MODULE - 3

Embedded Systems –
Definition
Embedded systems vs general computing systems
Classification of Embedded Systems
Major application areas of Embedded Systems
Elements of an Embedded System
Core of the Embedded System
Microprocessor vs Microcontroller
RISC vs CISC
Harvard vs Von-Neumann

Sensors and Interfacing –
Instrumentation and control systems
Transducers
Sensors
Actuators
LED and 7-Segment LED Display
Stepper Motor
Relay
Piezo Buzzer
Push Button Switch
Keyboard

Communication Interface
UART
Parallel Interface
USB
Wi-Fi
GPRS

DEFINITION

An embedded system is a microcontroller or microprocessor based system which is designed to perform a specific task in combination of both hardware and software.
E.g. Electronic Toys, Mobile Handsets, Washing Machines, Air Conditioners, Automotive Control Units, Set Top Box, DVD Player etc...
For example, a fire alarm; it will sense only smoke.
Laser printers; it only prints
Firmware: programming instructions, referred to as firmware, are stored in read-only memory (ROM).

Embedded Systems Vs General Computing Systems

General Computing System	Embedded Systems
It is microprocessor based system	It is microcontroller based system
A computer needs human interaction to perform tasks.	Embedded device does not need human interaction to perform tasks.
Architecture examples: Analog / Digital computer, Hybrid computer, Harvard /Von Neumann architecture, Reduced instruction set computer	**Architecture examples:** Small Scale Embedded System, Medium Scale Embedded Systems, Sophisticated or Complex Embedded Systems
It has 2 parts: Hardware and Software.	It has 3 parts: Hardware, Firmware and Software.
It can perform many tasks.	It performs specific tasks
(End user programmable) can be reprogrammed to for a new purpose.	(Not end user programmable) only for a specific set of purposes.
Computers are usually bigger in size with larger hardware and input output devices attached to it.	Embedded Devices are smaller in size than Computers, with limited hardware.
Power consumption is high	Power consumption is less

Classification of Embedded Systems

- ❑ Based on Generation
- ❑ Based on Complexity & Performance Requirements
- ❑ Based on deterministic behavior
- ❑ Based on Triggering

Based on Generation

First Generation

The earlier first-generation embedded systems were built around 8-bit microprocessors and 4-bit microcontrollers. Such embedded system possesses simple hardware and firmware developed using assembly code.
Ex: Digital telephone keypads, stepper motor control units.

Second Generation

After the evolution of the second generation embedded systems, the 8-bit processor and 4-bit controllers are replaced by 16-bit microprocessors and 8-bit microcontrollers. They are more powerful and complex compared to previous generation processors.
Ex: Data acquisition systems, SCADA systems.

ird Generation

mbedded Systems built around high performance 32-bit microprocessors and 16-bit microcontrollers. Hence, its operation has become much more powerful and complex than the second generation.

During this period, domain-specific processors/controllers like Digital Signal Processors (DSP), Application-Specific Integrated Circuits (ASICs) and the concept of instruction pipelining, embedded real-time operating system evolved into the embedded system industry.

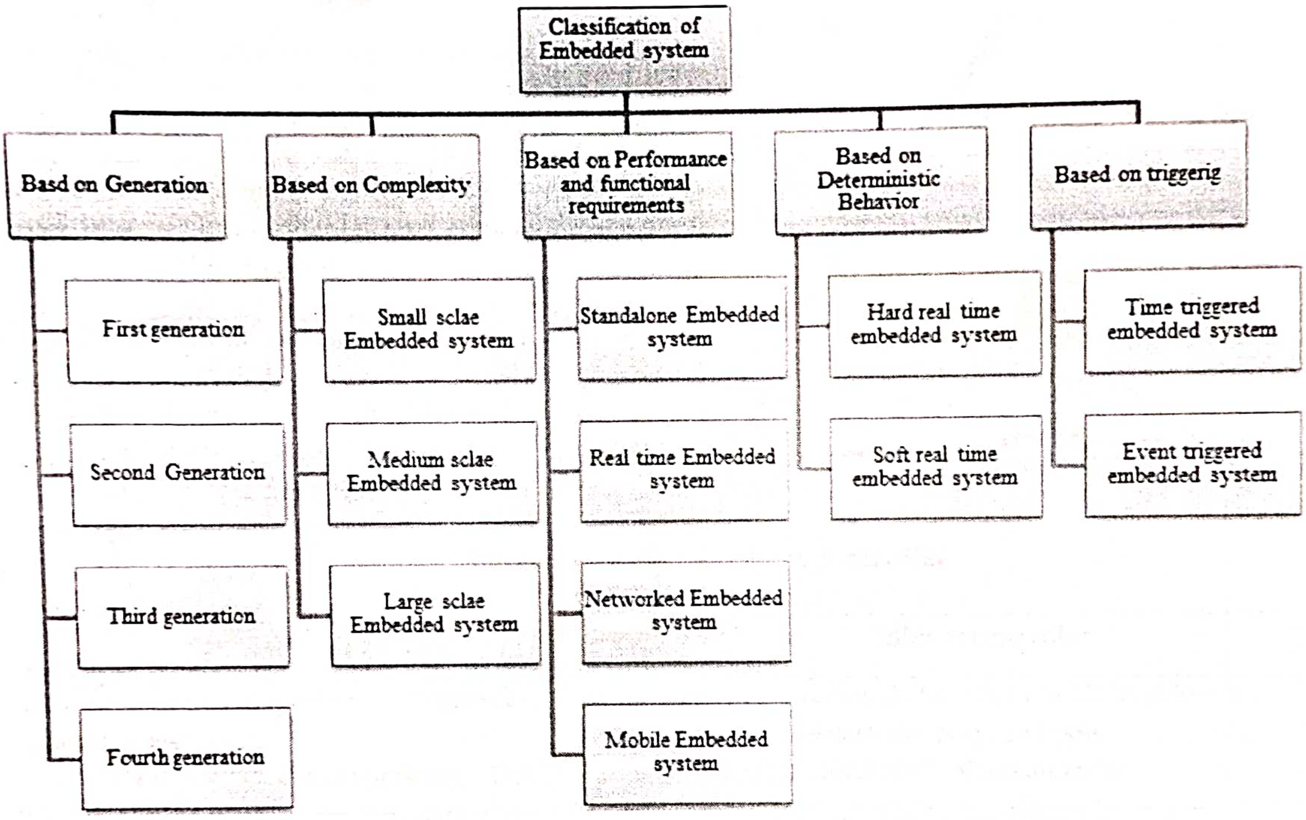
Ex: Robotics, industrial process control, embedded networking.

Fourth Generation

Embedded Systems built around System on Chips (SoCs), Re-configurable processors and multi-core processors, coprocessors also emerged into the embedded market to add more powerful performance

These systems also make use of the high-performance real-time operating system for their operation.

Ex: Smart devices, digital cameras, etc.



Based on Complexity and Performance Requirements

Small Scale Embedded Systems

Small Scale Embedded Systems are built with a single 8 or 16-bit microprocessor or controller. The main programming tools used are an editor, assembler, cross assembler and integrated development environment (IDE).

The hardware and software complexities in small-scale embedded system are very low. It may or may not contain an operating system for its functioning. An electronic toy is an example for a small-scale embedded system.

Medium Scale Embedded Systems

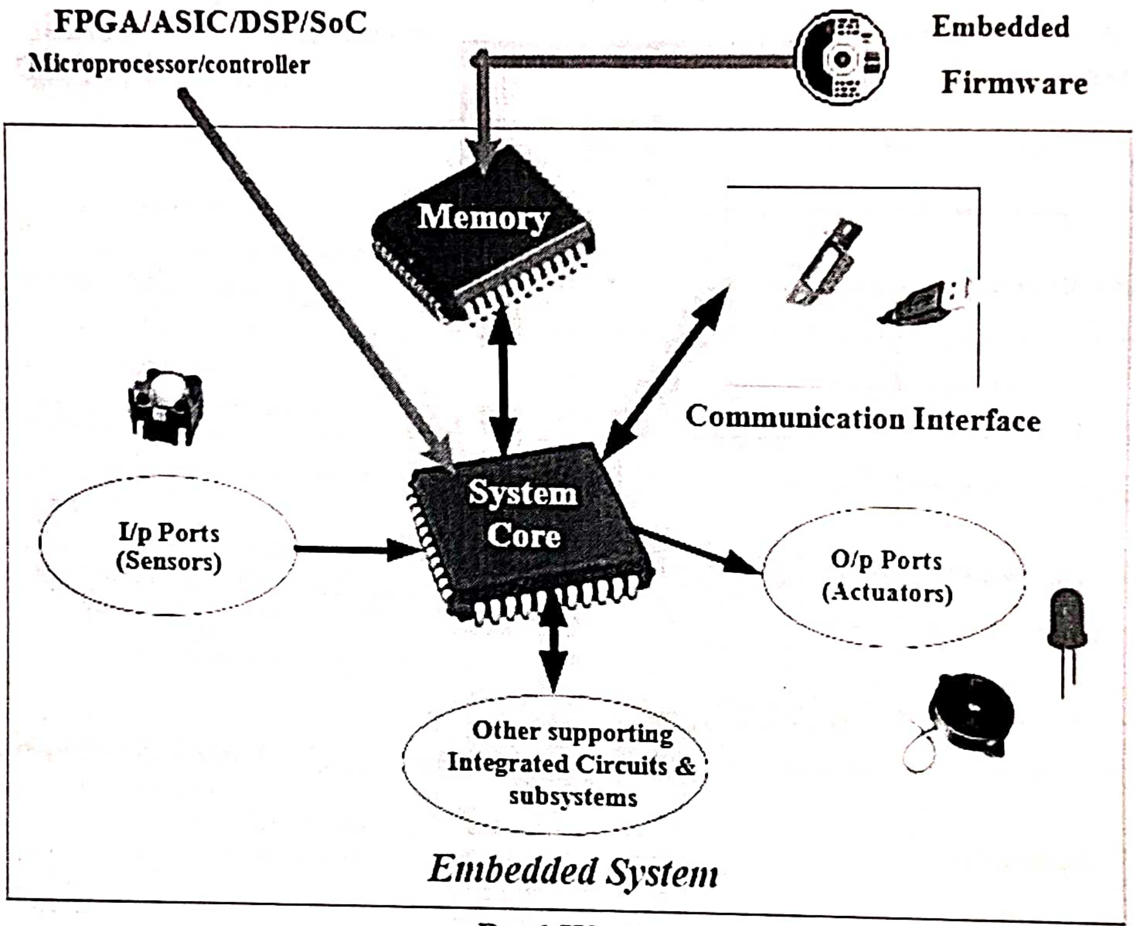
The Embedded system with medium performance 16-bit or 32-bit microprocessor or controller, ASICs or DSPs fall under the medium scale embedded systems. They have both hardware and software complexities.

The main programming tools used are C, C++, JAVA, Visual C++, RTOS, debugger, source code engineer tool, simulator and IDE.

Large scale Embedded Systems

The embedded systems have highly complex hardware and software, built around 32-bit or 64-bit processors/controllers, RISC processors, SoC, scalable and configurable processors. They are also called sophisticated embedded systems.

They are used for cutting-edge applications that need hardware and software co-design, where components have to be assembled into the final system. They also contain a high-performance real-time operating system for task scheduling, prioritization and management.



Real World

Major Application Areas of Embedded Systems

- **Consumer Electronics**:
  Cam-corders, Digital Cameras, Laptop, CCTV etc.
- **Household Appliances**:
  Television, DVD players, Washing machine, Fridge, Microwave Oven etc.
- **Home Automation and Security Systems**:
  Air conditioners, sprinklers, Intruder detection alarms, Closed Circuit Television Cameras, Fire alarms etc.
- **Automotive Industry**:
  Anti-lock breaking systems (ABS), Engine Control, Ignition Systems, Automatic Navigation Systems etc.

Telecom:
Cellular Telephones, Telephone switches, Handset Multimedia Applications etc.

- ❑ **Computer Peripherals:**
  Printers, Scanners, Fax machines etc.
- ❑ **Computer Networking Systems:**
  Network Routers, Switches, Hubs, Firewalls etc.
- ❑ **Health Care:**
  X-ray, Scanners, EEG, ECG, BP monitor, pulse monitor etc.
- ❑ **Measurement & Instrumentation:**
  Digital multi meters, Digital CROs, Logic Analyzers PLC systems etc.
- ❑ **Banking & Retail:**
  Automatic Teller Machines (ATM) and Currency counters,
- ❑ **Card Readers:**
  Barcode, Smart Card Readers, Hand held Devices etc.

The Core of the Embedded Systems

The core of the embedded system falls into any one of the following categories.

- ❑ General Purpose and Domain Specific Processors
  - ➢ Microprocessors
  - ➢ Microcontrollers
  - ➢ Digital Signal Processors (DSP)
- ❑ Programmable Logic Devices (PLDs)
- ❑ Application Specific Integrated Circuits (ASICs)
- ❑ Commercial off the shelf Components (COTS)

Microprocessor Vs Microcontroller

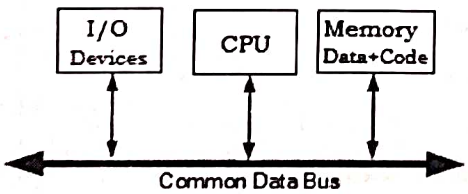
Microprocessor	Microcontroller
Consists of a CPU, performs Arithmetic and Logical operations **Ex: Intel 8086 microprocessor**	Highly integrated chip contains CPU, RAM, on chip ROM/flash memory, I/O ports **Ex: Intel 8051 microcontroller**
It is mainly used in Personal Computers	It is mainly used in an embedded system
Complex and expensive, with a large number of instructions to process.	Simple and inexpensive with less number of instructions to process.
Dependent Unit	Self contained unit
Consumes more power	Consumes less power
Limited power saving options	Includes lot of power saving features
Architecture is based on Von Neumann model	Architecture is based on Harvard architecture
Uses an external bus to interface to RAM, ROM, and other peripherals	Uses an internal controlling bus.

RISC	CISC
Reduced Instruction Set Computer.	Complex Instruction Set Computer.
Software centric design.	Hardware centric design.
Low power consumption.	High power consumption.
Requires more RAM	Requires a minimum amount of RAM
Simple decoding of instruction.	Complex decoding of instruction.
Processors are highly pipelined.	Processors are not pipelined or less pipelined.
Execution time is very less	Execution time is very high
Uses multiple registers.	Uses a single register.
It does not require external memory for calculations	It requires external memory for calculations
Compound addressing mode.	Limited addressing mode.
RISC architecture can be used with high-end applications like telecommunication, image processing, video processing, etc.	CISC architecture can be used with low-end applications like home automation, security system, consumer goods etc.
Small Code Size.	Large Code Size.
Fixed Instruction format (32-bit)	Varying formats (16 to 64 bits for each instruction).
Examples: ARM, PIC, Power Architecture, Alpha, AVR, ARC and the SPARC.	Examples: VAX, Motorola 68000 family, System/360, AMD and the Intel x86 CPUs.

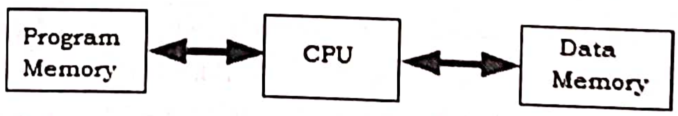
Microcontroller Architectures

1) Von-Neumann and 2) Harvard Architectures

The terms Harvard and Von-Neumann refers to the processor architecture design is depicted as shown below.



Von-Neumann Architecture



Harvard Architecture

VON NEUMANN ARCHITECTURE	HARVARD ARCHITECTURE
It is ancient computer architecture based on stored program computer concept.	It is modern computer architecture based model.
CPU is connected data memory (RAM) and program memory (ROM) by a single memory.	CPU is connected data memory (RAM) and program memory (ROM), separately.
CPU cannot access instructions and data at the same time.	CPU can access instructions and data at the same time.
Same physical memory address is used for instructions and data.	Separate physical memory address is used for instructions and data.
Common bus is used for data and instruction transfer.	Separate buses are used for data and instruction transfer.
The speed of execution is slower. It is because it is not capable of fetching the instructions and data both at the same time.	The overall speed of execution is faster. It is because the processor is capable of fetching both instructions and data at the very same time.
It is cheaper in cost.	It is costly.
Requires less hardware, but low performance.	Requires more hardware, but high performance.
It is used in personal computers and small computers.	It is used in microcontrollers and digital signal processing.

Sensors and Interfacing

Instrumentation and control systems

Instrumentation: Technology of measurement

An instrument is a device that measures or manipulates process physical variables such as flow, temperature, level, or pressure etc.

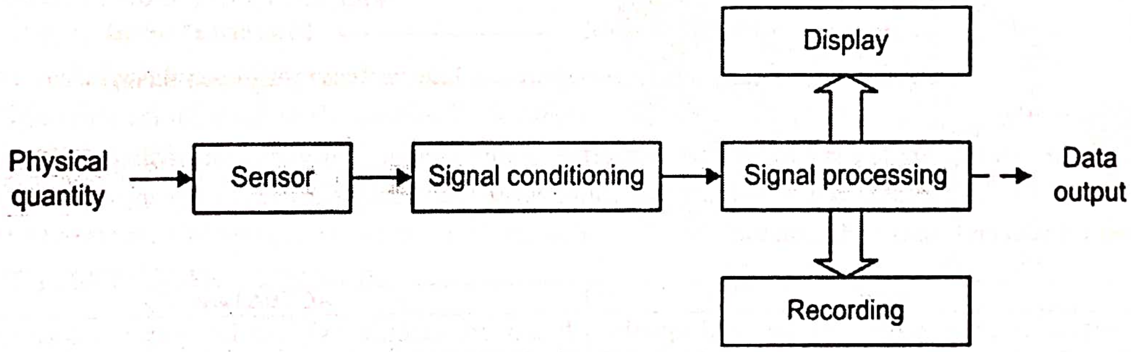


Fig. 1 Instrumentation system

Fig.1 shows the arrangement of an instrumentation system. The physical quantity to be measured (e.g. temperature) acts upon a sensor that produces an electrical output signal.

This signal is an electrical analogue of the physical input but there may not be a linear relationship between the physical quantity and its electrical equivalent.

Also, the output produced by the sensor may be small or may suffer from the presence of noise (i.e. unwanted signals). Therefore, further signal conditioning will be required before the signal will be at an acceptable level and in an acceptable form for signal processing, display and recording. The signal processing may use digital rather than analog signals for this purpose ADC may be required.

Control systems

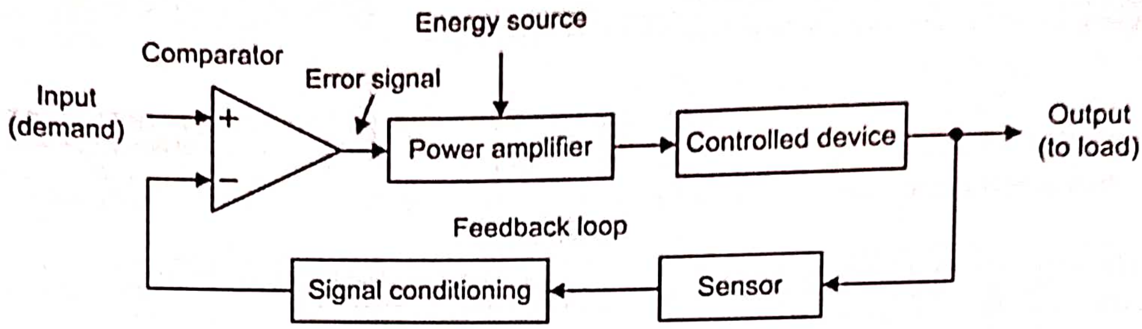


Fig.2 shows the arrangement of a control system

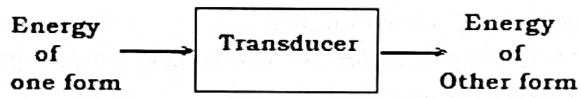
As seen from the fig.2, it uses negative feedback in order to regulate and stabilize the output. It thus becomes possible to set the input or demand (i.e. what we desire the output to be) and leave the system to regulate itself by comparing it with a signal derived from the output (via a sensor and appropriate signal conditioning).

A comparator is used to sense the difference in these two signals and where any changes detected, then input to the power amplifier is adjusted accordingly. This signal is referred to as an *error signal* (it should be zero when the output exactly matches the demand).

The input (demand) is often derived from a simple potentiometer connected across a stable DC voltage source while the controlled device can take many forms (Ex: a DC motor, linear actuator, heater, etc.).

Transducers

Transducers are devices that convert energy in the form (sound, light, heat, etc.,) into an equivalent electrical signal, or vice versa.



Transducer Type

Sensor: **(Physical form)** → **(Electrical form)**

Actuator: **(Electrical form)** → **(Physical form)**

- Examples:
  - A microphone is a transducer converting sound pressure variations into voltage or current.
  - A loudspeaker is a transducer that converts low frequency electric current into audible sounds.

Difference between Sensor and Actuator:

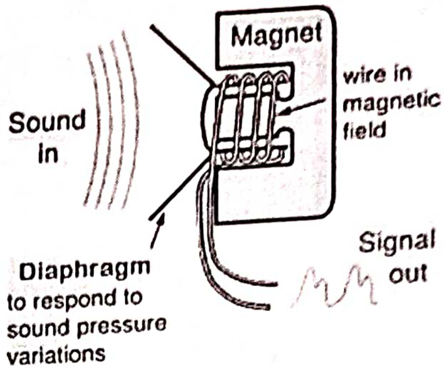
SENSOR	ACTUATOR
It converts physical quantity into electrical signals.	It converts electrical signals into physical quantity.
It takes input from environment.	It takes input from the electric or electronic system.
Sensor generated electrical signals.	Actuator generates heat, motion, vibration,etc.
It is placed at input port of the system.	It is placed at output port of the system.
It is used to measure the physical quantity.	It is used to measure the continuous and discrete process parameters.
It gives information to the system about environment.	It accepts command from the system to perform a function.
Example: Photo-voltaic cell which converts light energy into electrical energy.	Example: Stepper motor where electrical energy drives the motor.

Sensor

A sensor is a transducer which converts energy from physical form to electrical form for any measurement or control purpose. Sensors act as input device.

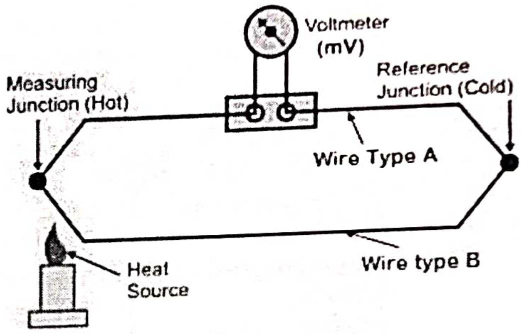
Example 1:

Physical quantity: **sound**
Input transducer: **Microphone**



Diaphragm attached to a coil is suspended in a magnetic field. Movement of the diaphragm causes current to be induced in the coil.

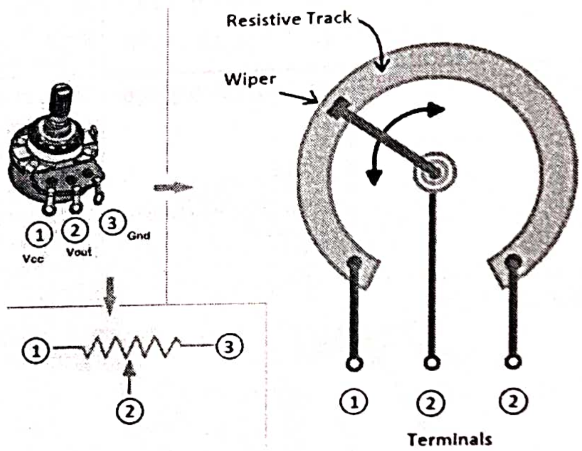
Example 2:
Physical quantity: **Temperature**
Input transducer: **Thermocouple**



Small e.m.f (mV) generated at the junction between two dissimilar metals (e.g. copper & constantan). Requires reference junction (cold) and compensated cables for accurate measurement.

Example 3:

Physical quantity: **Angular position**
Input transducer: **Rotary potentiometer**



Fine wire resistive element is wound around a circular frame. Slider (wiper) attached to the control shaft makes contact with the resistive element.
A stable DC voltage source is connected across the ends of the potentiometer. Voltage appearing at the slider will then be proportional to angular position.

Sensors can be categorized as either active or passive.
An *active sensor* generates a current or voltage output.
A *passive sensor* requires a source of current or voltage and it modifies this by virtue of a change in the sensor's resistance. The result may still be a voltage or current but it is not generated by the sensor on its own.
Sensors can also be classified as either *digital* or *analog*.
The output of a digital sensor can exist in only two discrete states, either 'ON' or 'OFF', 'LOW' or 'HIGH', 'logic 1' or 'logic 0', etc. The output of an analogue sensor can take of voltage or current levels. It is thus said to be continuously variable.
Transducers are also known as Input-Output subsystems. The I/O subsystem facilitates the interaction of the embedded system with external world. The interaction happens through the sensors and actuators connected to

the input and output ports of the embedded system. The sensors may not be directly interfaced to the i ports, instead they may be interfaced through signal conditioning and translating systems like ADC, Opt couplers etc.

Physical quantity	Input transducer	Notes
Liquid level	Float switch (see Fig. 15.7)	Simple switch element which operates when a particular level is detected.
	Capacitive proximity switch	Switching device which operates when a particular level is detected. Ineffective with some liquids.
	Diffuse scan proximity switch	Switching device which operates when a particular level is detected. Ineffective with some liquids.
Pressure	Microswitch pressure sensor (see Fig. 15.4)	Microswitch fitted with actuator mechanism and range-setting springs. Suitable for high-pressure applications.
	Differential pressure vacuum switch	Microswitch with actuator driven by a diaphragm. May be used to sense differential pressure. Alternatively, one chamber may be evacuated and the sensed pressure applied to a second input.
	Piezo-resistive pressure sensor	Pressure exerted on diaphragm causes changes of resistance in attached piezo-resistive transducers. Transducers are usually arranged in the form of a four active element bridge which produces an analogue output voltage.
Proximity	Reed switch (see Fig. 15.4)	Reed switch and permanent magnet actuator. Only effective over short distances.
	Inductive proximity switch	Target object modifies magnetic field generated by the sensor. Only suitable for metals (non-ferrous metals with reduced sensitivity).
	Capacitive proximity switch	Target object modifies electric field generated by the sensor. Suitable for metals, plastics, wood and some liquids and powders.
	Optical proximity switch (see Fig. 15.4)	Available in diffuse and through scan types. Diffuse scan types require reflective targets. Both types employ optical transmitters and receivers (usually infra-red emitting LEDs and photo-diodes or photo-transistors). Digital input port required.
Strain	Resistive strain gauge	Foil type resistive element with polyester backing for attachment to body under stress. Normally connected in full bridge configuration with temperature-compensating gauges to provide an analogue output voltage.
	Semiconductor strain gauge	Piezo-resistive elements provide greater outputs than comparable resistive foil types. More prone to temperature changes and also inherently non-linear.
Temperature	Thermocouple (see Fig. 15.2)	Small e.m.f. generated by a junction between two dissimilar metals. For accurate measurement, requires compensated connecting cables and specialized interface.
	Thermistor (see Fig. 15.6)	Usually connected as part of a potential divider or bridge. An analogue output voltage results from resistance changes within the sensing element.
	Semiconductor temperature sensor (see Fig. 15.6)	Two-terminal device connected as a current source. An analogue output voltage is developed across a series resistor of appropriate value.
Weight	Load cell	Usually comprises four strain gauges attached to a metal frame. This assembly is then loaded and the analogue output voltage produced is proportional to the weight of the load.
Vibration	Electromagnetic vibration sensor	Permanent magnet seismic mass suspended by springs within a cylindrical coil. The frequency and amplitude of the analogue output voltage are respectively proportional to the frequency and amplitude of vibration.

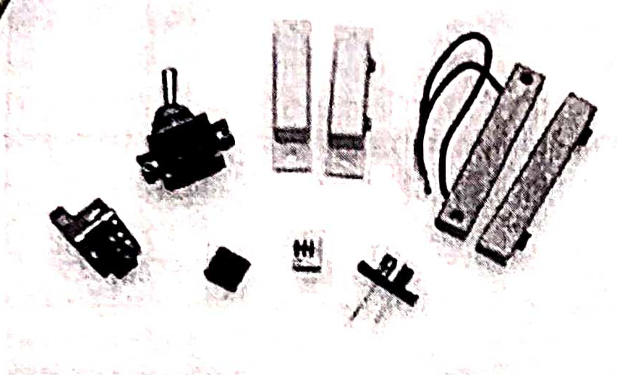


Figure 15.4 Various switch sensors

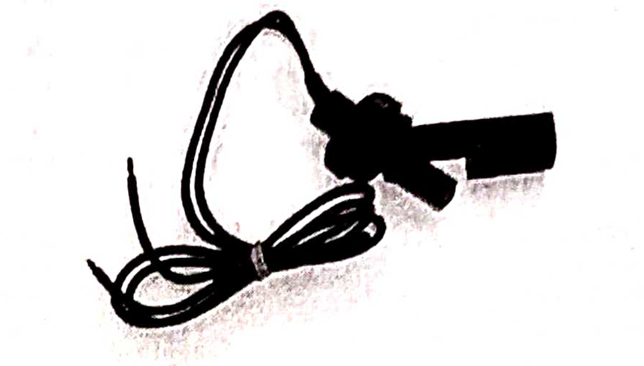


Figure 15.7 Liquid level float switch

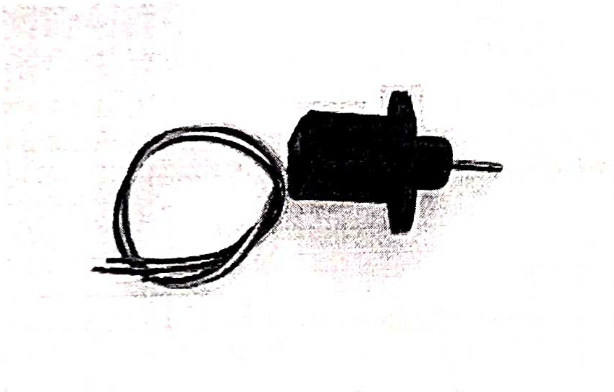


Figure 15.5 Resistive linear position sensor

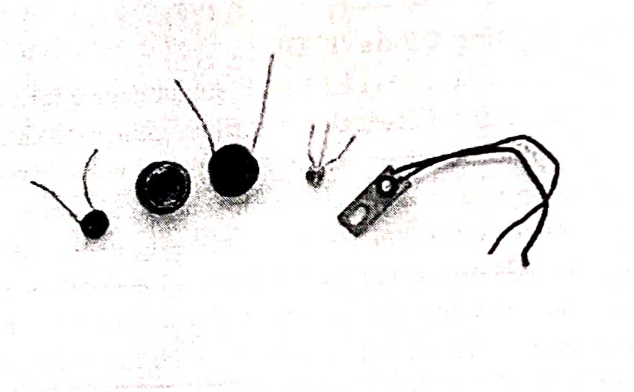


Figure 15.8 Various optical and light sensors

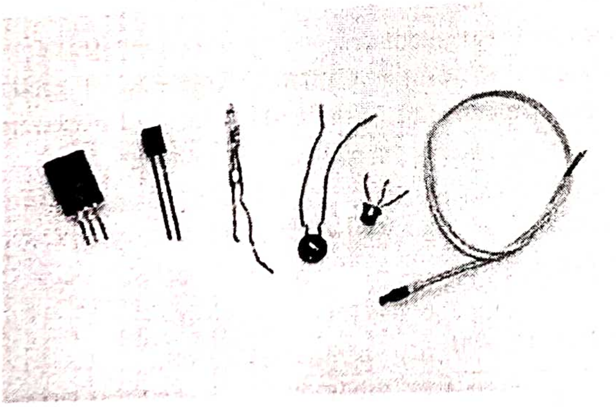


Figure 15.6 Various temperature and gas sensors

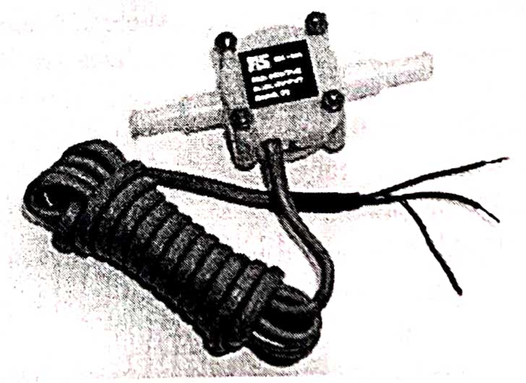


Figure 15.9 Liquid flow sensor (digital output)

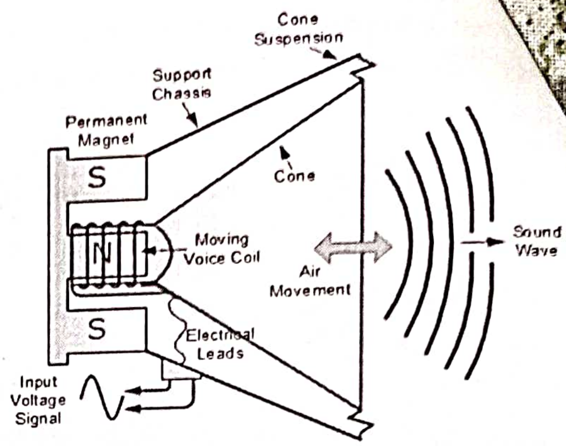
Actuator

Actuator is a transducer which converts electrical signals to corresponding physical action (motion). Actuator acts as an output device.

Example :1

Physical Quantity: **Sound** (pressure change)
Output transducer: **Loudspeaker**

Diaphragm attached to a coil is suspended in a magnetic field. Current in the coil causes movement of the diaphragm which alternately compresses and rarefies the air mass in front of it.

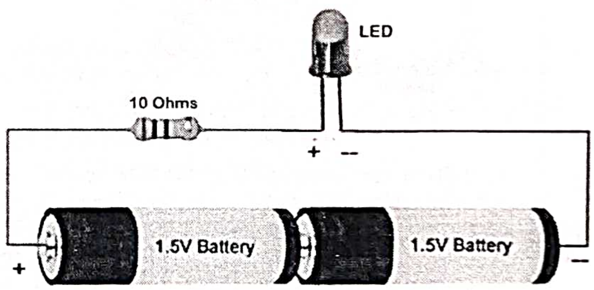


Examples: electric motors, stepper motors, jackscrews, electric muscular stimulators in robots,

Light Emitting Diode (LED)

Light Emitting Diode (LED) is an output device for visual indication in any embedded system. It can be used as an indicator for the status of various signals or situations. Typical examples are indicating the presence of power conditions like device ON, battery low or charging of battery etc.

LED is a p-n junction diode and it contains an anode (+) and a cathode (-). For proper functioning of the LED, the anode is connected to +ve terminal and cathode to the –ve terminal of supply voltage (forward bias condition). The current flowing through the LED must limited to a value below the maximum current that it can conduct. A resister is used in series between the power supply and the resistor to limit the current through it.



- LED can be interfaced to the port pin of microcontroller in two methods:
- **Method:1-** (current sourcing) Anode of LED is connected to port pin. Cathode is connected to ground (0V) through resistor. When port pin of microprocessor goes logic 1, the LED is forward biased and emits light. When the port pin goes 0, LED is *off.* That means port pin *sources* current to LED.
- **Method:2-** (current sinking) Cathode of LED is connected to port pin. Anode is connected to external supply through resistor. LED turns *on* when the port pin is at logic 0. Here port pin *sinks* current, such that the brightness of LED can be increased to the required level. See fig.3.

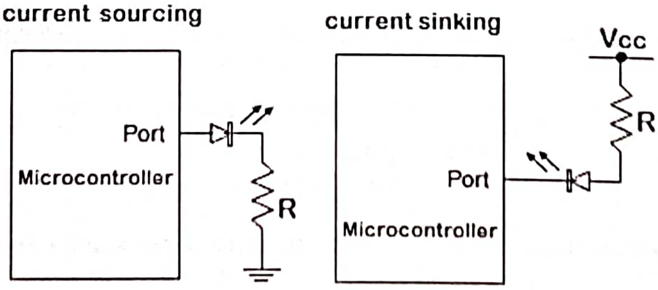


Fig.3 LED interfacing to the port pin of microcontroller

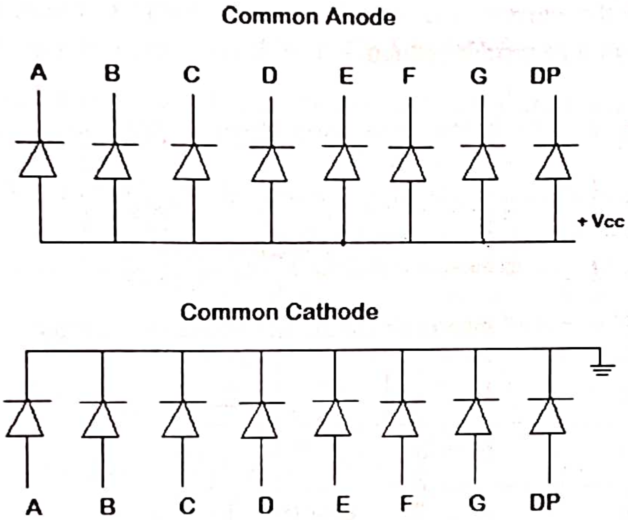
7-Segment LED Display

The 7 – segment LED display is an output device for displaying alpha numeric (0–9 and A–F) characters. It contains eight LED segments arranged in a special form. Out of the 8 LED segments, 7 are used for displaying alpha numeric characters. The LED segments are named A to G and the decimal point LED segment is named as DP. The LED Segments A to G and DP should be lit accordingly to display numbers and characters.

The 7 – segment LED displays are available in two different configurations, namely;

i) common anode and ii) common cathode

In the *common anode* configuration, the anodes of all LEDs connected together to +Vcc, and in the *common cathode* configuration, the cathodes of all LEDs connected together to ground as shown in the fig.4.



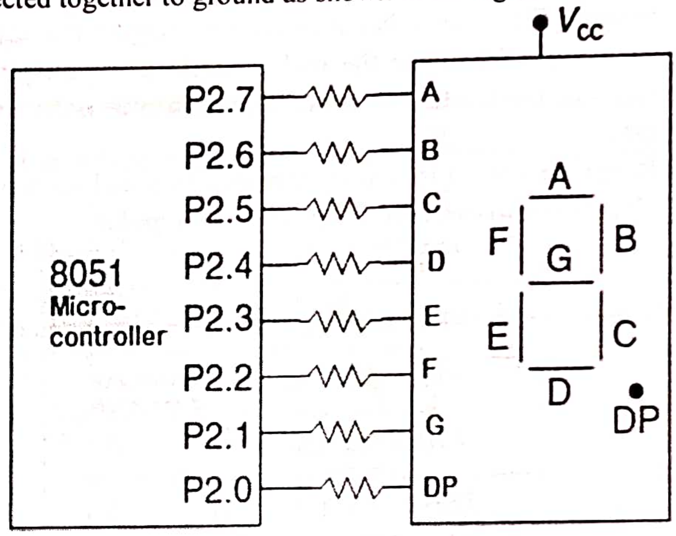


Fig.4 7-segment LED interfacing with microcontroller

Based on the configuration of the 7 – segment LED unit, the LED segment anode or cathode is connected to the port of microcontroller in the order. A segment to the least significant port pin and DP segment to the most significant port pin or vice versa. The current through each segment can be limited by connecting a resistor.

Number	Code	LED status							
		A	B	C	D	D	E	G	DP
0	03	0	0	0	0	0	0	1	1
1	9F	1	0	0	1	1	1	1	1
2	25	0	0	1	0	0	1	0	1
3	0D	0	0	0	0	1	1	0	1
4	D9	1	1	0	1	1	0	0	1
5	49	0	1	0	0	1	0	0	1
6	41	0	1	0	0	0	0	0	1
7	1F	0	0	0	1	1	1	1	1
8	01	0	0	0	0	0	0	0	1
9	19	0	0	0	1	1	0	0	1

7 segment Display for (0 - 9)

In order to display the required numbers (0 - 9) or HEX characters (A - F), the correct combination of segments need to be illuminated based on the type of configuration.

Stepper Motor

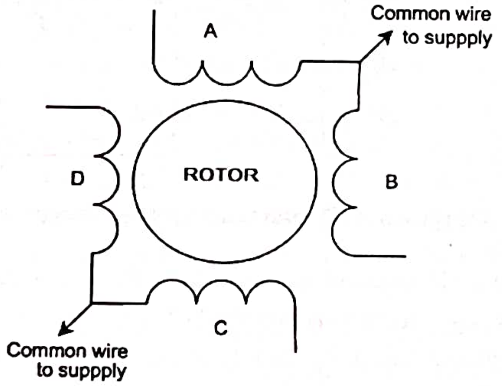
A Stepper motor is an electro-mechanical device which generates discrete rotation in response to dc electrical signals.

It differs from the normal DC motor in its operation. The DC motor produces continuous rotation on applying DC voltage whereas a stepper motor produces discrete rotation in response to the dc voltage applied to it. Stepper motors are widely used in industrial embedded applications, consumer electronic products, robotics control system, dot matrix printers, disk drives, etc.

Stepper motor rotor has a permanent magnet and the stator has four electromagnetic coils which remain stationary. Whenever the coils energized by applying the current, the electromagnetic field is created, resulting the rotation of rotor. Coils should be energized in a particular sequence to make discrete rotation of the rotor.

Based on the coil winding arrangements and to supply current in the coils two phase stepper motor is classified into two types: *Bipolar* and *Unipolar*.

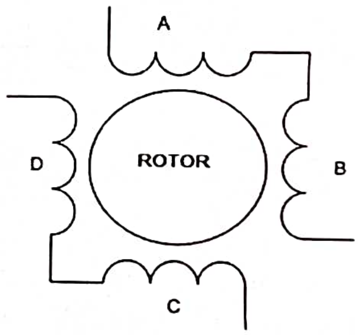
Unipolar



- Unipolar stepper motor contains two windings per phase and has common center connection.
- Current can flows only in a single direction.
- For reversing the motor requires simple circuitry.
- They have less torque and are less efficient.

Bipolar

The "polar" means electrical and magnetic polarity.



- Bipolar stepper motor contains single winding per phase and has no common center connection.
- Current can flow in both directions in all coils.
- For reversing the motor rotation the current flow through the windings is reversed dynamically.
- They have more torque and more efficient, requires complex circuitry.

The circuit diagram for interfacing stepper motor to 8051 is shown in the fig.5. Let P1.0, P1.1, P1.2 and P1.3 port pins are used for controlling the phases A, B, C and D of the stepper motor, respectively. ULN2003 is a current driver IC used for driving the individual phases of the stepper motor as it requires more than 60mA of current. Center tap of each windings of the stepper motor is shorted and connected to + 5V motor supply. It can be energized each winding coils (A, B, C and D) of the motor by sending electric pulses from microcontroller pins P1.0, P1.1, P1.2 and P1.3. Unipolar stepper motor is very common and popular because of its ease of use. There are different modes to drive a stepper motor in which full step drive is explained below.

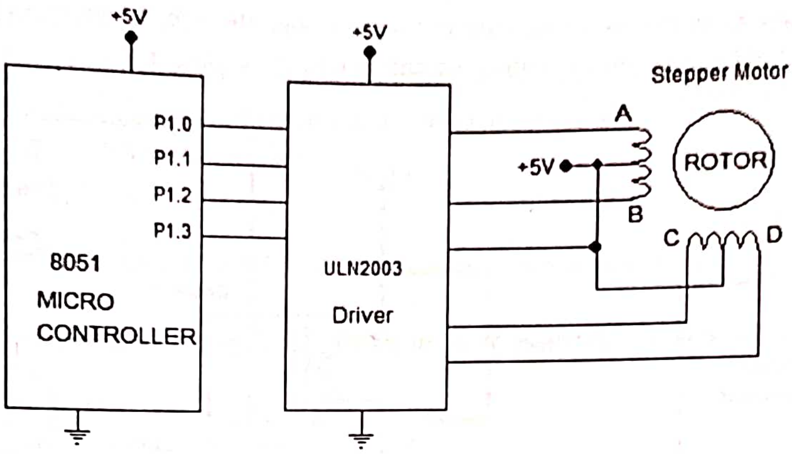


Fig.5 Interfacing of stepper motor with microcontroller

Full Step Drive: In this method two coils are energized at a time, produces more torque. Hence, the power consumption is also high.

The following table is showing the sequence of different coils energized in the following order.

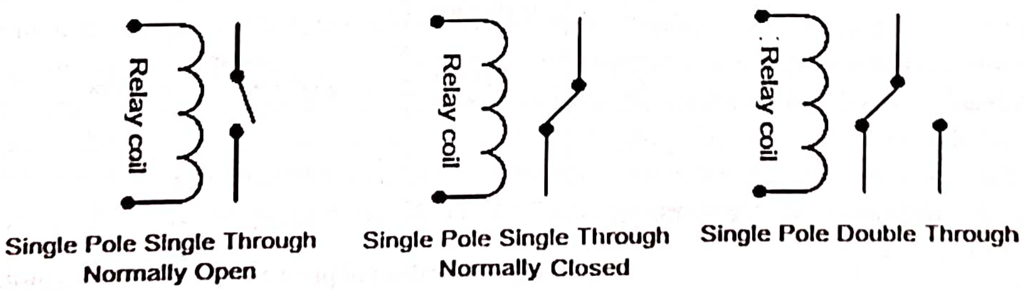
Steps	Coil A	Coil B	Coil C	Coil D	Coil sequence order
1	1	1	0	0	AB = ON, CD = OFF
2	0	1	1	0	BC = ON, AD = OFF
3	0	0	1	1	AB = OFF, CD = ON
4	1	0	0	1	AD = ON, BC = OFF

Working:
When the sequence is followed from step 1 to 4, stepper motor rotates in the *clock wise rotation*. That means, at first when the bit sequence 1100 is sent by the port 1 of microcontroller to the stepper motor, A and B windings are energized (ON) and C and D windings are de-energized (OFF). With an appropriate delay next bit sequences 0110, 0011 and 1101 are followed. Therefore, stepper motor completes one rotation by four steps in the clock wise direction.

Rotation of the stepper motor can be reversed by reversing the order from step 4 to 1.

Relay

An electro mechanical device which acts as a dynamic path selector for signals and power. Relay works on electromagnetic principle.



When a voltage is applied to the relay coil, current flows through the coil, which in turn generates a mag field. The magnetic field attracts the armature core and moves the contact point. The movement of the con point changes the path of power/signal.

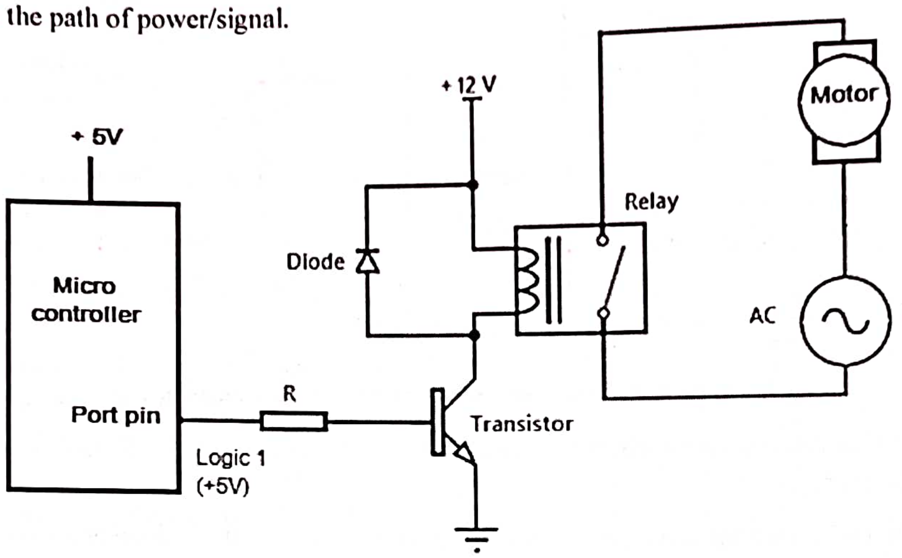


Fig. 6 Interface of relay with microcontroller

The relay is normally open or closed to operate high power units (motor) using a driver circuit (transistor) which is connected to the port pin of the microcontroller is shown in the fig.6. A transistor can be used as the relay driver.

The transistor is turned ON when logic 1 is available on the port pin of the microcontroller; in turn, the relay coil energized and normally closed. The relay is turned OFF by making logic 0 on the port pin. A diode is connected across the relay coil to protect the transistor from damage due to the *back emf* generated in the relay's inductive coil.

Piezo Buzzer

It is a piezoelectric device for generating audio indications in embedded applications. A Piezo buzzer contains a piezoelectric diaphragm which produces audible sound in response to the voltage applied to it. Buzzer can be used as an alarm or as a fire alarm or as an intruder alarm.

Piezoelectric buzzers are available in two types:

1. **Self-driving** and 2. **External driving**

External driving piezo buzzers supports the generation of different tones.

A Piezo buzzer can be interfaced to the port pin of the microcontroller through a driver transistor as shown in the fig. The transistor is turned ON when logic 1 is available on the port pin of the microcontroller; in turn, transistor drives the buzzer. Current will flow through the buzzer and a relative sound will be heard. The tone can be varied by applying a variable pulse train to the piezoelectric buzzer by programming the microcontroller. The model circuit is shown in the fig. 7.

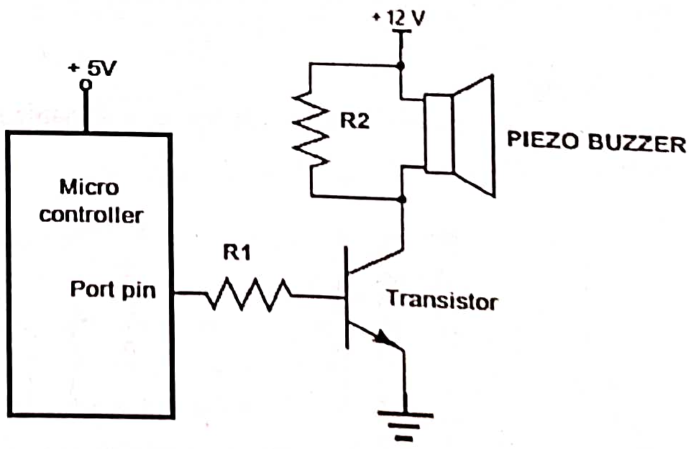


Fig. 7 Interface of piezo buzzer with microcontroller

Push button switch

The Push-button is a basic input device in the embedded system. It is used to control the operation of any output device using the microcontroller or control unit. Push button is used for generating a momentary pulse. Push button switch comes in two configurations:

1. Push to Make and

2. Push to Break

In the push to make configuration the switch is *normally in the open state* and makes a circuit contact when it is pushed or pressed.

In the Push to Break configuration, the switch *normally in the closed state* and breaks the circuit contact when it is pushed or pressed.

Interfacing a button switch with microcontroller (see fig. 8)

To interface a push-button with a port pin of the microcontroller, one end of the push-button is grounded, while the other end is connected to the port-pin of the microcontroller. The microcontroller port-pin is pulled HIGH with the help of a pull-up resistor R1.

When the button is open, the port pin receives logic 1 through the pull-up resistor and LED turns ON. When the button is pressed, the port pin receives logic 0, since it is connected to the ground directly via the closed push-button and LED turned OFF.

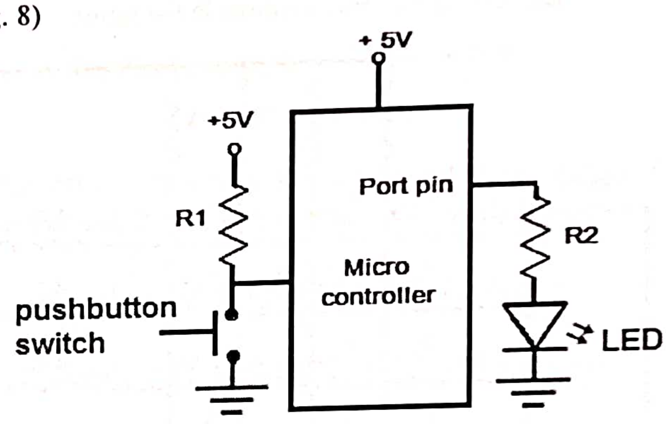


Fig.8 Interfacing a button switch with microcontroller

Keyboard

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. In case of handling large number of key requirements matrix keyboard is an optimum solution. Matrix keyboard greatly reduces the number of interface connections. Matrix keyboard connects the keys in a row column fashion. The key press in matrix keyboard is identified with row-column scanning technique.

For example, for interfacing 16 keys, in the direct interfacing technique 16 port pins are required, where as in the matrix keyboard only 4 columns and 4 rows are required for interfacing 16 keys.

4x4 matrix keyboard interface

4×4 matrix keypad consists of 4 rows and 4 columns and each switch is placed between the rows and columns. When a key is pressed, a row and a column make a contact. Otherwise, there is no connection between rows and columns.

In order to detect the pressed key, the following points to be noted worthy.

i) the rows configured as an output port, making each row logic 0 and
ii) the columns configured as an input port, making each column at logic 1
iii) If all rows = 0000, and all columns = 1111, detecting no key is pressed
iv) If one of the column bits read logic 0, detecting a key is pressed
v) To detect the key pressed, microcontroller undergoes scanning technique by reading each row

Scanning procedure:

i) If all column bits are 1111, no switch is pressed.

ii) If any key is pressed, one of the column bits read logic 0. To find the pressed key is in which row microcontroller sends logic 0 on the first row while other rows making logic 1, (say to find the key pressed in row-1: R_1= 0, $R_2R_3R_4$ = 111) and the column bits are read and then also if all column bits are 1111, no key is pressed.

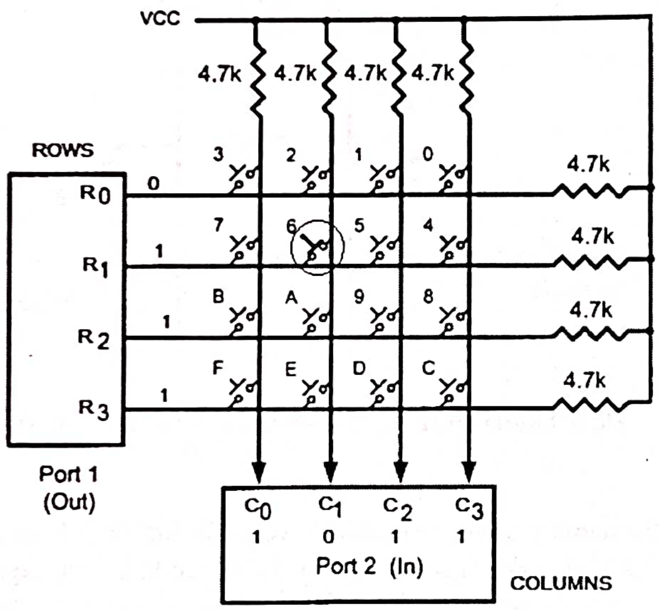
iii) Next, second row is made logic 0 and set all others rows at logic 1 (say to find the key pressed in row-2: R_2 = 0, $R_1R_3R_4$ = 111) and the column bits are read and if all column bits are 1111, no key is pressed.

iv) This process is repeated for all the rows until to find in which case the column bits and row bits are equal in logic.

Example: Scanning procedure to find the pressed key (6) for the fig.9.

a) At first assume all column bits $C_1C_2C_3C_4$ = 1111, when no key is pressed.

b) When key (6) is pressed, column bits become $C_1C_2C_3C_4$ = 1011, indicating that key (6) pressed is in the second column as shown in the figure.



Scanning rounds	Rows $R_1R_2R_3R_4$	Columns $C_1C_2C_3C_4$	Key detected
1	0 1 1 1	1 1 1 1	No
2	**1 0 1 1**	**1 0 1 1**	**Yes**
3	1 1 0 1	1 1 1 1	No
4	1 1 1 0	1 1 1 1	No

Fig.6 scanning procedure to find the pressed key (6)

c) To find the pressed key is found in which row, microcontroller will make logic 0 on the first row while other rows at logic 1 (i.e, R_1 = 0, $R_2R_3R_4$ = 111) as shown in the fig. and check all the column bits. Here all column bits are $C_1C_2C_3C_4$ = 1111, indicating the key pressed is not available in the first row.

d) Next, second row is made logic 0 and make all other rows at logic 1 (i.e, R_1 = 1, R_2 = 0, R_3R_4 = 11) and the column bits are read. For this scanning round, column bits become $C_1C_2C_3C_4$ = 1011. Indicating key (6) is in the second row - second column. See the table given above.

Debouncing

Since keys are mechanical devices, there is a possibility for de-bounce issues, (multiple key press effect for a single key press). To prevent this,

1. Hardware key de-bouncer circuits and
2. Software key de-bounce techniques available.

- The software key de-bouncing technique doesn't require any additional hardware and is easy to implement.
- In the software de-bouncing technique, on detecting a key press, the key is read again after a delay.
- If the key press is a genuine one, the same key status will remain on the second read also.

Pull-up resistors are connected to limit the current that flows to the Row line on a key press.

Communication Interface

Communication interface is an essential process for communicating with various subsystems of the embedded system and with the external world.
The communication interface can be viewed in two different perspectives; namely;
1. Device/board level communication interface (Onboard Communication Interface)
2. Product level communication interface (External Communication Interface)

1. Device/On board level communication interface

The communication channel which interconnects the various components within an embedded product is referred as *on-board level communication interface.*

Examples: Serial interfaces like I2C, SPI, UART, 1-Wire and Parallel bus interface

2. External Communication Interface

This level communication is responsible for data transfer between the embedded system and other devices or modules. It can be either wired media or wireless media and it can be a serial or parallel interface.

Examples:

i) For wireless communication interface: Infrared (IR), Bluetooth (BT), Wireless LAN (Wi-Fi), Radio Frequency waves (RF), GPRS etc.

ii) For wired communication interfaces: RS-232C/RS-422/RS 485, USB, Ethernet (TCP-IP), IEEE 1394 port, Parallel port etc.

[NOTE: Mobile Communication Equipment –
is an example of an embedded system with external communication interface]

On board level communication interface systems are classified into

i) I2C (Inter Integrated Circuit) Bus
ii) SPI (Serial Peripheral Interface) Bus
iii) UART (Universal Asynchronous Receiver Transmitter)
iv) 1-Wires Interface
v) Parallel Interface

Universal Asynchronous Receiver Transmitter (UART)

In UART communication, two UARTs will communicate directly with each other. Only two wires are needed to transmit data between two UARTs. Data flows from the TXD pin of the transmitting UART to the RXD pin of the receiving UART and vice versa, as shown in the fig.10.

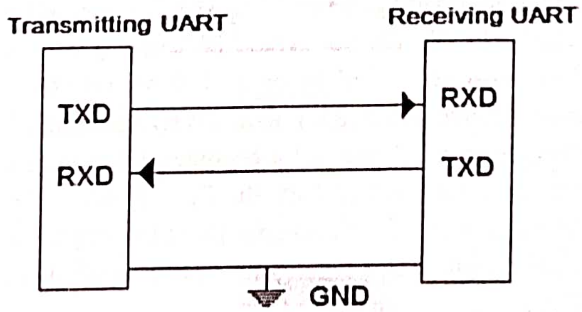
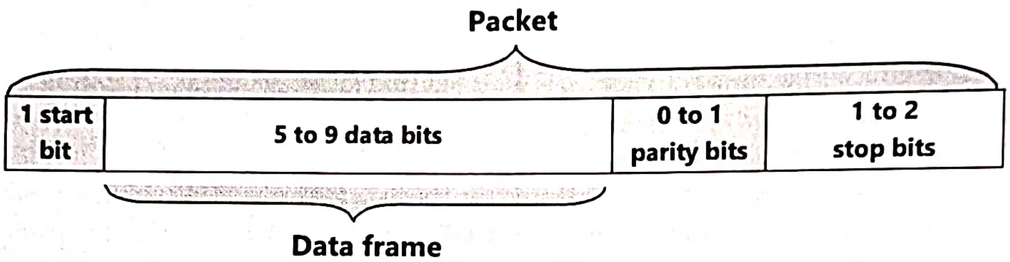


Fig. 10 Universal Asynchronous Receiver Transmitter (UART)

UART data transmission is an asynchronous form of serial data transmission, which means there is no c[illegible] signal to synchronize the output of transmitting UART and the input of receiving UART. Instead of a clo[illegible] signal, the transmitting UART consists *start* and *stop* bits with the *data packet* being transferred. The transmitting UART sends a single bit at a time. After sending one bit, the next bit is sent. In this way, all the data bits are sent to the receiver UART with a predefined *baud rate* (there will be a certain delay in transmitting each bit). For example, to send one byte of data at 9600 baud rate, each bit is sent at 108 µsec delay.

UART transmitted data is organized into *packets*. Each packet contains 1 start bit, 5 to 9 data bits (depending on the UART), an optional *parity* bit, and 1 or 2 stop bits as shown in the below figure.



Start bit: The UART data transmission line is normally held at a high voltage level when it's not transmitting data. To start the transfer of data, the transmitting UART pulls the transmission line from high to low for one (1) clock cycle. When the receiving UART detects the high to low voltage transition, it begins reading the bits in the data frame

Data Frame: The data frame contains the actual data being transferred. It can be 5 bits up to 8 bits long if a parity bit is used. If no parity bit is used, the data frame can be 9 bits long.

Parity bit: It counts the number of 1's and checks if the total is an even or odd number. If the parity bit = 0 (even parity), otherwise odd parity. The parity bit indicates if any data has changed during transmission. Bits can be changed by electromagnetic radiation, mismatched baud rates, or long-distance data transfers.

Stop Bits: To indicate the end of the data packet, the sending UART drives the data transmission line from a low to a high voltage for one to two bit(s) duration.

UART provides hardware hand-shaking signal support for controlling the serial data flow. It was used in IBM computers. Nowadays, even though USB has replaced UART in computers and other devices, it is still used in many serial data communication applications like GPS receivers, blue-tooth modules, GSM and GPRS modems, Wireless Communication Systems, RFID based applications etc.

Parallel Interface

It is used for communicating with peripheral devices (Ex: ADC) which are memory mapped to host of the system. Host has a parallel bus and it has the control over *read/ write* operation.

Communication through parallel bus is controlled by control signal interface between host and device. The control signals for communication includes *read* (RD), *write* (WR) signal and *Chip Set* (CS) signal.

The peripheral devices normally contain a CS line and it becomes active only when this line is asserted by the host processor through the address decoder circuit (See the fig.11). When the address is selected, CS line is activated by decoder circuit. The direction of data transfer (host to peripheral device or peripheral device to host) can be controlled through the control signal lines for 'Read' and 'Write'. Only the host processor has control over the 'Read' and 'Write' control signals.

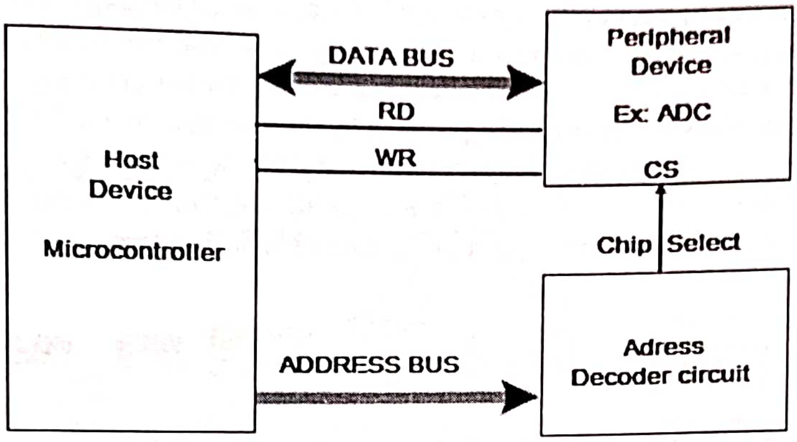


Fig. 11 Parallel Interface

Address and data bus width must match with the host device and the peripheral device. If the peripheral device wants to start communication, it can inform the same to microcontroller through the interrupts (An interrupt is a signal to the host processor raised by the peripheral device indicating an event that needs immediate attention). Parallel data communication offers highest speed for data transfer.

External Communication Interface

i) Universal Serial Bus (USB)

Universal Serial Bus (USB) is a wired high speed serial bus for data communication that enables universal communication between the peripheral devices and a host controller.

USB Topology: The USB communication system follows a star topology with a USB host at the center and one or more USB peripheral devices (or USB hosts connected to it) as shown in the fig.12. The USB host controller has an embedded hub called the *root hub*. A hub is a common connection point that allows multiple devices to connect in the network. A hub contains multiple ports. They are of two types: self powered and bus powered hubs. A USB host can support connections up to 127, including slave peripheral devices and other USB hosts. Each device receives unique address from the host (PC). USB transmits data in packet format. Each data packet has a standard format.

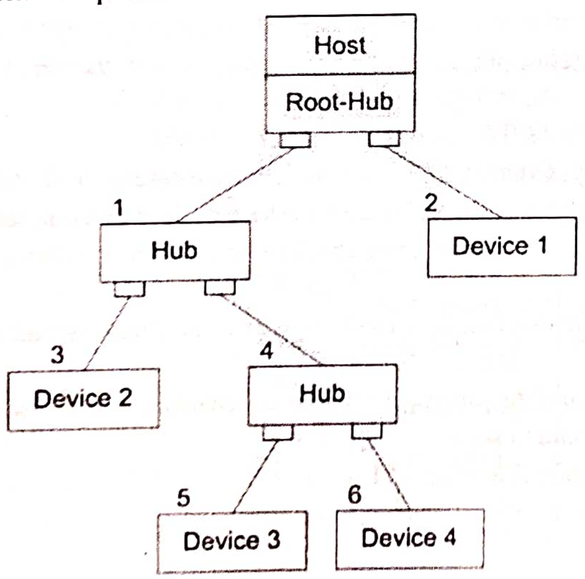
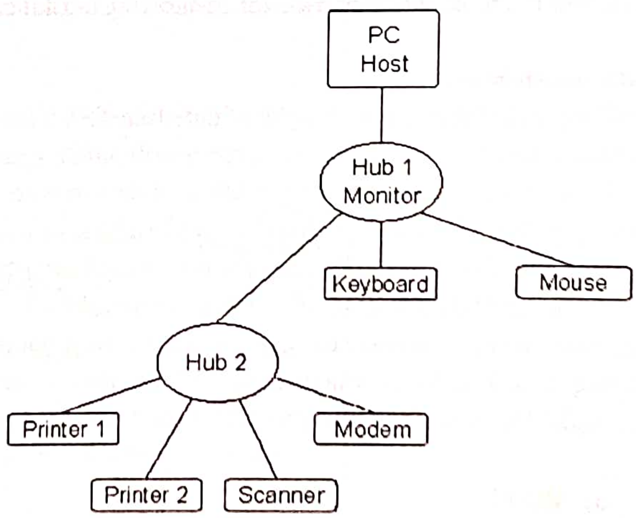


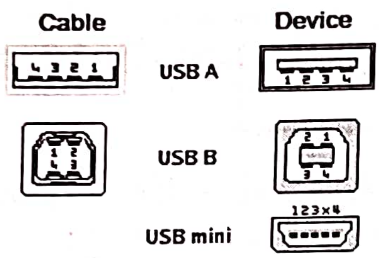
Fig.12 (a) USB tiered star topology



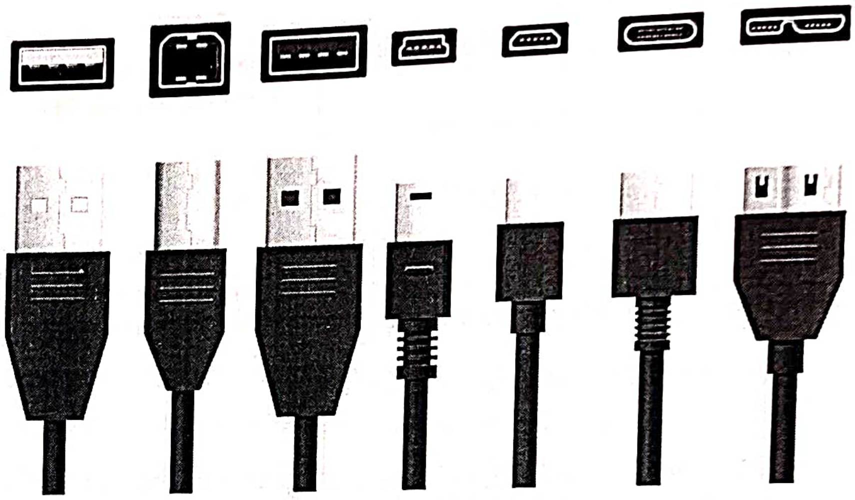
(b) Example of tiered star topology

Cable Description: The USB host controller is responsible for controlling the data communication, includi establishing connectivity with USB slave devices, packetizing and formatting the data packet.

For example, USB 2.0 cable, has four shielded wires. Red wire (1) for power supply voltage (+5V) and blac wire (4) for ground that allow the device to be powered by the host through the USB connection. The other two wires, green wire (3) is for the data (D+) upstream and white wire (2) is for data (D-) downstream are responsible for the transmission of the data, which is streamed in a form of a differential signal (between 0 and 3.3V). The USB cable supports communication distance of up to 5 meters.



Pin	Name	Cable color	Description
1	VCC	Red	+5 VDC
2	D-	White	Data -
3	D+	Green	Data +
4	GND	Black	Ground



USB Type A | USB Type B | USB 3.0 | USB Mini | USB Micro | USB Type C | USB Micro B

USB ports can be categorized based either on versions (USB 1.0, USB 2.0 and USB 3.0) or the type of USB cable (Connector type A, B, C.).

Type-A port (USB 2.0 standard) has a flat rectangle interface that inserts into a hub or USB host which transmits data and supplies power. Commonly used in desktop, laptop, media players, and gaming consoles.

Type-B connector is square with slanted exterior corners. Commonly used in printers, hard drives, and phones.

Type-C is used in major smart phone brands.

Mini and micro connectors used for connecting digital cameras, phones, laptops and tabs.

Data Transmission

USB supports four different types of data transfers, namely; Control, Bulk, Isochronous and Interrupt.

Control transfer is used by USB system software to query, configure and issue commands to the USB device.

Bulk transfer is used for sending a block of data to a device. It supports error checking and correction. Example: Transferring data from PC to a printer.

Isochronous data transfer is used for real time data communication. It doesn't support error checking and re-transmission of data in case of any transmission loss.

Interrupt transfer is used for transferring small amount of data. Interrupt transfer mechanism makes use of polling technique to see whether the USB device has any data to send.

Example: Devices like Mouse and Keyboard, which transmits fewer amounts of data.

ii) Wi-Fi

Wi-Fi is a wireless high-speed internet connection technique used to connect computers, tablets, smart phones and other accessing devices for internet. Wi-Fi uses radio waves sent from a wireless *router* to a nearby device, which translates the signal into a data format. Wi-Fi connection is established from the *access point* to e Wi-Fi enabled devices within a specific range as shown in the fig.13. There are four major types of Wi-Fi standards, operate in wide range of data speed and transmits on 2.4GHz or 5GHz frequency.

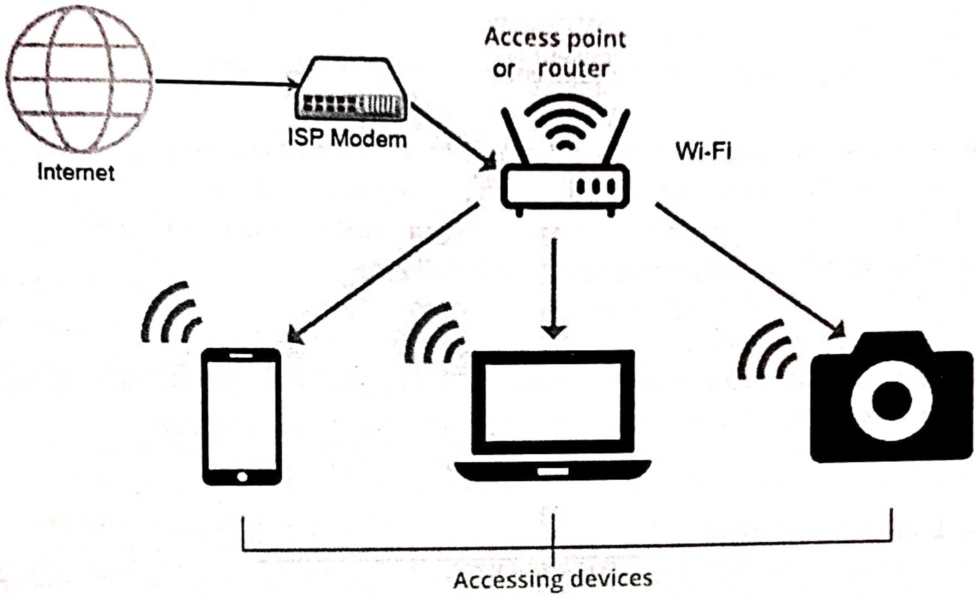


Fig.13 Wi-Fi network

Router: Wi-Fi router is the medium between the internet connection (WAN) and a wireless network. It acts as a base station. The router sends out radio waves through its antennas and broadcasts them into local area, allowing multiple devices access to the internet simultaneously.

The Wi-Fi router is responsible for

i) restricting the access to a network,
ii) assigning IP address to devices on the network,
iii) routing data packets to the intended devices on the network.

When Wi-Fi enabled device encounters a hotspot the device is then connected to that network wirelessly by configuring with the SSID and Password. Wi-Fi network is identified with a Service Set Identifier (SSID). If the network is security enabled, a password may be required to connect to a particular SSID.

For securing the data communication, Wi-Fi employs different security mechanisms like

- Wired Equivalency Privacy (WEP)
- Wireless Protected Access (WPA), etc.

SSID: It is a name that identifies the Wi-Fi network and differentiates one Wi-Fi from another Wi-Fi. When multiple wireless networks overlap in a certain location, SSIDs make sure that data gets sent to the correct destination.

Sl. No	Wi-Fi standards	Transmitting frequency	Data speed
1	IEEE802.11b	2.4GHz	11Mbps
2	IEEE802.11a	5GHz	54Mbps
3	IEEE802.11g	2.4GHz	100Mbps
4	IEEE802.11n	2.4GHz/5.0GHz	600Mbps

Wi-Fi supports data rates ranging from 1 Mbps to 1300Mbps (Growing towards higher rates as technology progresses), depending on the standards and access method. Depending on the type of antenna and usage location (indoor / outdoor), Wi-Fi offers a range of 100 to 1000 feet coverage area.

General Packet Radio Service (GPRS), 3G, 4G, LTE

GSM, stands for Global Systems for Mobile Communications, is basic standard bearer of 2G technologies. It is mainly used in mobile communication.
Features: Short Messaging System (SMS) was introduced; the data rate is only 9.6Kb/s.

General packet radio service (GPRS) is the first packet based mobile data service that available to users to get the phones online. Packet-based means the data is broken up into small blocks and transmitted over various channels. The data is then re-assembled into its original order at the receiving end.
GPRS supports a theoretical maximum transfer rate of 171.2kbps.
Features of GPRS: It's a 2G technology network.
MMS (Multimedia Messaging System)
It allowed subscribers to send videos, pictures, or sound clips to each other just like text messages.
GPRS also provided mobile handset the ability to surf the internet at dial-up speeds.
The limitation of GPRS: data cannot be sent while a voice call is in progress.

EDGE (Enhanced Data GSM Evolution)

- ✓ GPRS and EDGE are both 2G technology but EDGE is significantly faster with a download speed of up to 384Kbps. EDGE is sometimes called a 2.5G network

1G: Analog cellular telephony (voice only)

Introduced in 1980's
- ➢ It's speed was upto 2.4kbps
- ➢ It uses analog signal
- ➢ The frequency is typically 800 MHz band

Limitations:
- ➢ Poor voice quality
- ➢ Poor battery life
- ➢ Large phone size
- ➢ No security
- ➢ Limited capacity

2G: Digital cellular telephony

2G launched in the year 1991. Its capabilities were achieved by allowing multiple users on a single channel via multiplexing. It implemented the concept of CDMA and GSM technology.
- ➢ It uses digital signals.
- ➢ Data speed was upto 64kbps.
- ➢ It enables services such as text messages, picture messages and multi-media message (MMS).
- ➢ Internal roaming , conference calls, call hold and billing
- ➢ Provided better quality voice calls
- ➢ It used a bandwidth of 30 to 200 KHz

Limitations:
- ➢ It requires digital signals for communication.
- ➢ If there is no network coverage then digital signals would weak.
- ➢ These systems are unable to handle complex data such as videos.
- ➢ Reduced range

3G: High-speed digital cellular telephony (including video telephony)

3G Networks are based on WCDMA (Wideband Code Division Multiple Access). 3G network combines aspects of the 2G network with new technologies and protocols to deliver a significantly faster data rate.

- Phone calls
- Speed of up to 2 Mbps
- Global roaming
- Send/receive large email messages
- High speed web
- Navigation / maps
- Video conferencing
- TV streaming
- Electronic agenda meeting reminder

HSDPA (High-Speed Down-link Packet Access)

- It is based on the 3G network and an enhancement to 3G. Thus has a faster speed, download speed can be up to 14Mbps. HSDPA is sometimes called 3.5G.

4G: Fourth Generation Technology

It was released in 2008. IP-based "anytime, anywhere" voice, data, and multimedia telephony at faster data rates than 3G. The key technologies involved are MIMO (Multiple Input Multiple Output) and OFDM (Orthogonal Frequency Division Multiplexing). The most important 4G standards are Wi-MAX and LTE.

What is 4G LTE?

4G LTE is a "fourth generation long term evolution", capable of delivering a *very fast* and *secure internet* connection using a different radio interface together. LTE is the upgrade path for carriers with both GSM/UMTS networks and CDMA2000 networks.

Features of 4G LTE:

- Support gaming services, HD mobile TV, video conferencing, 3D TV
- High speed, high capacity and low cost per bit (Speeds of up to 20 Mbps or more.)
- Global and scalable mobile networks.
- Ad hoc and multi-hop networks
- download speed as high as 99.6Mbps

Generation	1G	2G	3G	4G
Period	1980-1990	1990-2000	2000-2010	2010-2020
Band width	150/900MHz	900MHz	100MHz	100MHz
Frequency	Analog Signal (30KHz)	1.8GHz (Digital)	1.6 - 2.0GHz	2 - 8GHz
Data rate	2Kbps	64Kbps	144Kbps – 2Mbps	100Mbps – 1Gbps
Characteristics	First wireless communication	Digital	Digital broadband, increased speed	High speed, all IP
Technology	Analog cellular	Digital cellular (GSM)	CDMA, UMTS,EDGE	LTE, Wi-Fi

WHAT IS 5G ?

5G is a packet switched wireless system with wide area coverage and high throughput. 5G wireless uses OFDM and millimeter wireless that enables data rate of 20 Mbps and frequency band of 2 to 8 GHz.

REVIEW QUESTIONS

1. Compare embedded systems and general computing systems. Also provide major application areas of Embedded Systems.
2. Write a note on classification of embedded systems.
3. Using suitable diagrams explain instrumentation and control systems.
4. Give the classification of transducers with examples.
5. Bring out the differences between RISC and CISC, Harvard & Von-Neumann.
6. Distinguish between microprocessor and microcontroller systems.
7. Explain the working, principle of operation and applications of stepper motor.
8. Define actuator and briefly describe the following actuators - LED, Piezo-buzzer.
9. With relevant diagrams explain the operation of relay, push button.
10. Define sensors and give its classification with examples.
11. Explain the different configurations of 7-segment LED Display.
12. Describe the matrix keyboard interfacing.
13. Bring out the main features of UART and USB.
14. Explain the following external communication interfaces: Wi-Fi
15. Bring out the main features of General Packet Radio Service (GPRS), 3G, 4G, LTE