MODULE 3

EMBEDDED SYSTEM COMPONENTS

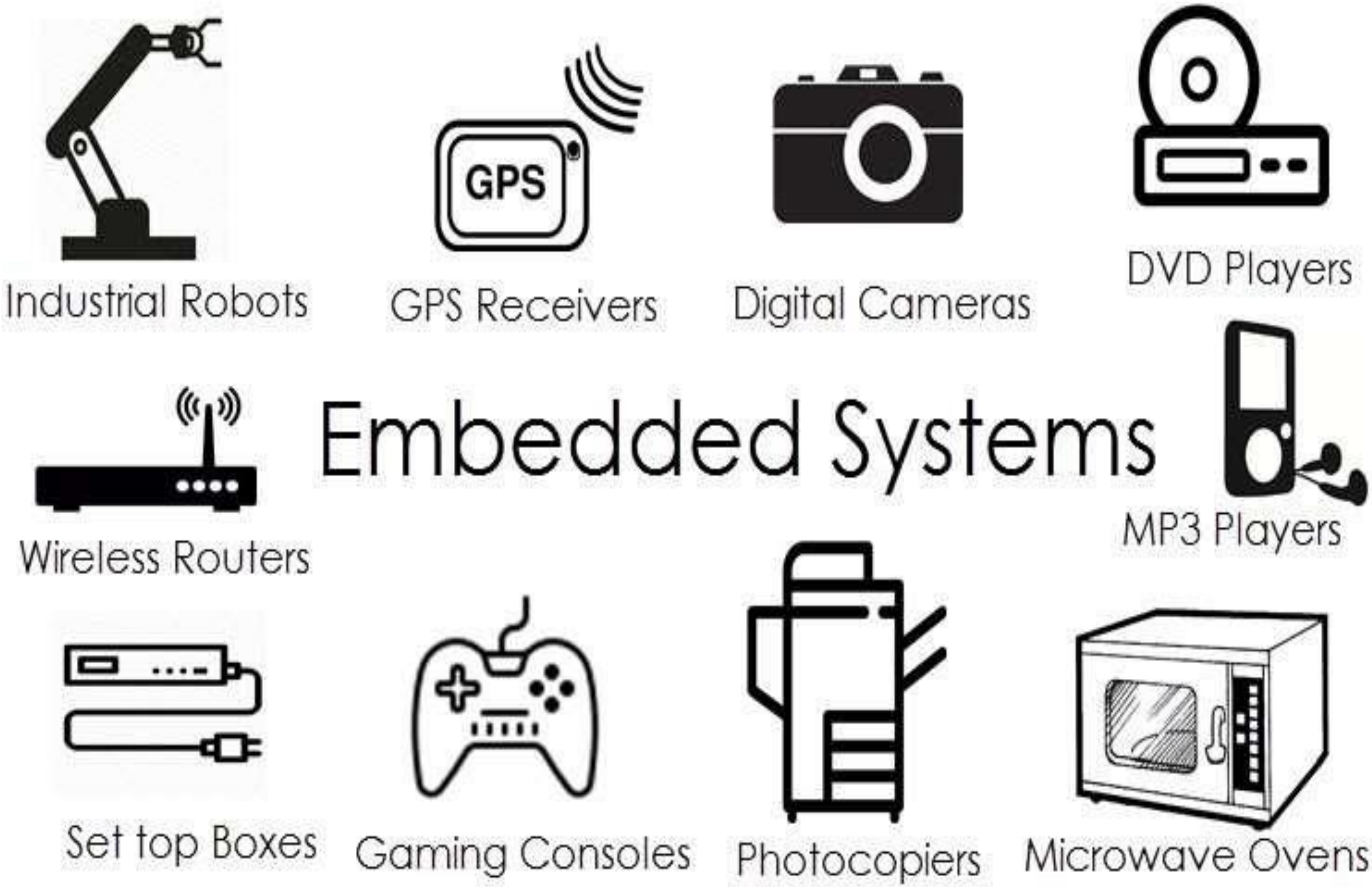
Learning Objectives

- Learn what an Embedded System is
- Learn the difference between Embedded Systems and General Computing Systems
- Know the history of Embedded Systems
- Learn the classification of Embedded Systems based on performance, complexity and the era in which they evolved
- Know the domains and areas of applications of Embedded Systems
- Understand the different purposes of Embedded Systems
- Analysis of a real life example on the bonding of embedded technology with human life

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 firmware (software).
- Every embedded system is unique and the hardware as well as the firmware 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.

Examples


Industrial Robots
GPS
GPS Receivers
Digital Cameras
DVD Players
Embedded Systems
Wireless Routers
MP3 Players
Set top Boxes
Gaming Consoles
Photocopiers
Microwave Ovens

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 system

The classification of embedded system can be

- **Based on Generation**
- **Based on Complexity & performance**
- **Based on Deterministic behavior**
- **Based on Triggering**

Based on Generation

1.First generation(1G)

- Built around 8bit microprocessor & 4bit microcontroller.
- Simple in hardware circuit with firmware developed in assembly code
- Examples: Digital telephone keypads.

2.Second Generation(2G)

- Built around 16-bit microprocessor & 8-bit microcontroller.
- They are more complex & powerful than first generation microprocessor &microcontroller.
- Some of the second generation embedded system contain operating system for their operation
- Examples: SCADA systems

3.Third generation(3G)

- Built around 32-bit microprocessor & 16-bit microcontroller.
- Concepts like Digital Signal Processors(DSPs),Application Specific Integrated Circuits(ASICs) evolved.
- Examples: Robotics, Media, etc.

4.Fourth generation(4G)

- Built around 64-bit microprocessor & 32-bit microcontroller.
- The concept of System on Chips (SoC), Multicore Processors evolved.
- Highly complex & very powerful.
- Examples: Smart Phones.

Based on Complexity & performance

1.Small-scale

- Simple in application need
- Performance not time-critical.
- Built around low performance & low cost 8 bit or 16 bit µp/µc.
- Example: an electronic toy

2.Medium-scale

- Slightly complex in hardware & firmware requirement.
- Built around medium performance & low cost 16 or 32 bit µp/µc.
- Usually contain operating system.
- Examples: Industrial machines.

3. Large-scale:

- Highly complex hardware & firmware.
- Built around 32 or 64 bit RISC µp/µc or PLDs or Multicore Processors.
- Response is time-critical.
- Examples: Mission critical applications.

Based on Deterministic behavior

- This classification is applicable for “Real Time” systems.
- The task execution behavior for an embedded system may be deterministic or non-deterministic.
- Based on execution behavior Real Time embedded systems are divided into Hard and Soft.

Based on Triggering

- Embedded systems which are “Reactive” in nature can be based on triggering.
- Reactive systems can be:
    - Event triggered
    - Time triggered

History of an Embedded system

- Embedded Systems were in existence even before the IT evolution.
- In the olden days embedded systems were built around the old vacuum tube and transistor technologies and the embedded algorithm was developed in low level languages.
- The first recognized modern embedded system is the Apollo Guidance Computer(AGC) developed by the MIT instrumentation laboratory for the lunar expedition.
- AGC was designed on 4K words of ROM & 256 words of RAM.
- The clock frequency of first microchip used in AGC was 1.024 MHz.
- It used 5000 Ics. The UI of AGC is known DSKY(display/keyboard) was used.
- The first mass-produced embedded system was the guidance computer for the Minuteman-I missile in 1961.
- It was the "AUTONETICS D-17" guidance computer, but using discrete transistor logic and a hard-disk for main memory.

Major Application Areas of Embedded System

The application areas and the products in the embedded domain are countless. A few of the important domains and products are listed below:

i. Consumer electronics: Camcorders, cameras, etc.

ii. Household appliances: Television, DVD players, washing machine, fridge, microwave oven, etc.

iii. Home automation and security systems: Air conditioners, sprinklers, intruder detection alarms, closed circuit television cameras, fire alarms, etc.

iv. Automotive industry: Anti-lock breaking systems (ABS), engine control, ignition systems, automatic navigation systems, etc.

v. Telecom: Cellular telephones, telephone switches, handset multimedia applications, etc.

vi. Computer peripherals: Printers, scanners, fax machines, etc.

vii. Computer Networking systems: Network routers, switches, hubs, firewalls, etc.

viii. Healthcare: Different kinds of scanners, EEG, ECG machines etc.

ix. Measurement & Instrumentation: Digital multi meters, digital CROs, logic analyzers PLC systems, etc.

x. Banking & Retail: Automatic teller machines (ATM) and currency counters, point of sales (POS).

xi. Card Readers: Barcode, smart card readers, hand held devices, etc.

Purpose of Embedded Systems

Embedded systems are used in various domains like consumer electronics, home automation, telecommunications, automotive industry, healthcare, control & instrumentation, retail and banking applications, etc. Within the domain itself, according to the application usage context, they may have different functionalities. Each embedded system is designed to serve the purpose of any one or a combination of the following tasks:

- Data collection/Storage/Representation
- Data Communication
- Data (signal) processing
- Monitoring
- Control
- Application specific user interface

1.Data Collection/Storage/Representation

- Embedded system 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 analog or digital.
- Embedded systems with analog data capturing techniques collect data directly in the form of analog signal.
- whereas embedded systems with digital data collection mechanism converts the analog signal to the digital signal using analog to digital converters.
- If the data is digital it can be directly captured by digital embedded system.
- **A digital camera is a typical example of an embedded**
- Images are captured and the captured image may be stored within the memory of the camera.
- The captured image can also be presented to the user through a graphic LCD unit.

2.Data communication

- Embedded data communication systems are deployed in applications from complex satellite communication to simple home networking systems.
- The transmission of data is achieved either by a wire-line medium or by a wire-less medium.
- Data can either be transmitted by analog means or by digital means.
- **Wireless modules-Bluetooth, Wi-Fi.**
- **Wire-line modules-USB, TCP/IP.**
- **Network hubs, routers, switches** are examples of dedicated data transmission embedded systems.

3.Data signal processing

- Embedded systems with signal processing functionalities are employed in applications demanding signal processing like speech coding,audio vidéo codec, transmission applications etc.
- **A digital hearing aid** is a typical example of an embedded system employing data processing.
- Digital hearing aid improves the hearing capacity of hearing impaired person.

4.Monitoring

- All embedded products coming under the medical domain are with monitoring functions.
- **Electro cardiogram machine** is intended to do the monitoring of the heartbeat of a patient but it cannot impose control over the heartbeat.
- Other examples with monitoring function are digital CRO, digital multi-meters, and logic analyzers.

5.Control

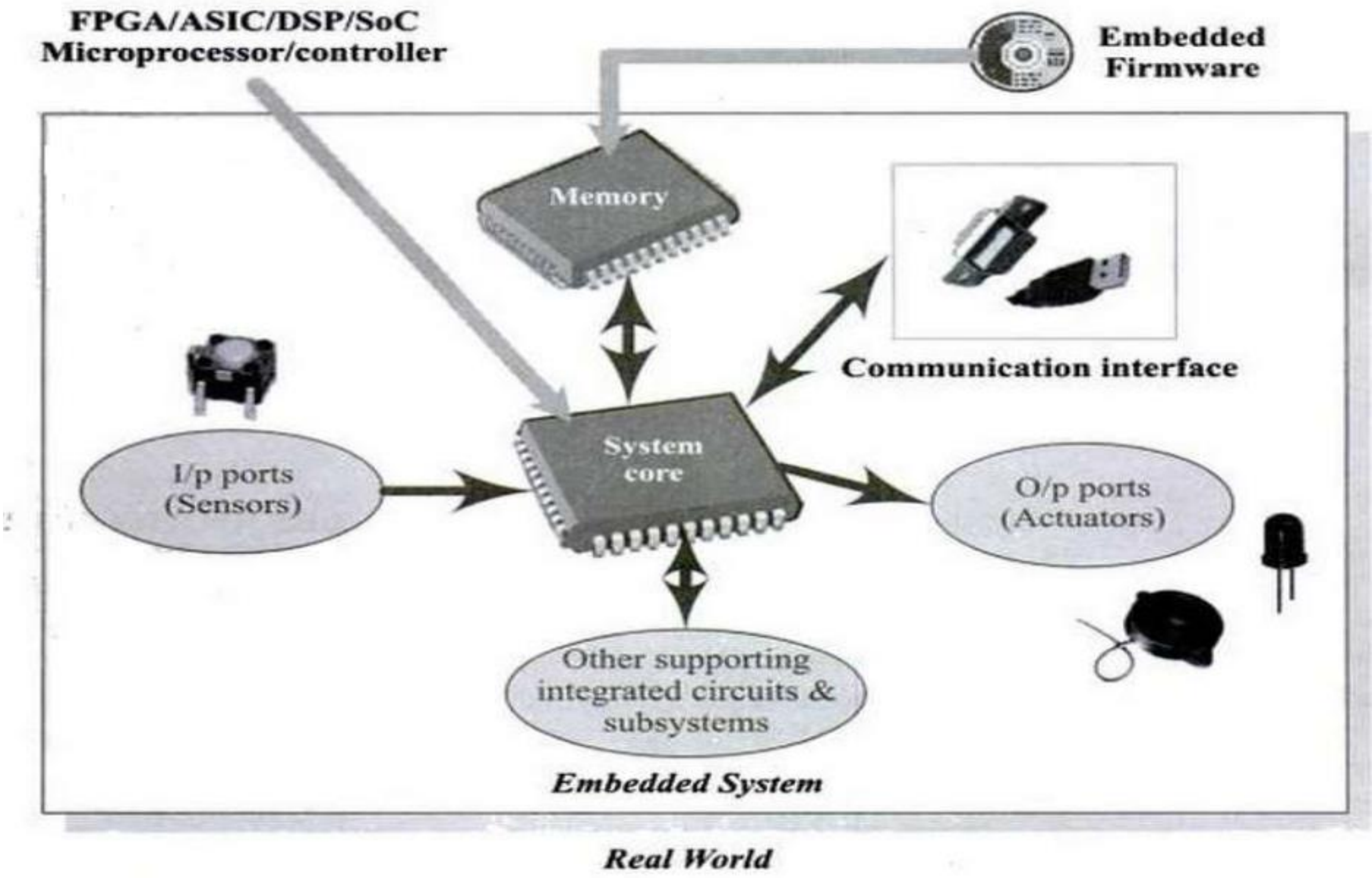
- Embedded systems with control functionalities impose control over some variables according to the changes in input variables.
- A system with control functionality contains both sensors and actuators.
- Sensors are connected to the input port for capturing the changes in environmental variable.
- Actuators connected to the output port are controlled according to the changes in the input variable.
- **Air conditioner** system used to control the room temperature to a specified limit is a typical example for **control** purpose.

.

6.Application specific user interface

- Buttons, switches, keypad, lights, bells, 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, system speaker, vibration alert etc.

Core of Embedded systems



Core of Embedded systems (Cont…)

- Embedded systems are domain and application specific and are built around a central core.
- The core of the embedded system falls into any of the following categories
- **General purpose and Domain Specific Processors**
  - Microprocessors
  - Microcontrollers
  - Digital Signal Processors
- **Application Specific Integrated Circuits. (ASIC)**
- **Programmable logic devices(PLD's)**
- **Commercial off-the-shelf components (COTs)**

General purpose and Domain Specific Processors

- Almost 80% of the embedded systems are processor/controller based.
- The processor may be microprocessor or a microcontroller or digital signal processor, depending on the domain and application.

Microprocessor	Microcontroller
❑A silicon chip representing a CPU	❑A highly integrated chip that contains CPU, RAM/ ROM, Timer, Storage and I/O ports
❑It is a dependent unit	❑It is a self contained unit.
❑Most of the time general purpose in design and operation	❑Mostly application-oriented or domain specific
❑Doesn’t contain built in IO port.	❑Most of the processor/controller contain multiple IO port
❑Targeted for high end market where performance is important.	❑Targeted for Embedded market where performance is not so critical.
❑Limited power saving options	❑Includes lot of power saving options

RISC	CISC
❑It contains lesser number of instructions.	❑It contains greater number of instructions.
❑Instruction pipelining and increased execution speed.	❑Instruction pipelining feature does not exist.
❑Orthogonal instruction set(allows each instruction to operate on any register and use any addressing mode)	❑Non-orthogonal set(all instructions are not allowed to operate on any register and use any addressing mode)
❑Operations are performed on registers only, only memory operations are load and store.	❑Operations are performed either on registers or memory depending on instruction.
❑A larger number of registers are available.	❑The number of general purpose registers are very limited.
❑Programmer needs to write more code to execute a task since instructions are simpler ones.	❑Instructions are like macros in C language. A programmer can achieve the desired functionality with a single instruction
❑It is single, fixed length instruction and Less silicon usage	❑It is variable length instruction and more silicon usage
❑Implemented With Harvard Architecture.	❑Implemented With both Harvard and Von-neumann Architecture.

Harvard architecture	Von-Neumann architecture
❑It has separate buses for instruction as well as data fetching.	❑It shares single common bus for instruction and data fetching.
❑Easier to pipeline, so high performance can be achieved.	❑Low performance as compared to Harvard architecture.
❑Comparatively high cost.	❑It is cheaper.
❑Since data memory and program memory are stored physically in different locations, no chances exist for accidental corruption of program memory.	❑Accidental corruption of program memory may occur if data memory and program memory are stored physically in the same chip.
❑No Memory alignment problem	❑Allows Self modifying codes

Digital Signal Processors

- DSP are powerful special purpose 8/16/32 bit microprocessor designed to meet the computational demands and power constraints of today's embedded audio, video and communication applications.
- DSP are 2 to 3 times faster than general purpose microprocessors in signal processing applications and implement algorithms in hardware which speeds up the execution.
- SOP(Sum of Products) calculation, convolution, FFT(Fast Fourier Transform), DFT(Discrete Fourier Transform), etc are some of the operation performed by DSP.
- DSP includes following key units:
  - Program memory: It is a memory for storing the program required by DSP to process the data.
  - Data memory: It is a working memory for storing temporary variables and data/signal to be processed.
  - Computational engine: It performs the signal processing in accordance with the stored program memory computational engine incorporated many specialized arithmetic units and each of them operates simultaneously to increase the execution speed. It also includes multiple hardware shifters for shifting operands and saves execution time.
  - I/O unit: It acts as an interface between the outside world and DSP. It is responsible for capturing signals to be processed and delivering the processed signals.
  - **Examples:** Audio video signal processing, telecommunication and multimedia applications.

Application Specific Integrated Circuits. (ASIC)

- ASICs is a microchip design to perform a specific and unique applications.
- Because of using single chip for integrates several functions there by reduces the system development cost.
- Most of the ASICs are proprietary (which having some trade name) products, it is referred as Application Specific Standard Products(ASSP).
- As a single chip ASIC consumes a very small area in the total system. Thereby helps in the design of smaller system with high capabilities or functionalities.

Programmable logic devices(PLD's)

- A PLD is an electronic component. It used to build digital circuits which are reconfigurable.
- A logic gate has a fixed function but a PLD does not have a defined function at the time of manufacture.
- PLDs offer customers a wide range of logic capacity, features, speed, voltage characteristics.
- PLDs can be reconfigured to perform any number of functions at any time.
- A variety of tools are available for the designers of PLDs which are inexpensive and help to develop, simulate and test the designs.

- PLDs having following two major types.

  1) CPLD(Complex Programmable Logic Device):CPLDs offer much smaller amount of logic up to 1000 gates.

  2) FPGAs(Field Programmable Gate Arrays): It offers highest amount of performance as well as highest logic density, the most features.

- **Advantages of PLDs**

  1) PLDs offer customer much more flexibility during the design cycle.

  2)PLDs do not require long lead times for prototypes or production parts because PLDs are already on a distributors shelf and ready for shipment.

  3)PLDs can be reprogrammed even after a piece of equipment is shipped to a customer.

Commercial off-the-shelf components(COTs)

- A Commercial off the Shelf product is one which is used 'as- is'.
- The COTS components itself may be develop around a general purpose or domain specific processor or an ASICs or a PLDs.
- The major advantage of using COTS is that they are readily available in the market, are chip and a developer can cut down his/her development time to a great extent.

- The major drawback of using COTS components in embedded design is that the manufacturer of the COTS component may withdraw the product or discontinue the production of the COTS at any time if rapid change in technology occurs.
- **Advantages of COTS**
  1) Ready to use
  2) Easy to integrate
  3) Reduces development time
- **Disadvantages of COTS:**
  1) No operational or manufacturing standard (all proprietary).
  2) Vendor or manufacturer may discontinue production of a particular COTS product.

Big-Endian vs. Little-Endian

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

Memory

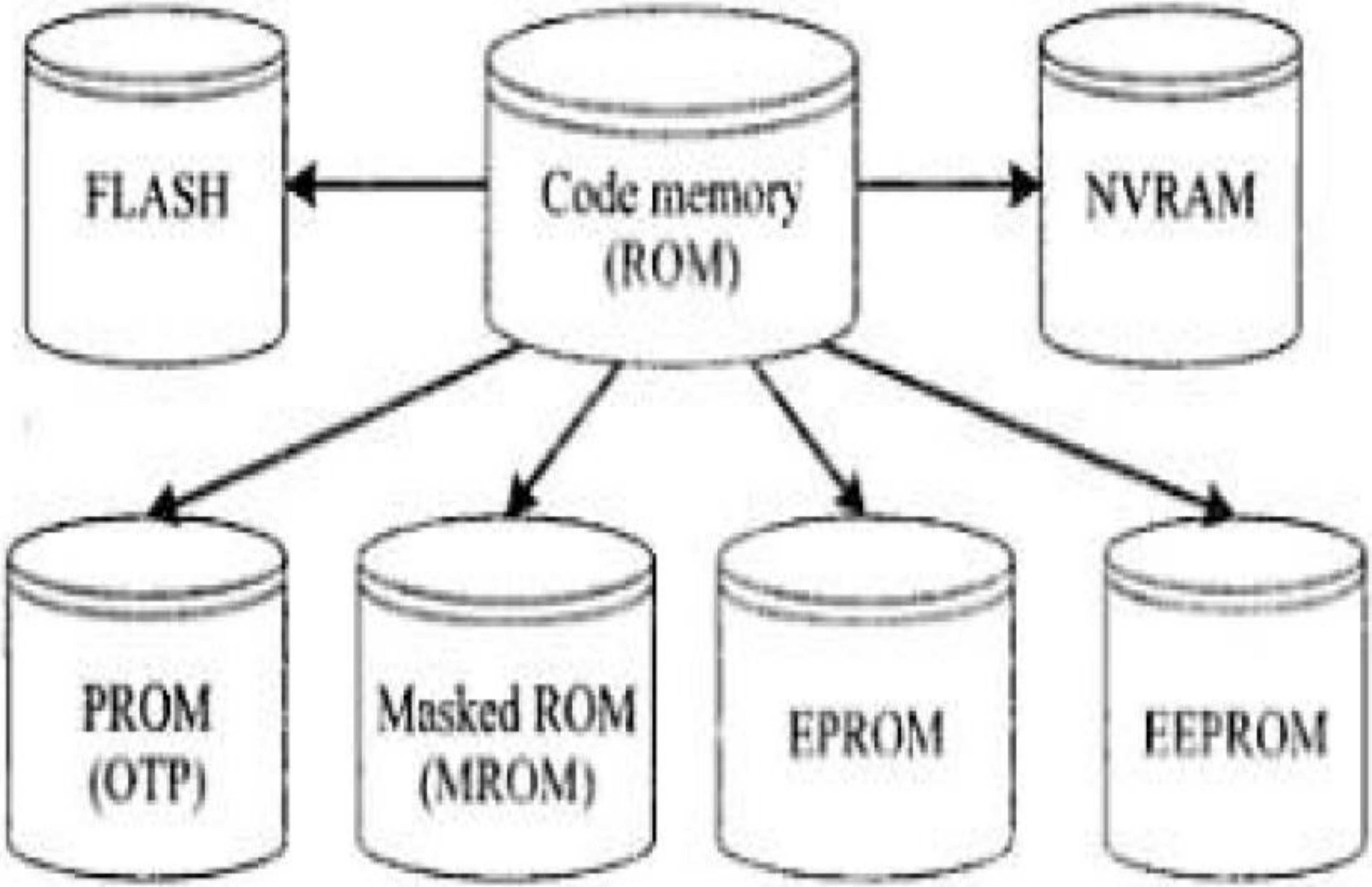
There are **different types of memory used in embedded system applications:**

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

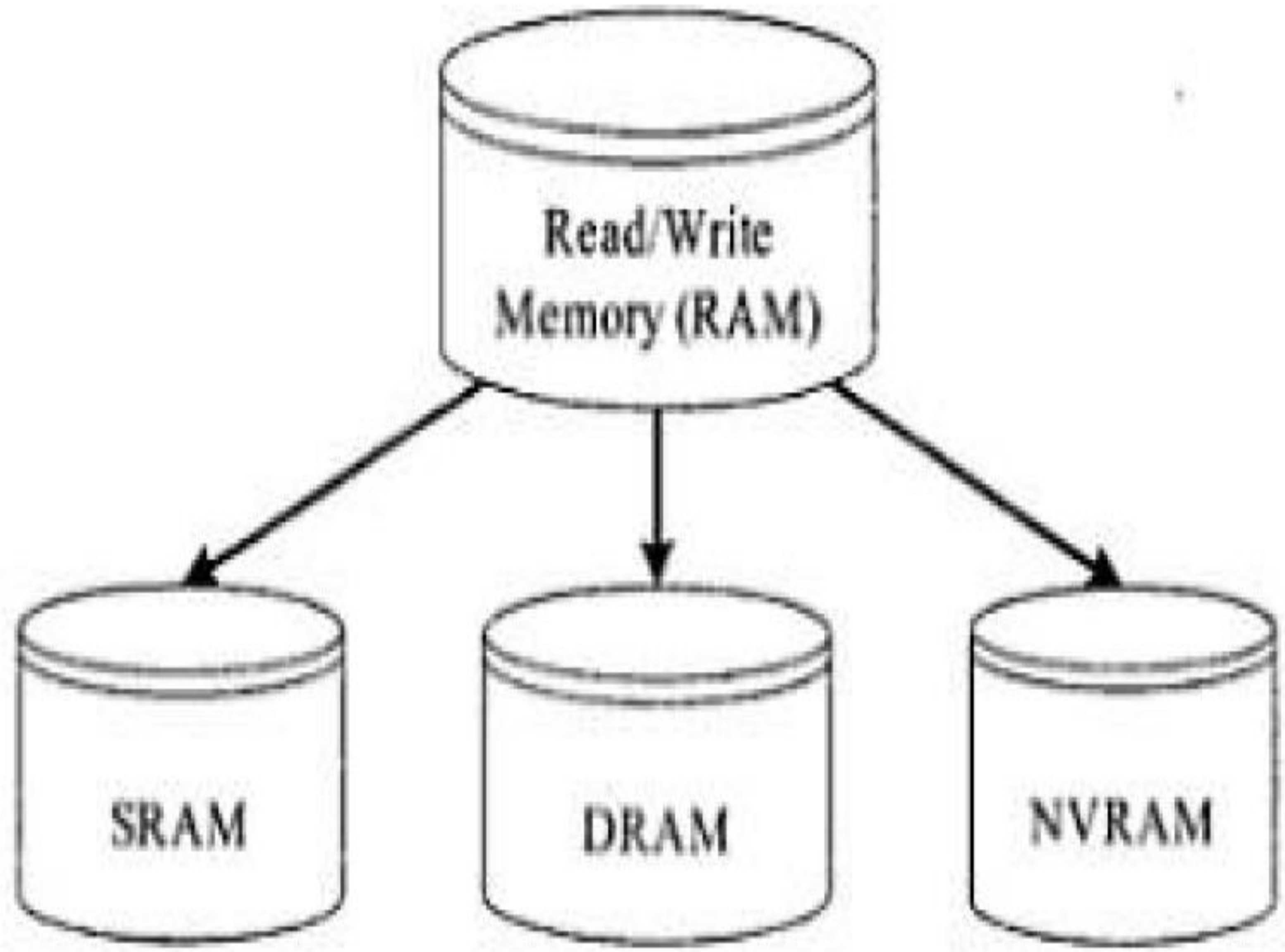
Read-Write Memory/Random Access Memory (RAM)

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


FLASH
Code memory
(ROM)
NVRAM
PROM
(OTP)
Masked ROM
(MROM)
EPROM
EEPROM

- **Mask ROM :** Masked ROM 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.
- **PROM/ OTP :** 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.
- **EPROM :** 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 reprogram ability, it needs to be taken out of the circuit board and put in a UV eraser device for 20 to 30 minutes. So it is a tedious and time-consuming process.

- **EEPROM :** 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 :** 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. Each sector/ page should be erased before re-programming.


Read/Write
Memory (RAM)
SRAM
DRAM
NVRAM

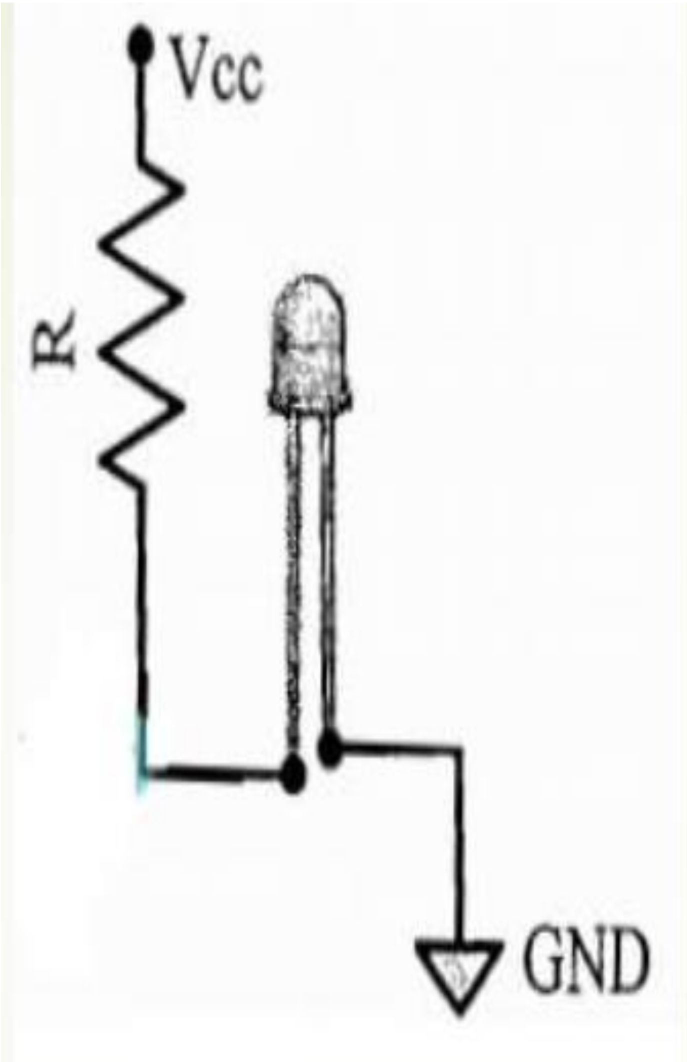
- **SRAM :** SRAM stores data in the form of voltage. They are made up of flip-flops. A flip-flop for a memory cell takes four or six transistors (or 6 MOSFETs) along with some wiring, four of the transistors are used for building the latch (flip-flop) part of the memory cell and two for controlling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

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.

Sensors ,Actuators and I/O Subsystem

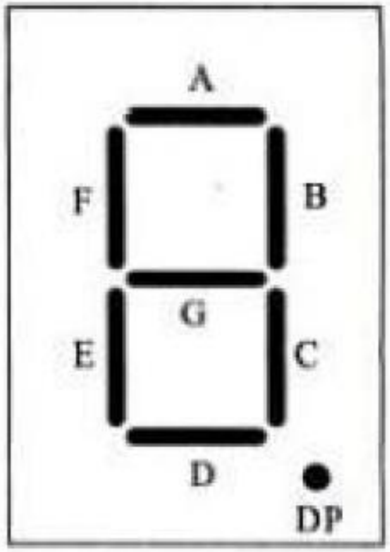
- A **Sensor** is a transducer device that converts energy from one form to another for any measurement or control purpose.
- An **Actuator** is a form of transducer device (mechanical or electrical) which converts signals to corresponding physical action (motion). Actuator acts as an output device.
- The **I/O subsystem** of the embedded system facilitates the interaction of the embedded system with the external world. The interaction happens through the sensors and actuators connected to the input and output ports respectively of the embedded system.
- Sensors and Actuators used in the embedded system are:
  1. **Light Emitting Diode (LED)**
  2. **7-Segment LED Display**
  3. **Optocoupler**
  4. **Stepper Motor**
  5. **Relay**
  6. **Piezo Buzzer**
  7. **Push Button Switch**
  8. **Programmable Peripheral Interface (PPI)**
  9. **Keyboard**

1.Light Emitting Diode (LED)



- LED is an important output device for visual indication in any embedded system. LED can be used as an indicator for the status of various signals or Situations.
- LED is a p-n junction diode and it contains an anode and a cathode.
- For proper functioning of the LED, the anode of it should be connected to +ve terminal of the supply voltage and cathode to the –ve terminal of the supply voltage.
- The current flowing through the LED must be limited to a value below the maximum current that it can conduct.
- A resistor is used in series between the power supply and the LED to limit the current through the LED.

2. Seven-Segment LED Display



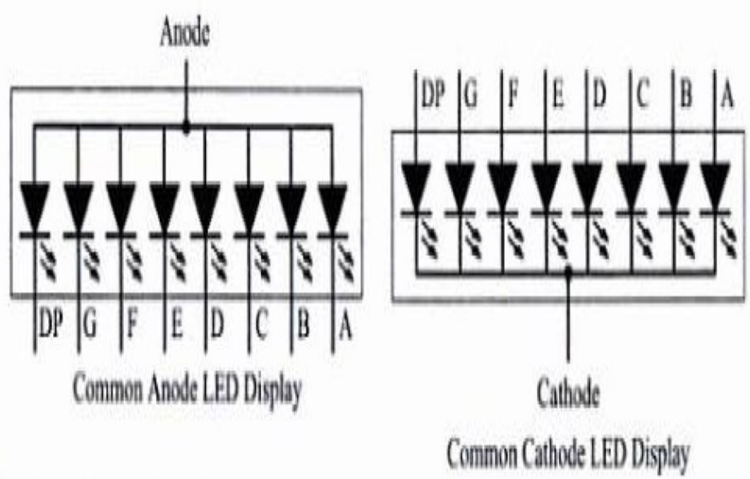


Fig. 2.18 Common anode and cathode configurations of a 7-segment LED Display

- The **7-segment LED display is an output device for displaying** alphanumeric characters. It contains 8 light-emitting diode (LED)
- Segments arranged in a special form. Out of the 8 LED segments, 7 are used for displaying alphanumeric characters and 1 is used for representing ‘decimal point’ in decimal number display.
- The 7-segment LED displays are available in two different configurations,namely;
- Common Anode and Common Cathode.
- In the common anode configuration, the anodes of the 8 segments are connected commonly.
- Whereas in the common cathode configuration, the 8 LED segments share a common cathode line.

3. Optocoupler

- Optocoupler is a solid state device to isolate two parts of a circuit.
- Optocoupler combines an LED and a photo-transistor.
- In electronic circuits, an optocoupler is used for suppressing interference in data communication, circuit isolation, high voltage separation, simultaneous separation and signal intensification, etc.
- Optocouplers can be used in either input circuits or in output circuits.
- Figure illustrates the usage of Optocoupler in input circuit and output circuit of an embedded system with a microcontroller as the system core.

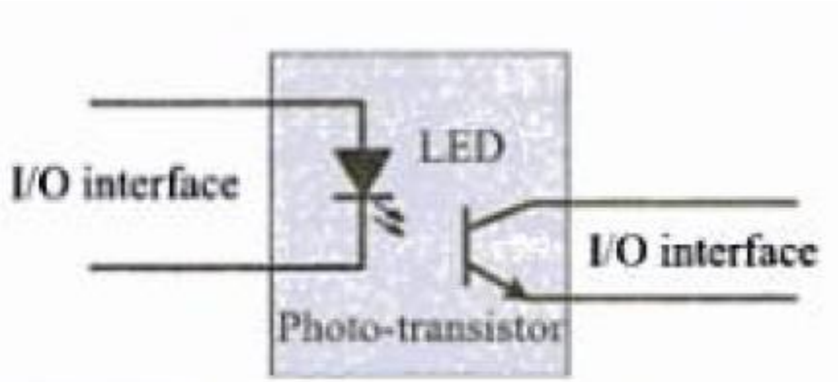


Fig. 2.16 An optocoupler device

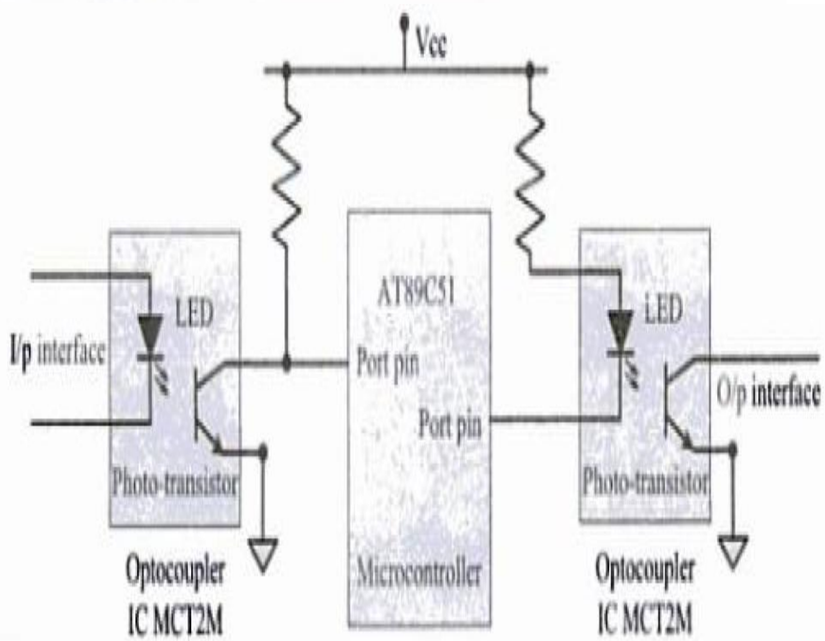
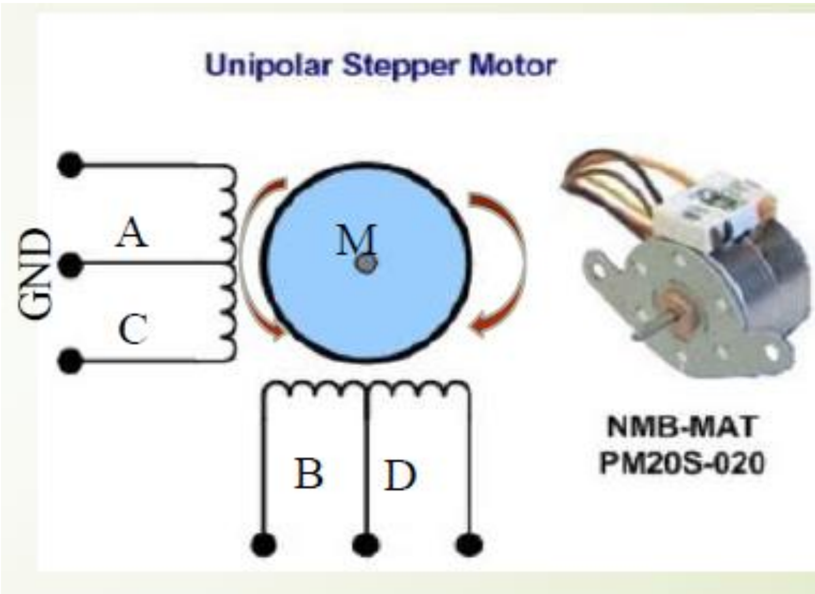
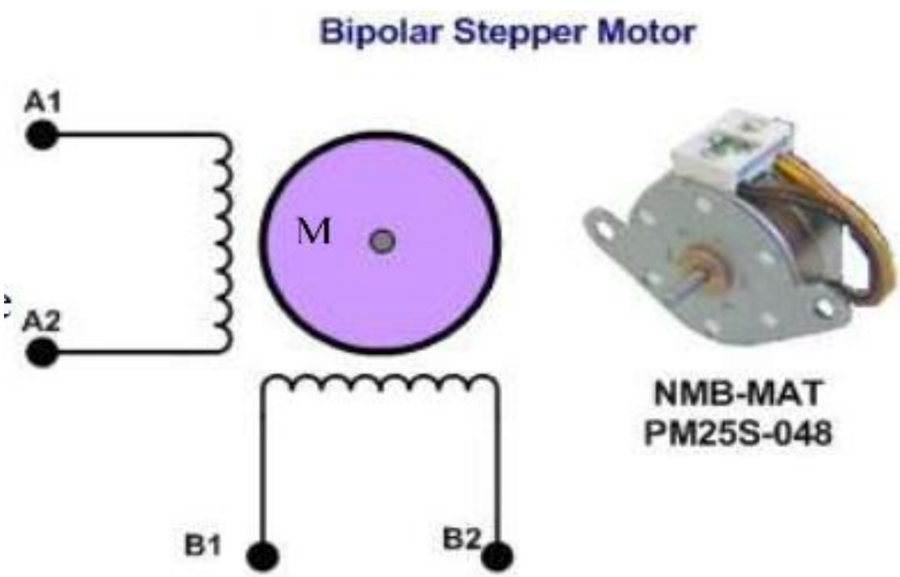


Fig. 2.17 Optocoupler in Input and Output circuit

4. Stepper Motor





- A stepper motor is an electro-mechanical device which generates discrete displacement (motion) in response to dc electrical signals.
- Stepper motors are widely used in industrial embedded applications, consumer electronic products and robotics control systems.
- Based on the coil winding arrangements, a two-phase stepper motor is classified into two.
  - Unipolar
  - Bipolar
- A unipolar stepper motor contains two windings per phase. The direction of rotation (clockwise or anticlockwise) of a stepper motor is controlled by changing the direction of current flow.
- Current in one direction flows through one coil and in the opposite direction flows through the other coil.

4. Stepper Motor(Cont…)

- A bipolar stepper motor contains single winding per phase. For reversing the motor rotation the current flow through the windings is reversed dynamically.
- It requires complex circuitry for current flow reversal. The different stepping modes supported by stepper motor are
  - **Full Step:** In the step mode both the phases are energized simultaneously.
  - **Wave Step:** In the wave step mode only one phase is energized at a time and each coils of the phase is energies alternatively.
  - **Half Step :** It uses the combination of wave and full step. It has the highest torque and stability.

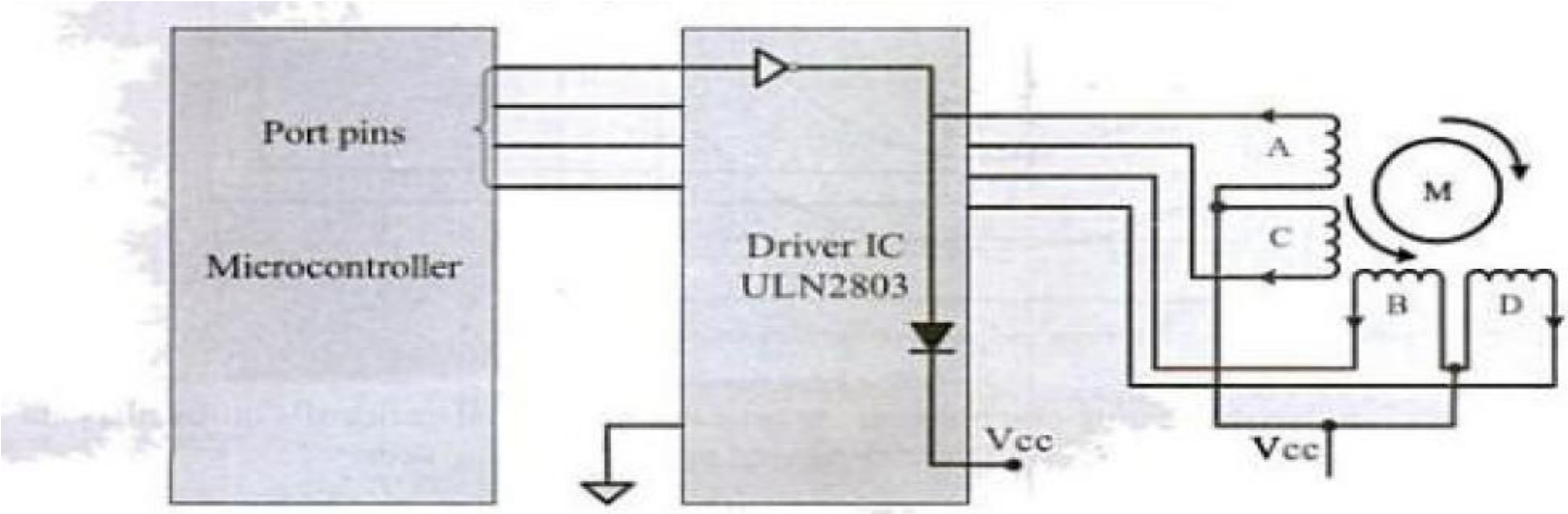


Fig. 2.20 Interfacing of stepper motor through driver circuit

4. Stepper Motor (Cont..)

Full Step

Step	Coil A	Coil B	Coil C	Coil D
1	H	H	L	L
2	L	H	H	L
3	L	L	H	H
4	H	L	L	H

Wave Step

Step	Coil A	Coil B	Coil C	Coil D
1	H	L	L	L
2	L	H	L	L
3	L	L	H	L
4	L	L	L	H

Half Step

Step	Coil A	Coil B	Coil C	Coil D
1	H	L	L	L
2	H	H	L	L
3	L	H	L	L
4	L	H	H	L
5	L	L	H	L
6	L	L	H	H
7	L	L	L	H
8	H	L	L	H

5. Relay

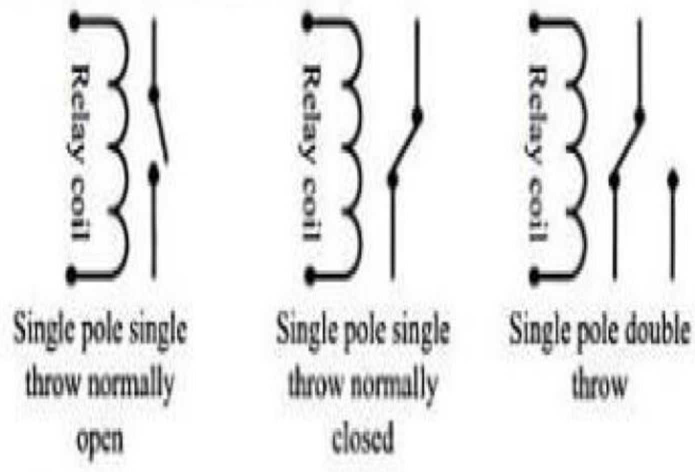


Fig. 2.21 Relay configurations

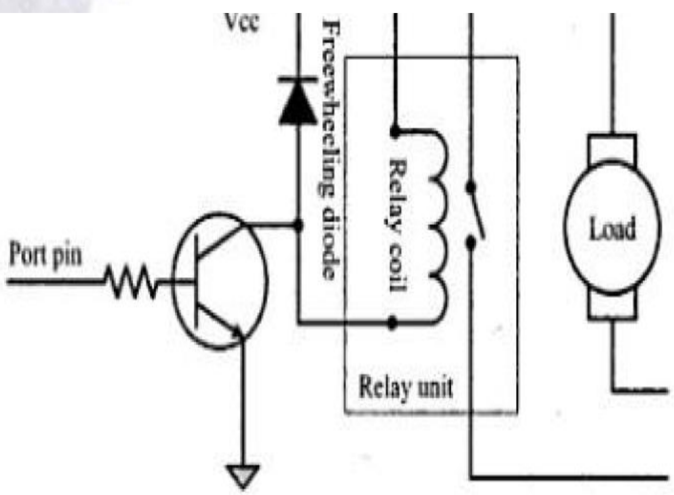


Fig. 2.22 Transistor based Relay driving circuit

- Relay is an electro-mechanical device. In embedded application, the 'Relay' unit acts as dynamic path selectors for signals and power.
- The 'Relay' unit contains a relay coil made up of insulated wire on a metal core and a metal armature with one or more contacts.
- Relay' works on electromagnetic principle. When a voltage is applied to the relay coil, current flows through the coil, which in turn generates a magnetic field.
- The magnetic field attracts the armature core and moves the contact point. The movement of the contact point changes the power/signal flow path.
- 'Relays' are available in different configurations.
- **Single Pole Single Throw:** configuration has only one path for information flow.
- **Closed Single Pole Single Throw:** the circuit is normally closed and it becomes open when the relay is energized.
- **Single Pole Double Throw Relay:** there are two paths for information flow and they are elected by energizing or de-energizing the relay.

6.Piezo Buzzer

Piezo buzzer is a piezoelectric device for generating audio indications in embedded application. A piezoelectric buzzer contains a piezoelectric diaphragm which produces audible sound in response to the voltage applied to it. Piezoelectric buzzers are available in two types: ‘Self-driving’ and ‘External driving’.

7. Push Button Switch

It is an input device. Push button switch comes in two configurations, namely “Push to Make’ and ‘Push to Break’. **In the ‘Push to Make’ configuration, the switch is normally in the open** state and it makes a circuit contact when it is pushed or pressed. **In the ‘Push to Break’ configuration, the switch is normally in the closed** state and it breaks the circuit contact when it is pushed or pressed.

8. Programmable peripheral interface(PPI)

These are used to extend the I/O capabilities of Microprocessor and Microcontroller.8255A is the popular PPI used which supports 24 I/O pins grouped into PORT A,PORT B and PORT C. Depending on the need these can be used very effectively.

9.Keyboard

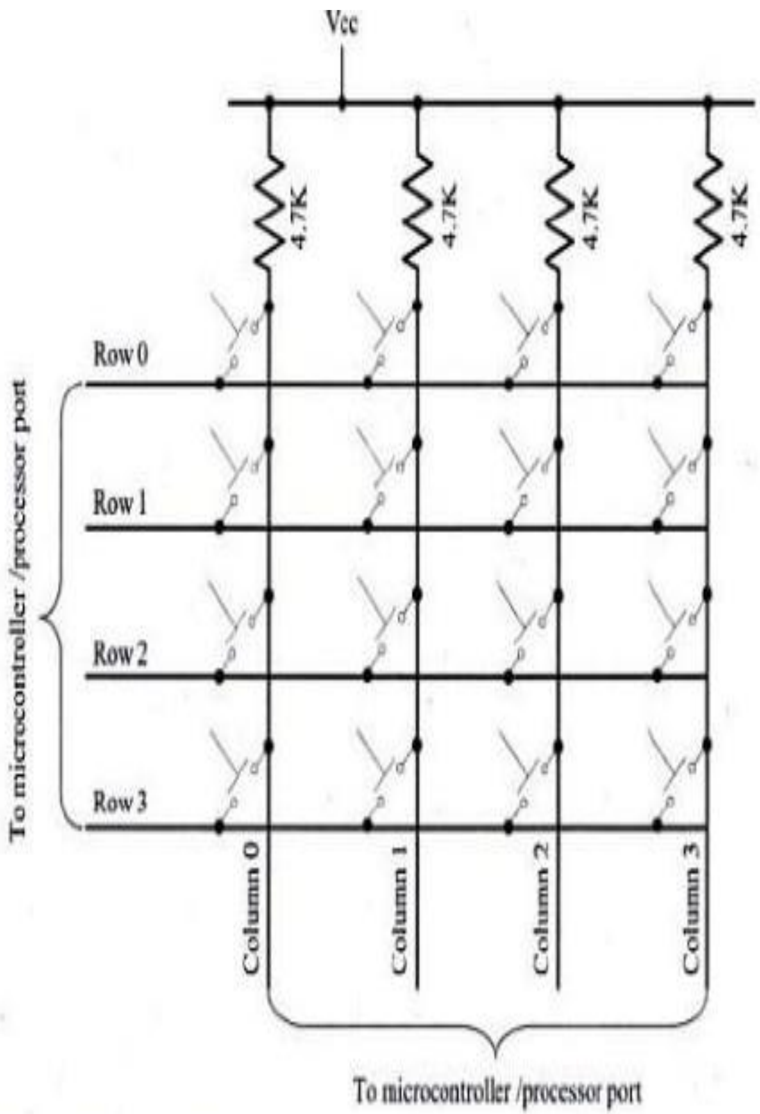


Fig. 2.24 Matrix keyboard Interfacing

- Keyboard is an input device for user interfacing.
- If the number of keys required is very limited, push button switches can be used and they can be directly interfaced to the port pins for reading.
- Matrix keyboard is an optimum solution for handling large key requirements. It greatly reduces the number of interface connections.
- For example, for interfacing 16 keys, in the direct interfacing technique 16 port pins are required, whereas the matrix keyboard only 8 lines are required. The 16 keys are arranged in a 4 column*4 row matrix.

Communication Interface

- For an embedded product, the communication interface can be viewed in two different perspectives; namely;
  - Device/board level communication interface (Onboard Communication Interface)
  - Product level communication interface (External Communication Interface).

- **Onboard Communication Interface**
  1. Inter Integrated Circuit (I2C) Bus
  2. Serial Peripheral Interface (SPI) Bus
  3. Universal Asynchronous Receiver Transmitter (UART)
  4. 1-Wire Interface
  5. Parallel Interface
- **External Communication Interface**
  1. RS-232 C & RS-485
  2. Universal Serial Bus (USB)
  3. IEEE 1394 (Firewire)
  4. Infrared (IrDA)
  5. Bluetooth (BT)
  6. Wi-Fi
  7. ZigBee
  8. General Packet Radio Service (GPRS)

Onboard Communication Interface

1.Inter Integrated Circuit (I2C) Bus

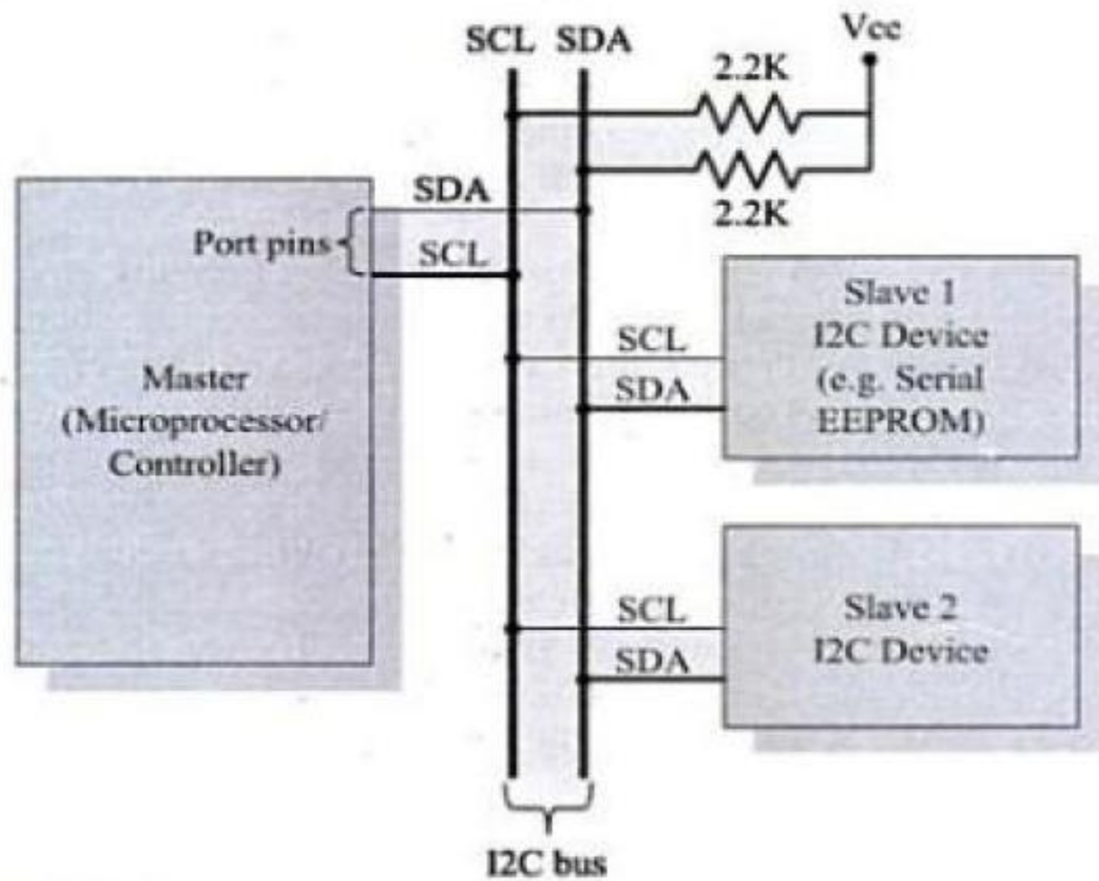


Fig. 2.26 I2C Bus Interfacing

I2C Protocol Communication (Steps)

1. Master device pulls the clock line (SCL) of the bus to ‘HIGH’.
2. Master device pulls the data line (SDA) ‘LOW’, when the SCL line is at logic ‘HIGH’.
3. Master sends the address (7 bit or 10 bit wide) of the ‘Slave’ device to which it wants to communicate, over the SDA line.
4. Master device sends read (1)or write(0) bit according to the requirement.
5. Master waits for the acknowledgement bit from the slave device.
6. The Slave device with the address requested by the master device responds by sending an acknowledge bit (Bit value =1) over the SDA line.
7. Upon receiving the acknowledge bit, master sends the 8bit data to the slave device over SDA line, if the requested operation is ‘Write to device’.
8. If the requested operation is ‘Read from device’, the slave device sends data to the master over the SDA line.
9. Master waits for the acknowledgement bit from the device upon byte transfer complete for a write operation and sends an acknowledge bit to the slave device for a read operation .
10. Master terminates the transfer by pulling the SDA line ‘HIGH’ when the clock line SCL is at logic ‘HIGH’(Indicating the „STOP" condition).

2.Serial Peripheral Interface (SPI) Bus

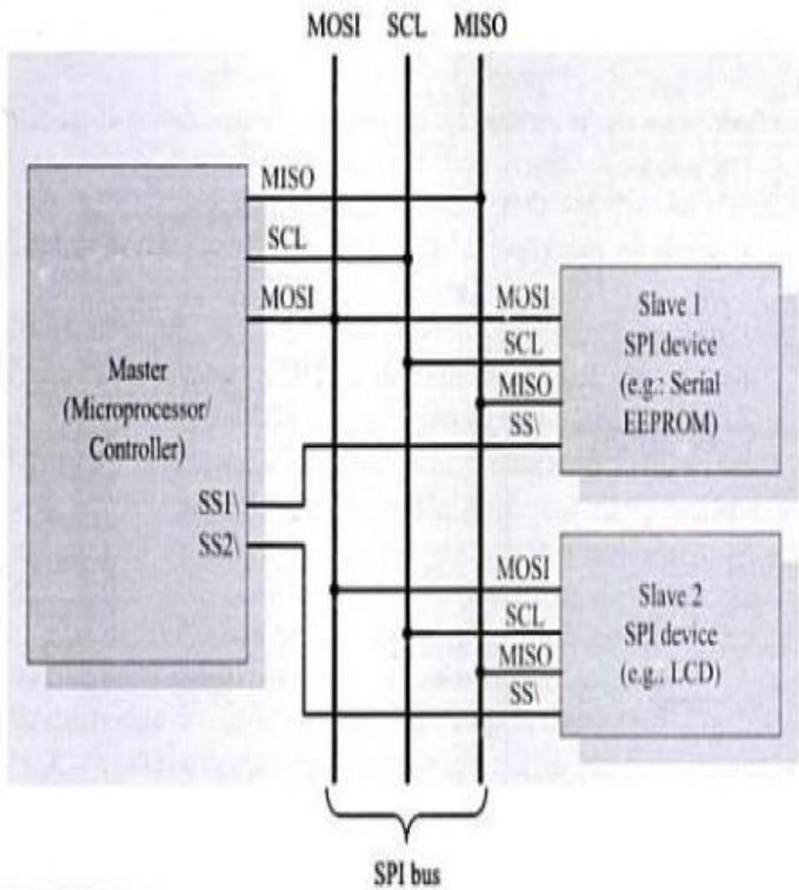
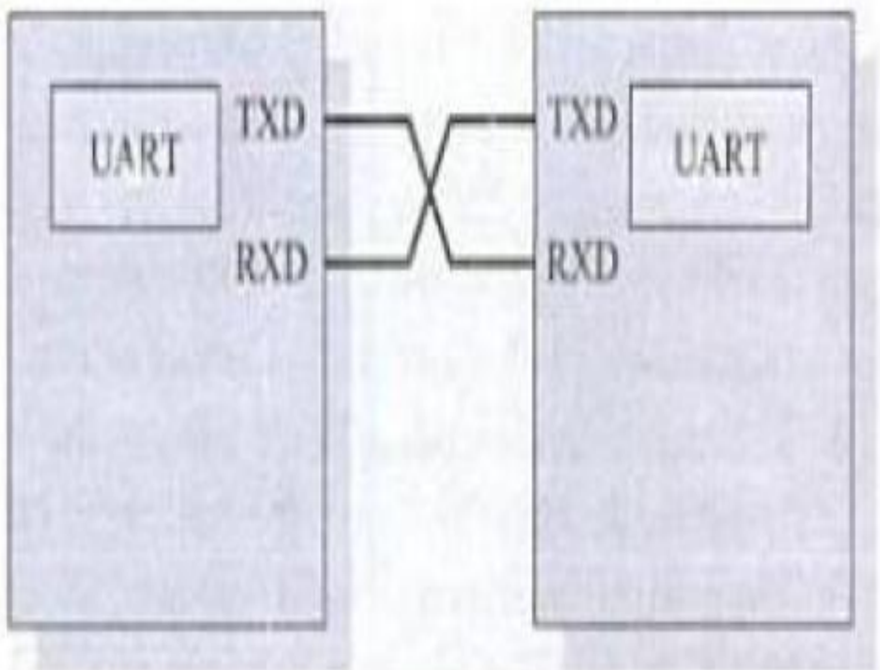


Fig. 2.27 SPI bus interfacing

- The Serial Peripheral Interface Bus (SPI) is a synchronous bidirectional full duplex four-wire serial interface bus.
- SPI is a single master multi-slave system.
- SPI requires four signal lines for communication. They are Master Out Slave In (MOSI), Master In Slave Out (MISO), Serial Clock (SCLK) and Slave Select (SS).
- When compared to I2C, SPI bus is most suitable for applications requiring transfer of data in 'streams'.

3.UART



TXD: Transmitter line
RXD: Receiver line

Fig. 2.28 UART Interfacing

- In UART Serial Communication, the data is transmitted asynchronously i.e. there is no clock or other timing signal involved between the sender and receiver. Instead of clock signal, UART uses some special bits called Start and Stop bits.
- These bits are added to the actual data packet at the beginning and end respectively. These additional bits allows the receiving UART to identify the actual data.
- When the receiving UART detects a start bit, it starts to read the incoming bits at a specific frequency known as the baud rate.

4. 1-Wire interface bus

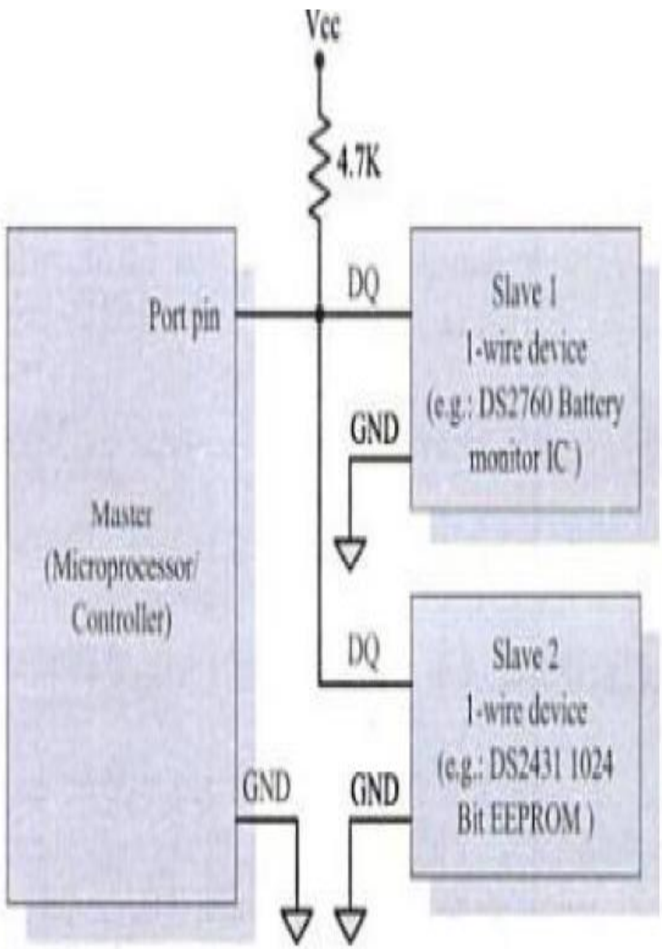


Fig. 2.29 1-Wire Interface bus

- Master device sends a 'Reset' pulse on the 1-wire bus.
- Slave device responds with 'presence' pulse.
- Master device sends ROM command followed by 64-bit address of the slave to which it wants to initiate a communication.
- Then master device send read/write command to or from the slave device.
- Master initiates read data/write data from the device or to the device.

5. Parallel interface

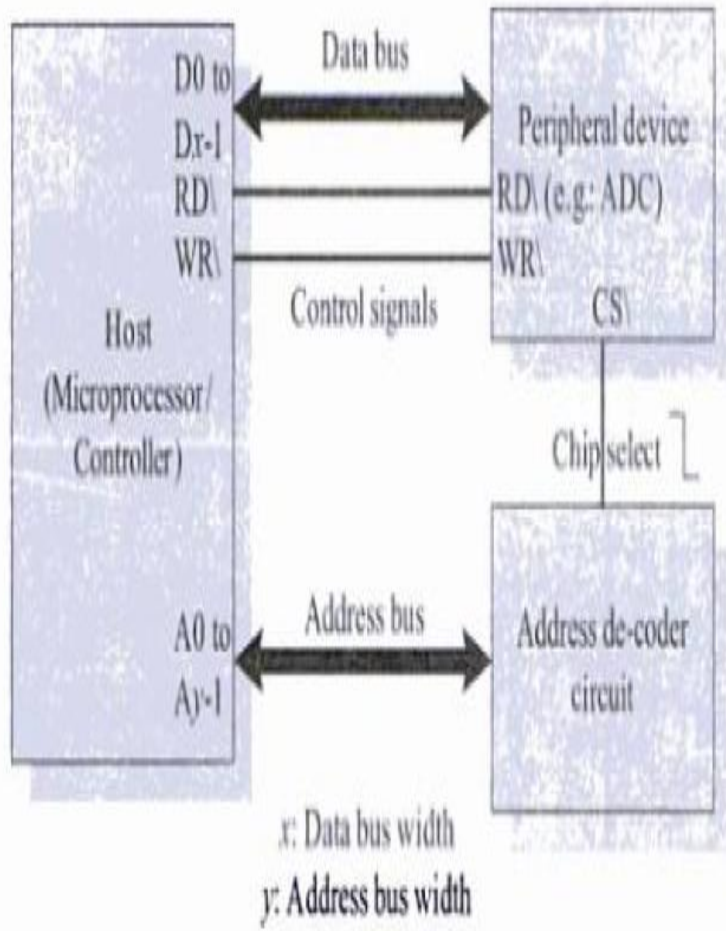


Fig. 2.30 Parallel Interface Bus

- In data transmission, parallel communication is a method of conveying multiple binary digits (bits) simultaneously.
- The communication channel is the number of electrical conductors used at the physical layer to convey bits.
- Parallel communication is and always has been widely used within integrated circuits, in peripheral buses, and in memory devices such as RAM.

External Communication Interfaces

1. **RS-232 C & RS-485:** RS-232C is a legacy, full duplex, wired, asynchronous serial communication interface. RS-232 supports two different types of connectors, namely; DB-9; 9-Pin connector and DB-25: 25-Pin connector. RS-232 supports only point-to-point communication and not suitable for multi-drop communication.RS-485 is the enhanced version of RS-422 and it supports multi-drop communication with up to 32 transmitting devices (drivers) and 32 receiving devices on the bus.

2. **Universal Serial Bus (USB):** Universal Serial Bus (USB) is a wired high speed serial bus for data communication. The USB host can support connections up to 127,including slave peripheral devices and other USB hosts.USB supports 4-types of data transfer namely

   Control transfer: used to query, configure and issue commands to the USB device

   Bulk transfer: used to send block of data.

   Isochronous data transfer: used for real time data communication.

   Interrupt transfer: to transfer small amount of data.

3. IEEE 1394 (Firewire):IEEE 1394 (Firewire) is a wired, isochronous high speed serial communication bus. It is also known as High Performance Serial Bus (HPSB).1394 is a popular communication interface for connecting embedded devices like Digital Camera, Camcorder, Scanners to desktop computers for data transfer and storage.

Unlike USB interface, IEEE 1394 doesn't require a host for communicating between devices. For example, you can directly connect a scanner with a printer for printing. The data rate supported by 1394 is far higher than the one supported by USB 2.0 interface. The 1394 hardware implementation is much costlier than USB implementation.

4. Infrared Data Association (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 we are very familiar with it. The remote control of your TV, VCD player, etc works on Infrared data communication principle.

5. **Bluetooth:** 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.It supports a data rate of up to 1 Mbps and a range of approximately 30 feet for data communication. 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.
6. **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.

7. **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.

   ZigBee device categories are as follows: **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.

8. **General Packet Radio Service (GPRS):** GPRS is a communication technique for transferring data over a mobile communication network like GSM. GPRS supports a theoretical maximum transfer rate of 17.2 kbps. The GPRS communication divides the channel into 8 timeslots and transmits data over the available channel.

Embedded Firmware

- Embedded firmware refers to the **control algorithm (Program instructions)** and/or the **configuration settings that an embedded system developer dumps** into the code (program) memory of the embedded system.
- There are various methods available for developing the embedded firmware.

  1.Write the program in **high level languages like Embedded C/C++ using an** Integrated Development Environment.

  2. Write the program in **Assembly language** using the instructions supported by your application’s target processor/controller.
- The process of converting the program written in either a high level language or processor/controller specific Assembly code to machine readable binary code is called **‘HEX File Creation’.**
- If the program is written in Embedded C/C++ using an IDE, **the cross compiler included** in the IDE converts it into corresponding processor/controller understandable ‘HEX File’.
- If we are following the Assembly language based programming technique, we can use the utilities supplied by the processor/controller vendors to convert the source code into ‘HEX File’.
- Embedded software development **process is tedious and time consuming.**

Reset Circuit

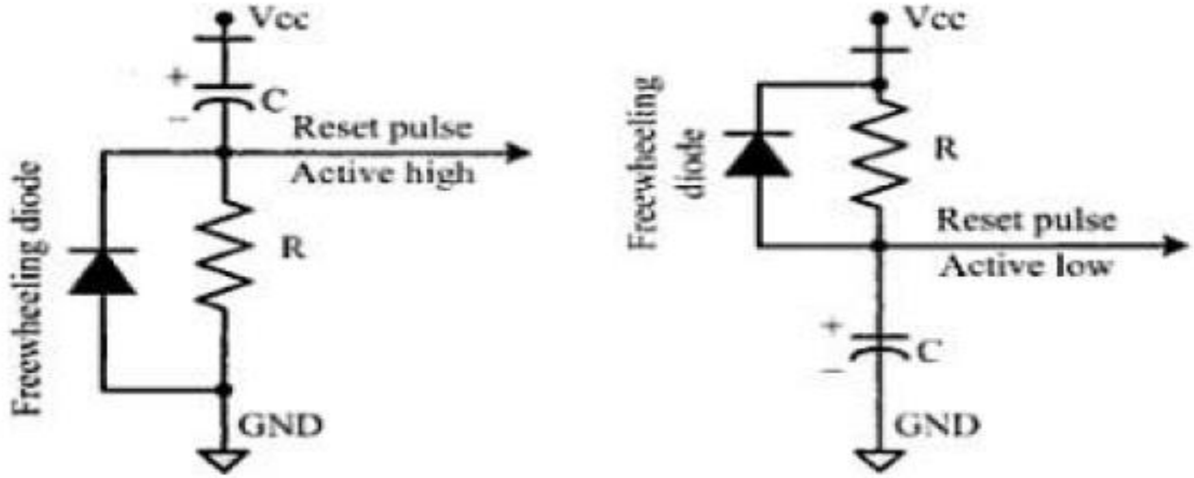


Fig. 2.35 **RC based reset circuit**

Reset Circuit

- The reset circuit is essential to ensure that the device is not operating at a voltage level where **the device is not guaranteed to operate, during system power ON. The reset signal brings** the **internal registers and the different hardware systems of the processor/controller to a** known state and starts the firmware execution from the reset vector.
- The reset signal can be either active high (The processor undergoes reset when the reset pin of the processor is at logic high) or active low (The processor undergoes reset when the reset pin of the processor is at logic low). Since the processor operation is synchronized to a clock signal, the reset pulse should be wide enough to give time for the clock oscillator to stabilize before the internal reset state starts.
- Some microprocessors/controllers contain built-in internal reset circuitry and they don’t require external reset circuitry. Figure illustrates a resistor capacitor based passive reset circuit for active high and low configurations. The reset pulse width can be adjusted by changing the resistance value R and capacitance value C.

Brown-out protection circuit

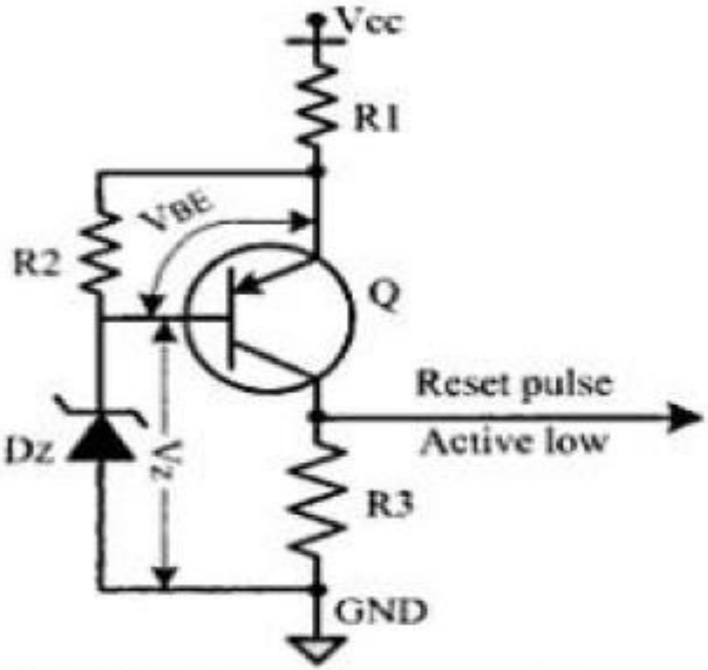
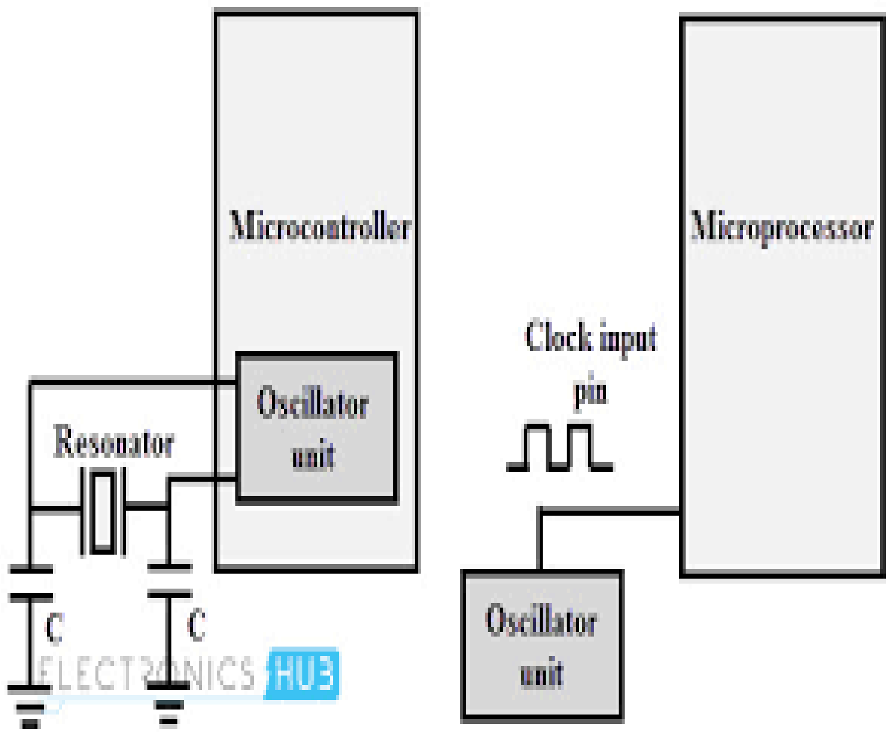


Fig. 2.36 Brown-out protection circuit with Active low output

- The brown-out protection circuit prevents the processor/controller from unexpected program execution behavior when the supply voltage to the processor/controller falls below a specified voltage.
- The zener diode Dz and transistors Q forms the heart of the circuit. The transistor conducts always when supply voltage Vcc is greater than the sum of V_{BE} and Vz (zener voltage)
- The transistor stop conducting when supply voltage falls below the sum of V_{BE} and Vz.
- Zener voltage must be selected with required voltage.
- The values of R1,R2 and R3 can be selected based on electrical characteristics.

Oscillator Unit(Quartz crystal oscillator)



- The Oscillator unit generates clock signals for synchronizing the operations of the processor.
- Clock pulses are required any digital systems. Microprocessor embedded system digital system. Some microcontrollers have on chip oscillator circuit and require only a crystal to be connected externally, while some require external oscillator circuit.
- The oscillator circuit generates the clock pulses **so that all the internal and external operations are synchronized.**

Real-Time Clock (RTC)

- Real-Time Clock (RTC) is a system component responsible for keeping track of time.
- RTC holds information like current time (in hours, minutes and seconds) in 12 hour/24 hour format, date, month, year, day of the week, etc. and supplies timing reference to the system.
- RTC is intended to function even in the absence of power.
- The RTC chip contains a microchip for holding the time and date related information and backup battery cell for functioning in the absence of power, in a single IC package.
- The RTC chip is interfaced to the processor or controller of the embedded system.
- For Operating System based embedded devices, a timing reference is essential for synchronizing the operations of the OS kernel.

Watchdog Timer

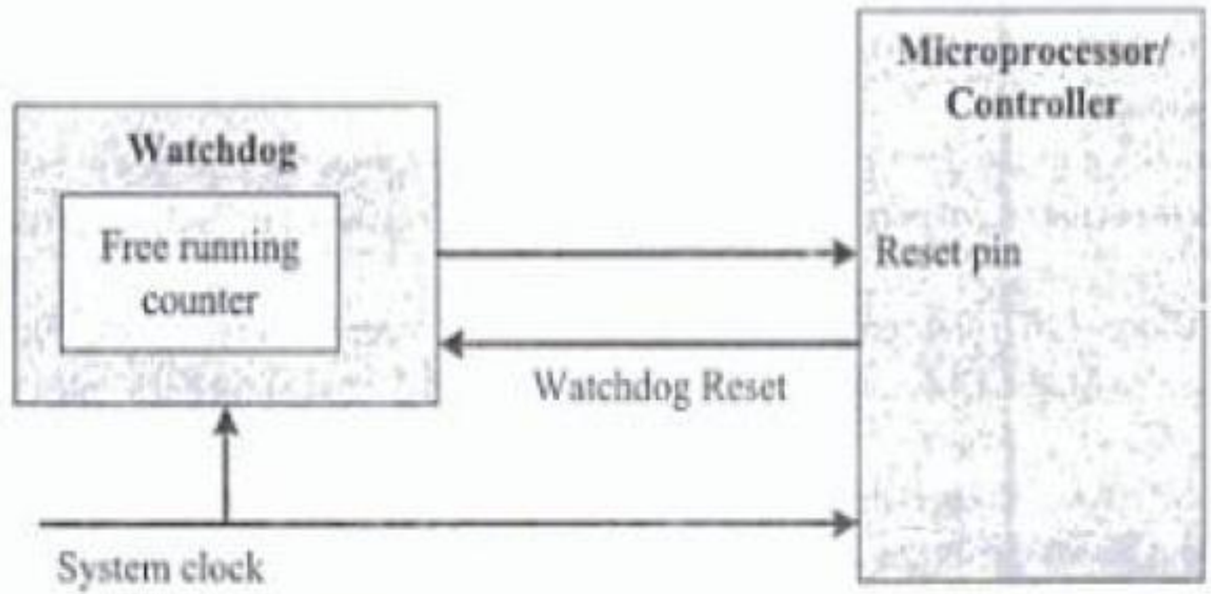


Fig. 2.38 **Watchdog timer for firmware execution supervision**

- A watchdog is to monitor the firmware execution and reset the system processor/microcontroller when the program execution hangs up or generates an Interrupt in case the execution time for a task is exceeding the maximum allowed limit.
- If the firmware execution doesn’t complete due to malfunctioning, within the time required by the watchdog to reach the maximum count, the counter will generate a reset pulse and this will reset the processor (if it is connected to the reset line of the processor).
- Most of the processors implement watchdog as a built-in component and provides status register to control the watchdog timer (like enabling and disabling watchdog functioning) and watchdog timer register for writing the count value.
- If the processor/controller doesn’t contain a built in watchdog timer, the same can be implemented using an external watchdog timer IC circuit.