Embedded System Components

MODULE 3

What is an embedded system?

- An embedded system is an electronic/electro-mechanical system designed to perform a specific function and a combination of both hardware and software.
- Every embedded system is unique and the hardware as well as the software is highly specialized to the application domain.
- Embedded systems are becoming an inevitable part of any product or equipment in all fields including household appliances, telecommunications, medical equipment, industrial control, consumer products, etc.

Embedded Systems vs General Computing Systems

Criteria	General Purpose Computing System	Embedded System
Contents	A system which is a combination of a generic hardware and a General Purpose Operating System for executing a variety of applications.	A system which is a combination of special purpose hardware and embedded OS for executing a specific set of applications.
OS	It contains a general purpose operating system (GPOS).	It may or not contain an operating system for functioning.
Alterations	Applications are alterable (programmable) by the user. (It is possible for the end user to re-install the OS and also add or remove user applications.)	The firmware of the embedded system is pre-programmed and it is non-alterable by the end-user.
Key factor	Performance is the key deciding factor in the selection of the system. Faster is better.	Application specific requirements (like performance, power requirements, memory usage, etc.) are key deciding factors.
Power Consumption	More	Less
Response Time	Not critical	Critical for some applications
Execution	Need not be deterministic	Deterministic for certain types of ES like 'Hard Real Time' systems.

Classification of Embedded Systems

- Some of the criteria used in the classification of embedded systems are:
  1. Based on generation
  2. Complexity and performance requirements
  3. Based on deterministic behaviour
  4. Based on triggering

Classification Based on Generation

- First Generation
- Second Generation
- Third Generation
- Fourth Generation
- Next Generation

Classification Based on Generation (continued)

- First Generation
  - Early embedded systems were built around 8-bit microprocessors like 8085 and Z80 and 4-bit microcontrollers.
  - Simple in hardware circuits with firmware developed in assembly code.
  - E.g.: Digital telephone keypads, stepper motor control units, etc.

Classification Based on Generation (continued)

- Second Generation
  - Embedded systems built around 16-bit microprocessors and 8-bit or 16-bit microcontrollers.
  - Instruction set were much more complex and powerful than the first generation.
  - Some of the second generation embedded systems contained embedded operating systems for their operation.
  - E.g.: Data acquisition systems, SCADA systems, etc.

Classification Based on Generation (continued)

- Third Generation
  - Embedded systems built around 32-bit microprocessors and 16-bit microcontrollers.
  - Application and domain specific processors/controllers like Digital Signal Processors (DSP) and Application Specific Integrated Circuits (ASICs) came into picture.
  - The instruction set of processors became more complex and powerful and the concept of instruction pipelining also evolved.
  - Dedicated embedded real time and general purpose operating systems entered into the embedded market.
  - Embedded systems spread its ground to areas like robotics, media, industrial process control, networking, etc.

Classification Based on Generation (continued)

- Fourth Generation
  - The advent of System on Chips (SoC), reconfigurable processors and multicore processors are bringing high performance, tight integration and miniaturisation into the embedded device market.
  - The SoC technique implements a total system on a chip by implementing different functionalities with a processor core on an integrated circuit.
  - They make use of high performance real time embedded operating systems for their functioning.
  - E.g.: Smart phone devices, Mobile Internet Devices (MIDs), etc.

Classification Based on Generation (continued)

- Next Generation
  - The processor and embedded market is highly dynamic and demanding.
  - The next generation embedded systems are expected to meet growing demands in the market.

Classification Based on Complexity and Performance

- Small-Scale Embedded Systems
- Medium-Scale Embedded Systems
- Large-Scale Embedded Systems

Classification Based on Complexity and Performance (continued)

- Small-Scale Embedded Systems
  - Simple in application needs and the performance requirements are not time critical.
  - E.g.: An electronic toy
  - Usually built around low performance and low cost 8-bit or 16-bit microprocessors/microcontrollers.
  - May or may not contain an operating system for its functioning.

Classification Based on Complexity and Performance (continued)

- Medium-Scale Embedded Systems
  - Slightly complex in hardware and firmware (software) requirements.
  - Usually built around medium performance, low cost 16-bit or 32-bit microprocessors/microcontrollers or digital signal processors.
  - Usually contain an embedded operating system (either general purpose or real time operating system) for functioning.

- Large-Scale Embedded Systems
  - Highly complex in hardware and firmware (software) requirements.
  - They are employed in mission critical applications demanding high performance.
  - Usually built around high performance 32-bit or 64-bit RISC processors/controllers or Reconfigurable System on Chip (RSoC) or multi-core processors and programmable logic devices.
  - May contain multiple processors/controllers and co-units/hardware accelerators for offloading the processing requirements from the main processor of the system.
  - Decoding/encoding of media, cryptographic function implementation, etc. are examples of processing requirements which can be implemented using a co-processor/hardware accelerator.
  - Usually contain a high performance real time operating system (RTOS) for task scheduling, prioritization and management.

Classification Based on Deterministic Behaviour

- Applicable for 'Real Time' systems.
- The application/task execution behaviour can be either deterministic or non-deterministic.
- Based on the execution behaviour, real time embedded systems are classified into Hard Real Time and Soft Real Time systems.

Classification Based on Triggering

- Embedded systems which are 'Reactive' in nature (like process control systems in industrial control applications) can be classified based on the trigger.
- Reactive systems can be either event-triggered or time-triggered.

Major Application Areas of Embedded Systems

1. Consumer electronics: Camcorders, cameras, etc.
2. Household appliances: Television, DVD players, washing machine, refrigerator, microwave oven, etc.
3. Home automation and security systems: Air conditioners, sprinklers, intruder detection alarms, closed circuit television cameras, fire alarms, etc.
4. Automotive industry: Anti-lock braking systems (ABS), engine control, ignition systems, automatic navigation systems, etc.
5. Telecom: Cellular telephones, telephone switches, handset multimedia applications, etc.

Major Application Areas of Embedded Systems

6. **Computer peripherals:** Printers, scanners, fax machines, etc.
7. **Computer Networking systems:** Network routers, switches, hubs, firewalls, etc.
8. **Healthcare:** Different kinds of scanners, EEG, ECG machines etc.
9. **Measurement & Instrumentation:** Digital multi meters, digital CROs, logic analyzers PLC systems, etc.
10. **Banking & Retail:** Automatic teller machines (ATM) and currency counters, point of sales (POS).
11. **Readers:** Barcode, smart card readers, handheld devices, etc.

Purpose of Embedded Systems

Each embedded system is designed to serve the purpose of any one or a combination of the following tasks:

i. Data collection/Storage/Representation
ii. Data Communication
iii. Data (signal) processing
iv. Monitoring
v. Control
vi. Application specific user interface

(i) Data Collection/Storage/Representation

- Embedded systems designed for the purpose of data collection performs acquisition of data from the external world.
- Data collection is usually done for storage, analysis, manipulation and transmission.
- Data can be either analog (continuous) or digital (discrete).
- Embedded systems with analog data capturing techniques collect data directly in the form of analog signal
- If the data is digital, it can be directly captured without any additional interface by digital embedded systems.
- The collected data may be stored directly in the system or may be transmitted to some other system, or it may be processed by the system or it may be deleted instantly after giving a meaningful representation.

(ii) Data Communication

- Embedded data communication systems are deployed in applications from complex satellite communication systems to simple home networking systems.
- The data collected by an embedded terminal is transmitted either by a wire-line medium or by a wire-less medium.
- The data collecting embedded terminal itself can incorporate data communication units like Wireless modules (Bluetooth, ZigBee, Wi-Fi, EDGE, GPRS, etc.) or wire-line modules (RS-232C, USB, TCP/IP, PS2,etc).
- Certain embedded systems act as a dedicated transmission unit between the sending and receiving terminals, offering sophisticated functionalities like **data packetizing**, **encrypting** and **decrypting** (Network hubs, routers, switches, etc.).

(iii) Data (Signal) Processing

- The data (voice, image, video, electrical signals and other measurable quantities) collected by embedded systems may be used for various kinds of data processing.
- Embedded systems with signal processing functionalities are employed in applications demanding signal processing like **speech coding**, **synthesis**, **audio video codec**, **transmission applications**, etc.
- A digital hearing aid is a typical example of an embedded system employing data processing.

(iv) Monitoring

- Almost all embedded products coming under **the medical domain** are with monitoring functions only.
- They are used **for determining the state of some variables using input sensors**.
- They cannot impose control over variables.
- Eg: Electrocardiogram (ECG) machine for monitoring the heartbeat of a patient, measuring instruments like digital CRO, digital multimeters, logic analyzers., etc. used in control & instrumentation applications.

(v) Control

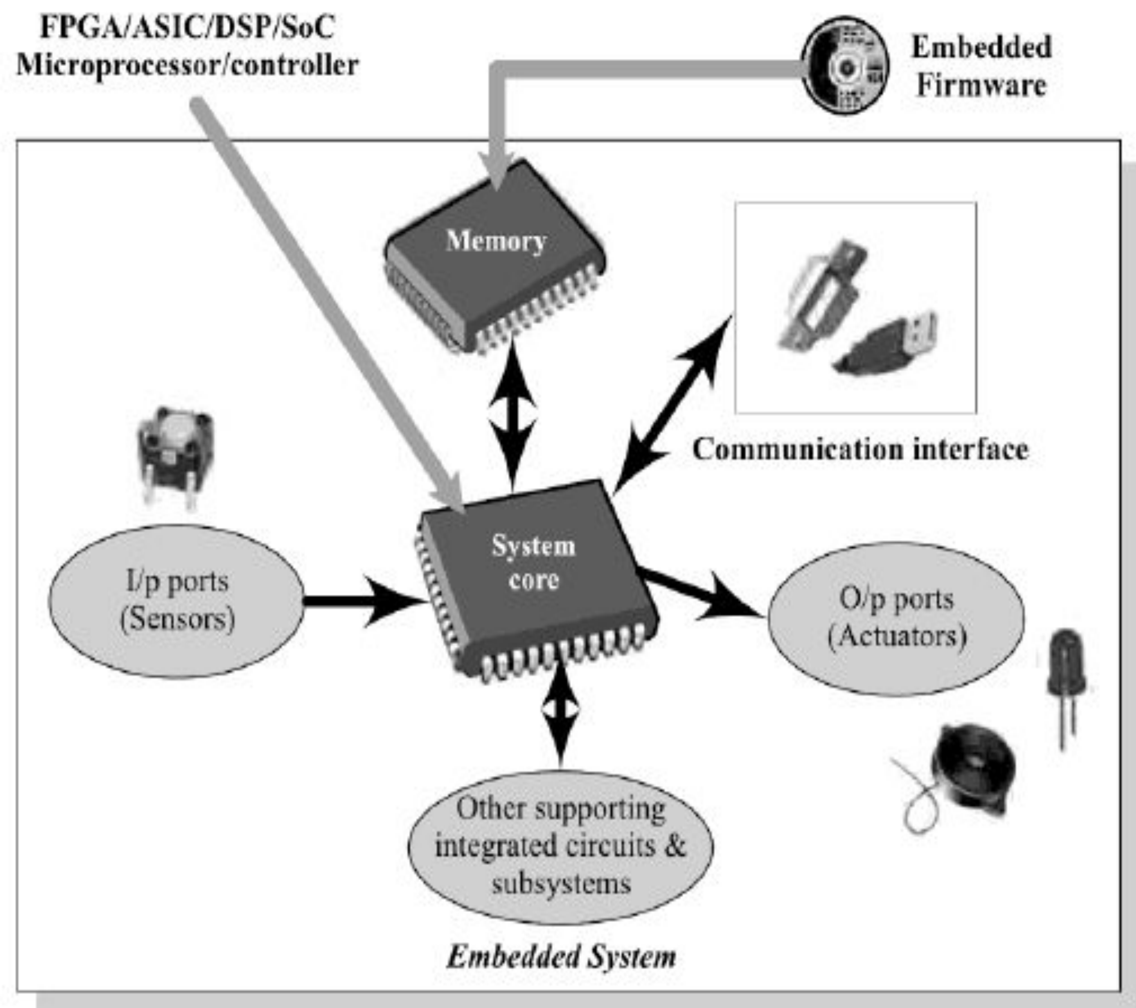
- Embedded systems with control functionalities impose control over some variables according to the changes in input variables.
- A system with control functionality contains both sensors and actuators.
- Air conditioner system used in our home to control the room temperature to a specified limit is a typical example for embedded system for control purpose.

(vi) Applications specific user interface

- Buttons, switches, keypad, lights, speakers, display units, etc. are application-specific user interfaces.

- Mobile phone is an example of application specific user interface. In mobile phone, the user interface is provided through the keypad, graphic LCD module, system speaker, vibration alert, etc.

A Typical Embedded System



A Typical Embedded System (continued)

- It contains a single chip controller, which acts as the master brain of the system.
- The controller can be
  - ✓ A microprocessor or
  - ✓ A microcontroller or
  - ✓ A Field Programmable Gate Array (FPGA) device or
  - ✓ A Digital Signal Processor (DSP) or
  - ✓ An Application Specific Integrated Circuit (ASIC)/Application Specific Standard Product (ASSP)

A Typical Embedded System (continued)

- An embedded system can be viewed as a reactive system.
- The control is achieved by processing the information coming from the sensors and user interfaces, and controlling some actuators that regulate the physical variable.
- Key boards, push button switches, etc. are examples for common user interface input devices.
- LEDs, liquid crystal displays, piezoelectric buzzers, etc. are examples for common user interface output devices for a typical embedded system.

A Typical Embedded System (continued)

- The memory of the system is responsible for holding the control algorithm and other important configuration details.
- For most of embedded systems, the memory for storing the algorithm or configuration data is of fixed type, which is a kind of Read Only Memory (ROM).
  - It is not available for the end user for modifications
  - The memory is protected from unwanted user interaction by implementing some kind of memory protection mechanism.
  - The most common types of memories used in embedded systems for control algorithm storage are OTP, PROM, UVEPROM, EEPROM and FLASH.
- Sometimes the system requires temporary memory for performing arithmetic operations or control algorithm execution and this type of memory is known as "working memory".
  - Random Access Memory (RAM) is used in most of the systems as the working memory.
  - Various types of RAM like SRAM, DRAM and NVRAM are used for this purpose.

A Typical Embedded System (continued)

- Apart from these, communication interface is essential for communicating with various subsystems of the embedded system and with the external world.
- The communication interfaces may be used to achieve onboard (I2C, SPI, UART, parallel bus interface, etc.) or external communication (wireless interfaces like Infrared, Bluetooth, Wi-Fi, etc.)

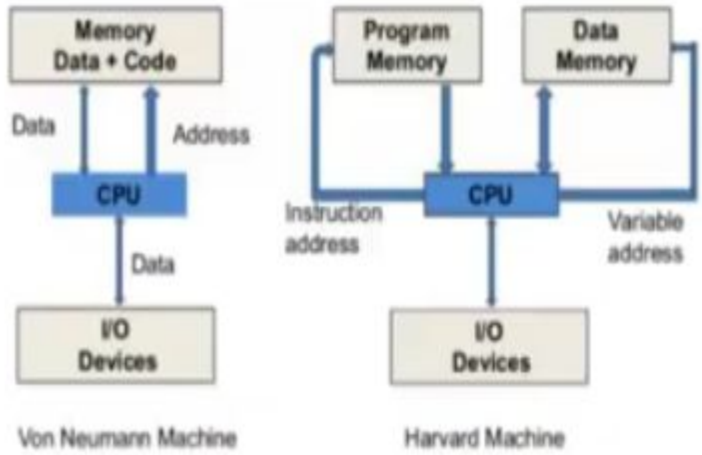
Core of Embedded Systems

- General Purpose and Domain Specific Processors
    i. Microprocessors
    ii. **Microcontrollers**
    iii. Digital Signal Processors
- Application Specific Integrated Circuits (ASICs)
- Programmable Logic Devices (PLDs)
- Commercial off-the-shelf Components (COTS)

RISC vs CISC

RISC	CISC
Lesser number of instructions	Greater number of Instructions
Instruction pipelining and increased execution speed	Generally **no instruction pipelining feature**
Orthogonal instruction set	**Non-orthogonal** instruction set
Operations are performed **on registers only**, the only memory operations are load and store.	Operations are performed **on registers or memory** depending on the instruction.
A large number of registers are available.	**Limited** number of general purpose registers.
Programmer **needs to write more code to execute a task** since **the instructions are simpler ones.**	Instructions are like macros in **C language**. A programmer can achieve the desired functionality with a single instruction which in turn provides the effect of using **more simpler single instructions** in RISC.
Single, fixed length instructions	**Variable length instructions**
Less silicon usage and pin count	**More** silicon usage since more additional decoder logic is required to implement the complex instruction decoding.
With **Harvard Architecture**	**Harvard or Von-Neumann Architecture**

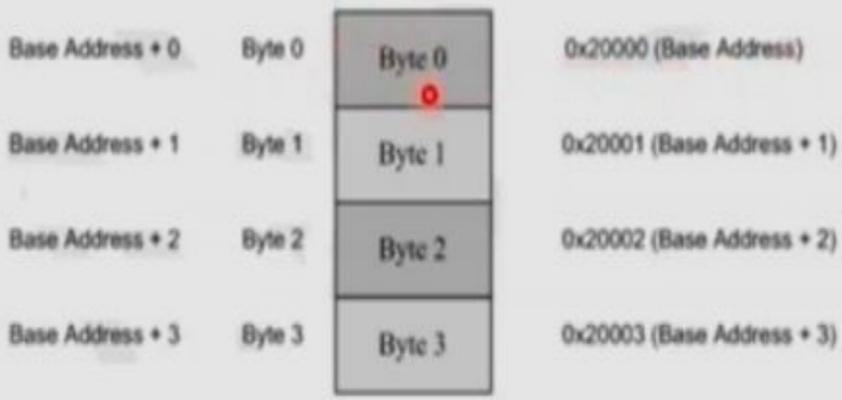
Harvard vs Von-Neumann Architecture



Harvard	Von-Neumann
Separate buses for instruction and data fetching	Single shard bus for instruction and data fetching
Easier to pipeline, so high performance can be achieved	Comparatively, low performance
Comparatively high cost	Cheaper
No chances for accidental corruption of program memory	Memory in same chip, chances of accidental corruption of program memory.

Big-Endian vs. Little-Endian Processors/Controllers

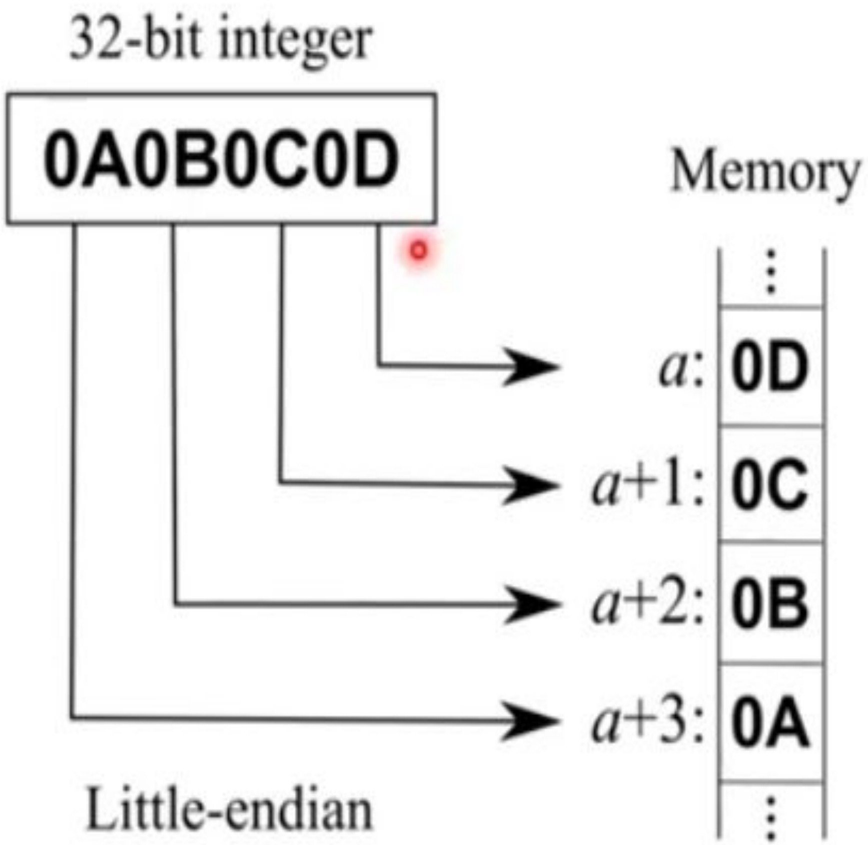
- **Endianness** specifies the order in which a sequence of bytes are stored in computer memory.



Little-endian Operation

- **Little-endian** is an order in which the "little end"/ the lower-order byte of the data (least significant value in the sequence) is stored in memory at the lowest address. (The little end comes first.)

Little-endian example



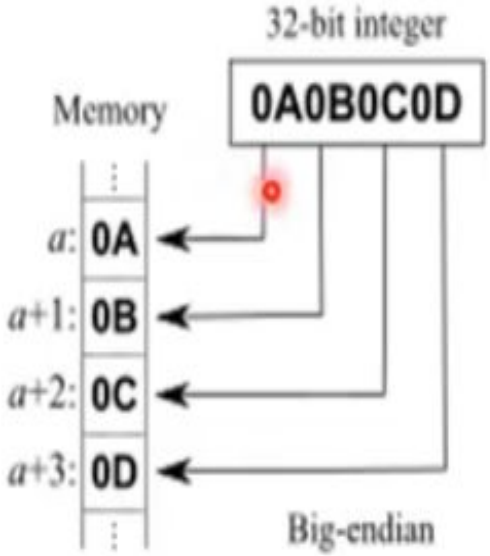
Big-Endian vs. Little-Endian Processors/Controllers

- **Endianness** specifies the order in which a sequence of bytes are stored in computer memory.

Big-endian Operation

- **Big-endian** is an order in which the "big end" / the higher-order byte of the data (most significant value in sequence) is stored in memory at the lowest address. (The big end comes first.)

Big-endian example




MEMORY

- **Memory** is an **important part** of a processor/controller based embedded systems.
- Some of the processors/controllers contain **built in memory** and this memory is referred as **on-chip memory**.
- Others do not contain any memory inside the chip and requires **external memory to be connected** with the controller/processor to store the control algorithm. It is called **off-chip memory**.

Program Storage Memory (ROM)

- Masked ROM (MROM)
- Programmable Read Only Memory (PROM)/ (OTP)
- Erasable Programmable Read Only Memory (EPROM)
- Electrically Erasable Programmable Read Only Memory (EEPROM)
- FLASH

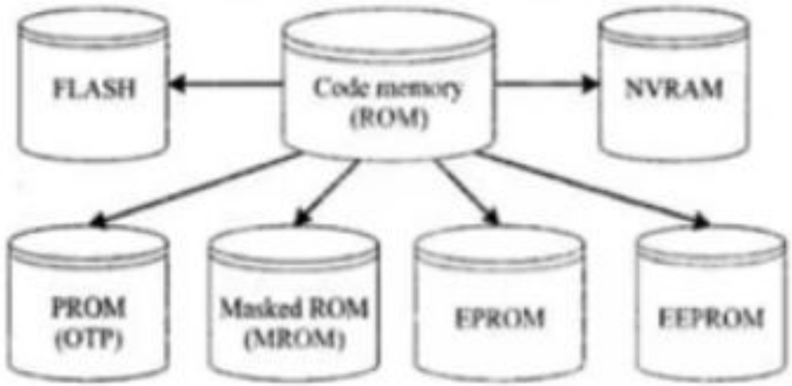
Read-Write Memory/Random Access Memory (RAM)

- Static RAM (SRAM)
- Dynamic RAM (DRAM)
- NVRAM

Types of memory

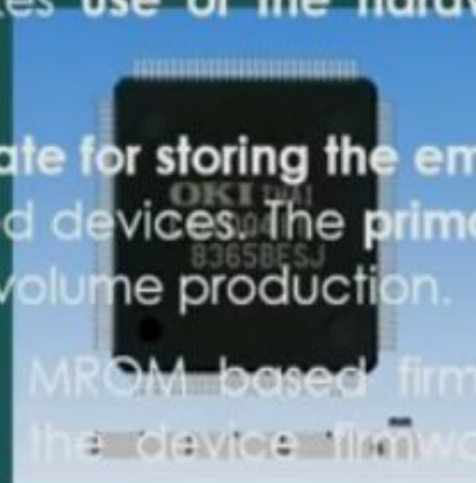
Program Storage Memory (ROM)

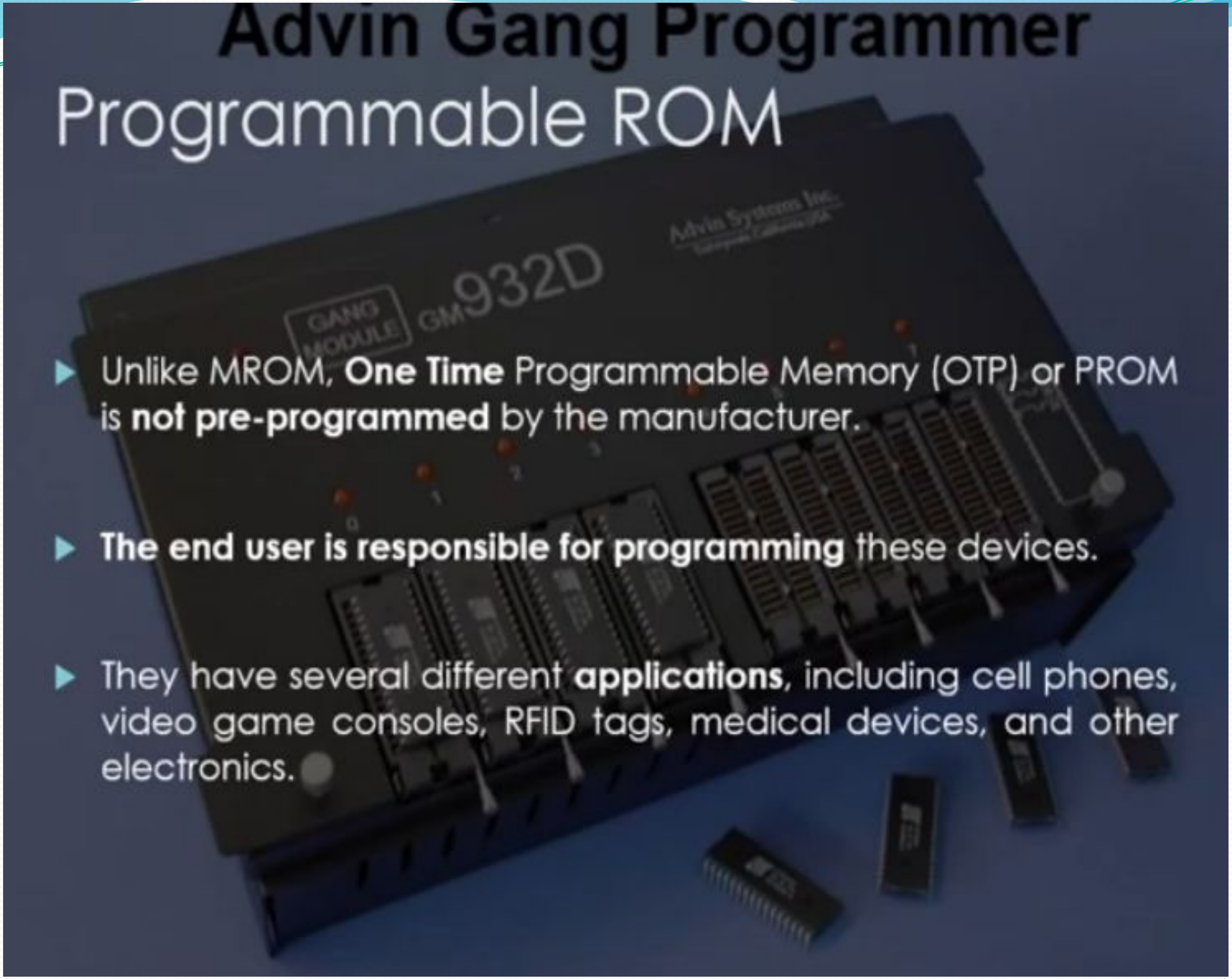
- **The program memory or code storage memory** of an embedded system **stores the program instructions,** and it can be classified into **different types** as per the block diagram representation.
- The code memory **retains its contents** even **after the power to it is turned off**. It is generally known as **nonvolatile** storage memory.



Masked ROM

- It is a **static ROM** which comes **programmed** into an integrated circuit by its manufacturer.
- Masked ROM makes **use of the hardwired technology** for storing data.
- It is a **good candidate for storing the embedded firmware** for low-cost embedded devices. The **primary advantage** of this is low cost for high volume production.
- The **limitation** with MROM based firmware storage is the inability to modify the device firmware against firmware upgrades.
- They are **used** in network operating systems, server operating systems, storing of fonts for laser printers, sound data in electronic musical instruments.





Programmable ROM

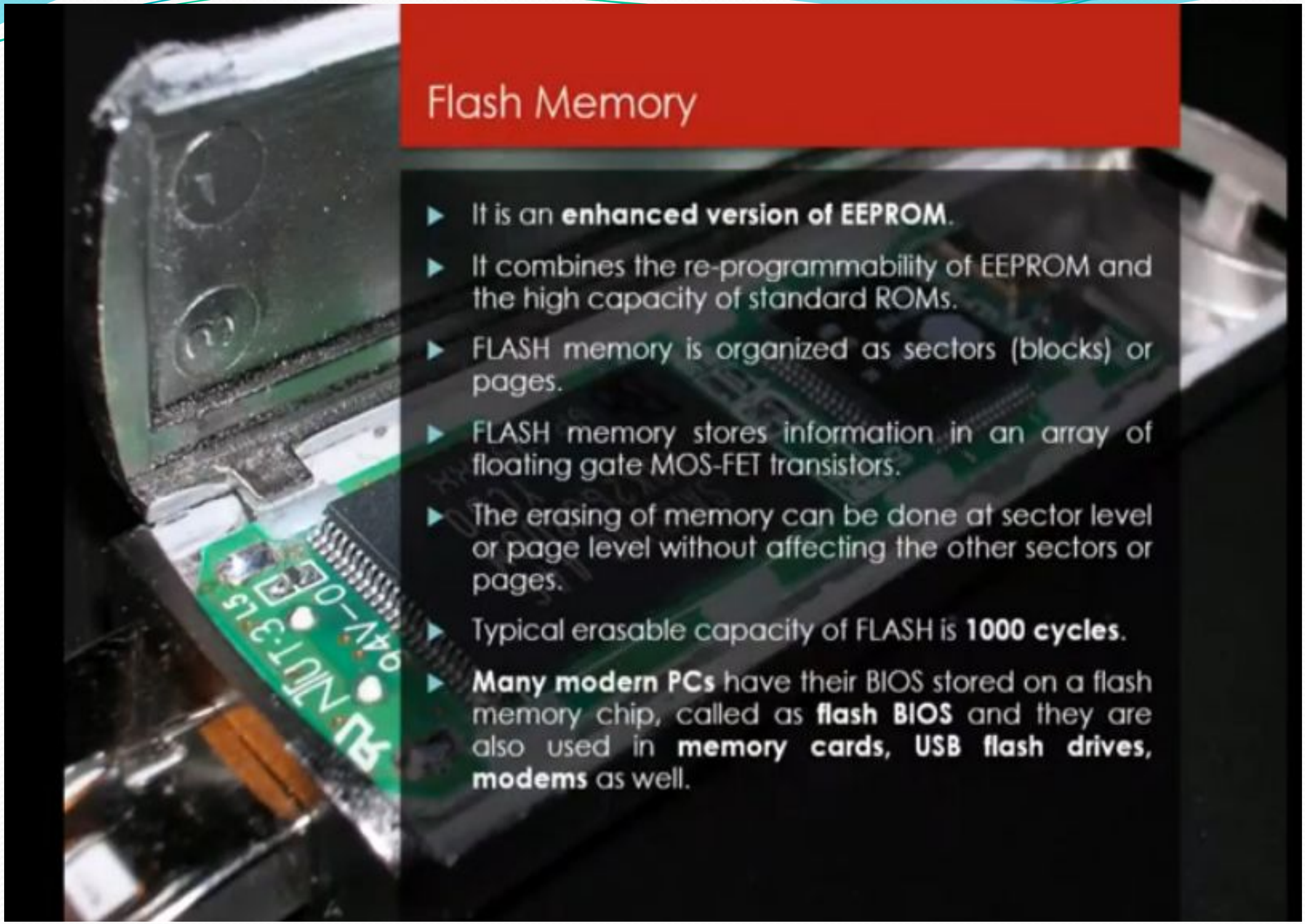
- Unlike MROM, **One Time** Programmable Memory (OTP) or PROM is **not pre-programmed** by the manufacturer.
- **The end user is responsible for programming** these devices.
- They have several different **applications**, including cell phones, video game consoles, RFID tags, medical devices, and other electronics.

Erasable PROM

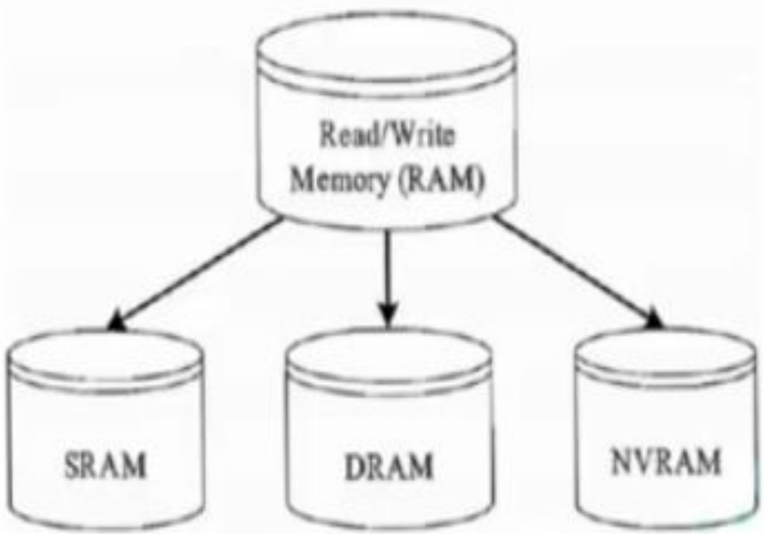
- EPROM gives the **flexibility to re-program** the same chip.
- EPROM stores the bit information **by charging the floating gate** of an FET and contains **a quartz crystal window** for erasing the stored information.
- Even though the EPROM chip is flexible in terms of reprogrammability, it needs to be taken out of the circuit board and put in a UV eraser device for 20 to 30 minutes.

Electrically EPROM

- The information contained in the EEPROM memory **can be altered** by using electrical signal at the register/Byte level.
- They **can be erased and reprogrammed** in-circuit.
- These chips include **a chip erase mode,** and, in this mode, they **can be erased in a few milliseconds**.
- It provides **greater flexibility** for system design. The only **limitation** is their **capacity is limited** when compared with the standard ROM (a few kilobytes).
- It is **used** for storing the computer system BIOS.


Flash Memory
It is an **enhanced version of EEPROM**.
It combines the re-programmability of EEPROM and the high capacity of standard ROMs.
FLASH memory is organized as sectors (blocks) or pages.
FLASH memory stores information in an array of floating gate MOS-FET transistors.
The erasing of memory can be done at sector level or page level without affecting the other sectors or pages.
Typical erasable capacity of FLASH is **1000 cycles**.
Many modern PCs have their BIOS stored on a flash memory chip, called as **flash BIOS** and they are also used in **memory cards, USB flash drives, modems** as well.

Read-Write Memory/Random Access Memory (RAM)

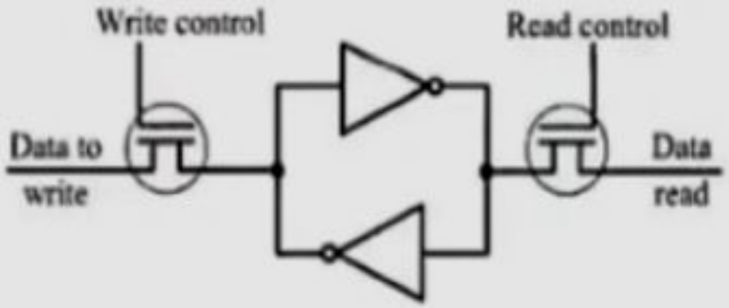


- The **Random-Access Memory (RAM)** is the **data memory or working memory** of the controller/processor.
- Controller/processor can **read** from it and **write** to it.
- RAM is **volatile**, meaning when the power is turned off, all the contents are destroyed.
- RAM generally falls into **three categories**: Static RAM (SRAM), dynamic RAM (DRAM) and nonvolatile RAM (NVRAM).

Visualization of SRAM cell

- The implementation in its simpler form can be visualized as **two-cross coupled inverters** with read/write control through transistors.

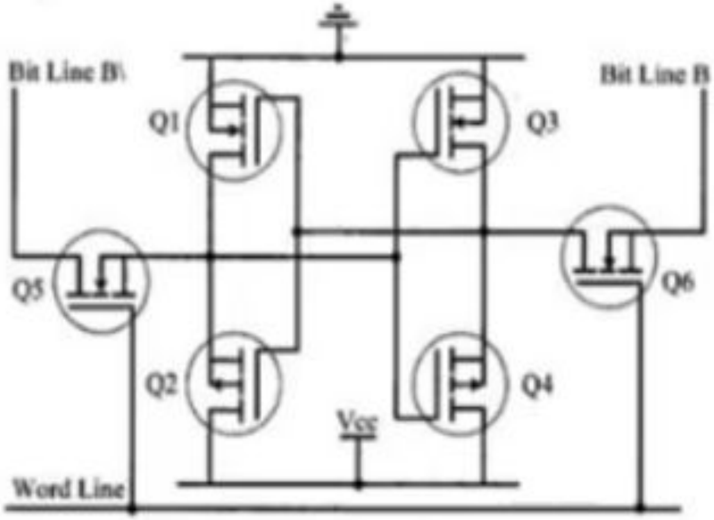
- The **four transistors in the middle** form the cross-coupled inverters.



- The major limitations of SRAM are **low capacity** and **high cost**.

Static RAM (SRAM)

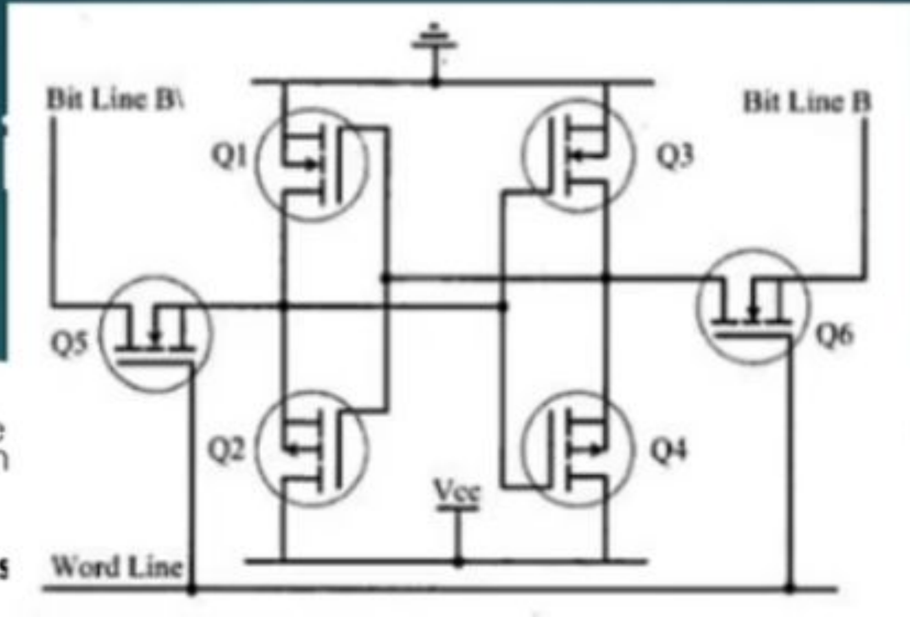
- SRAM stores data in **the form of voltage**.
- They are made up of **flip-flops**.
- A flip-flop for a memory cell takes **four or six transistors** (or 6 MOSFETs) along with some wiring, four of the transistors are used **for building the latch (flip-flop) part of the memory cell** and two **for controlling the access**.
- SRAM is **fast in operation** due to its resistive networking and switching capabilities.

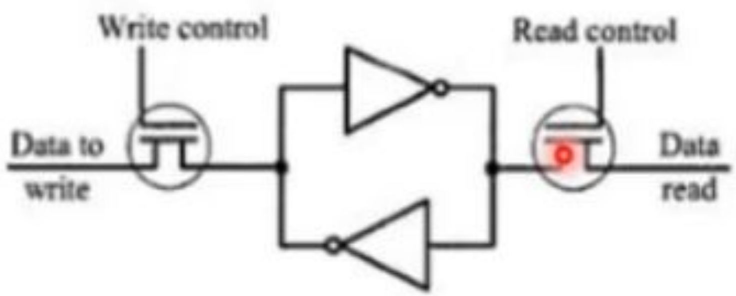


Visuali

- The impleme ... ss coupled **inverters** with

- The **four trans**





- The major limitations of SRAM are **low capacity** and **high cost**.

Dynamic RAM (DRAM)

- DRAM stores data in **the form of charge**. They are made up of **MOS transistor gates**.
- The advantages of DRAM are its **high density** and **low cost** compared to SRAM.
- The disadvantage is that since the information is stored as charge it gets **leaked off with time** and to **prevent this they need to be refreshed periodically**.
- Special circuits called DRAM controllers are used for the **refreshing operation**.
- The refresh operation is done periodically **in milliseconds** interval.
- The MOSFET acts as the gate for the incoming and outgoing data whereas the capacitor acts as the bit storage unit.

SRAM and DRAM technology

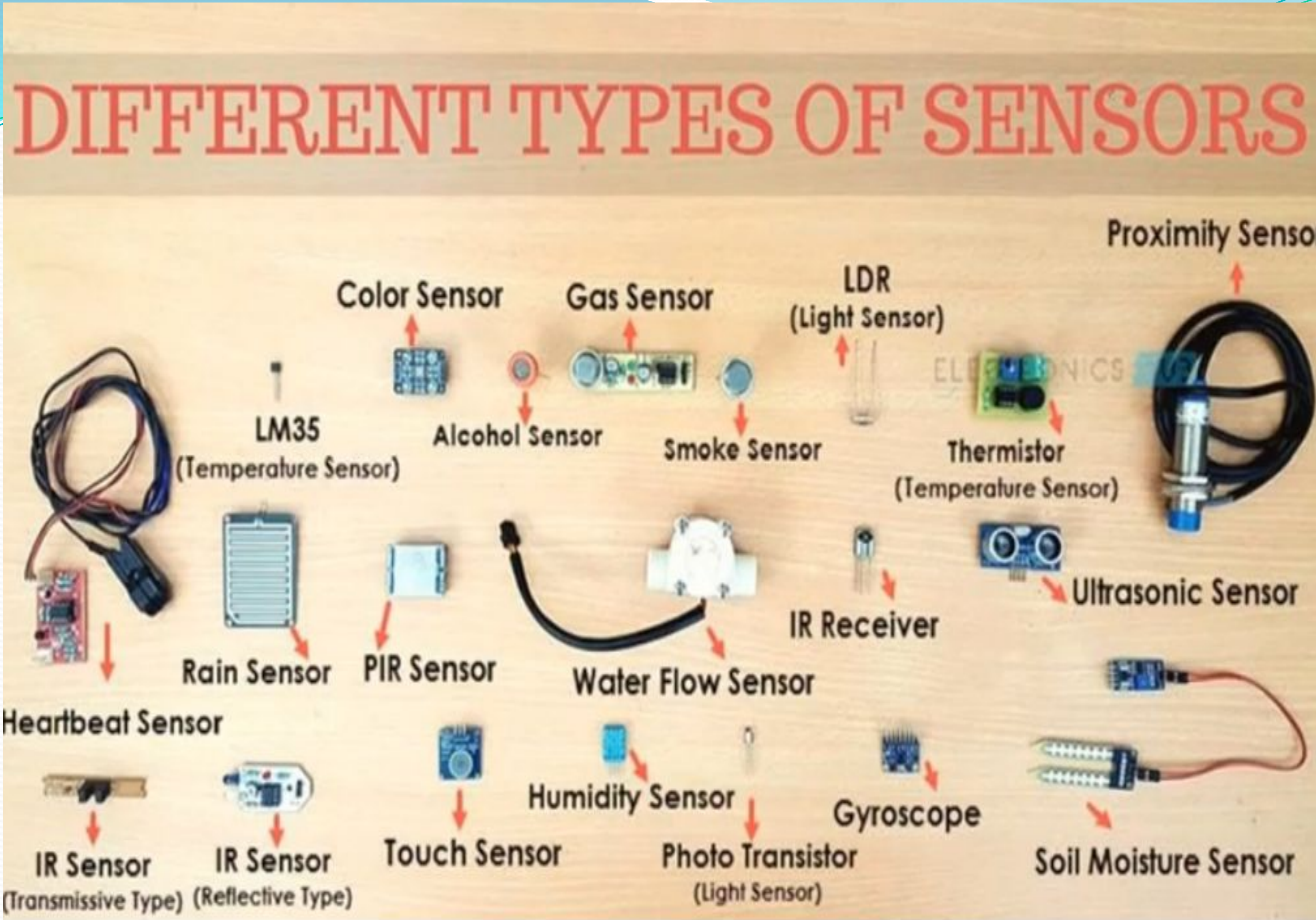
SRAM Cell	DRAM Cell
Made up of 6 CMOS transistors (MOSFET)	Made up of a MOSFET and a capacitor
Doesn't require refreshing	Requires refreshing
Low capacity (Less dense)	High capacity (Highly dense)
More expensive	Less expensive
Fast in operation. Typical access time is 10 ns.	Slow in operation due to refresh requirements. Typical access time is 60 ns. Write operation is faster than read operation.

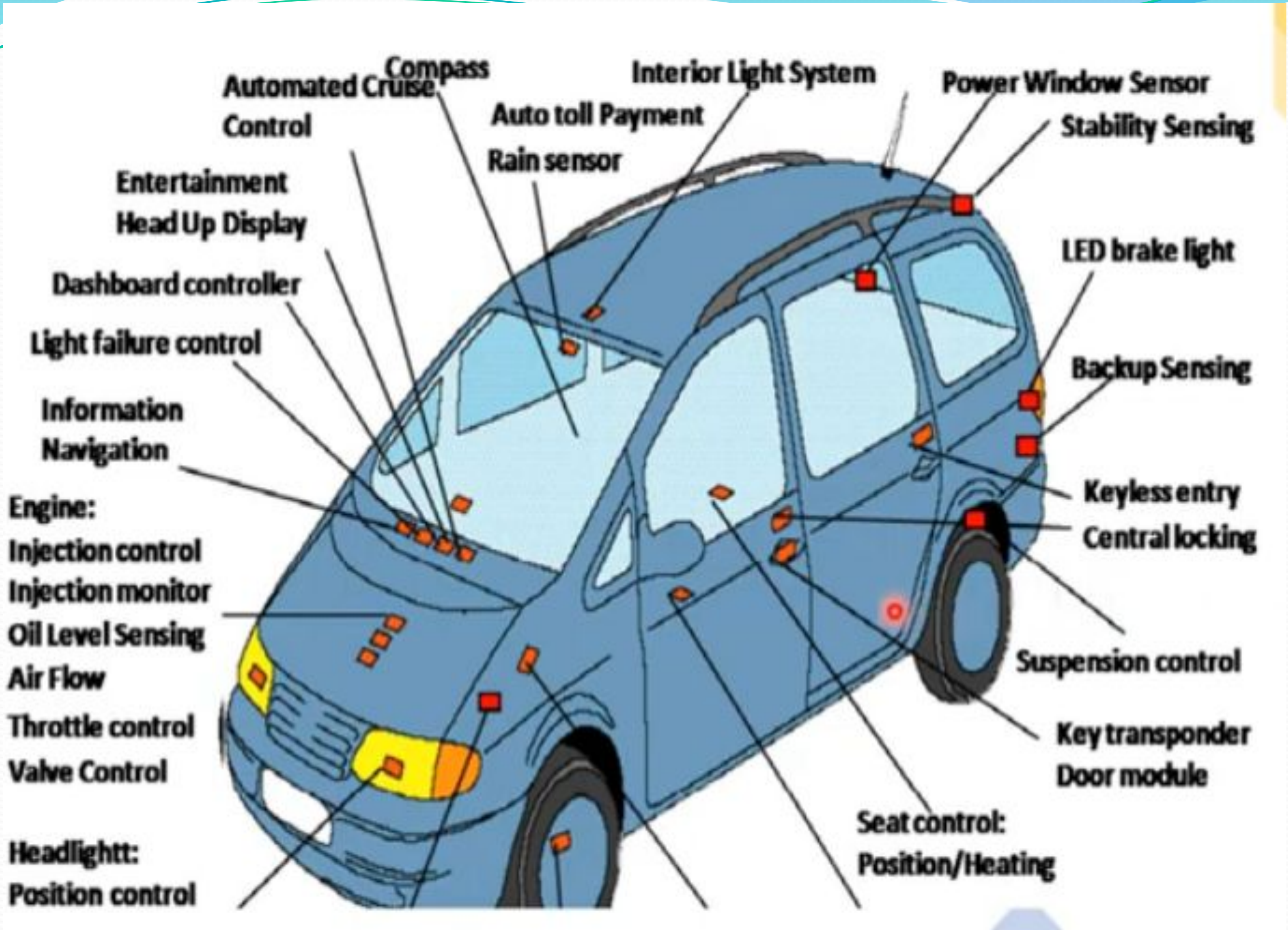
Non-volatile RAM

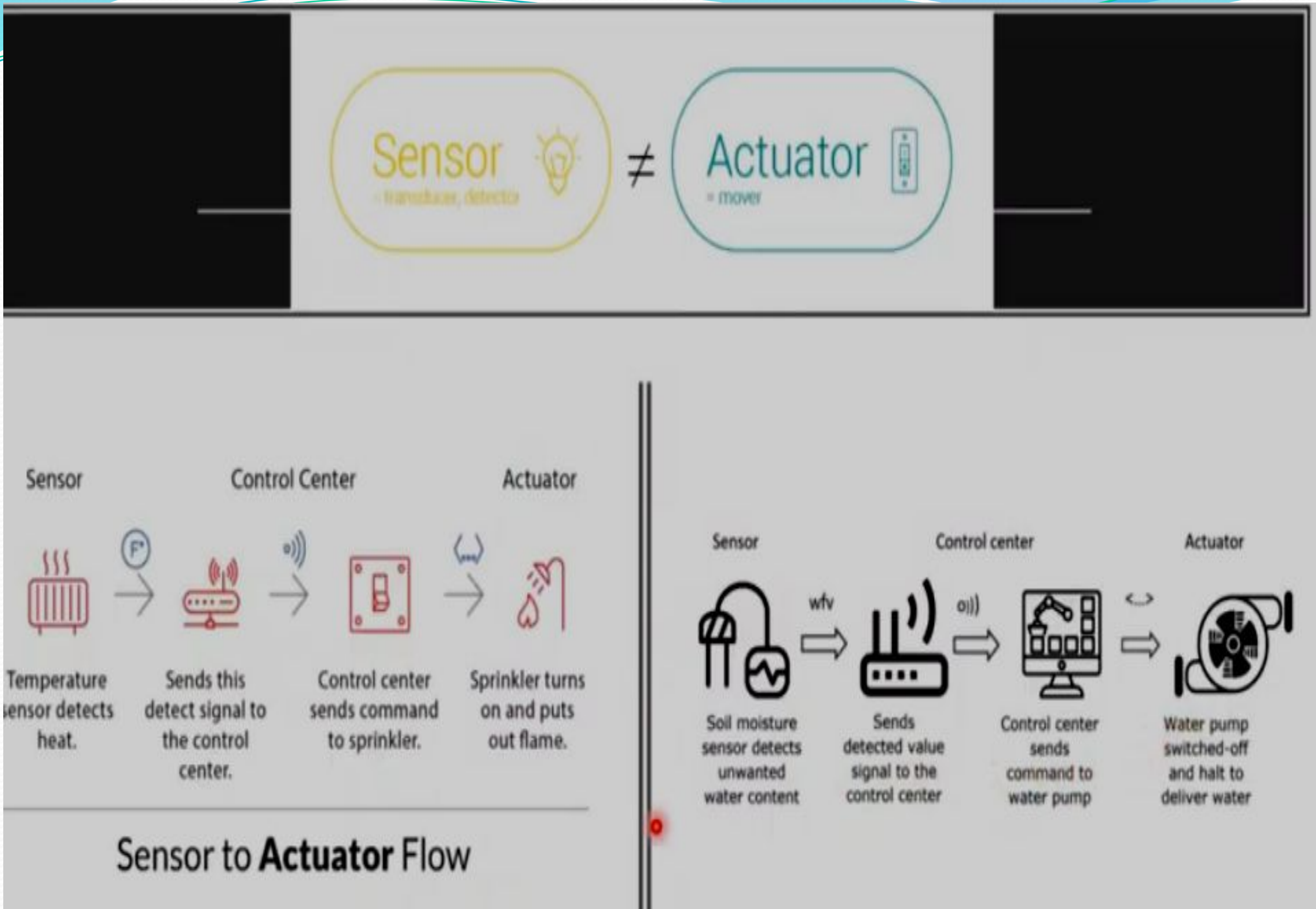
- Non-volatile RAM is a **random-access memory with battery backup**.
- It contains **static RAM based memory** and **a minute battery** for providing supply to the memory in the absence of external power supply.
- The memory and battery are **packed together** in a single package.
- NVRAM is used for the **non-volatile storage of results of operations**.
- The **life span** of NVRAM is expected to be around 10 years.
- **DS1744 from Maxim/Dallas** is an example for 32 KB NVRAM.

Sensors and Actuators

- *A **sensor** is a transducer device that converts energy from one form to another for any measurement or control purpose.*
- The changes in system environment or variables are detected by the sensors connected to the input port of the embedded system.
- ***Actuator** is a form of transducer device (mechanical or electrical) which converts signals to corresponding physical action (motion).*
- Actuator acts as an output device.
- If the embedded system is designed for any controlling purpose, the system will produce some changes in the controlling variable to bring the controlled variable to the desired value.
- It is achieved through an actuator connected to the output port of the embedded system.

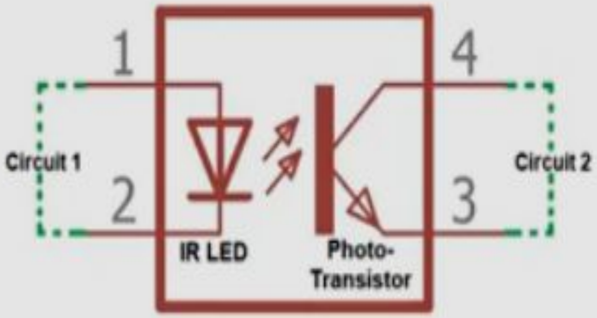

DIFFERENT TYPES OF SENSORS
Proximity Sensor
Color Sensor
Gas Sensor
LDR
(Light Sensor)
LM35
(Temperature Sensor)
Alcohol Sensor
Smoke Sensor
Thermistor
(Temperature Sensor)
Ultrasonic Sensor
IR Receiver
Rain Sensor
PIR Sensor
Water Flow Sensor
Heartbeat Sensor
Humidity Sensor
Gyroscope
Touch Sensor
Photo Transistor
(Light Sensor)
Soil Moisture Sensor
IR Sensor
(Transmissive Type)
IR Sensor
(Reflective Type)

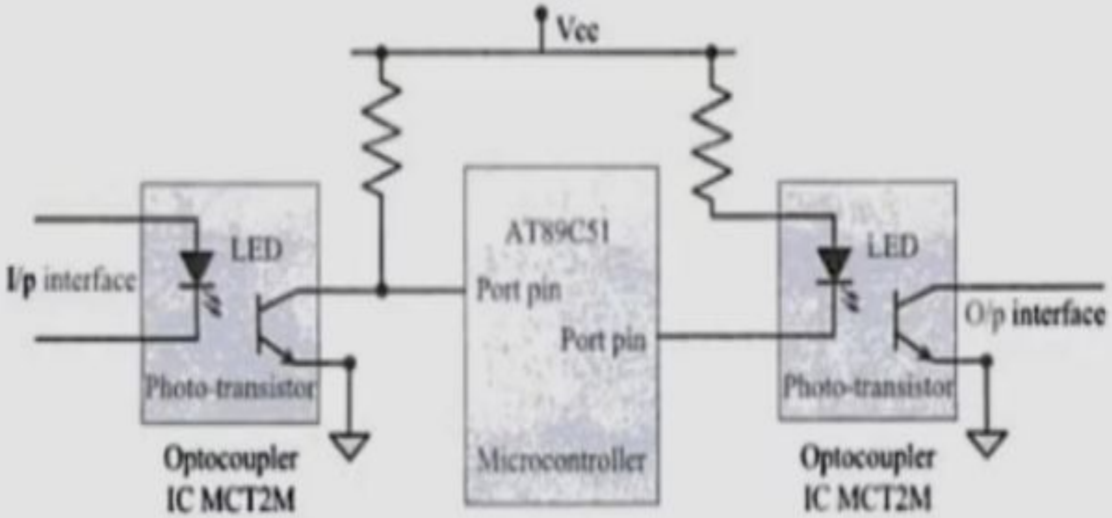

Compass
Automated Cruise Control
Interior Light System
Power Window Sensor
Auto toll Payment
Stability Sensing
Rain sensor
Entertainment
Head Up Display
LED brake light
Dashboard controller
Light failure control
Backup Sensing
Information
Navigation
Keyless entry
Central locking
Engine:
Injection control
Injection monitor
Oil Level Sensing
Air Flow
Throttle control
Valve Control
Suspension control
Key transponder
Door module
Seat control:
Position/Heating
Headlightt:
Position control


Sensor
Actuator
= mover
Sensor
Control Center
Actuator
Temperature ensor detects heat.
Sends this detect signal to the control center.
Control center sends command to sprinkler.
Sprinkler turns on and puts out flame.
Sensor to Actuator Flow
Sensor
Control center
Actuator
Soil moisture sensor detects unwanted water content
Sends detected value signal to the control center
Control center sends command to water pump
Water pump switched-off and halt to deliver water

Optocoupler

- Optocoupler is a solid-state device to isolate two parts of a circuit.
- Optocoupler combines an LED and a photo-transistor in a single housing (package).





Communication Interface

Communication Interface

- Communication interface is essential for communicating with various subsystems of the embedded system and with the external world.
- For an embedded product, the communication interface can be viewed in two different perspectives:
  - Onboard Communication Interface (Device/board level communication interface)
    - E.g.: Serial interfaces like I2C, SPI, UART, 1-Wire, etc and parallel bus interface.
  - External Communication Interface (Product level communication interface)
    - E.g.: Wireless interfaces like Infrared (IR), Bluetooth (BT), Wireless LAN (Wi-Fi), Radio Frequency waves (RF), GPRS, etc. and wired interfaces like RS-232C/RS-422/RS-485, USB, Ethernet IEEE 1394 port, Parallel port, CF-II interface, SDIO, PCMCIA, etc.

Onboard Communication Interfaces

- An embedded system is a combination of different types of components (chips/devices) arranged on a printed circuit board (PCB).
- Onboard Communication Interface refers to the different communication channels/buses for interconnecting the various integrated circuits and other peripherals within the embedded system.
- E.g.: Serial interfaces like I2C, SPI, UART, 1-Wire, etc and parallel bus interface

Onboard Communication Interfaces

Onboard Communication Interface refers to the different communication channels/buses for interconnecting the various integrated circuits and other peripherals within the embedded system.

The **various interfaces** for onboard communication are:

i. Inter Integrated Circuit (I2C) Bus

ii. Serial Peripheral Interface (SPI) Bus

iii. Universal Asynchronous Receiver Transmitter (UART)

iv. 1-Wire Interface

v. Parallel Interface

Inter Integrated Circuit (I2C) Bus

- The Inter Integrated Circuit Bus (I2C or I^2C Pronounced 'I square C') is a synchronous bi-directional half duplex two wire serial interface bus.
  - (Half duplex - one-directional communication at a given point of time)
- The concept of I2C bus was developed by Philips Semiconductors in the early 1980s.
- The original intention of I2C was to provide an easy way of connection between a microprocessor/microcontroller system and the peripheral chips in television sets.
- The I2C bus comprise of two bus lines:
  - Serial Clock (SCL line) – responsible for generating synchronisation clock pulses
  - Serial Data (SDA line) – responsible for transmitting the serial data across devices

Inter Integrated Circuit (I2C) Bus (continued)

- The following bus interface diagram illustrates the connection of master and slave devices on the I2C bus.

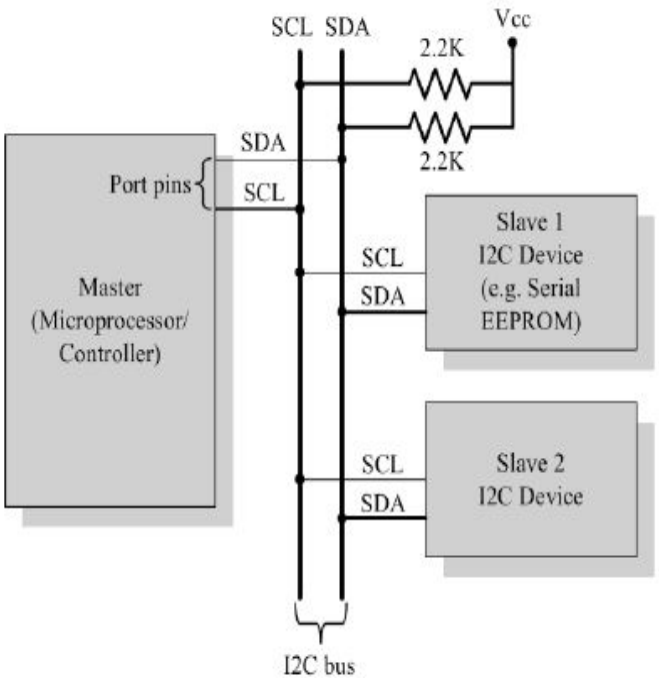


Fig: I2C Bus Interfacing

- I2C bus is a shared bus system to which many number of I2C devices can be connected.
- Devices connected to the I2C bus can act as either 'Master' or 'Slave'.
  - The 'Master' device is responsible for controlling the communication by initiating/terminating data transfer, sending data and generating necessary synchronisation clock pulses.
  - 'Slave' devices wait for the commands from the master and respond upon receiving the commands.
- 'Master' and 'Slave' devices can act as either transmitter or receiver.
- Regardless whether a master is acting as transmitter or receiver, the synchronisation clock signal is generated by the 'Master' device only.
- I2C supports multi masters on the same bus.

Inter Integrated Circuit (I2C) Bus

1. Master pulls SCL to HIGH
2. Master pulls SDA to LOW ('Start' condition)
3. Master sends address of Slave to be communicated over SDA line
4. Clock pulses generated at SCL to synchronize
5. Master sends Read or Write bit
6. Selected Slave sends acknowledge bit over SDA line
7. Master sends/receives 8-bit data over the SDA line
8. Acknowledgement by Master/Slave
9. Master terminates operation by pulling SDA line HIGH

- I2C bus supports three different data rates:
  - Standard mode (Data rate up to 100kbits/sec (100 kbps))
  - Fast mode (Data rate up to 400kbits/sec (400 kbps))
  - High speed mode (Data rate up to 3.4Mbits/sec (3.4 Mbps))

Serial Peripheral Interface (SPI) Bus

- The Serial Peripheral Interface Bus (SPI) is a synchronous bi-directional full duplex four-wire serial interface bus.
- The concept of SPI was introduced by Motorola.
- SPI is a single master multi-slave system.
  - There can be more than one masters, but only one master device can be active at any given point of time.
- SPI requires four signal lines for communication. They are:
  - Master Out Slave In (MOSI) – Signal line carrying the data from master to slave device. It is also known as Slave Input/Slave Data In (SI/SDI)
  - Master In Slave Out (MISO) – Signal line carrying the data from slave to master device. It is also known as Slave Output (SO/SDO)
  - Serial Clock (SCLK) – Signal line carrying the clock signals
  - Slave Select (SS) – Signal line for slave device select. It is an active low signal

Serial Peripheral Interface (SPI) Bus (continued)

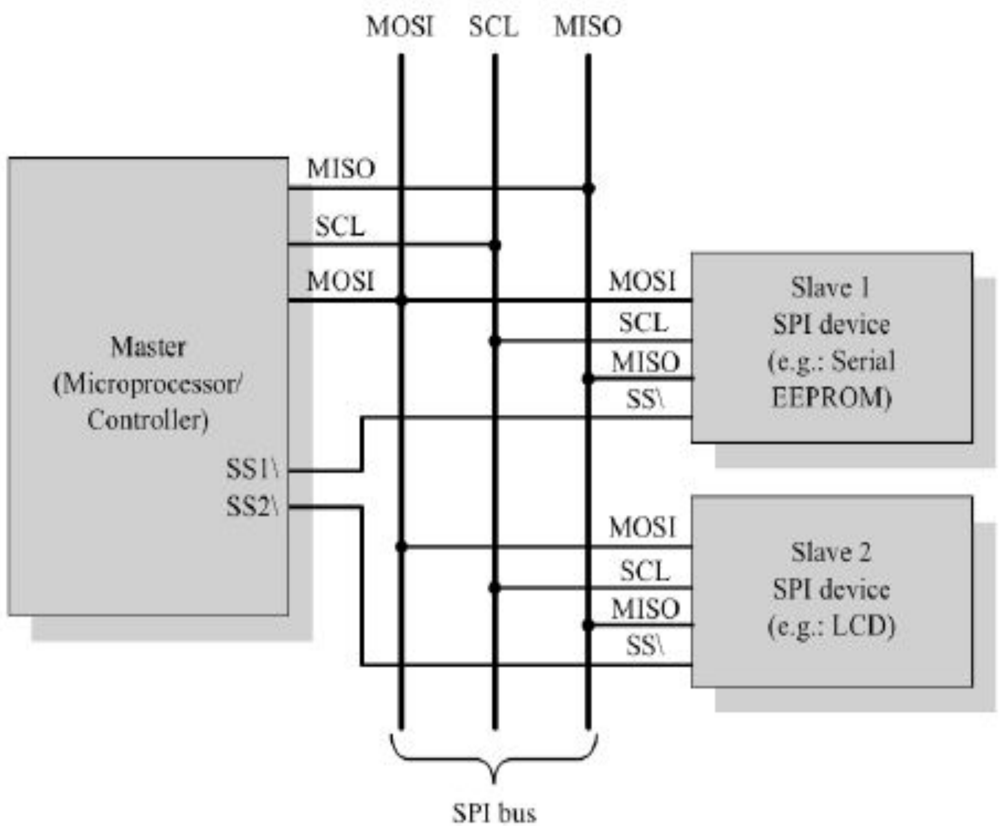


Fig: SPI Bus Interfacing

Serial Peripheral Interface (SPI) Bus (continued)

- The master device is responsible for generating the clock signal.
- It selects the required slave device by asserting the corresponding slave device's slave select signal 'LOW'.
- The data out line (MISO) of all the slave devices when not selected floats at high impedance state.
- The serial data transmission through SPI bus is fully configurable.
  - SPI devices contain a certain set of registers for holding these configurations.
  - The control register holds the various configuration parameters like master/slave selection for the device, baud rate selection for communication, clock signal control, etc.
  - The status register holds the status of various conditions for transmission and reception.

Serial Peripheral Interface (SPI) Bus (continued)

- SPI works on the principle of 'Shift Register'.
- The master and slave devices contain a special shift register for the data to transmit or receive.
  - The size of the shift register is device dependent. Normally it is a multiple of 8.
- During transmission from the master to slave, the data in the master's shift register is shifted out to the MOSI pin and it enters the shift register of the slave device through the MOSI pin of the slave device.
- At the same time the shifted out data bit from the slave device's shift register enters the shift register of the master device through MISO pin.
- In summary, the shift registers of 'master' and 'slave' devices form a circular buffer.
- When compared to I2C, SPI bus is most suitable for applications requiring transfer of data in ' streams '.
- The only limitation is SPI doesn't support an acknowledgement mechanism.

EXTERNAL COMMUNICATION INTERFACE

- The External Communication Interface refers to the different communication channels/buses used by the embedded system to communicate with the external world.

- The various interfaces for external communication are:

i. RS-232 C & RS-485

ii. Universal Serial Bus (USB)

iii. IEEE 1394 (Firewire)

iv. **Infrared (IrDA)**

v. **Bluetooth (BT)**

vi. **Wi-Fi**

vii. **ZigBee**

viii. General Packet Radio Service (GPRS)

INFRARED (IRDA)

- **Infrared Data Association (IrDA)** is the regulatory body – Defines and license specifications for IR data communications.
- **Infrared** is a serial, **half duplex**, line of sight based wireless technology for data communication between devices.
- It is in use from the olden days of communication.
- It supports point to point and point to multi-point communication.
- Depending on speed – SIR, MIR, FIR, VFIR, UFIR, GIR
- The remote control of your TV, VCD player, etc. works on Infrared data communication principle.

Infrared (IrDA)

- Infrared (IrDA) is a serial, half duplex, line of sight based wireless technology for data communication between devices.
- It is in use from the olden days of communication and you may be very familiar with it.
  - E.g.: The remote control of TV, VCD player, etc. works on Infrared.
- Infrared communication technique uses infrared waves of the electromagnetic spectrum for transmitting the data.
- It supports point-point and point-to-multipoint communication, provided all devices involved in the communication are within the line of sight.
- The typical communication range for IrDA lies in the range 10 cm to 1 m.
- The range can be increased by increasing the transmitting power of the IR device.

Infrared (IrDA) (continued)

- IR supports data rates ranging from 9600bits/second to 16Mbps.
- Depending on the speed of data transmission IR is classified into:
  - Serial IR (SIR) – supports data rates ranging from 9600bps to 115.2kbps.
  - Medium IR (MIR) – supports data rates of 0.576Mbps and 1.152Mbps.
  - Fast IR (FIR) – supports data rates up to 4Mbps.
  - Very Fast IR (VFIR) – supports high data rates up to 16Mbps.
  - Ultra Fast IR (UFIR) – targeted to support a data rate up to 100Mbps.
- IrDA is a popular interface for file exchange and data transfer in low cost devices.
- IrDA was the prominent communication channel in mobile phones before Bluetooth's existence.

Infrared (IrDA) (continued)

- IrDA communication involves a transmitter unit for transmitting the data over IR and a receiver for receiving the data.
- Infrared Light Emitting Diode (LED) is the IR source for transmitter and at the receiving end a photodiode acts as the receiver.
- Both transmitter and receiver unit will be present in each device supporting IrDA communication for bidirectional data transfer.
  - Such IR units are known as 'Transceiver'.
- Certain devices like a TV remote control always require unidirectional communication and so they contain either the transmitter or receiver unit.
  - The remote control unit contains the transmitter unit and TV contains the receiver unit.

Infrared (IrDA) (continued)

- Infrared Data Association (IrDA) is the regulatory body responsible for defining and licensing the specifications for IR data communication.
- IR communication has two essential parts: a physical link part and a protocol part.
  - The physical link is responsible for the physical transmission of data between devices supporting IR communication
  - Protocol part is responsible for defining the rules of communication.
- The physical link works on the wireless principle making use of Infrared for communication.
- The IrDA specifications include the standard for both physical link and protocol layer.
- The IrDA control protocol contains implementations for Physical Layer (PHY), Media Access Control (MAC) and Logical Link Control (LLC).

BLUETOOTH (BT)

- **Bluetooth** is a low cost, low power, short range **wireless** technology for data and voice communication.
- Bluetooth supports point-to-point (device to device) and point-to-multipoint (device to multiple device broadcasting) wireless communication.
- A Bluetooth device can function as either master or slave. When a network is formed with one Bluetooth device as master and more than one device as slaves, it is called a **Piconet**. A **Piconet** supports a maximum of seven slave devices.
- Bluetooth is the favorite choice for short range data communication in handheld embedded devices.
- Bluetooth technology is very popular among cell phone users as they are the easiest communication channel for transferring ringtones, music files, pictures, media files, etc. between neighboring Bluetooth enabled phones.
- It supports a data rate of **up to 24 Mbps** and a range of approximately **100 feets** for data communication.

Bluetooth (BT) (continued)

- Bluetooth communication has two essential parts – a physical link part and a protocol part.
  - The physical link is responsible for the physical transmission of data between devices supporting Bluetooth communication
  - The protocol part is responsible for defining the rules of communication.
- The physical link works on the wireless principle making use of RF waves for communication.
- Bluetooth enabled devices essentially contain a Bluetooth wireless radio for the transmission and reception of data.

Bluetooth (BT) (continued)

- The rules governing the Bluetooth communication is implemented in the 'Bluetooth protocol stack'.
  - The Bluetooth communication IC holds the stack.
- Each Bluetooth device will have a 48 bit unique identification number.
- Bluetooth communication follows packet based data transfer.
- Bluetooth supports point-to-point (device to device) and point-to-multipoint (device to multiple device broadcasting) wireless communication.
- The point-to-point communication follows the master-slave relationship.
- A Bluetooth device can function as either master or slave.
- When a network is formed with one Bluetooth device as master and more than one device as slaves, it is called a Piconet.
  - A Piconet supports a maximum of seven slave devices.

Bluetooth (BT)

- Bluetooth is a low cost, low power, short range wireless technology for data and voice communication.
- Bluetooth was first proposed by Ericsson in 1994.
- Bluetooth operates at 2.4GHz of the Radio Frequency spectrum and uses the Frequency Hopping Spread Spectrum (FHSS) technique for communication.
- It supports a data rate of up to 1Mbps and a range of approximately 30 feet for data communication.

Bluetooth (BT) (continued)

- Bluetooth is the favourite choice for short range data communication in handheld embedded devices.
- Bluetooth technology is very popular among cell phone users as they are the easiest communication channel for transferring ringtones, music files, pictures, media files, etc. between neighbouring Bluetooth enabled phones.
- The Bluetooth standard specifies the minimum requirements that a Bluetooth device must support for a specific usage scenario.
- The Generic Access Profile (GAP) defines the requirements for detecting a Bluetooth device and establishing a connection with it.
  - All other specific usage profiles are based on GAP.
  - Serial Port Profile (SPP) for serial data communication, File Transfer Profile (FTP) for file transfer between devices, Human Interface Device (HID) for supporting human interface devices like keyboard and mouse are examples for Bluetooth profiles.
- The specifications for Bluetooth communication is defined and licensed by the standards body 'Bluetooth Special Interest Group (SIG)'.

Wi-Fi

- **Wi-Fi or Wireless Fidelity** is the popular wireless communication technique for networked communication of devices.
- Wi-Fi is intended for network communication, and it supports Internet Protocol (IP) based communication.
- It is essential to have device identities in a multipoint communication to address specific devices for data communication.
- **Wi-Fi based communications** require an intermediate agent called Wi-Fi router/Wireless access point to manage the communications.
- Wi-Fi supports data rates ranging from 1 Mbps to 150 Mbps and offers a range of 100 to 300 feet.

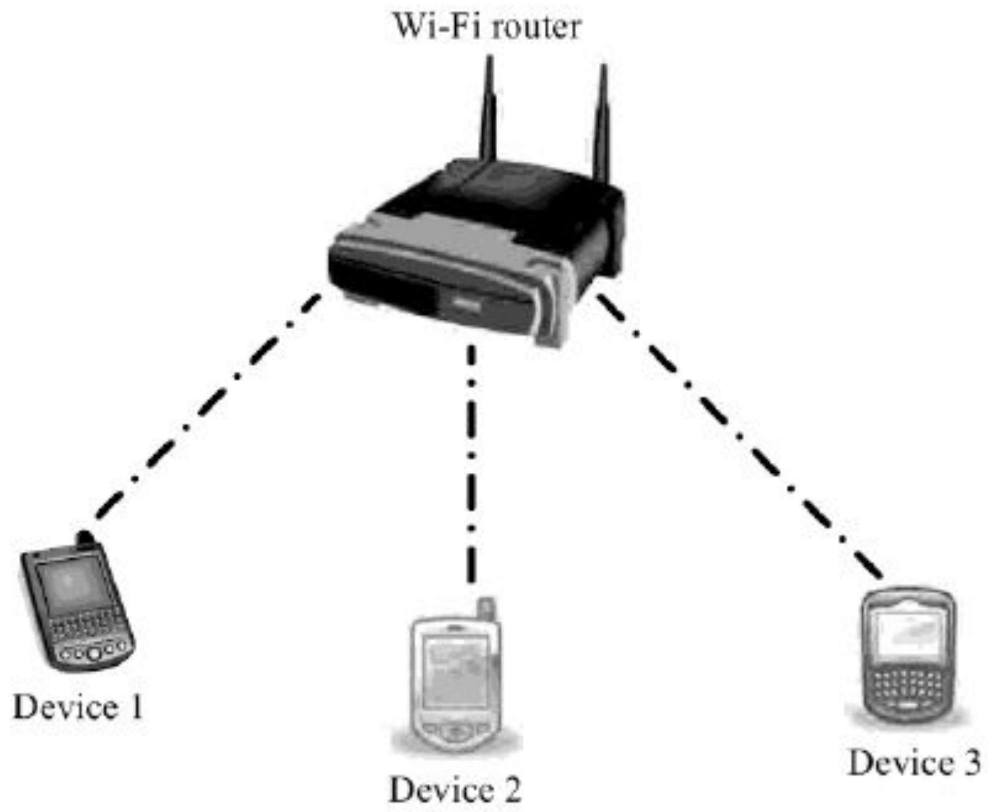


Fig: Wi-Fi Network

Wi-Fi

- Wi-Fi or Wireless Fidelity is the popular wireless communication technique for networked communication of devices.
- Wi-Fi follows the IEEE 802.11 standard.
- Wi-Fi is intended for network communication and it supports Internet Protocol (IP) based communication.
- It is essential to have device identities in a multipoint communication to address specific devices for data communication.
- In an IP based communication each device is identified by an IP address, which is unique to each device on the network.

Wi-Fi (continued)

- Wi-Fi based communications require an intermediate agent called Wi-Fi router/Wireless Access point to manage the communications.
- The Wi-Fi router is responsible for restricting the access to a network, assigning IP address to devices on the network, routing data packets to the intended devices on the network.
- Wi-Fi enabled devices contain a wireless adaptor for transmitting and receiving data in the form of radio signals through an antenna.
- The hardware part of it is known as Wi-Fi Radio.
- Wi-Fi operates at 2.4 GHz or 5 GHz of radio spectrum and they co-exist with other ISM band devices like Bluetooth.

Wi-Fi (continued)

- For communicating with devices over a Wi-Fi network, the device when its Wi-Fi radio is turned ON, searches the available Wi-Fi network in its vicinity and lists out the Service Set Identifier (SSID) of the available networks.
- If the network is security enabled, a password may be required to connect to a particular SSID.
- Wi-Fi employs different security mechanisms like Wired Equivalency Privacy (WEP), Wireless Protected Access (WPA), etc. for securing the data communication.
- Wi-Fi supports data rates ranging from 1 Mbps to 1.73 Gbps depending on the standards (802.11a/b/g/n) and access/modulation method.
- Depending on the type of antenna and usage location (indoor/outdoor), Wi-Fi offers a range of 100 to 300 feet.

ZIGBEE

- **ZigBee** is a low power, low cost, wireless network communication protocol based on the IEEE 802.15.4-2006 standard.
- ZigBee is targeted for low power, low data rate and secure applications for Wireless Personal Area Networking (WPAN).
- ZigBee operates worldwide at the unlicensed bands of Radio spectrum, mainly at 2.400 to 2.484 GHz, 902 to 928 MHz and 868.0 to 868.6 MHz.
- ZigBee supports an operating distance of up to 100 meters and a data rate of 20 to 250Kbps.

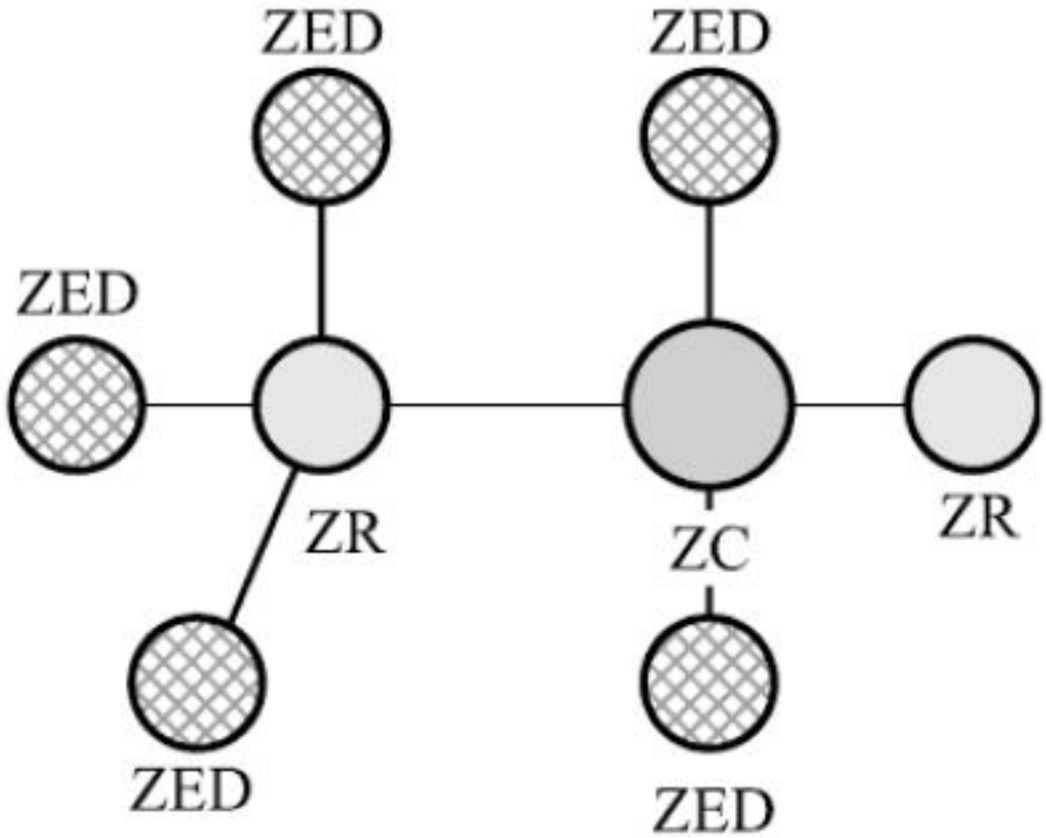


Fig: A ZigBee network model

- In the ZigBee terminology, each ZigBee device falls under any one of the following ZigBee device category:
- ZigBee Coordinator (ZC)/Network Coordinator
  - The ZigBee coordinator acts as the root of the ZigBee network.
  - The ZC is responsible for initiating the ZigBee network and it has the capability to store information about the network.
- ZigBee Router (ZR)/Full function Device (FFD)
  - Responsible for passing information from device to another device or to another ZR.
- ZigBee End Device (ZED)/Reduced Function Device (RFD):
  - End device containing ZigBee functionality for data communication.
  - It can talk only with a ZR or ZC and doesn't have the capability to act as a mediator for transferring data from one device to another.

ZigBee (continued)

- ZigBee is primarily targeting application areas like home & industrial automation, energy management, home control/security, medical/patient tracking, logistics & asset tracking and sensor networks & active RFID.
- Automatic Meter Reading (AMR), smoke detectors, wireless telemetry, HVAC control, heating control, lighting controls, environmental controls, etc. are examples for applications which can make use of the ZigBee technology.
- The specifications for ZigBee is developed and managed by the ZigBee Alliance, a non-profit consortium of leading semiconductor manufacturers, technology providers, OEMs and end-users worldwide.

THANK YOU