liquid. (TOP of THE DISC)
- Lacquer side is only 30 μm thick.
- Reading side is covered by thicker opaque plastic material.
  CD-ROM is removable.
- The recording side is unlikely to be damaged through the reading side.
- So if it is damaged it will change recorded bumps

CD-ROM

WORM

- Contains a thin **sheet of metal sandwiched between two pieces of transparent material**.
- One form uses **tellurium between two pieces of glass**
- Metal surface starts life clear
- Then information is written by **burning small pits** into the surface with a high intensity laser beam.

WORM

- Difference with magnetic disk - self study
- Advantage and disadvantage - self study

Optical Positioning

- There are two aspects to the positioning of the laser beam when reading data optically
  - Focus control
  - Track following

Optical Positioning: Focus Control

- Focusing of laser beam on the surface depends on the distance between **objective lens and the surface**
- Position of lens is controlled by **voice coil mechanism** similar to adjust the moving head in magnetic disk.
- In order to provide the required depth of focus
    - Sensors must be arranged such that, one **sensor is actually four separate sensors** arranged in a diamond shape
    - A cylindrical lens is placed between **beam splitter** and **sensor array**

Optical Positioning: Focus Control

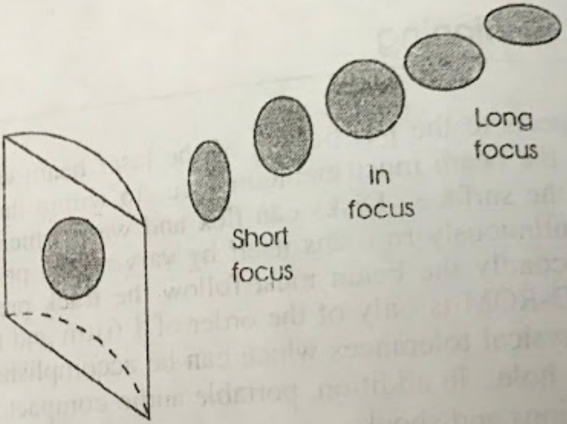


Figure 5.32 *Cylindrical lens to detect focus*

Optical Positioning: Focus Control

- The cylindrical lens:
  - Alters the shape of the cylindrical laser beam
  - When in focus- appears as a circle on the sensor array
  - When out of focus- appears as elliptical image on the sensor array
- The **output of the top and bottom sensors is summed and compared with the output sum of the left and right sensors**
- In focus, the sum is same.
- Out of focus, the sum differs.
- Then the lens is adjusted.

Optical Positioning: Focus Control

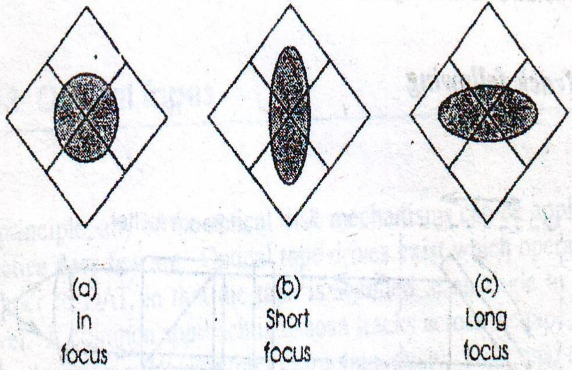


Figure 5.31 *Four-segment sensor to detect focus adjustments*

Optical Positioning: Track following

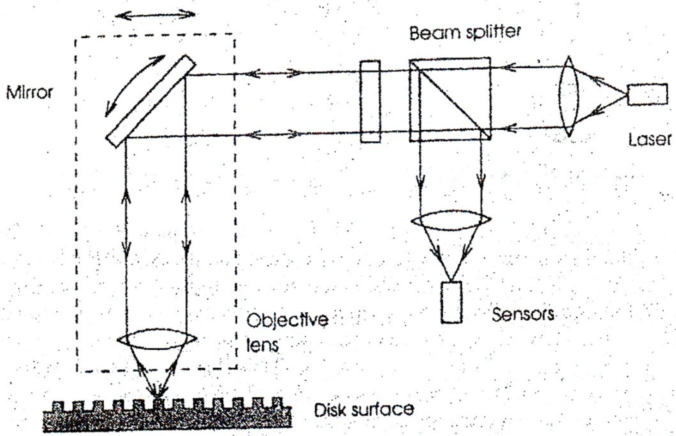


Figure 5.33 *Laser track following mechanism*

Optical Positioning: Track following

- For sensing track **two additional laser beams focused on either side of the main beam**
- These will hit the gap between the tracks and will always be totally reflected back.
- If the **track deviate slightly**, the one of the side beams will encounter track and be partially reflected
- The difference in strength can be used to control the carriage and mirror.

WORM - READ and WRITE

- Writing a WORM is same as the READ operation of a optical storage.
- The power of the laser beam is increased and its heat burns a pit on the surface.
- For READ operation, WORM follows same as the optical storage discussed earlier, only the power of the laser beam is decreased from that of the WRITE operation.

Performance Enhancers-Caches

Performance Enhancers-Caches

- Working principle:
  - Access to a store is not entirely random
  - At any point in time a small number of blocks are regularly accessed
- Types:
  - Read Cache
  - Write Cache
  - Read/Write Cache

Read Cache

- Most commonly used
- Retains a copy of each block when it is read from backing storage.
- When full, the oldest copies are replaced.
- Whenever a block is demanded, **cache is searched first**.
  - If it is present in cache, sent to the computer without searching the backing store
  - As the cache is faster, performance is enhanced.
- **Loses its content when power is removed**, so the first few disk accesses will be directly to the backing store
- Sophisticated algorithms anticipate future disk read
  - When directed to read a block, may also read a few blocks ahead into the cache assuming that a file has been placed in contiguous blocks.

Write Cache - Type

- Two types:
  - Write Through Cache
  - Write Deferred Cache

Write Through Cache

- Keeps a copy in the cache of each block written to backing store
- Only advantage to this occurs if the written block is likely to be read back-quite a common phenomenon

Write Deferred Cache

- Also keeps copy in the cache but delays write to the backing store for a short time
- Actually at first any block is written to cache instead of backing store-much faster
- Later, when the computer is idle, the block is written to the backing store
- If a block is written again, before it left the cache, then the new copy overwrites the one in the cache

Write Deferred Cache

Disadvantage:

- If power is lost, then the blocks will never reach to the backing store
- Consequences may vary, depending upon the block's contents
- A block of user's data file lost means- the file is slightly earlier version
- But, if the block belongs to **disk directory-consequences** become serious.

Solution:

- Some caches able o recognize whether **a block should be cached or not**
- Computer system may also employ **protected power supply with backup** which in case of power loss, takes over for at least time required to empty the cache

Read/Write Cache

- Combination of the previous two
- May employ two different fixed size regions of memory
- Or a single larger memory which serves for both on demand
    - At peak reading times, more memory is given to the read cache
    - At times of large writing activity, the write cache would be given a larger share

Performance Enhancers-Compression

Performance Enhancers-Compression

- Compression attempts to re-code data in such a manner that less space is required on the backing store
- Compression is performed by a program which acts as an interface between the computer system and backing store
- When data is retrieved from backing store it must be uncompressed to its original form

Compression Techniques

Two types of techniques:

- First one uses the fact, that in any **file certain bytes or sequences of bytes occur more frequently than others**. So, it can be performed by a program or built into the hardware of the disk or tape.
- Second uses the fact that, **because files are stored as a sequence of fixed sized units**, the last unit in each file contains unused space

Compression Techniques

1. The first technique exploits- the inherent redundancy that exists in most files
2. As an example, many text files consist largely of the **26 lower case alphabetic characters.**
3. Each character is stored as a byte- a byte can encode 256 characters
4. The file could be recorded by assigning 26 short codes to a-z and using the usual code for the rest
5. Only **5 bits** are needed for short codes instead of 8
6. Overhead penalty will be there to keep track of which are short code and which are long

Compression Techniques

1. Many algorithms are there that look for either common sequences of data
2. Repetition of characters, such as a sequence of the character ‘0’ can be detected
3. Then replaced with a count of length of the sequence and one copy of the character

Compression Techniques

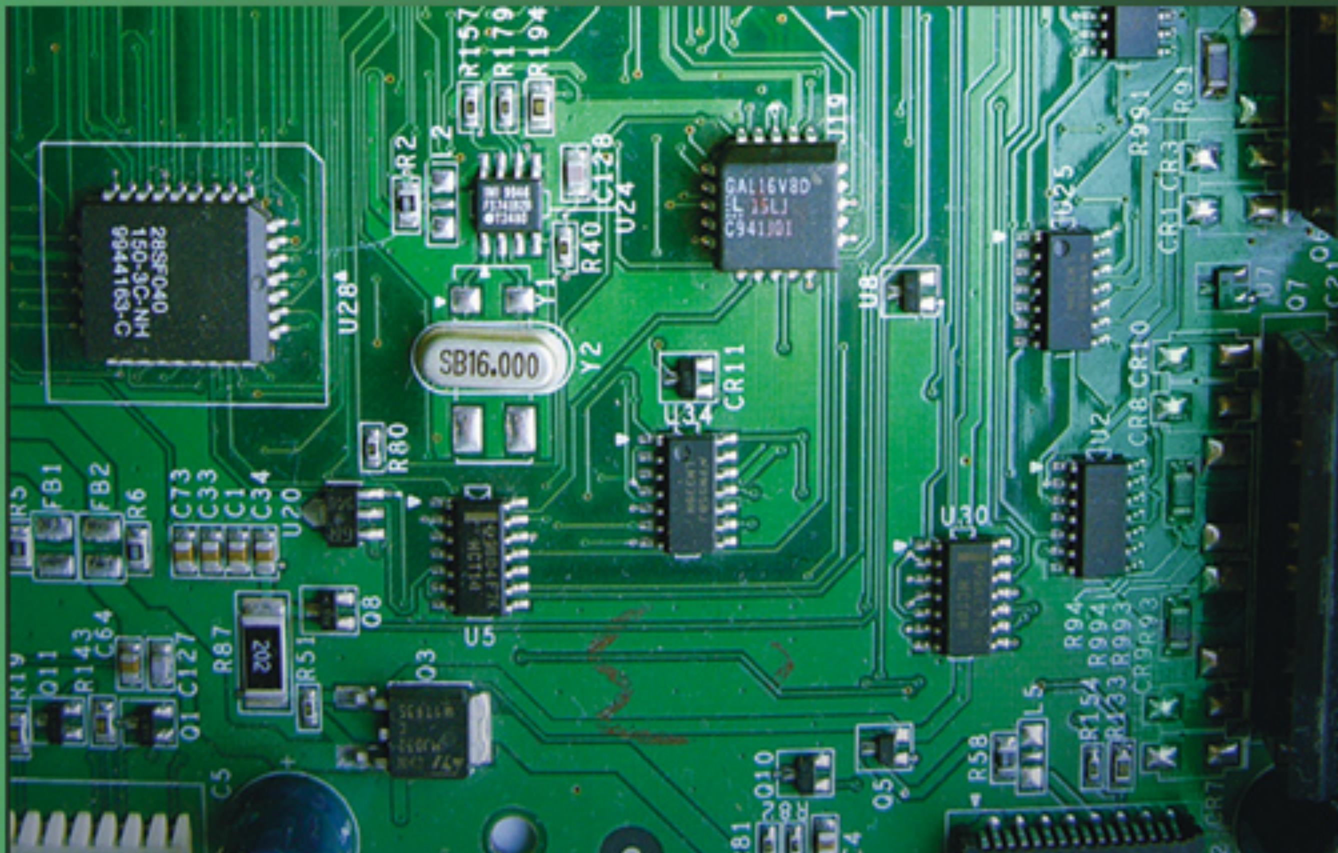
1. Second approach- With a **block size of 512 bytes**, on average 256 bytes will be unused for each file.
2. In practice, waste is much more
3. Because most operating system group blocks into clusters and use cluster as the minimum storage of unit
4. Cluster size is typically between 4 and 32
5. Reason behind this- consecutive blocks can be read/written in sequence more quickly
6. Compression programs
7. Store all files in one large file
8. Maintain a list of the original files' names, their starting position in the large file and their length

The End

The Intel Microprocessors

8086/8088, 80186/80188, 80286, 80386, 80486 Pentium, Pentium Pro Processor, Pentium II, Pentium 4, and Core2 with 64-bit Extensions

Architecture, Programming, and Interfacing



EIGHTH EDITION

Barry B. Brey

PEARSON

Chapter 13: Direct Memory Access and DMA-Controlled I/O

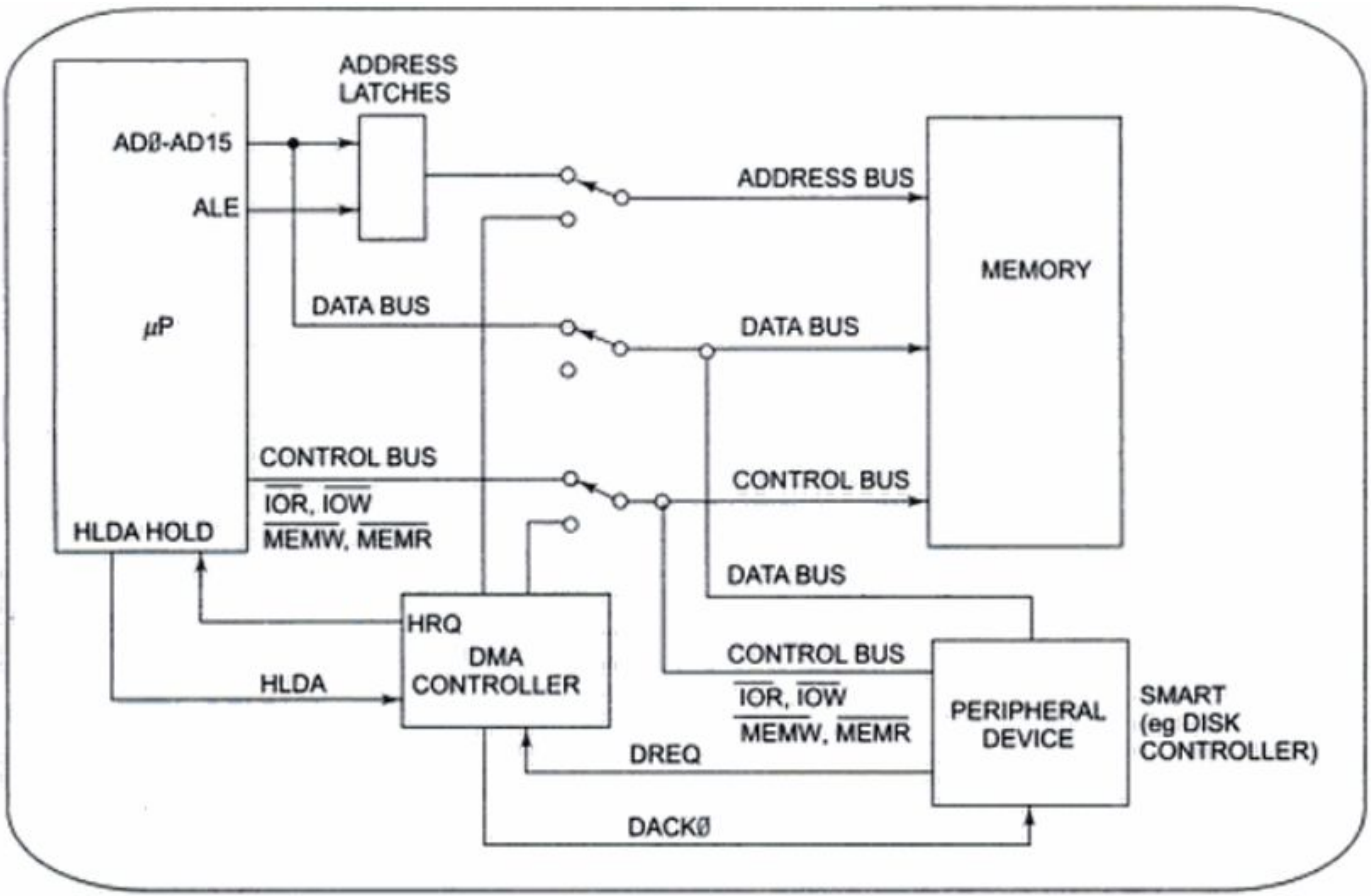
Introduction

- The DMA I/O technique provides direct access to the memory while the microprocessor is temporarily disabled.
- This allows data to be transferred between memory and I/O at a speed limited by the speed of the memory component and the DMA controller (33 to 150 M-bytes transfer rates).
- The common purposes of DMA are **DRAM refresh**, **video displays for refreshing the screen**, **disk memory system write and read** and also **high-speed memory-to-memory transfer**.

Introduction

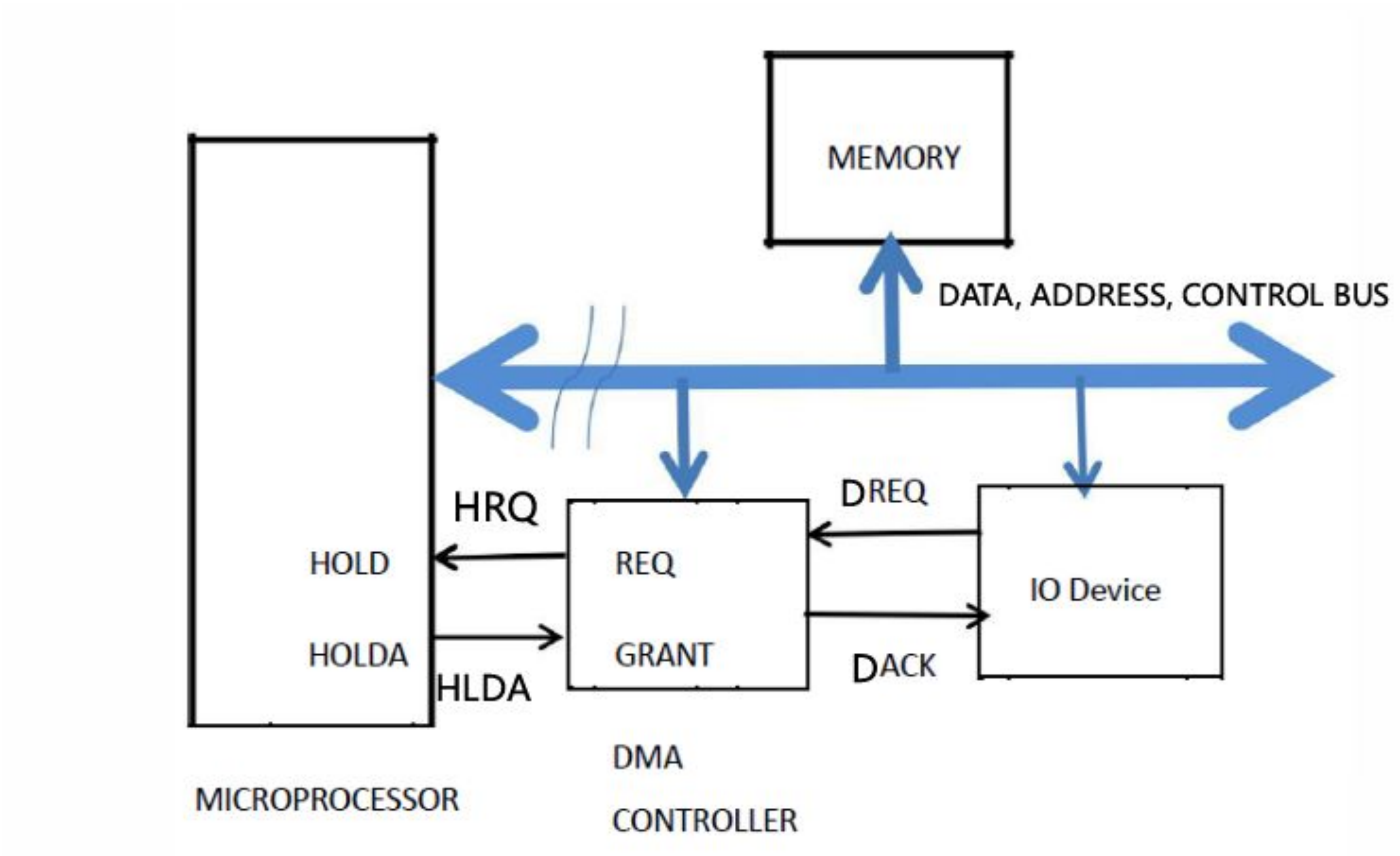
- DMA is designed by Intel to transfer data at the fastest rate.
- It allows the device to **transfer the data directly to/from memory without any interference of the CPU**.
- Using a **DMA controller**, the device requests the CPU to hold its data, address and control bus, so the device is free to transfer data directly to/from the memory.
- The DMA data transfer is initiated only after **receiving HLDA signal from the CPU**.
- **Programmed I/O and Interrupt driven** I/O are the alternative methods of data transferring against Direct Memory Access(DMA)

Figure: Block diagram showing how a DMA controller operates in a microcomputer system.



Ref.- Microprocessors & Interfacing by Douglas V. Hall, 2nd Edition, Pp. 348-351)

Figure: Block diagram showing how a DMA controller operates in a microcomputer system.



BASIC DMA OPERATION

- Two control signals are used to request and acknowledge a direct memory access (DMA) transfer in the **microprocessor-based system.**
  - the HOLD pin is an input used to request a DMA action
  - the HLDA pin is an output that acknowledges the DMA action

BASIC DMA OPERATION

- when the processor recognizes the hold, it stops executing software and enters hold cycles
- HOLD input has higher priority than **INTR or NMI**
- the only microprocessor pin that has a higher priority than a HOLD is the RESET pin

BASIC DMA OPERATION

- Initially, when any device has to send data between the device and the memory, the device has to **send DMA request (DRQ) to DMA controller**.
- The DMA controller **sends Hold request (HRQ) to the CPU** and waits for the CPU to assert the HLDA.
- Then the **microprocessor leaves the contro**l over the data bus, address bus, and control bus; and **acknowledges the HOLD request through HLDA signal.**
- Now the CPU is in HOLD state and the DMA controller has to manage the operations over buses between the CPU, memory, and I/O devices.

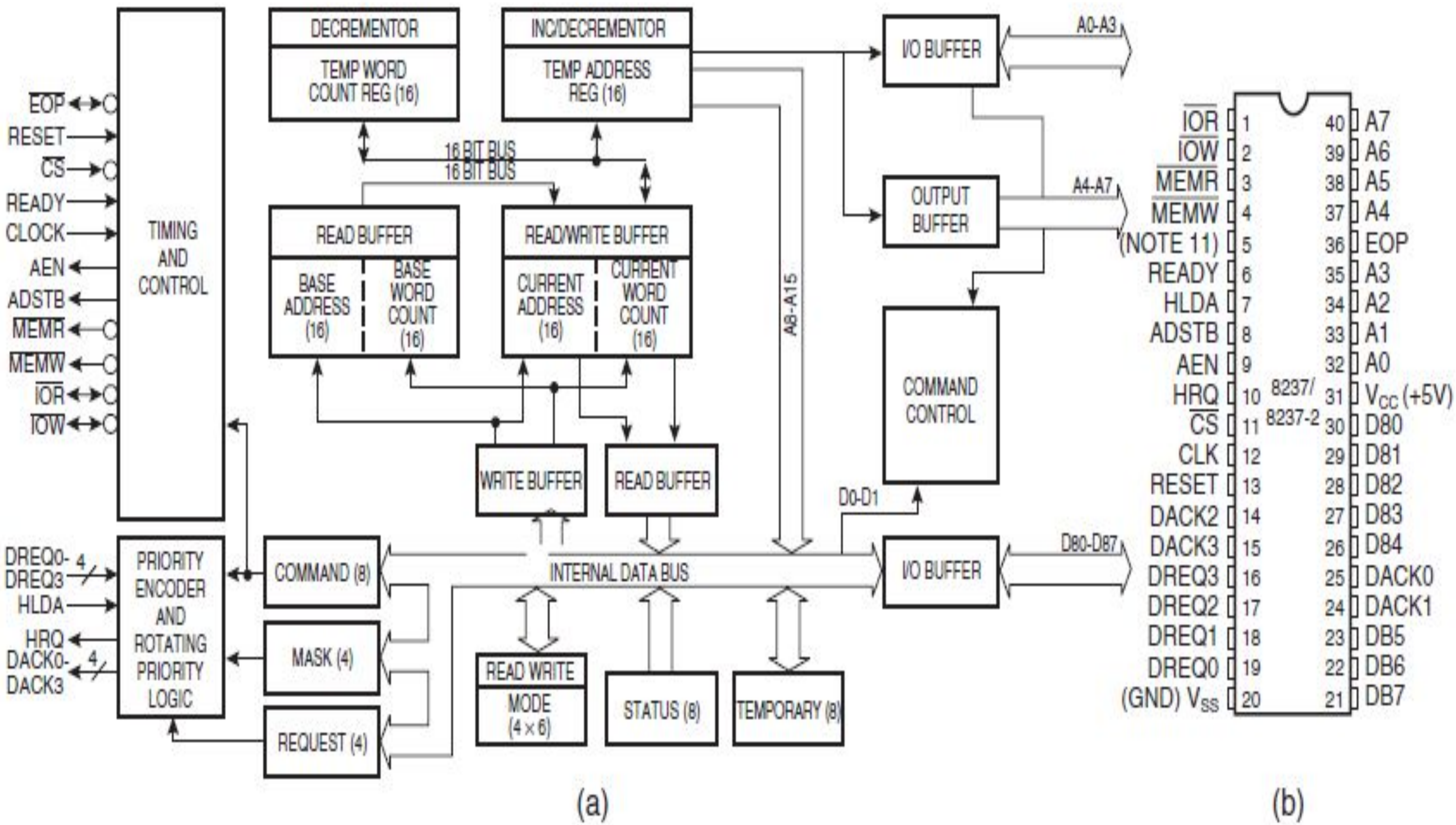
Basic DMA Definitions

- Direct memory accesses normally occur between an I/O device and memory without the use of the microprocessor.
  - a **DMA read** transfers data **from the memory to the I/O device**
  - **A DMA read** causes both the MRDC and IOWC signals to activate simultaneously
  - A **DMA write** transfers data **from an I/O device to memory**
  - **A DMA write causes the** MWTC and IORC signals to both activate
- Memory & I/O are controlled simultaneously.
- The data transfer speed is determined by the speed of the **memory device or a DMA controller** that often controls DMA transfers.

THE 8237 DMA CONTROLLER

- The 8237 supplies memory & I/O with **control signals and memory address** information during the DMA transfer.
  - actually a special-purpose microprocessor whose job is high-speed data transfer between memory and I/O
- It has **four channels,** which can be used over four I/O devices.
- Figure 13–3 shows the pin-out and block diagram of the 8237 programmable DMA controller.

Figure 13–3 The 8237A-5 programmable DMA controller. (a) Block diagram and (b) pin-out. (Courtesy of Intel Corporation.)



THE 8237 DMA CONTROLLER

- 8237 is not a discrete component in modern microprocessor-based systems.
  - it appears within many system controller chip sets
- 8237 is a four-channel device compatible with 8086/8088, adequate for small systems.
  - expandable to any number of DMA channel inputs
- 8237 is capable of DMA transfers at rates up to 1.6M bytes per second.
  - each channel is capable of addressing a full 64K-byte section of memory and transfer up to 64K bytes with a single programming

8237 Internal Registers

CAR

- The **current address register** holds a 16-bit memory address used for the DMA transfer.
  - each channel has its own current address register for this purpose
- When a byte of data is transferred during a DMA operation, CAR is either **incremented or decremented**.
  - depending on how it is programmed

8237 Internal Registers

CWCR

- The **current word count register** programs a channel for the number of bytes (up to 64K) transferred during a DMA action.
- The number loaded into this register is one less than the number of bytes transferred.
  - for example, if a 10 is loaded to CWCR, then 11 bytes are transferred during the DMA action

8237 Internal Registers

BA and BWC

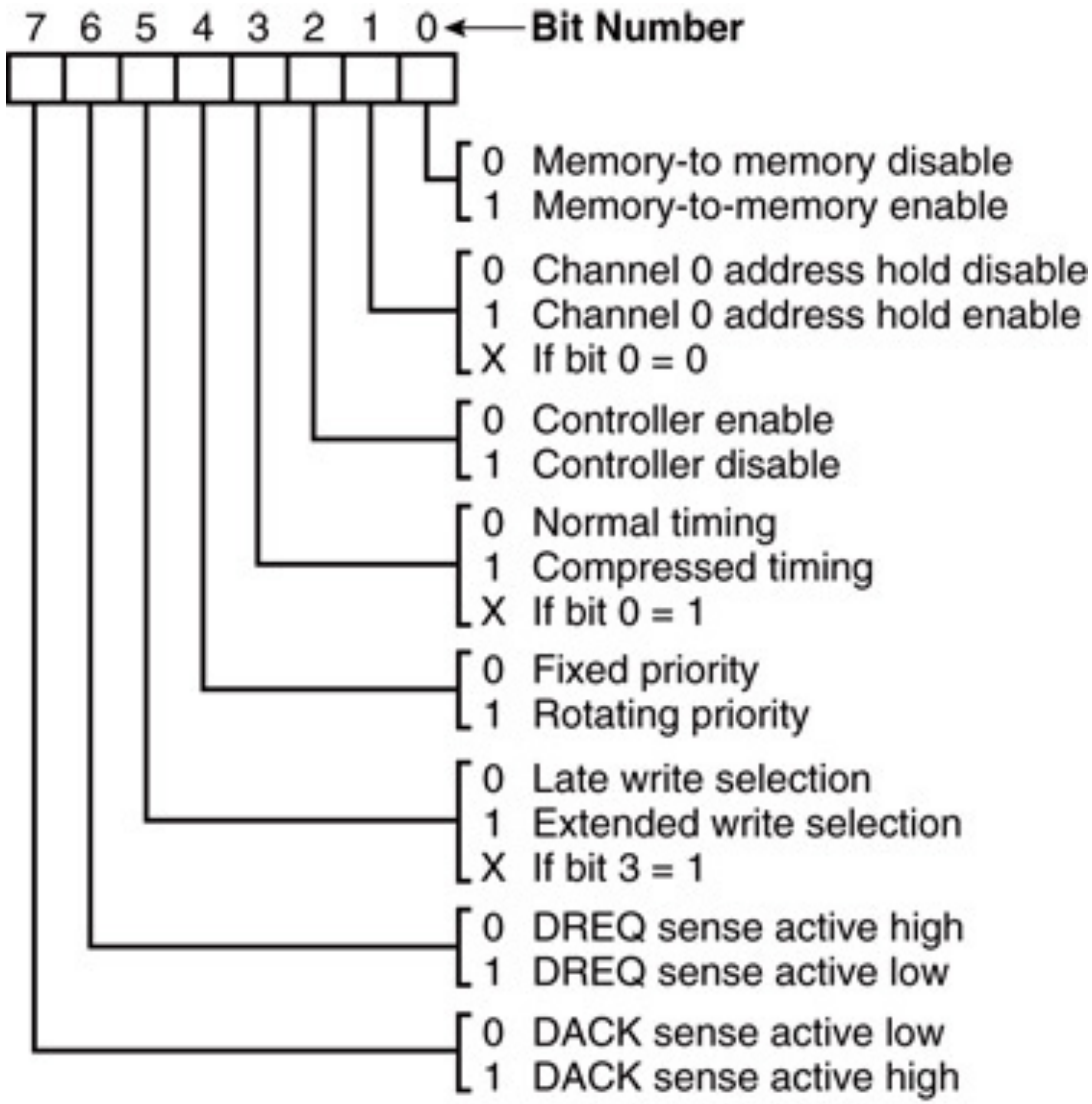
- The **base address** (BA) and **base word count** (BWC) registers are used when auto-initialization is selected for a channel.
- In auto-initialization mode, these registers are used to reload the CAR and CWCR after the DMA action is completed.
  - allows the same count and address to be used to transfer data from the same memory area

8237 Internal Registers

CR

- The **command register** programs the operation of the 8237 DMA controller.
- The register uses bit position 0 to select the memory-to-memory DMA transfer mode.
  - memory-to-memory DMA transfers use DMA **channel 0 to hold the source address**
  - DMA **channel 1 holds** the **destination address**

Figure: 8237A-5 command register. (Courtesy of Intel Corporation.)

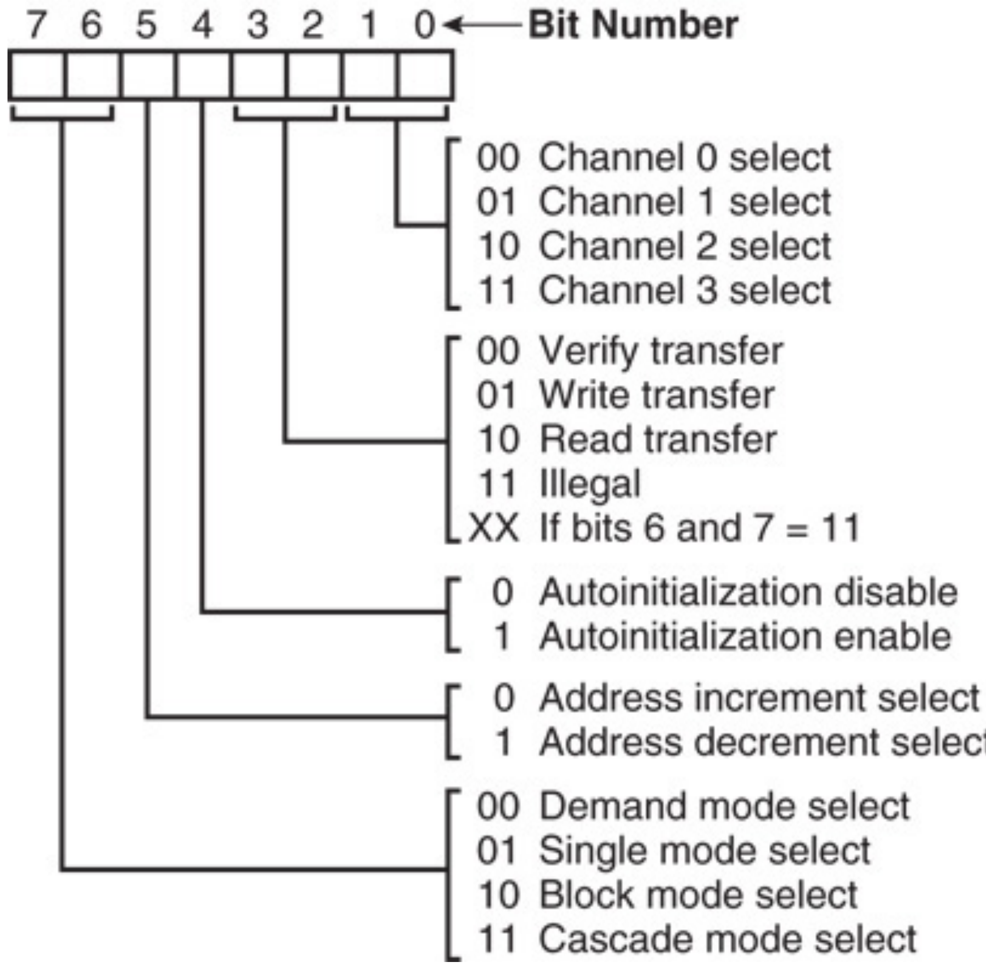


8237 Internal Registers

MR

- The **mode register** programs the mode of operation for a channel.
- Each channel has its **own mode register** as selected by bit positions 1 and 0.
  - remaining bits of the mode register select operation, **auto-initialization, increment/decrement, and mode for the channel**

Figure 13–5 8237A-5 mode register. (Courtesy of Intel Corporation.)

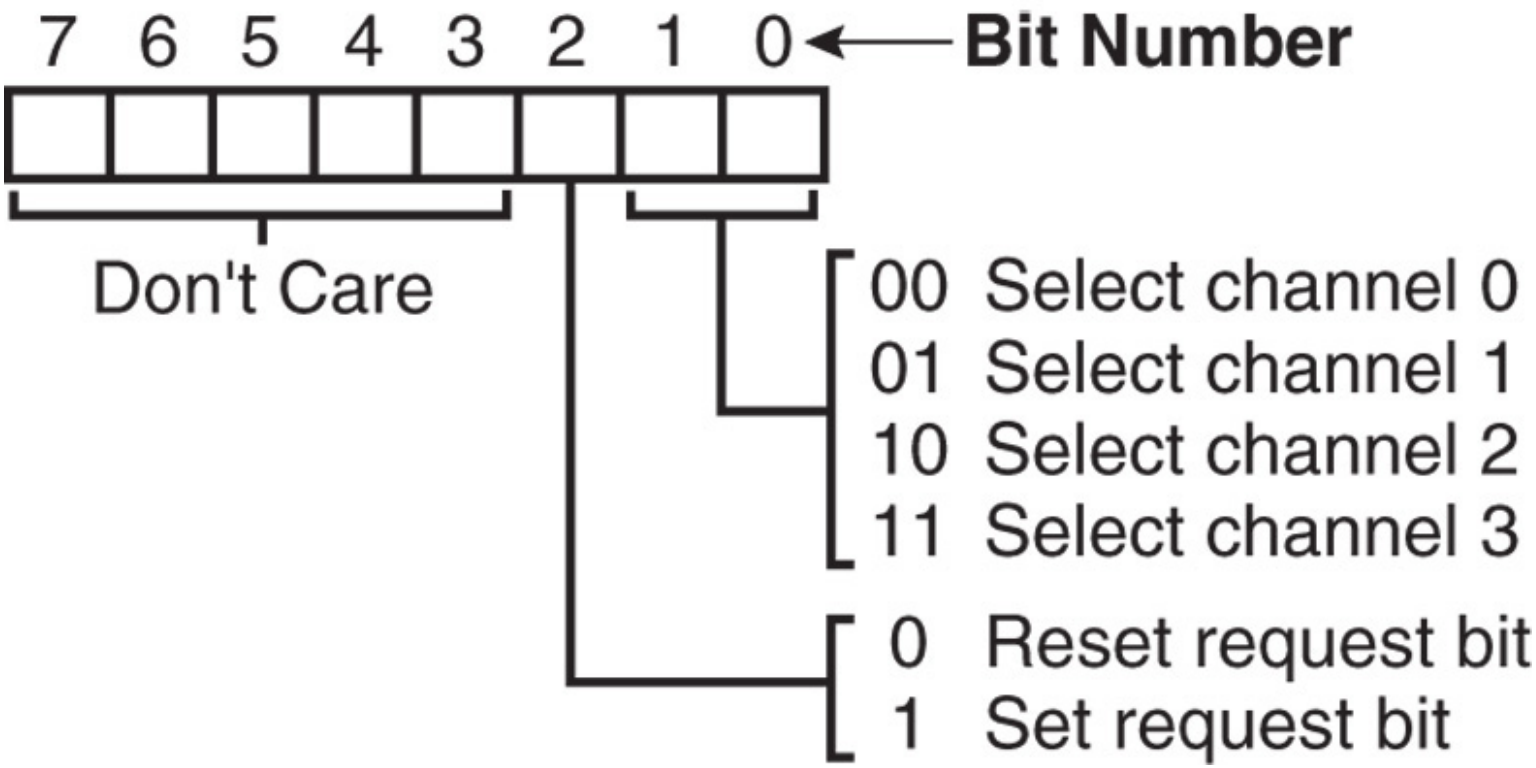


8237 Internal Registers

BR

- The **bus request register** is used to request a DMA transfer via software.
  - very useful in **memory-to-memory** transfers, where an external signal is not available to begin the DMA transfer

Figure 13–6 8237A-5 request register. (Courtesy of Intel Corporation.)

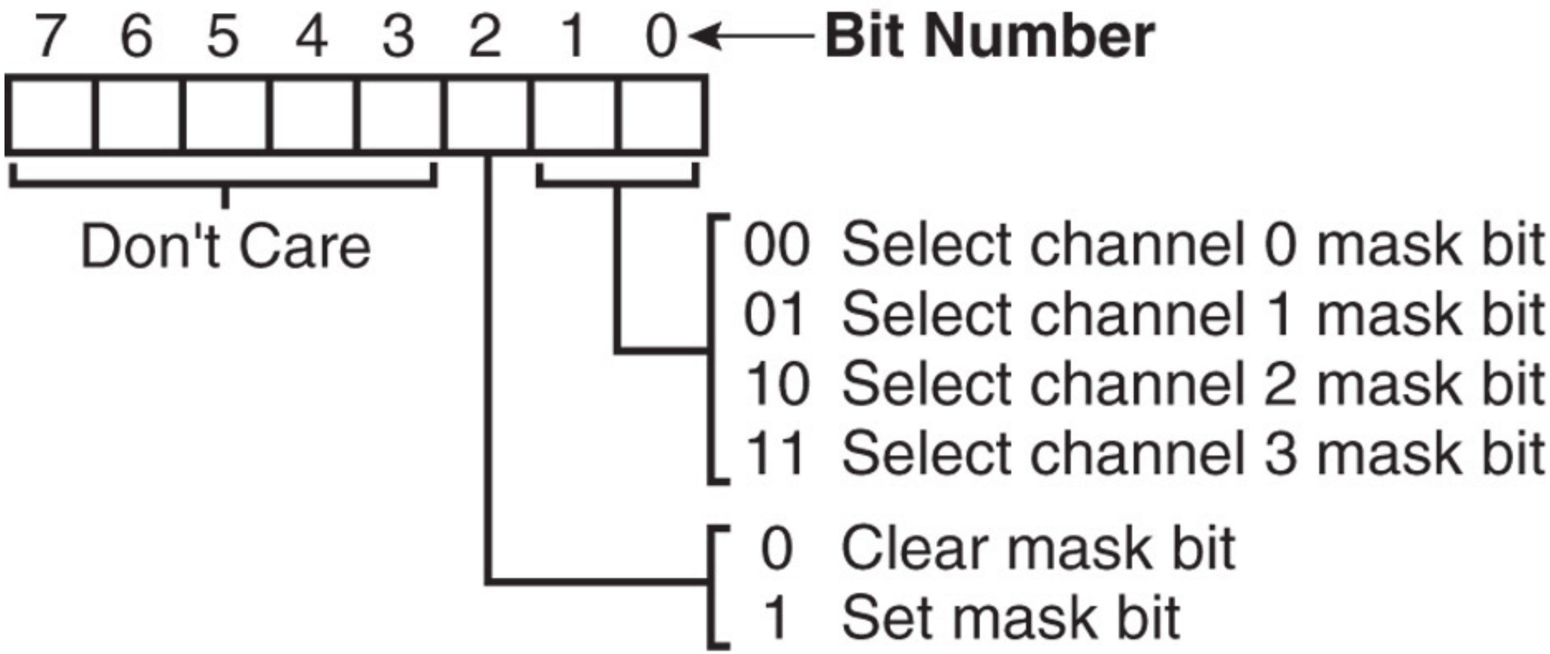


8237 Internal Registers

MRSR

- The **mask register set/reset** sets or clears the channel mask.
  - if the mask is set (1), the **channel is disabled**
  - the **RESET signal** sets **all channel masks to disable them**

Figure 13–7 8237A-5 mask register set/reset mode. (Courtesy of Intel Corporation.)

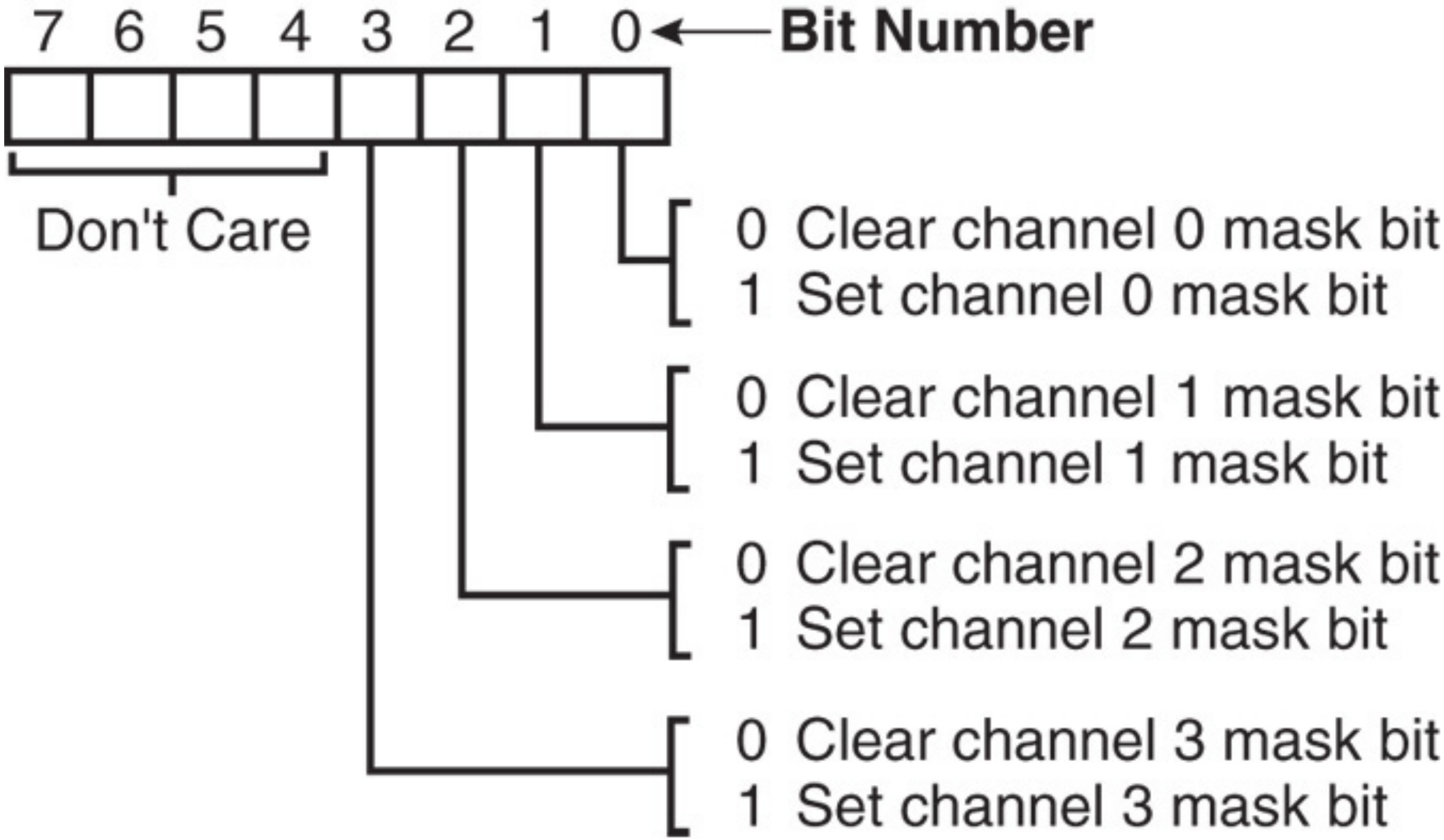


8237 Internal Registers

MSR

- The **mask register** clears or sets all of the masks with one command instead of individual channels, as with the MRSR.

Figure 13–8 8237A-5 mask register. (Courtesy of Intel Corporation.)

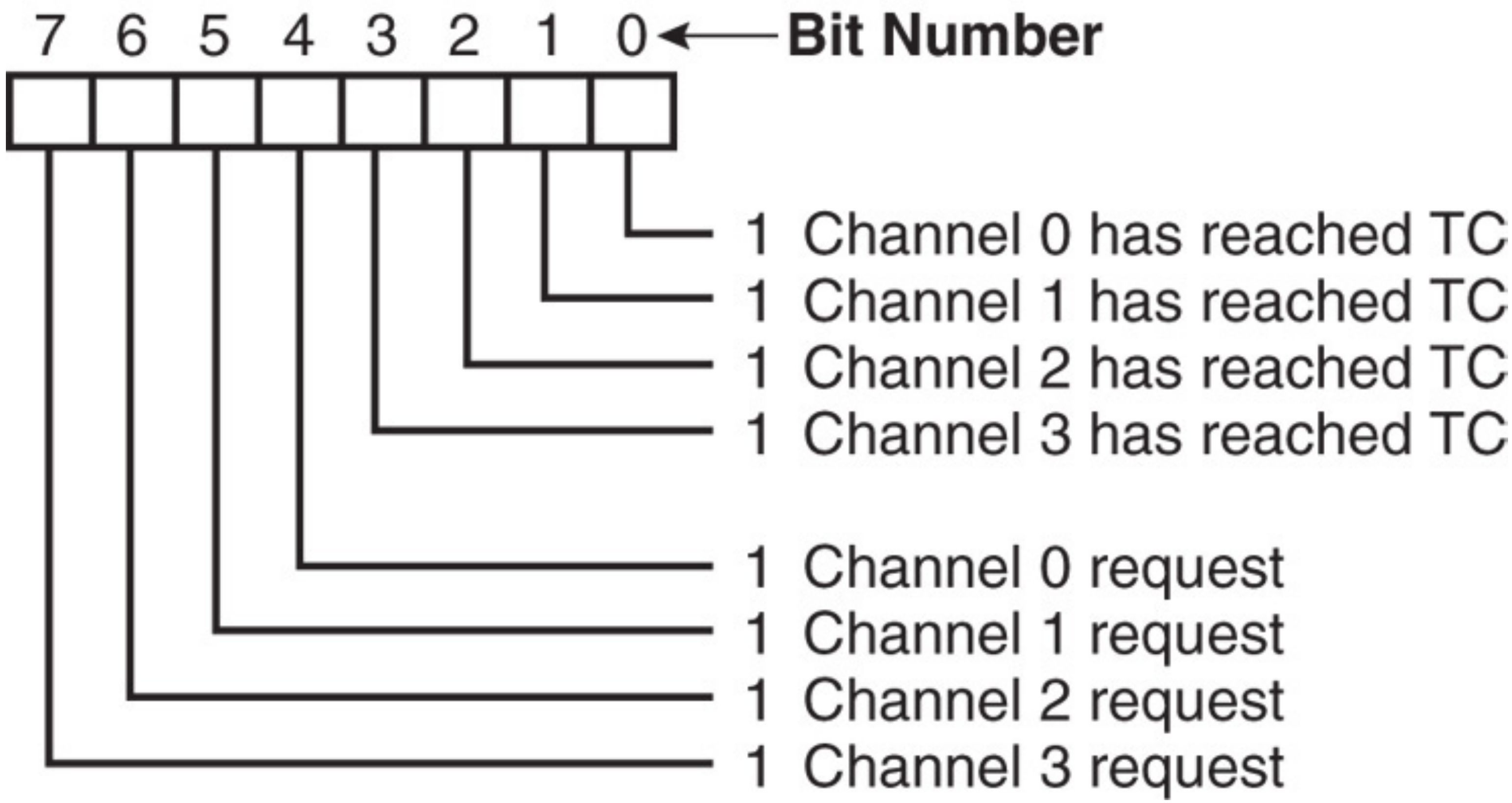


8237 Internal Registers

SR

- The **status register** shows status of **each DMA channel**.
- The **TC bits** indicate if the channel has reached **its terminal count** (transferred all its bytes).
- When the terminal count is reached, the DMA transfer is terminated for most modes of operation.
  - the **request bits indicate** whether the **DREQ input** for a given channel is **active**

Figure 13–9 8237A-5 status register. (Courtesy of Intel Corporation.)



Software Commands

- **Three software commands** are used to control the operation of the 8237.
- These commands **do not have a binary bit pattern**, as do various control registers within the 8237.
  - **a simple output to the correct port number** enables the software command
- Fig 13–10 shows **I/O port assignments** that access all registers and the software commands.

Software Commands

Master Clear (Reset) Command

- Clears the 8237, disables all DMA channels, and resets the internal state.
- Issued by writing to the appropriate command port.

Clear Mask Register Command

- Unmasks (enables) all DMA channels (allows them to request transfers).

All Mask Command

- Masks (disables) all DMA channels (blocks DMA requests).

Figure 13–10 8237A-5 command and control port assignments. (Courtesy of Intel Corporation.)

Signals						Operation
A3	A2	A1	A0	$\overline{IOR}$	$\overline{IOW}$	
1	0	0	0	0	1	Read Status Register
1	0	0	0	1	0	Write Command Register
1	0	0	1	0	1	Illegal
1	0	0	1	1	0	Write Request Register
1	0	1	0	0	1	Illegal
1	0	1	0	1	0	Write Single Mask Register Bit
1	0	1	1	0	1	Illegal
1	0	1	1	1	0	Write Mode Register
1	1	0	0	0	1	Illegal
1	1	0	0	1	0	Clear Byte Pointer Flip/Flop
1	1	0	1	0	1	Read Temporary Register
1	1	0	1	1	0	Master Clear
1	1	1	0	0	1	Illegal
1	1	1	0	1	0	Clear Mask Register
1	1	1	1	0	1	Illegal
1	1	1	1	1	0	Write All Mask Register Bits

Software Commands

Clear the first/last flip-flop

- Clears the first/last (F/L) flip-flop within 8237.
- The F/L flip-flop selects which byte (low or high order) is read/written in the current address and current count registers.
  - if F/L = 0, the low-order byte is selected
  - if F/L = 1, the high-order byte is selected
- Any read or write to the address or count register automatically toggles the F/L flip-flop.

Programming the Address and Count Registers

- Figure 13–11 shows I/O port locations for programming the count and address registers for each channel.
- The state of the **F/L flip-flop determines whether the LSB or MSB is programmed.**
  - if the state is unknown, count and address could be programmed incorrectly
- It is important to disable the DMA channel before the address and count are programmed.

Figure 13–11 8237A-5 DMA channel I/O port addresses. (Courtesy of Intel Corporation.)

Channel	Register	Operation	Signals $\overline{CS}$	$\overline{IOR}$	$\overline{IOW}$	A3	A2	A1	A0	Internal Flip-Flop	Data Bus DB0-DB7
0	Base and Current Address	Write	0	1	0	0	0	0	0	0	A0-A7
			0	1	0	0	0	0	0	1	A8-A15
	Current Address	Read	0	0	1	0	0	0	0	0	A0-A7
			0	0	1	0	0	0	0	1	A8-A15
	Base and Current Word Count	Write	0	1	0	0	0	0	1	0	W0-W7
			0	1	0	0	0	0	1	1	W8-W15
	Current Word Count	Read	0	0	1	0	0	0	1	0	W0-W7
			0	0	1	0	0	0	1	1	W8-W15
1	Base and Current Address	Write	0	1	0	0	0	1	0	0	A0-A7
			0	1	0	0	0	1	0	1	A8-A15
	Current Address	Read	0	0	1	0	0	1	0	0	A0-A7
			0	0	1	0	0	1	0	1	A8-A15
	Base and Current Word Count	Write	0	1	0	0	0	1	1	0	W0-W7
			0	1	0	0	0	1	1	1	W8-W15
	Current Word Count	Read	0	0	1	0	0	1	1	0	W0-W7
			0	0	1	0	0	1	1	1	W8-W15
2	Base and Current Address	Write	0	1	0	0	1	0	0	0	A0-A7
			0	1	0	0	1	0	0	1	A8-A15
	Current Address	Read	0	0	1	0	1	0	0	0	A0-A7
			0	0	1	0	1	0	0	1	A8-A15
	Base and Current Word Count	Write	0	1	0	0	1	0	1	0	W0-W7
			0	1	0	0	1	0	1	1	W8-W15
	Current Word Count	Read	0	0	1	0	1	0	1	0	W0-W7
			0	0	1	0	1	0	1	1	W8-W15
3	Base and Current Address	Write	0	1	0	0	1	1	0	0	A0-A7
			0	1	0	0	1	1	0	1	A8-A15
	Current Address	Read	0	0	1	0	1	1	0	0	A0-A7
			0	0	1	0	1	1	0	1	A8-A15
	Base and Current Word Count	Write	0	1	0	0	1	1	1	0	W0-W7
			0	1	0	0	1	1	1	1	W8-W15
	Current Word Count	Read	0	0	1	0	1	1	1	0	W0-W7
			0	0	1	0	1	1	1	1	W8-W15

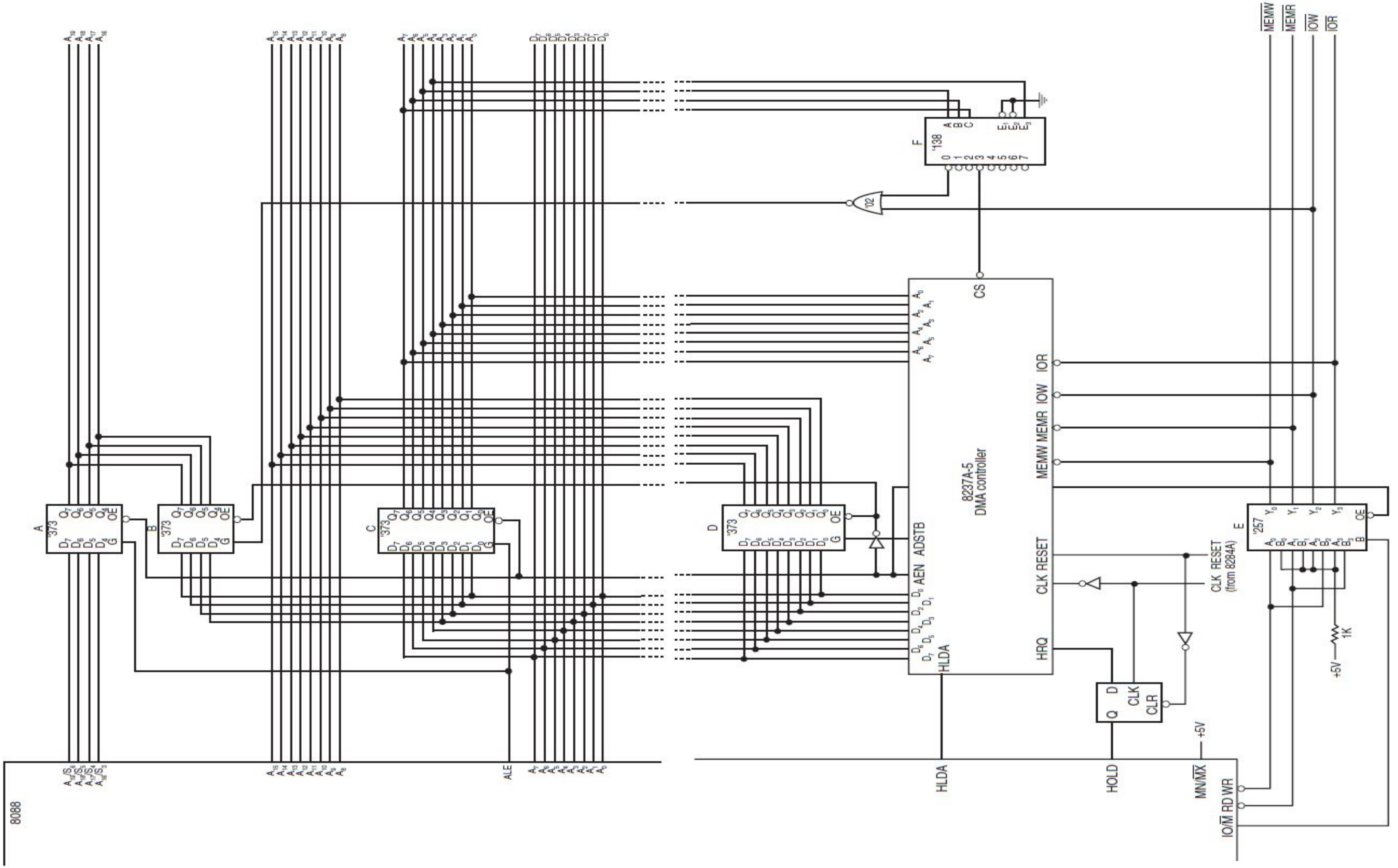
Programming the 8237

- **Four steps** are required to program the 8237:
  - (1) The F/L flip-flop is cleared using a **clear F/L command**
  - (2) the **channel is disabled**
  - (3) LSB & MSB of the address are programmed
  - (4) LSB & MSB of the count are programmed
- Once these **four operations** are performed, the channel is **programmed and ready to use.**
  - additional programming is required to select the mode of operation before the channel is enabled and started

The 8237 Connected to the 80X86

- The address enable (AEN) output of 8237 controls the output pins of the latches and outputs of the 74LS257 (E).
  - during normal operation (AEN=0), latches A & C and the multiplexer (E) provide address bus bits A_{19}–A_{16} and A_7–A_0
- See Figure 13-12.

Figure 13–12 Complete 8088 minimum mode DMA system.



The 8237 Connected to the 80X86

- The multiplexer provides the system control signals as long as the 80X86 is in control of the system.
  - during a DMA action (AEN=1), latches A & C are disabled along with the multiplexer (E)
  - latches D and B now provide address bits A_{19}–A_{16} and A_{15}–A_{8}
- Address bus bits A_{7}–A_{0} are provided directly by the 8237 and contain part of the DMA transfer address.
- The DMA controller provides control signals.

The 8237 Connected to the 80X86 (*Summary*)

Normal Operation	DMA Operation
AEN=0,	AEN=1,
A, C, E $\Rightarrow$ ON	B, D $\Rightarrow$ ON
A $\Rightarrow$ A_{19}-A_{16}	B $\Rightarrow$ A_{19}-A_{16}
Directly $\Rightarrow$ A_{15}-A_{8}	D $\Rightarrow$ A_{15}-A_{8}
C $\Rightarrow$ A_{7}-A_{0}	Directly $\Rightarrow$ A_{7}-A_{0}

Memory-to-Memory Transfer with the 8237

- Memory-to-memory transfer is much more powerful
- 8237 requires only 2.0 μs per byte, which is over twice as fast as a software data transfer.
- This is not true if an 80386, 80846, or Pentium is in use in the system.

Sample Memory-to-Memory DMA Transfer

- Suppose contents of memory locations 10000H–13FFFH are to be transferred to locations 14000H–17FFFH.
  - accomplished with the DMA controller
- Example 13–1 shows the software required to initialize the 8237 and **program latch B** in Figure 13–12 for this DMA transfer.

EXAMPLE 13–1

```
;A procedure that transfers a block of data using the 8237A
;DMA controller in Figure 13-12. This is a memory-to-memory
;transfer.

;Calling parameters:
;     SI = source address
;     DI = destination address
;     CX = count
;     ES = segment of source and destination

LATCHB EQU   10H
CLEARF EQU   7CH
CH0A   EQU   70H
CH1A   EQU   72H
CH1C   EQU   73H
MODE   EQU   7BH
CMMD   EQU   78H
MASKS  EQU   7FH
REQ    EQU   79H
STATUS EQU   78H

TRANS PROC   NEAR USES AX

    MOV    AX,ES                   ;program latch B
    MOV    AL,AH
    SHR    AL,4
    OUT    LATCHB,AL
    OUT    CLEARF,AL               ;clear F/L

    MOV    AX,ES                   ;program source address
    SHL    AX,4
    ADD    AX,SI
    OUT    CH0A,AL
    MOV    AL,AH
    OUT    CH0A,AL

    MOV    AX,ES                   ;program destination address
    SHL    AX,4
    ADD    AX,DI
    OUT    CH1A,AL
    MOV    AL,AH
    OUT    CH1A,AL
```

EXAMPLE 13–1 (Cont.)

```
        MOV   AX,CX               ;program count
        DEC   AX
        OUT   CH1C,AL
        MOV   AL,AH
        OUT   CH1C,AL

        MOV   AL,88H              ;program mode
        OUT   MODE,AL
        MOV   AL,85H
        OUT   MODE,AL

        MOV   AL,1                ;enable block transfer
        OUT   CMMD,AL

        MOV   AL,4                ;start DMA
        OUT   REQ,AL

        .REPEAT                   ;wait for completion
              IN  AL,STATUS
        .UNTIL AL & 1
        RET

TRANS ENDP
```

Thank You

PEARSON

Military Institute of Science and Technology
Department of Computer Science and Engineering

Sensors and Transducers

Course Name: Computer Interfacing
Course Code: CSE-405

Book Reference

- Microprocessor and Interfacing- Douglas V. Hall (Chapter 10)

Introduction

- A sensor is a physical device that detects or **senses changes in any physical quantity or energy- heat, light, pressure, humidity, motion**
- **A sensor's output is analog in nature**.
- **Active sensors** can **power themselves** (eg, piezoelectric crystal), whereas **passive sensors require external power**.
- A **transducer** is a physical device that **converts one form of energy into another**.
- **Electrical transducer** converts other forms of energy into an **electrical signal.**
- Transducer is made of **a sensor and a signal conditioning circuit**.
- **A transducer can generate analog as well as a digital output.**

Types of sensors

- Light Sensors:
  - ➔ Photodiode
  - ➔ LDR
  - ➔ Solar Cell
- Force Sensors:
  - ➔ Strain Gauge
  - ➔ Load Cell
- Pressure Sensors
  - ➔ Pressure Gauge
  - ➔ Manometer
  - ➔ LVDT
- Temperature Sensors:
  - ➔ RTD
  - ➔ Thermistor

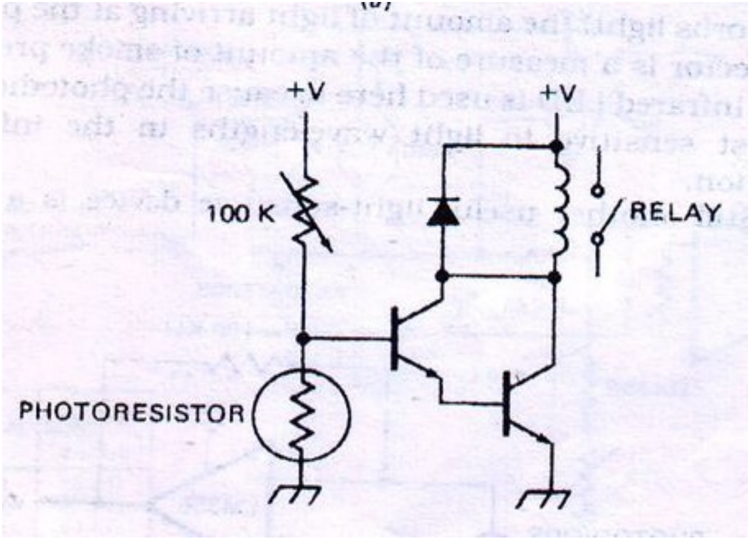
Light Sensors

Light Dependent Resistor (LDR)

- A glass window allows light to fall on a zigzag pattern of **cadmium sulfide** whose **resistance depends on the amount of light present**
- Resistance vary from **15 MΩ (in dark) to 15 kΩ** (in bright light)
- Advantages-
  - ➢ Inexpensive
  - ➢ Durable
  - ➢ Sensitive
- Disadvantages
  - ➢ Photoresistor have very **slow response time**
  - ➢ **Not stable with temperature**
  - ➢ May get damaged if temperature gets too high
- Application : light measurement need not be precise- e.g- streetlight

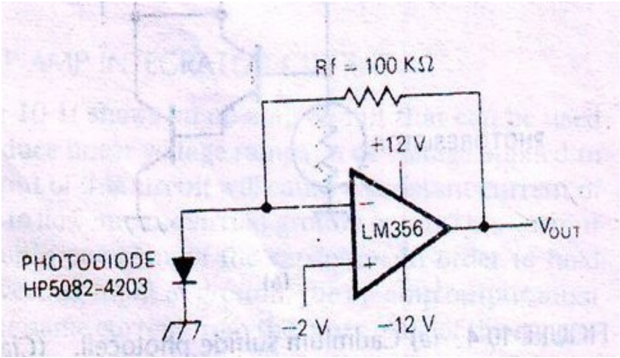
LDR—Working Principle in Streetlight

- Streetlight contain a photoresistor, a transistor drive and a mechanical relay
- In the dark, **resistance of photoresistor goes up**
- Increases **the voltage on the base of the transistor**
    - The transistor turns on
    - Turns on relay
    - Switches on the lamp



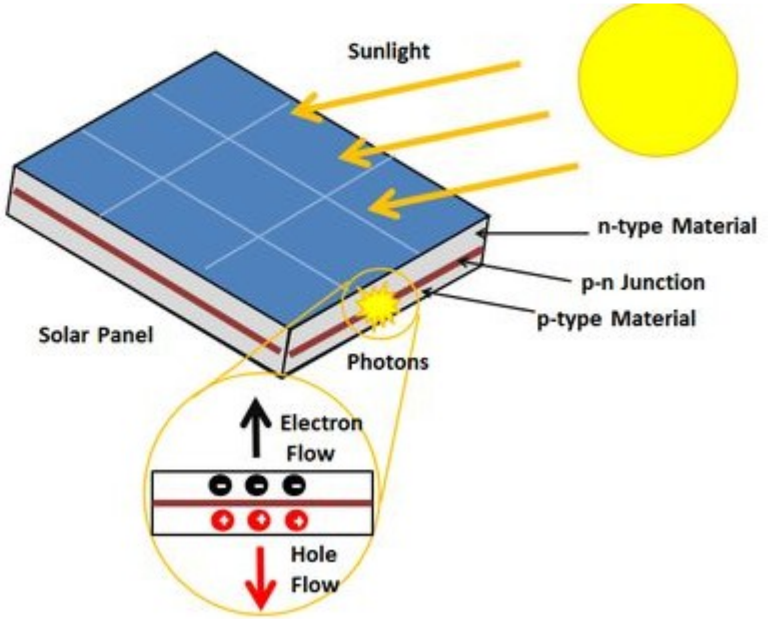
Photodiode

- Light is allowed to fall on the junction of a specially **constructed silicon diode**.
- **Reverse leakage current** increases linearly as the amount of **light increases**
- **Leakage current is converted into proportional voltage**
- This can be used for **smoke detection.**
- Put **gallium arsenide infrared LED** on one side and **photodetector circuit** on the other side or the smokestack
- Smoke absorbs light- **the amount of light arriving the photodetector is the measure of smoke**.
- **Infrared LED is used**—photodiode is **more sensitive** to light wavelength in the infrared region.



Solar Cell

- **Very heavily doped silicon PN Junctions**
- Light shining on the solar cell causes a **reverse current to flow**.
- **Solar cells can be connected in series** and parallel array to produce a solar power supply



Temperature Sensors

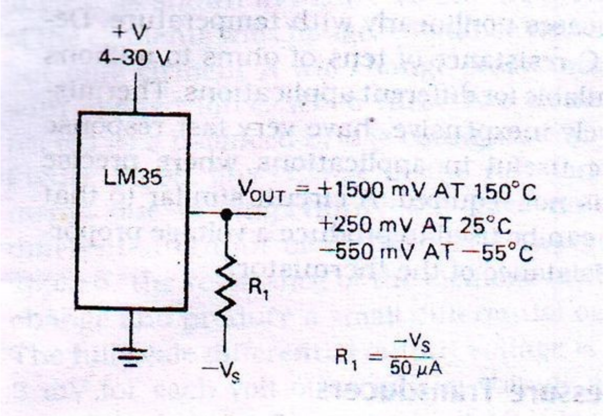
Semiconductor Temperature Sensors

Two types of semiconductor temperature sensors

- **Temperature-Sensitive Voltage Sources**
- **Temperature-Sensitive Current Sources**

Temperature-Sensitive Voltage Sources

- Example: **LM35**
- **Voltage output increases by 10 mV** for **each degree celsius in temperature increases**
- Output adjust to **0 V for 0° C**.
- Sensor gives output for temperature range of **-55° to +150°C**



Thermocouples

- A **thermocouple** is a **junction** between **two different metals** that **produces a voltage** related to a **temperature difference**.
- **When two different metals are put in contact , a small voltage is produced between them—the Seebeck effect**
- Developed voltage depends on
  - **Type of metals used**
  - **Temperature**
- The developed **voltage increases between 7 and 75 µV** for **each degree celsius increase** in temperature- depending on the metal
- A thermocouple junction made of **iron and constantan** is called a type ‘J Thermocouple’
  - Temperature range : -184 to +760 degree Celsius
- Thermocouple junction can be made of **platinum and alloy of platinum**-rhodium
  - Temperature range : 0 to 1600 degree Celsius

Thermocouples - 3 main problems

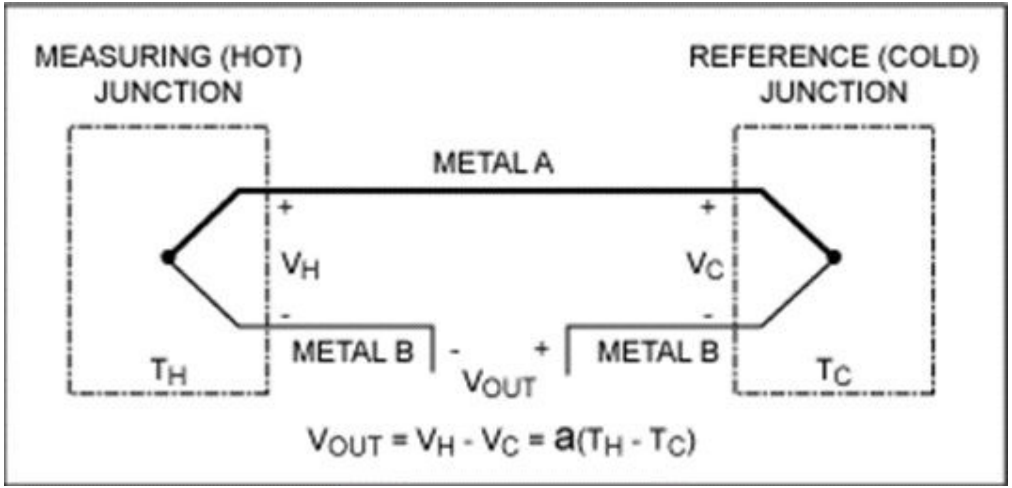
1. **Output is very small and need to be amplified** a great deal
2. **A reference junction of same metal must be connected in series** with the junction used to make the **measurement**

- Because thermocouple reading is meaningless without comparing it to another, more accurate temperature sensor, **usually a reference**
- The difference between the voltages across the two thermocouples goes as input to the amplifier
- It's impossible to tell which thermocouple is causing a change in voltage if both **temperatures are changing**.
    - One end of the thermocouple—the hot junction—is thermally attached to the object to be measured
    - The other end of the thermocouple—the cold junction—is placed into an ice bath to hold it at a constant temperature

Thermocouples - 3 main problems

3. Output voltages do not change linearly with temperature

- Analog circuitry is needed which changes the gain of an amplifier according to the value of the signa



Thermocouple math problem

Suppose the measured output voltage between measuring junction and reference junction using thermocouple is 3.41mV. Reference junction temperature is 22 C. What will be the temperature of the measuring junction? Also find the seebeck

Thermoelectric Voltage in mV

°C	0	1	2	3	4	5	6	7	8	9	10
0	0.000	0.039	0.079	0.119	0.158	0.198	0.238	0.277	0.317	0.357	0.397
10	0.397	0.437	0.477	0.517	0.557	0.597	0.637	0.677	0.718	0.758	0.798
20	0.798	0.838	0.879	0.919	0.960	1.000	1.041	1.081	1.122	1.163	1.203
30	1.203	1.244	1.285	1.326	1.366	1.407	1.448	1.489	1.530	1.571	1.612
40	1.612	1.653	1.694	1.735	1.776	1.817	1.858	1.899	1.941	1.982	2.023
50	2.023	2.064	2.106	2.147	2.188	2.230	2.271	2.312	2.354	2.395	2.436
60	2.436	2.478	2.519	2.561	2.602	2.644	2.685	2.727	2.768	2.810	2.851
70	2.851	2.893	2.934	2.976	3.017	3.059	3.100	3.142	3.184	3.225	3.267
80	3.267	3.308	3.350	3.391	3.433	3.474	3.516	3.557	3.599	3.640	3.682
90	3.682	3.723	3.765	3.806	3.848	3.889	3.931	3.972	4.013	4.055	4.096
100	4.096	4.138	4.179	4.220	4.262	4.303	4.344	4.385	4.427	4.468	4.509
110	4.509	4.550	4.591	4.633	4.674	4.715	4.756	4.797	4.838	4.879	4.920
120	4.920	4.961	5.002	5.043	5.084	5.124	5.165	5.206	5.247	5.288	5.328
130	5.328	5.369	5.410	5.450	5.491	5.532	5.572	5.613	5.653	5.694	5.735
140	5.735	5.775	5.815	5.856	5.896	5.937	5.977	6.017	6.058	6.098	6.138
150	6.138	6.179	6.219	6.259	6.299	6.339	6.380	6.420	6.460	6.500	6.540
160	6.540	6.580	6.620	6.660	6.701	6.741	6.781	6.821	6.861	6.901	6.941
170	6.941	6.981	7.021	7.060	7.100	7.140	7.180	7.220	7.260	7.300	7.340
180	7.340	7.380	7.420	7.460	7.500	7.540	7.579	7.619	7.659	7.699	7.739
190	7.739	7.779	7.819	7.859	7.899	7.939	7.979	8.019	8.059	8.099	8.138
200	8.138	8.178	8.218	8.258	8.298	8.338	8.378	8.418	8.458	8.499	8.539
210	8.539	8.579	8.619	8.659	8.699	8.739	8.779	8.819	8.860	8.900	8.940
220	8.940	8.980	9.020	9.061	9.101	9.141	9.181	9.222	9.262	9.302	9.343
230	9.343	9.383	9.423	9.464	9.504	9.545	9.585	9.626	9.666	9.707	9.747
240	9.747	9.788	9.828	9.869	9.909	9.950	9.991	10.031	10.072	10.113	10.153
250	10.153	10.194	10.235	10.276	10.316	10.357	10.398	10.439	10.480	10.520	10.561
260	10.561	10.602	10.643	10.684	10.725	10.766	10.807	10.848	10.889	10.930	10.971
270	10.971	11.012	11.053	11.094	11.135	11.176	11.217	11.259	11.300	11.341	11.382
280	11.382	11.423	11.465	11.506	11.547	11.588	11.630	11.671	11.712	11.753	11.795
290	11.795	11.836	11.877	11.919	11.960	12.001	12.043	12.084	12.126	12.167	12.209
300	12.209	12.250	12.291	12.333	12.374	12.416	12.457	12.499	12.540	12.582	12.624
310	12.624	12.665	12.707	12.748	12.790	12.831	12.873	12.915	12.956	12.998	13.040
320	13.040	13.081	13.123	13.165	13.206	13.248	13.290	13.331	13.373	13.415	13.457
330	13.457	13.498	13.540	13.582	13.624	13.665	13.707	13.749	13.791	13.833	13.874
340	13.874	13.916	13.958	14.000	14.042	14.084	14.126	14.167	14.209	14.251	14.293
350	14.293	14.335	14.377	14.419	14.461	14.503	14.545	14.587	14.629	14.671	14.713
360	14.713	14.755	14.797	14.839	14.881	14.923	14.965	15.007	15.049	15.091	15.133
370	15.133	15.175	15.217	15.259	15.301	15.343	15.385	15.427	15.469	15.511	15.554
380	15.554	15.596	15.638	15.680	15.722	15.764	15.806	15.849	15.891	15.933	15.975
390	15.975	16.017	16.059	16.102	16.144	16.186	16.228	16.270	16.313	16.355	16.397
400	16.397	16.439	16.482	16.524	16.566	16.608	16.651	16.693	16.735	16.778	16.820
410	16.820	16.862	16.904	16.947	16.989	17.031	17.074	17.116	17.158	17.201	17.243
420	17.243	17.285	17.328	17.370	17.413	17.455	17.497	17.540	17.582	17.624	17.667
430	17.667	17.709	17.752	17.794	17.837	17.879	17.921	17.964	18.006	18.049	18.091
440	18.091	18.134	18.176	18.218	18.261	18.303	18.346	18.388	18.431	18.473	18.516

Thermocouple math problem

Suppose the measured output voltage between measuring junction and reference junction using thermocouple is 3.41mV. Reference junction temperature is 22 C. What will be the temperature of the measuring junction? Also find the seebeck

We know, Vout = Vh - Vc

Vout = a (Th - Tc)

Vout=3.41mV

Tc=22 degree C

Th=?

a=?

From table,

When T=22 °C, V=0.879 mV

So, Vc=0.879 mV

Thus, Vout=Vh-Vc

3.41=Vh-0.879

Vh=4.289mV

So, From table, when V=4.289 approx., T= 104 or 105 C So, Th= 105 (4.289 close to 4.303 thann 4.262)

So, a=3.41/(105-22) = 0.0411

Resistance Temperature Detector (RTD)

- **Resistance Temperature Detector** is another commonly used Temperature Sensor
- It is actually a **resistor** which **change value with a change in temperature**
- It consists of a **wire or a thin film of platinum or a nickel wire**
- Response of RTDs is **non-linear** but they have excellent
    - **Stability** (This is the sensor's ability to maintain a consistent output when a constant input is applied)
    - **Repeatability** (is the sensor's ability to give the same output or reading under repeated identical conditions.)
- It is useful for measures in the range of **-250 to +850° C**
- **The resistance of an RTD increases with an increase in temperature**
- Change in the **resistance** is **converted into a proportional voltage**
- **Used where precise temperature measurement is needed**

Thermistor

- Thermistors consists of **semiconductor materials**
- **Resistance decreases nonlinearly with temperature**
- Relatively inexpensive
- Very fast response time
- **Applicable where precise measurement is not required**

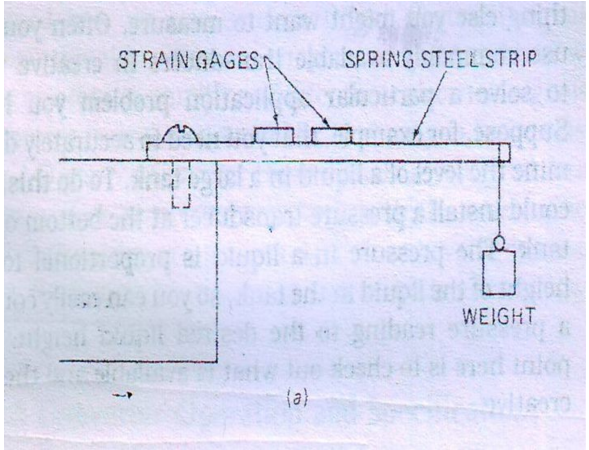
Force and Pressure Sensors

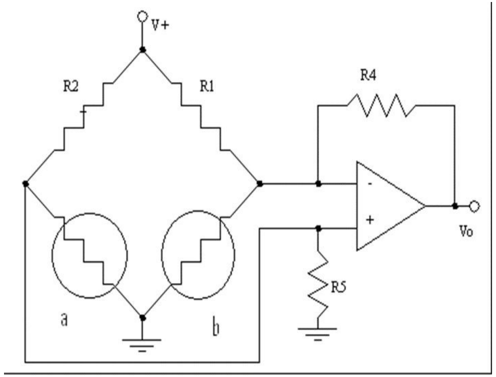
Strain Gages

- It is a small **resistor** whose **value changes when its length is changed**
- It is made of **thin wire, thin foil or semiconductor material**
- **One end of a piece of spring steel is attached to a fixed surface**
- A **strain gage is glued on the top of the flexible bar**
- The **force or weight to be measured is applied to the unattached end** of the bar
- As the applied force **bend the bar**, the **strain gage is stretched, increasing its resistance**
- Since the **amount that the bar is bent is directly proportional to the applied force**
- The **change** in the **resistance will be proportional to the applied force**

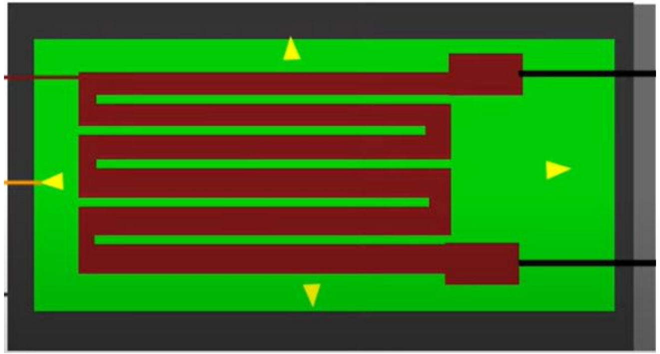
Strain Gages

- If current is passed through the strain gage, **then the change in voltage across the strain gage will be proportional to the applied force**
- Unfortunately the **resistance of strain gage** element also **changes with the temperature**
- **Two Strain gage elements are mounted at right angles**
- **If the elements are connected in a balanced** –bridge configuration, **any change in the resistance due to temperature will have no effect on the** differential output **of the bridge**

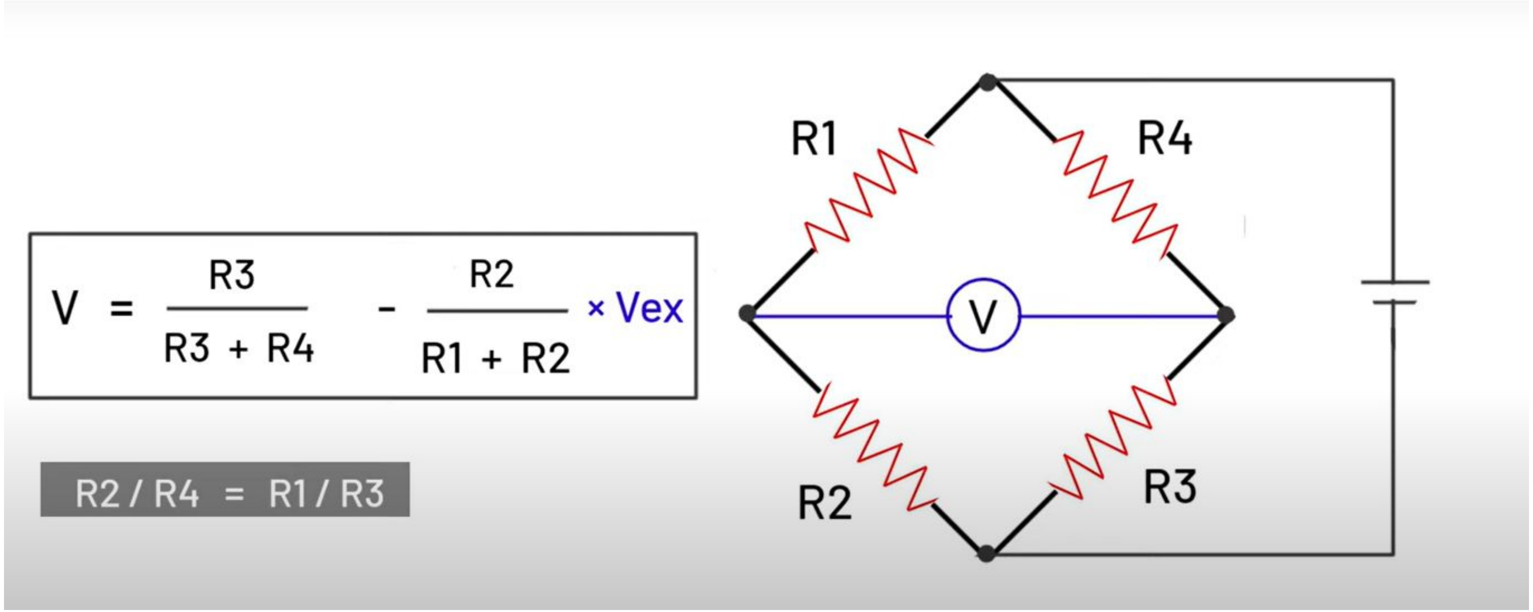




Strain Gages



Strain Gages



Strain Gages

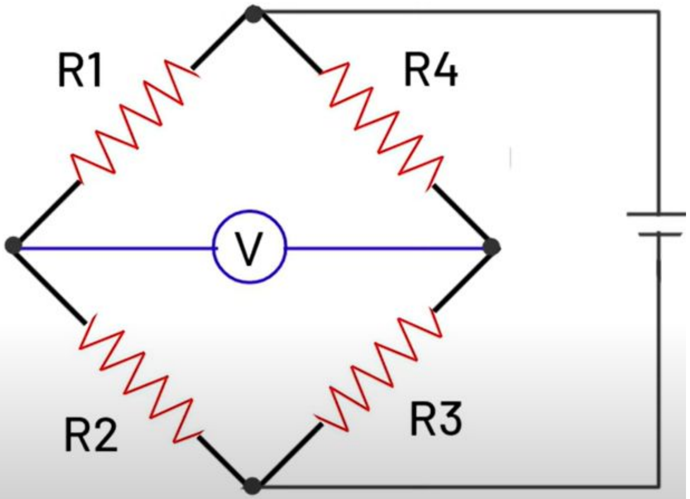
- gauge factor

$$\boxed{K = \frac{\Delta R / R}{\Delta L / L}}$$

K - gauge factor

DR - change in initial resistance

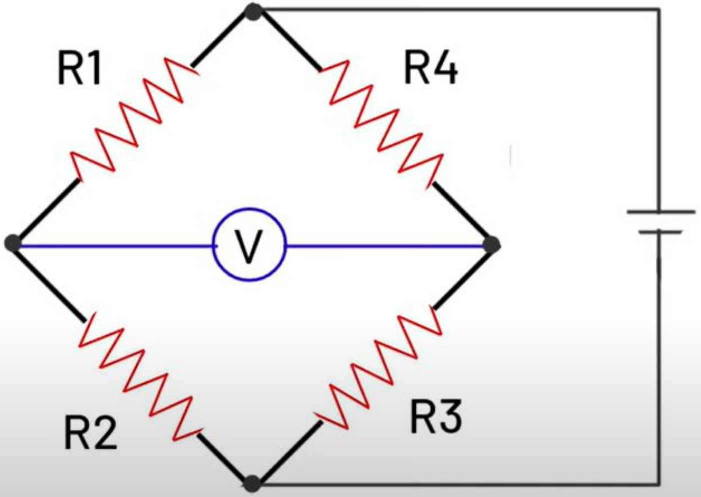
R - Initial resistance without strain



Strain Gages

As we know ,

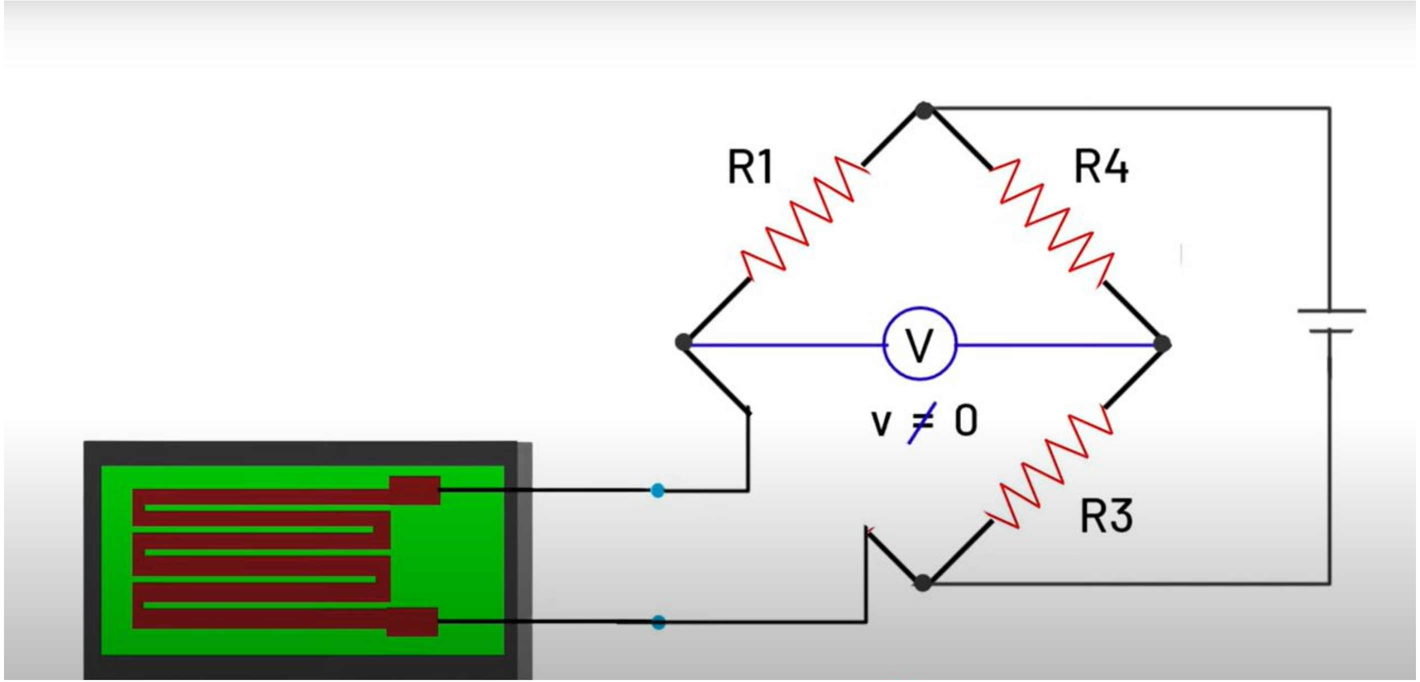
$$R = \frac{\rho L}{A}$$



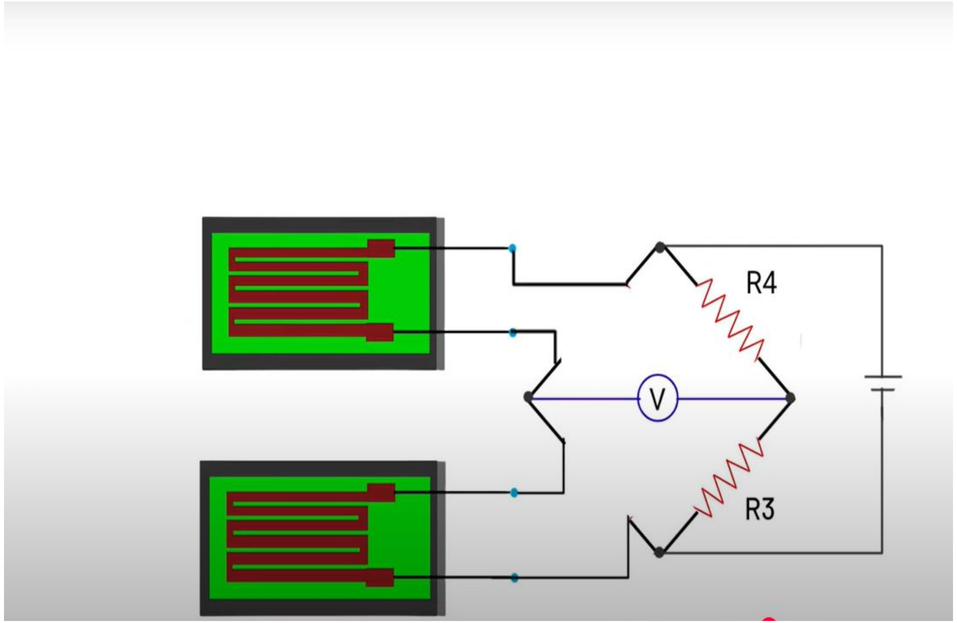
R = is directly proportional to length (L)

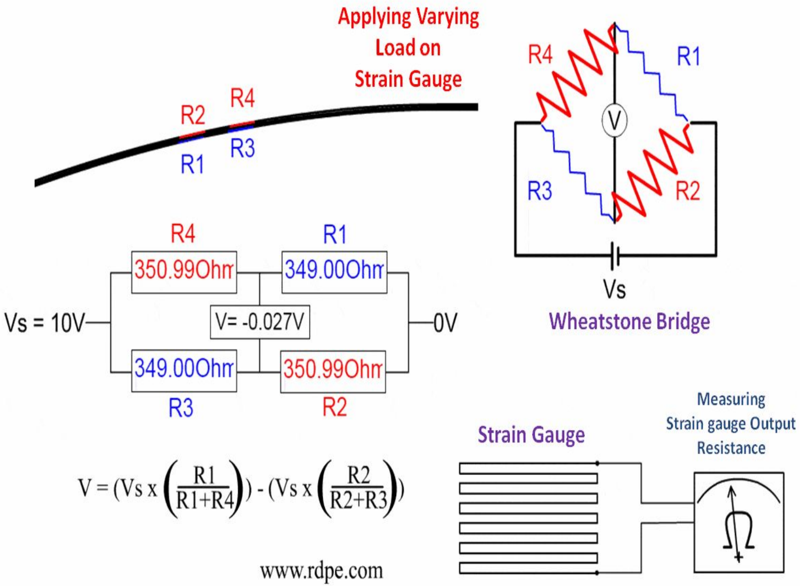
R = is inversely proportional to cross section area (A)

Strain Gages

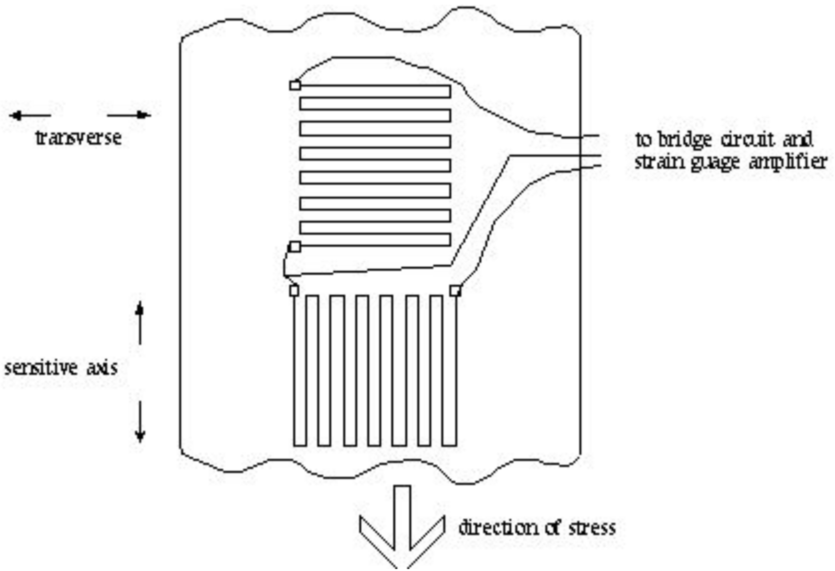


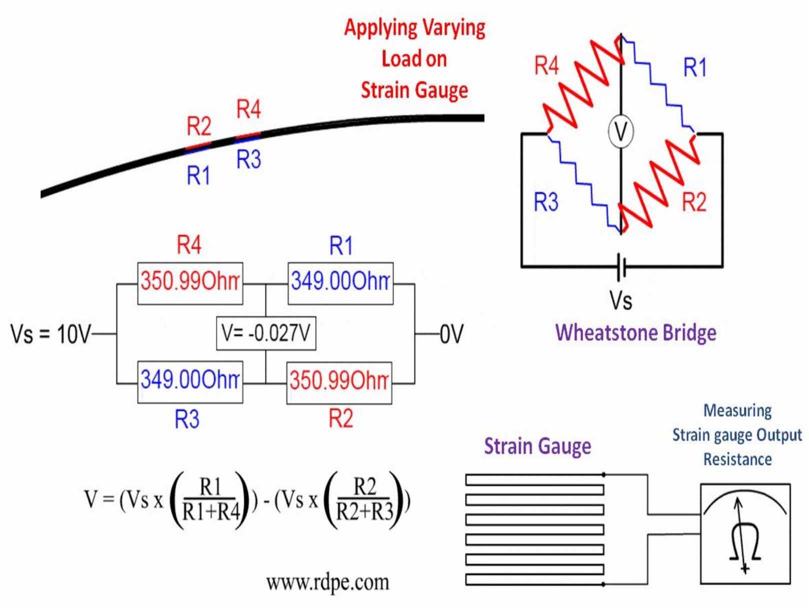
Strain Gages





Strain Gages



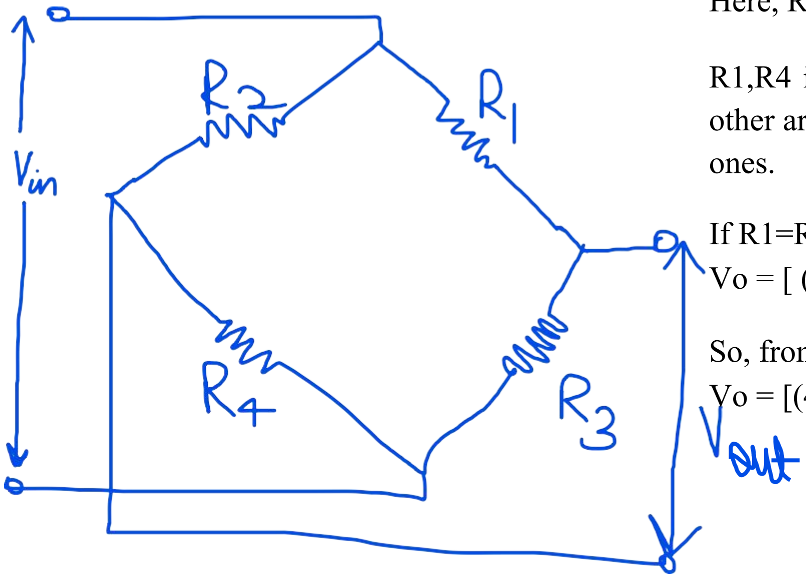


Math

For a balance bridge: stretch resistor are 4Ω and compressed resistor are 2Ω. Vin is 5V. Find Vout.

Math

For a balance bridge: **stretch resistor** are 4Ω and **compressed resistor** are 2Ω. Vin is 5V. Find Vout.



Vo = [(R4 / R4+R2) - (R3 / R3+R1)] * Vin
Here, R4 > R3

R1,R4 is a pair. R2,R3 is a pair. If one of the pair are the compressed resistors, then the other are stretched resistors. Stretched resistors have greater resistance than the compressed ones.

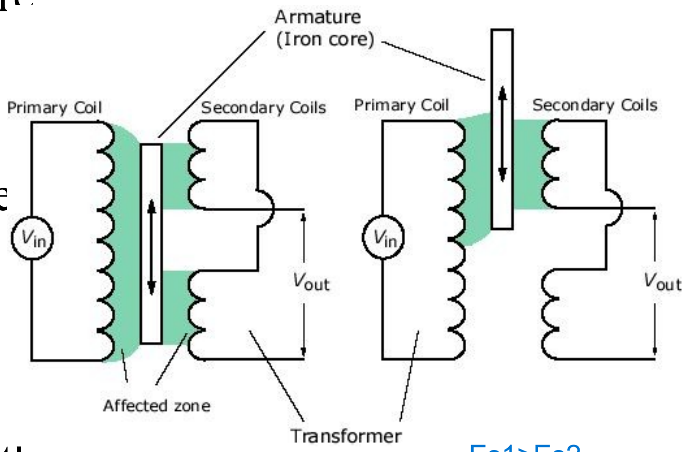
If R1=R4 and R2=R3, then
Vo = [(stretched / stretched+compressed) - (compressed / stretched+compressed)] * Vin

So, from the question,
Vo = [(4/ 4+2) - (2 / 4+2)] * 5 = 1.67V

Linear Variable Differential Transformers

- The phase relationship between the output signal and the input signal is an indication of which direction the core moved from the center position.
- The amplitude of the output signal is linearly proportional to how far the core moves from the cente position.

  Thus an LVDT can be used directly in this form to measure displacement or position.
- If we add a spring so that a force is required to move the core, then the voltage out of LVDT will be proportional to the force applied to the core.



Es1=Es2
So Vout=0

Es1>Es2
So Vout = 0

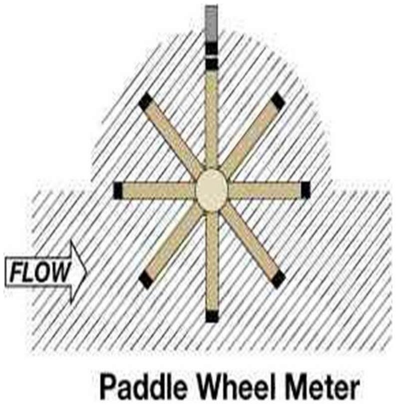
Linear Variable Differential Transformers

- LVDT is a transducer often used to measure force, pressure or position
- Basic structure of an LVDT consists of
    - **Three coils of wire wound (1 Primary, 2 secondaries**).
    - **A movable iron core (armature).**
- An ac excitation signal of 20 kHz is applied to the primary.
- The secondaries are connected in opposition such that the voltage induced in one opposes the voltage induced in the other.
- If the core is centered , then the induced voltages are equal and cancel each other.
- So there is no net output voltage.
- If the coil is moved off center, coupling to one secondary coil will be stronger, so that the coil will produce a greater output voltage.
- Result- a net output voltage.

Flow Sensors

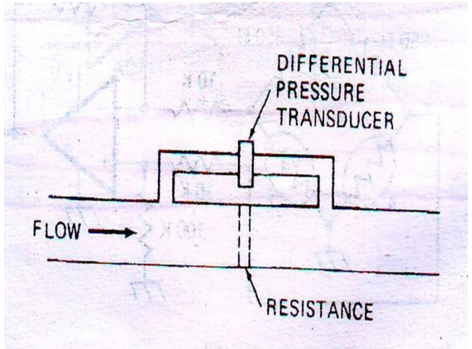
Paddle Wheel

- We can use a paddle wheel in the flow.
- The rate at which the paddle **wheel turns is proportional to the rate of flow** of a liquid or gas.
- An **optical encoder** can be attached to the shaft of the paddle wheel to **produce digital information as to how fast the paddle wheel is turning**.
- **Optical encoder converts motion into a sequence of digital pulses**.



Differential pressure Transducer

- A wire mesh or screen is put in the pipe to create some resistance.
- **Flow through this resistance produces a difference in pressure between the two sides of the screen**.
- The pressure transducer gives an **output proportional to the difference in the pressure between the two sides of the resistance**.
- Thus the **output of a pressure transducer is proportional to the flow of a liquid or gas through the pipe.**



Thank You

Military Institute of Science and Technology Department of Computer Science and Engineering

Analog Interfacing and Industrial Control (Part 2)

Course Name: Computer Interfacing
Course Code: CSE-405

Book Reference

- Microprocessor and Interfacing- Douglas V. Hall (Chapter 10)

A/D Converter

A/D Converter

- The function of an A/D converter is to produce **a digital word** which represents the **magnitude of some analog voltage or current**

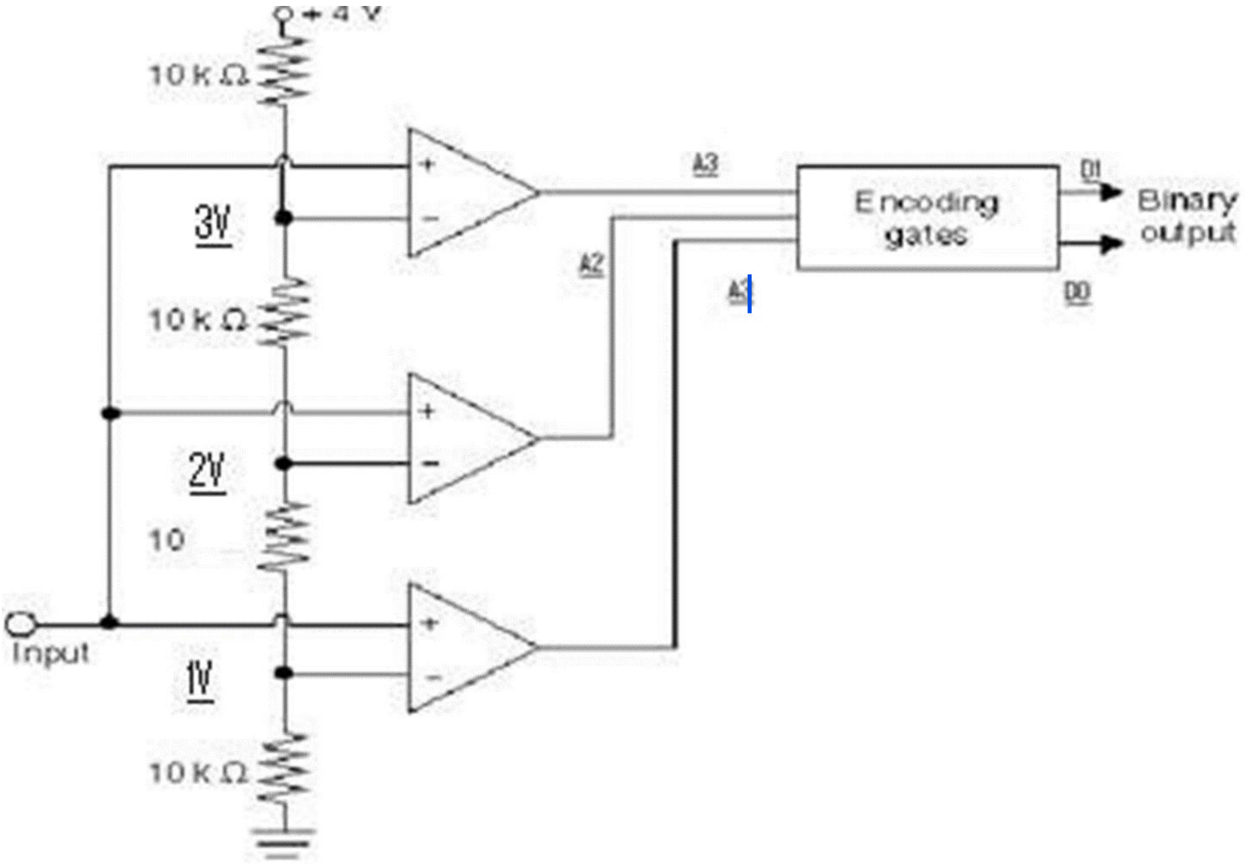
A/D converter Specification

- Resolution: Same as D/A Converter
- Accuracy: Same as D/A Converter
- Linearity: Same as D/A Converter
- Conversion Time
- This is simply the **time it takes the converter** to **produce a valid output binary code** for an applied input voltage
- When we refer to a converter as **high speed** , we mean that it has **a short conversion time**

Parallel A/D Converter

- A voltage divider sets reference voltages on the inverting inputs of each of the comparators.
- The voltage at the top of the divider represents the **full scale value** for the comparators.
- The voltage to be converted is applied to the **non inverting inputs** of all the comparators in parallel
- If the **input voltage on a comparator is greater than the reference voltage** on the inverting input, **the output of the comparator will go high**
- The outputs of the comparators then give us a digital representation of the voltage level of the input signal
- **Example:**
- With an input voltage of 2.6V , the outputs of comparators A1 and A2 will be high

Parallel A/D Converter



Parallel A/D Converter

Advantage:

- It is the **fastest type of ADC** because the conversion is performed **simultaneously** through a set of comparators, hence referred as **flash type ADC**. Typical conversion time is 100ns or less.
- The construction is simple and **easier to design**

Disadvantage:

- It is not suitable for **higher number of bits**. **The number of comparators needed to produce a result with a reasonable amount of resolution**
- To produce a converter with N bits of resolution we need **(2^N- 1)comparators**

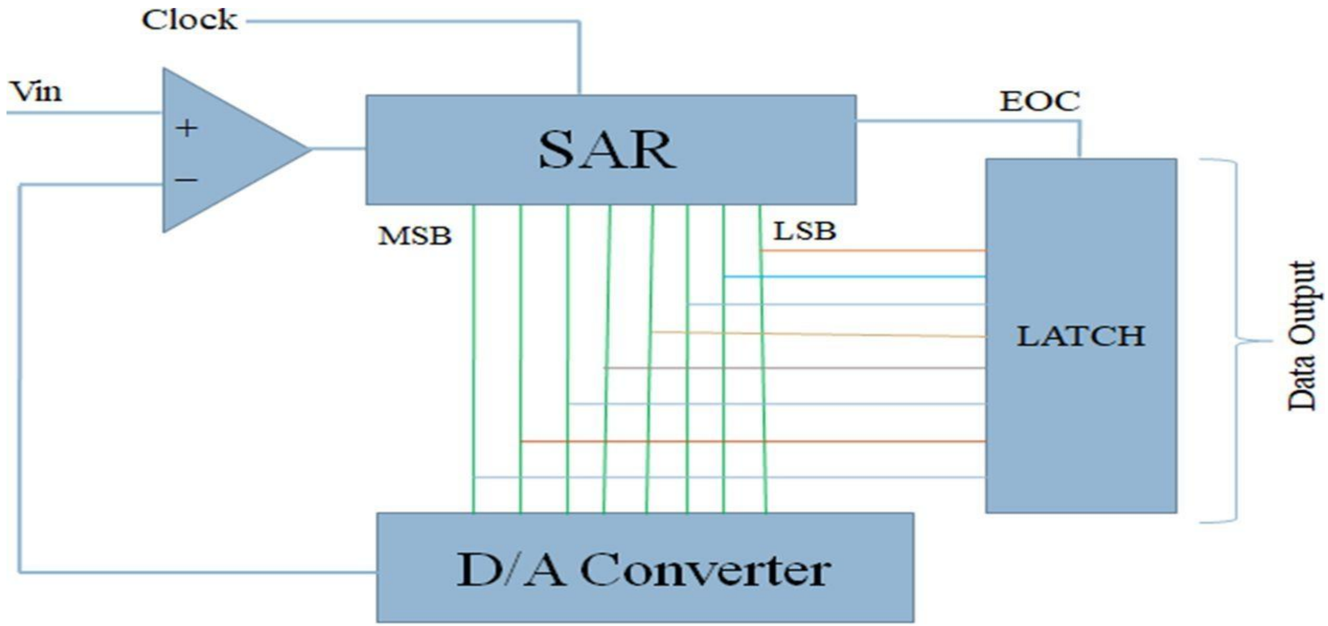
Successive Approximation A/D Converter

- The heart of this converter is ***Successive Approximation Register (SAR)***
- On the **first clock pulse**, at the **start of a conversion cycle**, the **SAR outputs a high on its most-significant bit to the D/A converter**
- The **D/A converter and the amplifier convert this to a voltage** and **apply it to one input of a comparator**
- If this **voltage (Vdac>Vin) is higher than the input voltage** on **the other input of the comparator**, the **comparator output will go low** and tell the **SAR to turn off** that bit- because it is too large
- If the **voltage from the D/A converter (Vdac<Vin) is less than the input voltage**, then the **comparator output will be high** , which tells the **SAR to keep that bit on**
- When the **next clock pulse occurs**, the SAR will **turn on the next most- significant bit to the D/A converter**

Successive Approximation A/D Converter

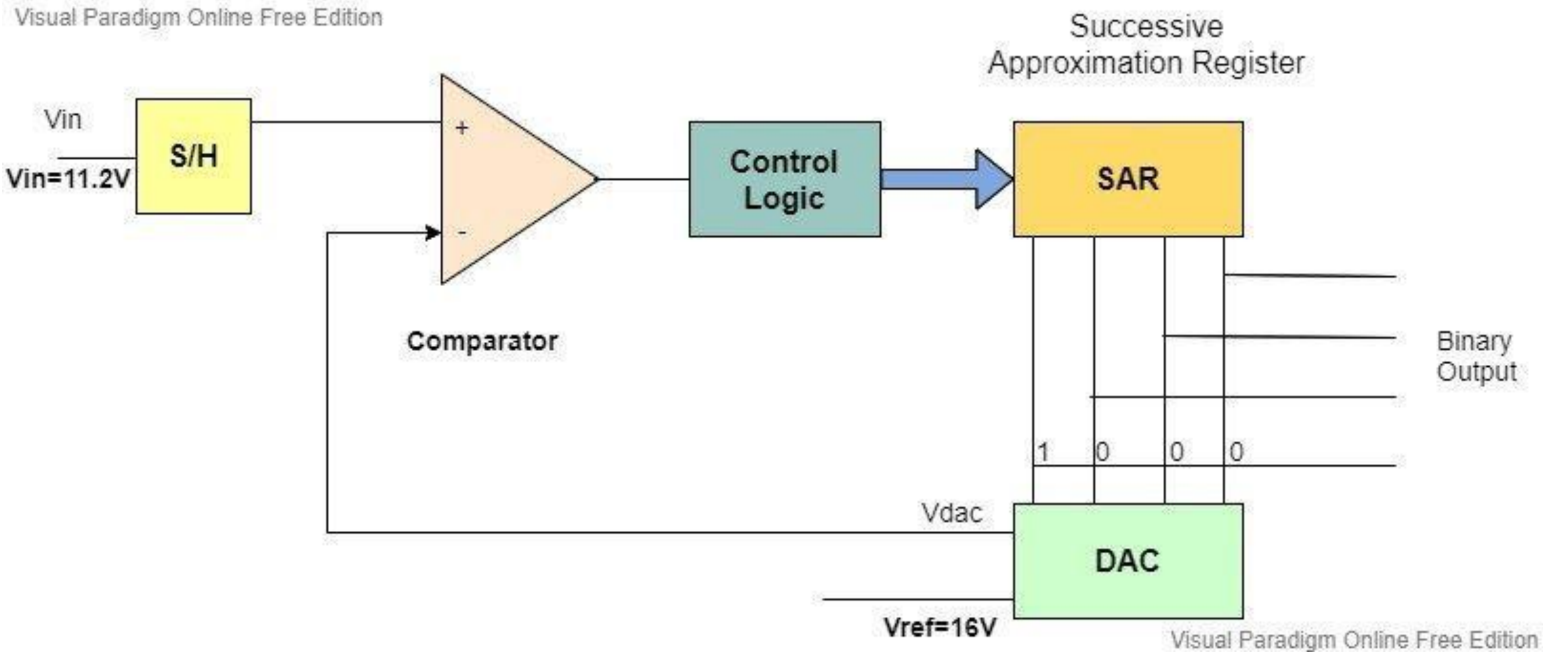
- Based on the answer this produces from the comparator , the **SAR will keep or Reset this bit.**
- **SAR will proceed in this way on down to the Least-significant bit** adding each bit to the total in turn and using the signal from the comparator to decide whether to keep that bit in the result.
- When the conversion is complete , **binary results will be now on the parallel outputs of SAR.**
- SAR sends out an **End-of-conversion(EOC) signal** to indicate the conversion is complete.
- **EOC is used to strobe the binary result into latches** , where it can be read by a microcomputer.

8 Bit SAR A/D Converter



8 Bit Successive approximation A/D converter

4 bit SAR A/D Converter



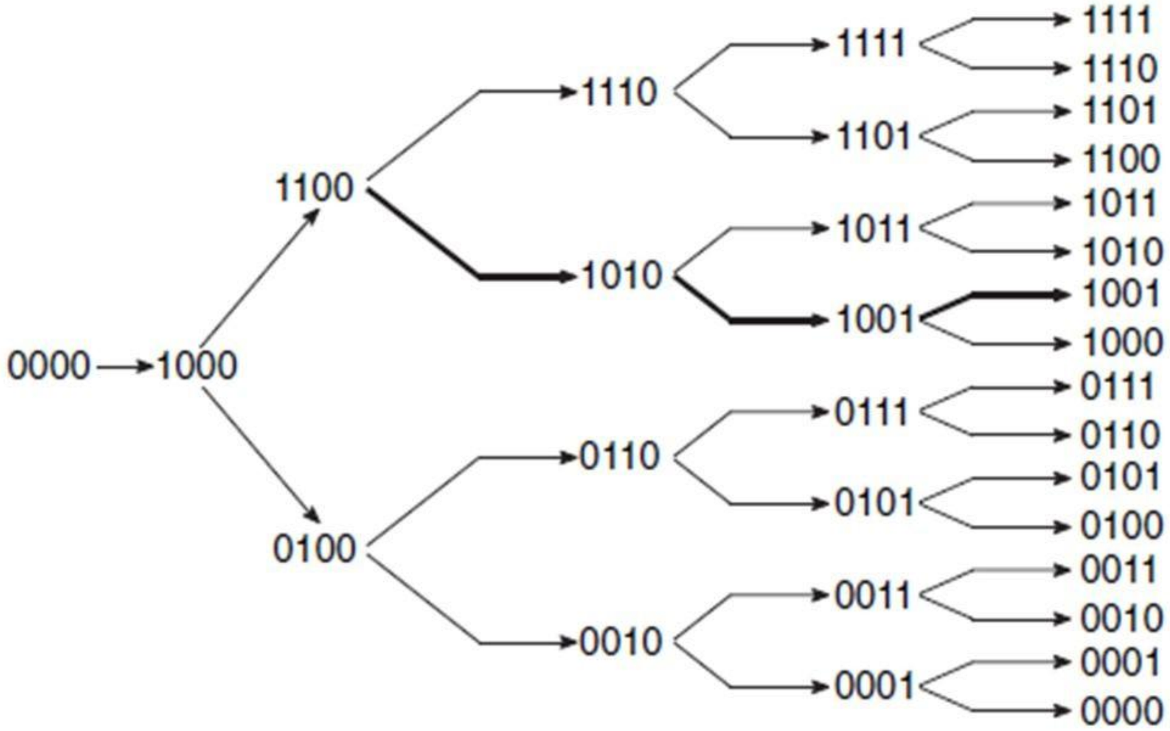
4 bit SAR A/D Converter: Steps

- We set bits one by one and use the formula: $V_{DAC} = V_{ref}\left(\frac{d_3}{2}+\frac{d_2}{4}+\frac{d_1}{8}+\frac{d_0}{16}\right)$
- At each trial the DAC outputs V_{DAC} for the tentative code and the comparator tells us whether $V_{in} > V_{DAC}$ (so keep the trial bit = 1) or $V_{in} < V_{DAC}$ (clear the trial bit = 0).
- **Step 1 (test d3 = 1)**
  - Trial code: 1000 → d3=1, d2=d1=d0=0.
  - V_{DAC}=8.0 V.
  - Compare: V_{in}=11.2 V>8.0 → keep d3=1.
  - Current tentative code = 1000.
- **Step 2 (test d2=1)**
  - Trial code: 1100 → d3=1,d2=1,d1=0,d0=0.
  - V_{DAC}=12.0 V.
  - Compare: V_{in}=11.2 V<12.0 → clear, d2=0.
  - Current code = 1000.

4 bit SAR A/D Converter: Steps

- **Step 3 (test d1 = 1)**
  - Trial code: 1010 → d3=1, d2=0, d1=1, d0=0.
  - V_{DAC}=10.0 V.
  - Compare: V_{in}=11.2 V>10.0 → keep d1=1.
  - Current code = 1010.
- **Step 4 (test d0=1)**
  - Trial code: 1011 → d3=1,d2=0,d1=1,d0=1.
  - V_{DAC}=11.0 V.
  - Compare: V_{in}=11.2 V>11.0 → keep, d0=1.
  - Current code = 1011.

Final 4-bit word: 1011



Conversion process in a successive approximation type A/D converter.

Successive Approximation A/D Converter

Advantages:

- Conversion time is very small.
- Conversion time is constant and independent of the amplitude of the analog input signal VA.

Disadvantages:

- Circuit is complex.
- The conversion time is more compared to flash type ADC.

Dual Slope A/D Converter

Self Study

Microcomputer-Based SmartScale

Microcomputer-Based Smart Scale

- **A load cell** converts **applied weight into a proportional electrical signal**
- It is then **amplified and converted into a digital value**
- Can be **read by the microprocessor and sent to the attached display**
- The **user can enter price per pound** by using the keyboard.
  - Shown on the display
- When the user presses compute key on the keyboard
  - The microprocessor multiplies the weight times the price per pound
  - and displays the computed price
- After holding the price display long enough for the user to read it, the scale goes back to reading the weight and displaying it

Microcomputer-Based Smart Scale

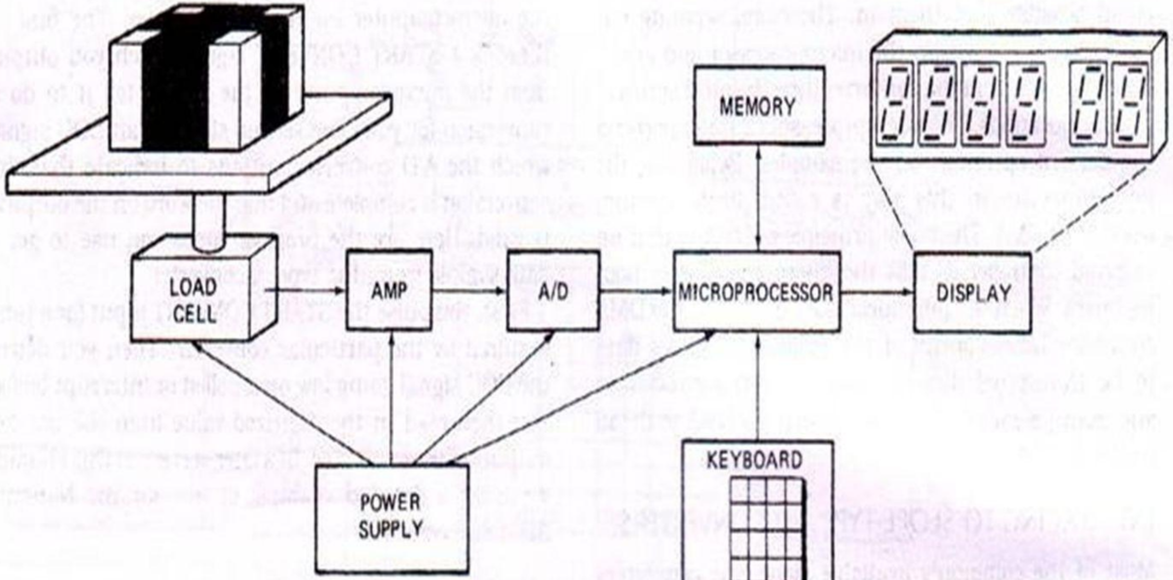


FIGURE 10-22 Block diagram of microcomputer-based smart scale.

Microprocessor-Based Industrial Control System

Industrial Process Control

- One area in which **microprocessors and microcomputers** have had a major impact is **industrial process control**
- Process control involves **first measuring system variables** such as **motor speed, temperature, the flow of reactants, the level of liquid in a tank, the thickness of a material** etc
- The **output of the controller** then adjusts the value of each variable until it is equal to a predetermined value called a ***set point***
- **System controller** must maintain each variable as close as possible to its **set point value**
- It must compensate as **quickly and accurately** as possible for any change in the variable caused by, for example , increased load on the motor

Controlling the Speed of a dc motor

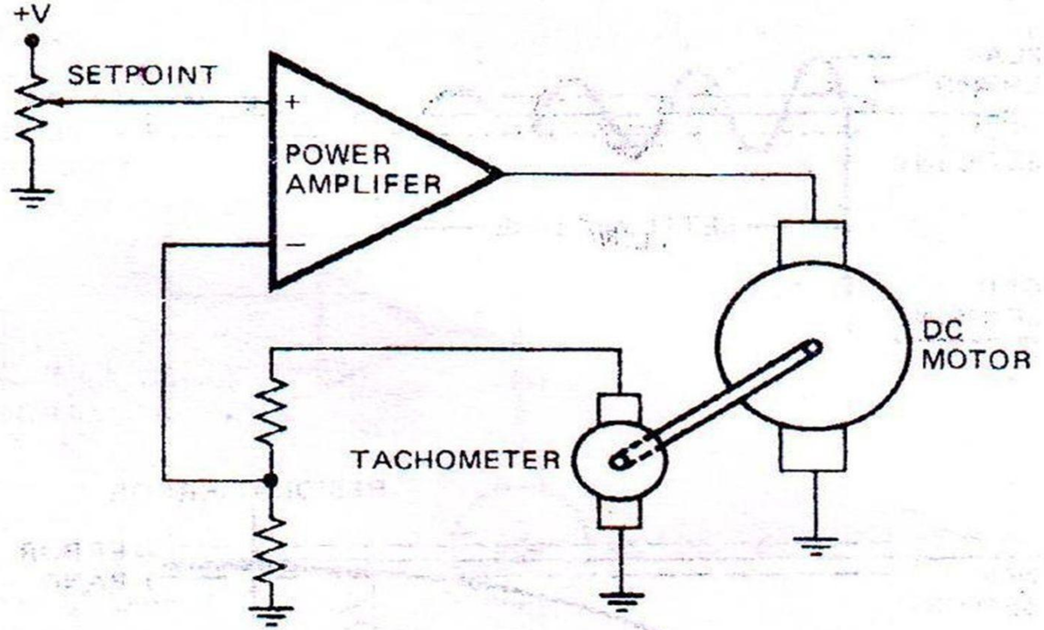


FIGURE 10-27 Circuit for controlling speed of dc motor using feedback from tachometer.

Controlling the Speed of a dc motor

- To the shaft of the motor a **dc generator or tachometer is attached**
- It puts out **a voltage proportional to the speed of the motor**
- A fraction of the tachometer output voltage is **fed back to the inverting input** of the **power amplifier driving the motor**
- In the non-inverting input of the amplifier, **a positive voltage is applied as set point**
- **When the power is turned on, the motor accelerates until the** voltage fed back from the tachometer is **nearly equal to the set point voltage**

Controlling the Speed of a dc motor

- If the load on **the motor is increased**, the motor **will initially slow down**
    - **Voltage output from the tachometer will decrease**
    - **Increase the difference** in voltage between the inputs of the amplifiers
    - **Cause it to drive more current in the motor**
- Now, **the increased current will increase the speed of the motor** to nearly the speed it had before the increased load has added
- The similar reaction takes place if the load on the motor is decreased

Problems

Overshoot

- When the **set point is changed**
- In this case, the variable- motor speed(for example) **overshoots** the new set point and bounces ups and downs for a while.
- The **time it takes the bouncing to settle within a specified error range or error band is called the *settling time***
- This **type of response is referred to as *underdamped***
- Damping can be added

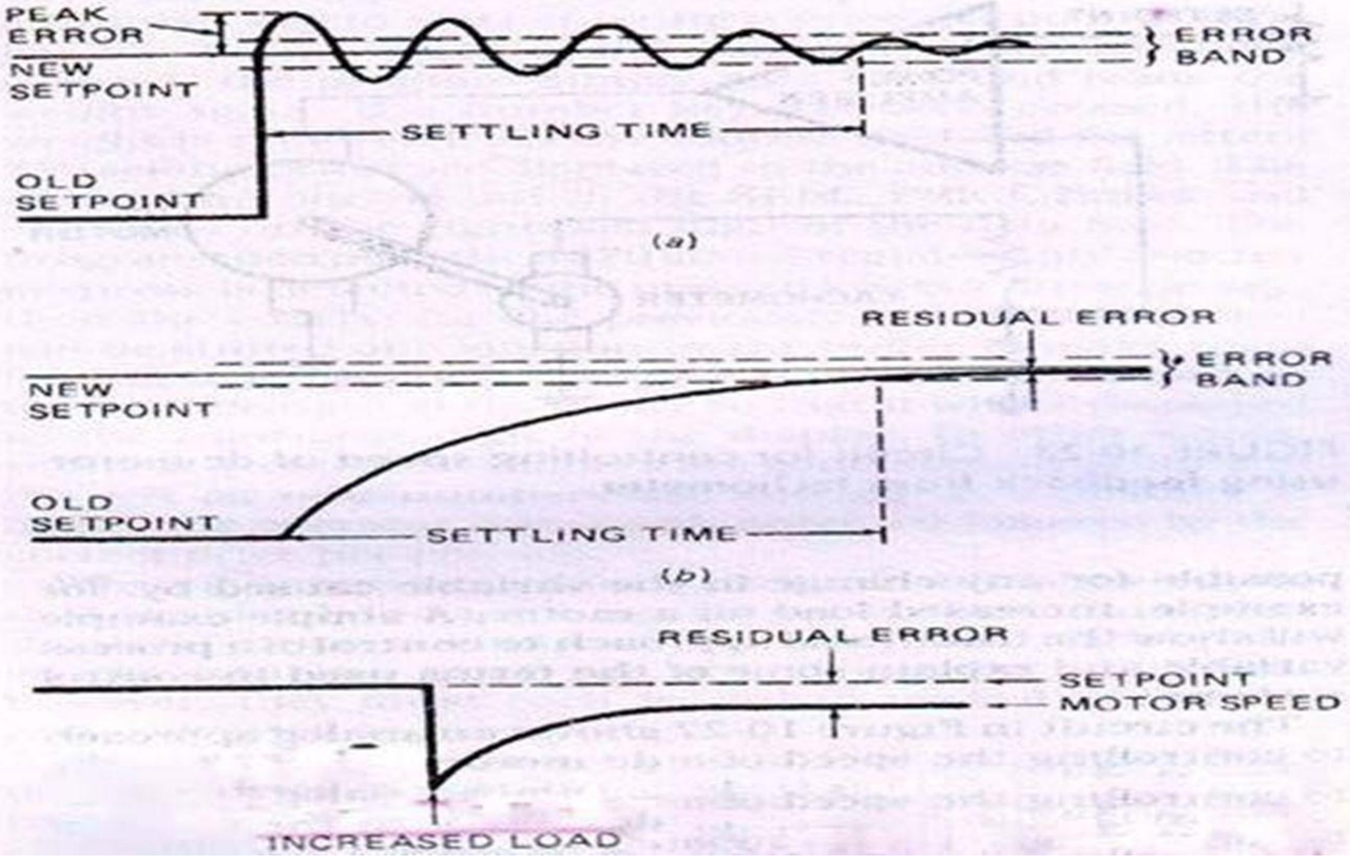
Problems

Undershoot

- If too much damping is added
- This type of response is referred to as an **over-damped response**
- The difficulty is that **it takes a long time for the variable to reach the new set point**

Residual Error

- There is some **noticeable difference between the set point and the voltage fed back from the tachometer**
- This **difference is amplified by the gain of the amplifier** to produce the **additional drive for the motor**
- For stability reason, gain of many control systems cannot be too high
- Therefore some residual error always exists between the set point and actual output



PEAK
ERROR
NEW
SETPOINT
ERROR
BAND
SETTLING TIME
OLD
SETPOINT
(a)
RESIDUAL ERROR
ERROR
BAND
NEW
SETPOINT
OLD
SETPOINT
SETTLING TIME
(b)
RESIDUAL ERROR
SETPOINT
MOTOR SPEED
INCREASED LOAD

Solution- PID Controller

- Circuits with more **complex feedback are used**
- Here the **power amplifier is an adder with 4 inputs**
- The **current supplied to the summing point of the adder** by the **set point input produces the basic drive output current**
- If there is **no difference between the set point an the feedback voltage** from the tachometer- **other 3 inputs do not supply any current**
- **Amplifier 1 compares the set point value with the feedback voltage** from the tachometer
- If speed of the **motor is at set point value**
  - **Amplifier 1 output will be zero**
  - **Amplifier 2,3,4 will contribute no current** to the summing junction of the power amplifier.

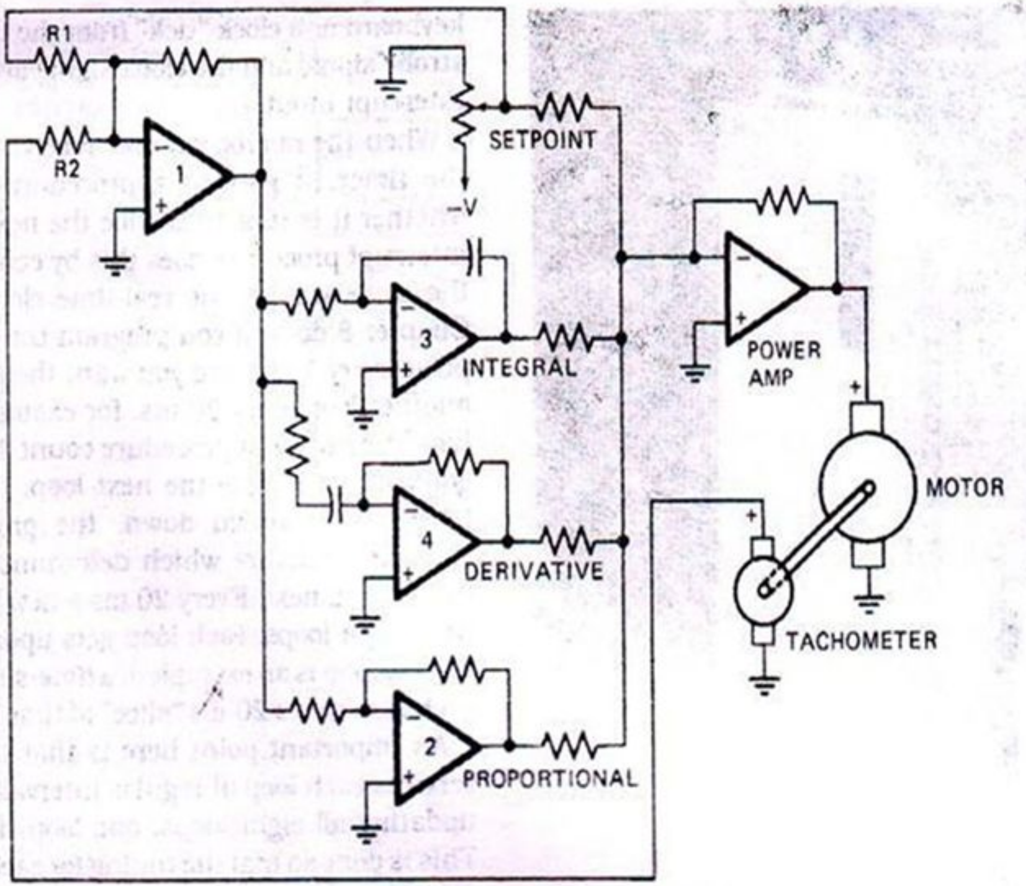


FIGURE 10-29 Circuit showing proportional, integral, and derivative feedback control.

PID Controller

- **If speed of the motor is not at set point value**
  - **Amplifier 1** now **has some output**
  - Amplifier **2,3,4 produces three types of feedback** of this error signal to the summing junction of the power amp
- **Amplifier 3 provides integral feedback**
  - The cure for residual error
- **Amplifier 4 improves the response time** of the system
  - Provides **derivative feedback**
  - Produces **signal, proportional to the rate of the change of the error signal**
  - If the load on the system is suddenly changed, this circuit will give a quick feedback to try to correct the error

PID Controller

- By using combination of these types of feedback , a system can be adjusted for optimum response to changes in load or set point.
- **Amplifier 2 provides proportional feedback**.
- **Process control loops** thatuses all these three types of feedback are called ***proportional Integral derivative*** **or PID control loops**.

Thank You

Military Institute of Science and Technology Department of Computer Science and Engineering

Analog Interfacing and Industrial Control

Course Name: Computer Interfacing

Course Code: CSE-405

Book Reference

- Microprocessor and Interfacing- Douglas V. Hall (Chapter 10)

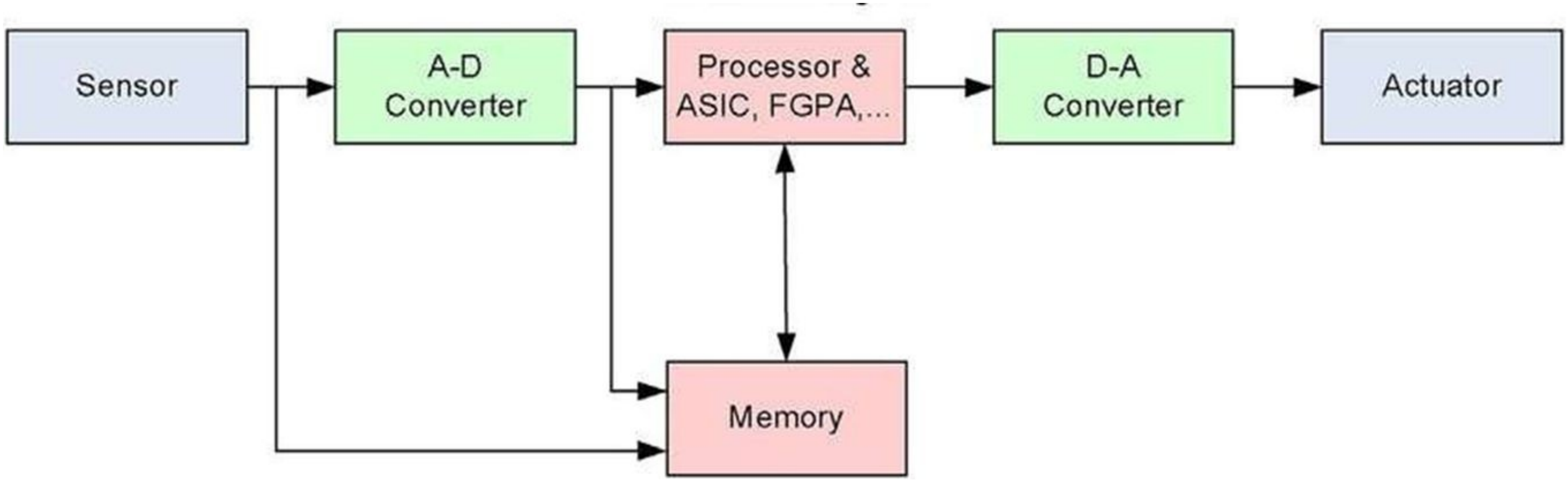
Embedded System

Introduction

- A computer hardware system having software embedded in it.
- A **microcontroller or microprocessor based system** which is designed to **perform a specific task.**
- either as an independent system or as a part of a large system.
- At the core is an integrated circuit designed to carry out computation for **real-time operations.**
- An embedded system has three components
  - It has hardware.
  - It has application software.
  - It has **Real Time Operating system (RTOS)** that defines the way the system works.
  - It sets the rules during the execution of application program.

https://www.digi.com/blog/post/examples-of-embedded-systems

Basic Structure of An Embedded System



Basic Structure of An Embedded System

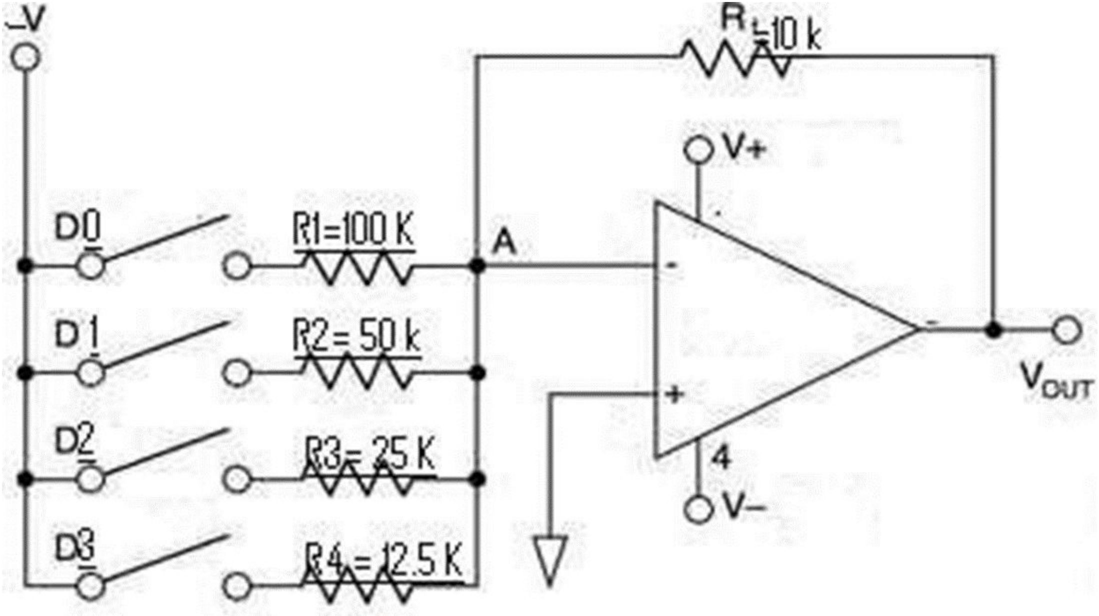
- **Sensor:** It measures the **physical quantity and converts it to an electrical signal** which can be read by an observer or by any **electronic instrument like an A/D** converter. A sensor stores the measure**d quantity to the memory.**
- **A-D Converter:** An analog-to-digital converter **converts the analog signal sent by the sensor** into a digital signal.
- **Processor & ASICs: Processors process the data to measure the output** and store it to the memory.
- **D-A Converter:** A **digital-to-analog converter converts the digital data fed by the processor to analog data**
- **Actuator:** An actuator converts an electrical signal into physical action (movement, heat, pressure, rotation, etc.). It executes commands from the processor to control the physical world.

D/A Converter

D/A Converter: Operation

- The purpose of a Digital to Analog converter is to **convert a binary word to a proportional current or voltage.**
- In a simple 4 bit D/A converter , **inverting input of the circuit is referred to as the summing point.**
- When one of the switches is closed, **current will flow from -V to the summing point**.
- The equation of O/P voltage, $\mathbf{V_{out} = - Rf/ R (V_{ref})}$
- If we close the switch D0 , $\mathbf{V_{out} = - 10/ 100(-5) = 0.5\ V}$
- If we close the switch D0 & D1 , $\mathbf{V_{out} = - 10/ 33.33(-5) = 1.5\ V}$
- The point here is that, **binary weighted resistors produce currents which are summed by the op amp to produce a proportional output voltage**
- The heart of **a D/A converter is a set of binary weighted current sources** which **can be switched on or off according to a binary word applied to its input**

D/A Converter



D/A Converter: Operation

- Let the 4 bits be D3, D2, D1, D0 (MSB → LSB) and input resistors be R4 for the MSB, R3, R2, R1(LSB). Let Rf be the feedback resistor and Vref is the reference voltage. The formula for output voltage is:

$$V_{\text{out}} = -V_{\text{ref}}\, R_f \left(\frac{D_0}{R_1} + \frac{D_1}{R_2} + \frac{D_2}{R_3} + \frac{D_3}{R_4} \right)$$

- **Full-scale voltage**

  Full-scale here means **all bits = 1**. Plug all Di=1. So the formula is:

$$V_{\text{FS (general)}} = -V_{ref}\, R_f \left(\frac{1}{R_1} + \frac{1}{R_2} + \frac{1}{R_3} + \frac{1}{R_4} \right)$$

- The FS amplitude depends on the sum of the inverse input resistances and on the ratio Rf to those Ri
- If you change Rf, you scale the whole output up/down; if you change the Ri weights you change the contribution of each bit.

Binary Weighted D/A Converter: Operation

- A very common design choice sets the input resistors in exact binary ratios and makes the feedback resistor equal to the MSB resistor. For 4 bits:
- Choose a base resistor Rref and set,

  R4 (MSB)=2Rf, R3=2^2Rf, R2= 2^3Rf, R1=2^4 Rf (LSB)
- Now substitute into the general formula:

$$V_{\text{out}} = -V_{\text{ref}}\,R_{\text{ref}}\left(\frac{D_0}{16R_{\text{ref}}} + \frac{D_1}{8R_{\text{ref}}} + \frac{D_2}{4R_{\text{ref}}} + \frac{D_3}{2R_{\text{ref}}}\right)$$
$$= -V_{\text{ref}}\left(\frac{D_0}{16} + \frac{D_1}{8} + \frac{D_2}{4} + \frac{D_3}{2}\right)$$

Use this formula when Rref not given

- Now, in this case the full scale voltage:

$$V_{\text{FS}} = -V_{\text{ref}}\left(\frac{1}{2} + \frac{1}{4} + \frac{1}{8} + \frac{1}{16}\right) = -V_{\text{ref}}\left(1 - \frac{1}{16}\right) = -V_{\text{ref}}\frac{15}{16}$$

- Here, LSB (0001) step value is: $\text{LSB step} = \dfrac{-V_{\text{ref}}}{16}$
- So, $V_{\text{FS}} = -(V_{\text{ref}} - \text{LSB step})$

D/A Converter: Characteristics

Resolution

- This is determined by the **number of bits in the input binary word**.
- A converter with **8 binary inputs has 2^8 output levels.**
- Thus its resolution is **1 part in 256**.

Full scale output voltage

- The analog output voltage produced by a DAC when its digital input code is set to the maximum valid code (all 1's)
- In case of binary weighted D/A converter, the maximum output voltage of a converter will always have a value 1 least significant bit (LSB) less than the named value.

D/A Converter: Example

Example:

Q) An n bit D/A converter has **full scale output of 5V**. **But actually it is 4.6875V**. **what is the value of n? (Binary Weighted D/A converter)**

A) Actual output = full scale output - (full scale output / 2^n) 4.6875 = 5 - 5/(2^n)-1

- So, n=4
- value of 1 LSB = (10V)/4096 = 2.44mV
- The highest voltage output for the converter when it is properly adjusted will then be : (10 – 0.0024)V or 9.9976V.

D/A Converter: Characteristics

Accuracy

- The accuracy specification for a D/A converter is a comparison between the actual output and the expected output
- It is **specified as a percentage of the actual full-scale output voltage or current**

Example

- If the converter has a output of **10V and ± 0.2 percent accuracy**
- Maximum error for any output will be **0.002 * 10V or 20mV**

Linearity

- Linearity is the measure of **how much the output ramp deviates from a straight line** as the converter is stepped from no switches on to all switches on.
- **Ideally the deviation** of the output from a straight line **should be no greater than ± ½ the value of LSB to maintain overall accuracy**

D/A Converter: Characteristics

Settling Time

- When we change the binary word applied to the input of a converter , the output will change to the appropriate new value
- The output may overshoot the correct value, “ring” for a while before finally settling down to the correct value
- The time the output takes to get within ± ½ **LSB of the final value is called settling time**
- This specification is important, because if a converter is operated at too high frequency, it may not have time to settle to one value before it is switched to the next

D/A Converter: Interfacing to Microcomputers

- The inputs of the D/A circuit can be connected directly to the microcomputer output port.
- Thus interfacing an 8 bit converter involves simply connecting the inputs of the converter to an output port.
- For some application we need to interface a 12 bit converter to a microcomputer.
- If we are working with a system which has an 8 bit data bus
  - The first thought might be – to connect the lower 8 inputs of the 12 bit converter to one output port and upper 4 bits to another port
  - In that case, we could send the lower 8 bits with one write operation and the upper 4 bits with another write operation
  - Time between the two writes introduces a potential problem in this approach

D/A Converter: Interfacing to Microcomputers

Example:

- Suppose we want to change the pattern of a 12 bit converter from 0000 1111 1111 to 0001 0000 0000
- When we will write the lower 8 bits, the output will go from 0000 1111 1111 to 0000 0000 0000
- When we will write the upper 4 bits the output will go back up to the desired 0001 0000 0000
- The point here is that for the time between the two writes the output will go to an unwanted value
- In many systems this could be disastrous
- **The cure of this problem is to put latches on the input lines**
- The latches can be loaded separately and then strobed together to pass all 12 bits to the D/A converter at the same time

Thank You

CSE 405
(Computer Interfacing)

Printer Interface

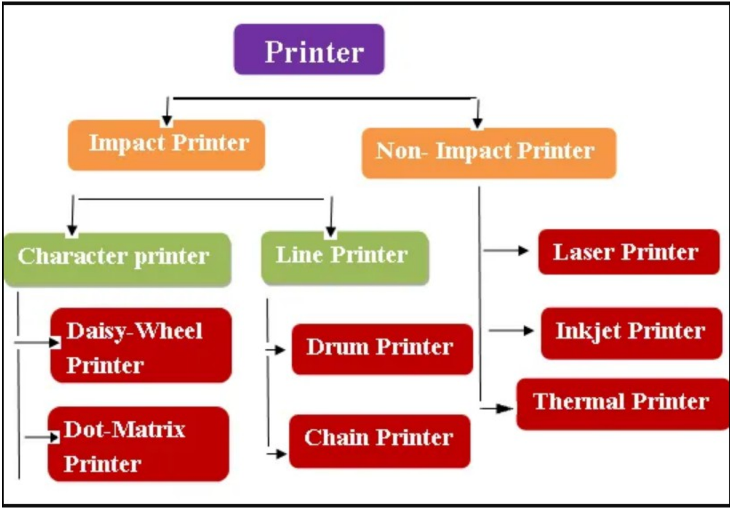
(Cook and White)

Outline

Printer

- An external hardware device responsible for taking computer data and generating a hard copy of that data.
- Printers are one of the most commonly used peripherals and they print text and still images on the paper.

Types of printer



Dot Matrix Printer

- The term dot matrix refers to the process of **placing dots** to form an image.
- Its speed is usually **30 to 550 characters per second** (cps).
- This is the cheapest and the most **noisy printer and has a low print quality**.

Dot Matrix Printer- Working Principle

- The dot matrix **forms images one character** at a time as the print head moves across the paper.
- Uses **tiny pins to hit an ink ribbon** and **the pape**r much as a typewriter does.
- This printer **arranges dots to form characters** and all kinds of images.
- **9 to 24 vertical column pins** are contained in a rectangular print head.
- When print head moves across the paper, **pins are activated to form a dotted character image.**
- These printers can produce carbon copies along with the originals.

Dot Matrix Printer

- Advantage:
  - Inexpensive
  - Low per page cost
  - Energy efficient
- Disadvantage:
  - Noisy
  - Low resolution
  - Limited fonts flexibility
  - Poor quality graphics output

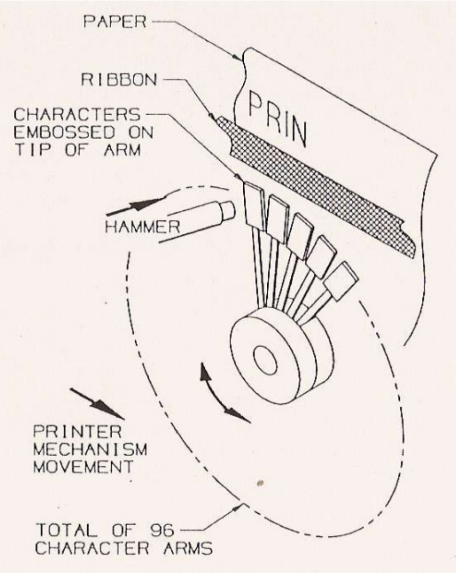
Daisy Wheel Printer

- Daisy wheel printers **print only characters and symbols** and cannot print graphics.
- They are generally slow with a printing speed of about **10 to 75 characters per second.**

Daisy Wheel Printer- Working Principle

- very similar to typewriters.
- A circular printing element (known as daisy wheel) contains all text, numeric characters and symbols mould on each petal on the circumference of the circle.
- The printing element rotates rapidly with the help of **a servo motor** until the desired letter is facing the paper.
- Then the **hammer strikes the disk**, forcing the character to hit **an ink ribbon**, leaving an impression of the character on the paper.

Daisy Wheel Printer- Working Principle



Laser Printer

Made for

- High speed
- High volume production

Critical component:

- Photosensitive drum
- Sensitive to light
- In **dark- it has a high resistance**
- Acts as a capacitor which is charged by charging wire

Other components:

- Laser
- Polygonal mirror
- Toner
- Corona wire
- Fusing roller

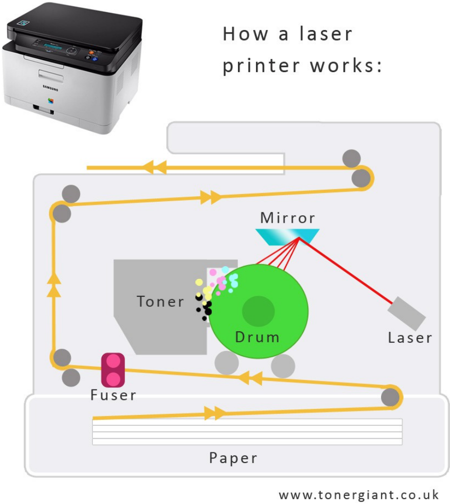
Laser Printer

- Laser printers use very **advanced technology** and produce a high quality output.
- Laser printers can also produce high quality graphics images.
- Resolution is **600 to 1200dpi.**

Laser Printer- Working Principle

- Paper is fed and **the drum rotates.**
- A laser beam conveys information from the computer to a **rotating mirror** and thus **an image is created on the drum**.
- **The charges on the drum are ionized** and the toner sticks to the drum.
- **Toner** is transferred from drum to paper.
- Heat is applied to fuse the toner on the paper.

Laser Printer- Working Principle



Laser Printer- Limitations

- Wasteful in small operations
- working process

Inkjet Printer

- Computer sends the **printer a digital map of every dot of the image or text.**
- The printer's brain decides which **nozzles will fire** and how much ink for each dot.
- Rollers grab a sheet of paper and move it underneath the print head.
- The paper stops **exactly where the first line of printing will star**t.
- The Print Head Glides Back and Forth: A small carriage holding microscopic nozzles slides left to right, then back again.
- Think of it like a very fast paintbrush moving across the page.
- Ink Is Shot Out as Tiny Droplets
- Thermal bubble: A tiny heater warms the ink in a nozzle - ink forms a bubble -bubble pops and shoots a droplet.
- Each droplet lands exactly where the computer told it.

Inkjet Printer

- Ink Dries Almost Instantly: Special quick-dry ink soaks slightly into the paper fibers and stays put.
- By the time the sheet exists, it’s already dry to the touch.
- inkjet printer process

Inkjet Printer

- Advantages:
  - High resolution output
  - Energy Efficient
- Disadvantages:
  - Expensive
  - Time consuming incase of graphics printing
  - Special paper required for high resolution output.

End of Computer Interfacing!!

CSE 405
(Computer Interfacing)

Barcode

(Computer Peripherals by Cook & White)

Outline

1. Barcode
1. Codes
1. Error Handling

Introduction

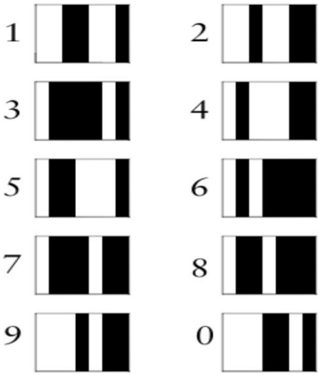
- Giving every item that you want to classify its own, unique number and then simply printing the number on the item so an electronic scanning device can read it.
- Trouble with decimal numbers:
  - A misprinted number can show a different one.
  - For example, eight could look like a three to a computer or a six is identical to nine if you turn it upside down—which could cause all sorts of chaos at the checkout if you scanned the wrong way up.

Representation

- Looking at a barcode, can't make head or tail of it- where one number ends and another one begins.
- But it's simple, really.
- Each digit in the product number is given the same amount of horizontal space: **exactly 7 units**.
- Then, to represent any of the numbers from zero through nine, we **simply color those seven units with a different pattern of black and white stripes.**

Representation

- The number one is represented by coloring in two white stripes, **two black stripes, two white stripes, and one black stripe**
- The first part of a barcode tells you the country where it was issued.
- The next part reveals the manufacturer of the product.
- The final part of the barcode identifies the product itself.
- Most products carry a simple barcode known as the UPC (universal product code)



Barcode Scanner

- Reading barcode involves determining the widths of the dark and light bars across the pattern.
- Elements of bar code may be only fractions of a millimeter wide
- the reader has to focus to at least this resolution if a clean output is to be produced

Barcode Scanner

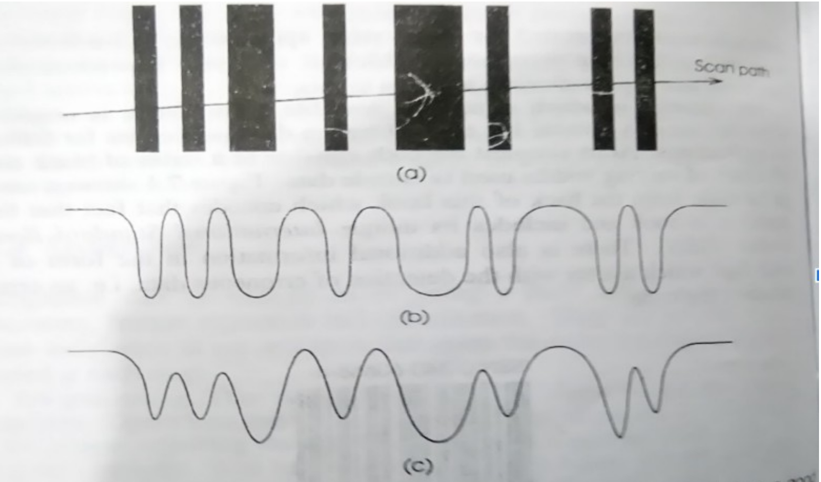
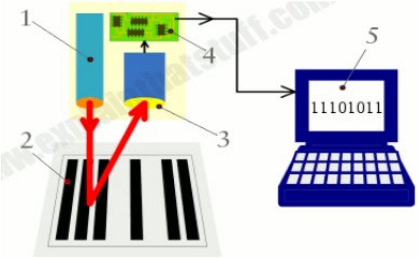


Figure 7.5 Scanning a bar code: (a) the scan path; (b) output from a good resolution scanner; (c) output from a poor resolution scanner

Barcode Scanner

- Barcode scanners have to be able to read the black-and-white zebra lines on products extremely quickly and feed that information to a computer or checkout terminal, which can identify them immediately using a product database.
- Lets' assume, each **black line corresponding to a one** and **each white line a zero.**



Barcode Scanner

- Scanning **head shines LED or laser** light onto barcode.
- Light reflects back off barcode into a **light-detecting electronic component called a photoelectric cell**
- White areas of the barcode reflect most light; black areas reflect least.
- As the scanner moves past the barcode, the **cell generates a pattern of on-off pulses** that correspond to the black and white stripes.
- An **electronic circuit** attached to the scanner converts these on-off pulses into binary digits (zeros and ones).
- The **binary digits are sent to a computer attached** to the scanner, which detects the code as 11101011.

Types of Barcode Scanner

- In some scanners, there's a single photoelectric cell and, as you move the scanner head past the product, the cell detects each part of the black-white barcode in turn.
- In more sophisticated scanners, **there's a whole line of photoelectric cells and the entire code is detected in one go.**

Types of Barcode Scanner

- **Pen Scanner:** Need to run it across the barcode so it can reach each block of black or white in turn
- **Wand Scanner:** A wand scanner, the photocells read the entire code at once.
- **Laser Scanner:** The laser beam being bounced around at high-speed by a spinning wheel so it reads products (literally) in a flash

The best barcode scanners are so accurate that they make only one **mistake in something like 70 million** pieces of scanned information!

Types of Barcode Scanner

Coding of Barcode

- UPC- Universal Product Coding: UPC-A (Universal
- Product Code) is assigned and managed by **GS1**
- EAN-European Article Number

Numbers to be coded consist of

- **Prefix (1 digit for UPC, 2 for EAN): Product category**

  Example:

  -3 = Pharmaceuticals

  -5 = Coupons

- **Manufacturer number (5 digits)**
- **Item reference(5 digits)**:

  -Specific to the **product**

  -Assigned by the manufacturer

- **Check digit (1 digit)**:

  -Calculated using the other 11 digits

  -Detect errors during scanning

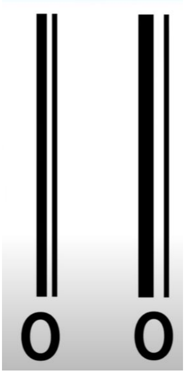
UPC-A

- **12 digit** number
- **UPC prefix** denotes the **product category**, 0 for grocery, 3 for pharmaceuticals etc
- The scan able area of every UPC-A barcode follows the pattern **SLLLLLLMRRRRRRE**
- where **S (start) and E (end)** guard patterns are represented the same way **(101)**
- **M (middle) represented by 5 bit : 01010**
- L (left) and R (right) sections collectively represent the 12 numerical digits that make each UPC-A unique.
- The total width for a digit is always 7 modules; consequently, UPC-A 12-digit number requires a total of $7 \times 12 = 84$ modules.
- A complete UPC-A is **95 modules** wide: 84 modules for the digits, combined with **11 modules** for the S (start), M (middle), and E (end) guard patterns.

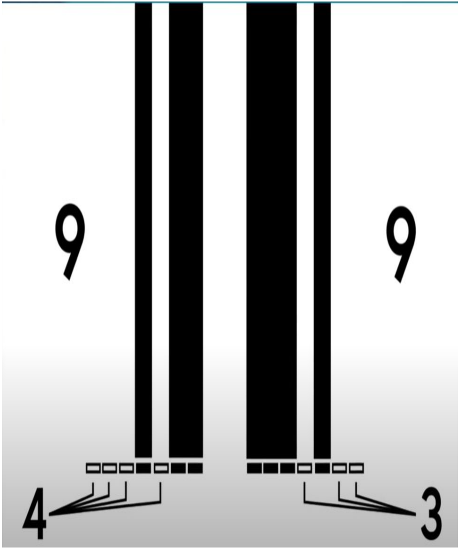
UPC-A

- **Codes** used on either side of the **Middle (M)** are **different** so that the direction of scan can be determined
- A scanner can determine whether it is scanning a symbol from left-to-right or from right-to-left
- The scanner will first see digit representation, **if it has odd** parity **then scanning is left-to-right**, or if it has even parity then scanning is right-to-left.
- With the parity/direction information, an **upside-down** symbol will not confuse the scanner.

UPC-A



- The number of black bars (representing binary 1s) is always **odd in odd parity**
- The **number of black bars** (representing binary 1s) **is always even in even parity.**



Code

- ‘Left Hand A’ Encoding:
  - Used for the digits on the **left side** of the middle point
  - **Left-Hand A pattern** uses **odd parity**
  - The number of black bars (representing binary 1s) is always **odd**.
- ‘Left Hand B’ Encoding:
  - Used for the digits on the **left side** of the middle point
  - **Left-Hand B pattern** uses **even parity**
  - The number of black bars (representing binary 1s) is always even in even parity.
- For UPC: **left hand digits are always encoded using ‘Left** Hand A’ and right hand digits using “**Right** Hand” encoding

Code

- ‘Right Hand’ Encoding:
  - Used for right-side digits (8–13)
  - Even parity: Each 7-bit pattern has an even number of 1s (black bars)
  - It is also the bitwise inverse (NOT) of Left hand A-code patterns, and mirrored

Digit	L-code (Odd parity)	R-code (Even parity)
0	0001101 (**3 ones**)	1110010 (**4 ones**)

Code

EAN:

- EAN-13 (European Article Number) is a 13-digit barcode system.
- The left-hand side contains the **first 7 digits**
- The **first digit defines** how the following **six digits** will be encoded.
- Can be either **left-Hand A** (odd parity) or **Left-Hand B** (even parity).

Code

- EAN:
  - From 00 to 09 is not used in EAN so UPC can also be read by EAN-13 scanners.
  - EAN-13 barcodes start with a first digit of **0** follow the same Left-Hand A encoding for the first six digits
  - **UPC-A barcodes to be read as valid EAN-13 barcodes** by simply adding a leading zero.

Character Set for Coding

Table 7.1 *UPC/EAN character set ('1' represents a dark band)*

Number	Left Hand A	Left Hand B	Right Hand
0	0001101	0100111	1110010
1	0011001	0110011	1100110
2	0010011	0011011	1101100
3	0111101	0100001	1000010
4	0100011	0011101	1011100
5	0110001	0111001	1001110
6	0101111	0000101	1010000
7	0111011	0010001	1000100
8	0110111	0001001	1001000
9	0001011	0010111	1110100

Table 7.2 *Code for the first prefix character in EAN13*

Number	Prefix 2	Data 1	Data 2	Data 3	Data 4	Data 5
0	A	A	A	A	A	A
1	A	A	B	A	B	B
2	A	A	B	B	A	B
3	A	A	B	B	B	A
4	A	B	A	A	B	B
5	A	B	B	A	A	B
6	A	B	B	B	A	A
7	A	B	A	B	A	B
8	A	B	A	B	B	A
9	A	B	B	A	B	A

Character Set for Coding

Codering EAN-13

cijfer	L-code	G-code	R-code
0	0001101	0100111	1110010
1	0011001	0110011	1100110
2	0010011	0011011	1101100
3	0111101	0100001	1000010
4	0100011	0011101	1011100
5	0110001	0111001	1001110
6	0101111	0000101	1010000
7	0111011	0010001	1000100
8	0110111	0001001	1001000
9	0001011	0010111	1110100

EAN13 prefix

7	8	0	3	4	0
lefthand	lefthand	lefthand	lefthand	lefthand	lefthand
A	B	B	A	B	A
0111011	0001001	0100111	0111101	0011101	0001101

The last six digits are used to code the righthand portion:

6	0	6	5	8	2
righthand	righthand	righthand	righthand	righthand	righthand
1010000	1110010	1010000	1001110	1001000	1101100

Calculating the Checksum

1. Add the values of the digits in odd positions.
2. Multiply this result by 3.
3. Add the values of the digits in even positions.
4. Sum the results of steps 2 and 3.
5. The check character is the smallest number, which, when added to the result in step 4, produces a multiple of 10.

Calculating the Checksum

1. Procedure is same for both UPC and EAN.
2. Just UPC does not have the first digit (as UPC is 12-digit code) So it finds a first digit value of zero
3. This does not affect the calculation of check digit

Checksum

[redacted]

Assume the barcode data = 01234567890X and it is UPC code. Calculate the checksum.

- Since it consists of 12 digits,first digit will be considered as '0' [0-012345-67890-X]
- 0 + 2 + 4 + 6 + 8 + 0 = 20
- 20 X 3 = 60
- 0+1 + 3 + 5 + 7 + 9 = 25
- 60 + 25 = 85
- 85 + X = 90 (next highest multiple of 10), therefore X = 5 (checksum)


End of
Slides