

DEFINITION OF AN EMBEDDED SYSTEM

An embedded system is a combination of 3 types of components: a. Hardware b. Software c. Mechanical Components and it is supposed to do one specific task only.

Example 1: Washing Machine

- A washing machine from an embedded systems point of view has: a. Hardware: Buttons, Display & buzzer, electronic circuitry. b. Software: It has a chip on the circuit that holds the software which drives controls & monitors the various operations possible. c. Mechanical Components: the internals of a washing machine which actually wash the clothes control the input and output of water, the chassis itself.

Example 2: Air Conditioner

- An Air Conditioner from an embedded systems point of view has: a. Hardware: Remote, Display & buzzer, Infrared Sensors, electronic circuitry. b. Software: It has a chip on the circuit that holds the software which drives controls & monitors the various operations possible. The software monitors the external temperature through the sensors and then releases the coolant or suppresses it. c. Mechanical Components: the internals of an air conditioner the motor, the chassis, the outlet, etc An embedded system is designed to do a specific job only.

- Example: a washing machine can only wash clothes, an air conditioner can control the temperature in the room in which it is placed.

The hardware & mechanical components will consist all the physically visible things that are used for input, output, etc. An embedded system will always have a chip (either microprocessor or microcontroller) that has the code or software which drives the system

EMBEDDED SYSTEM & GENERAL PURPOSE COMPUTER

The Embedded System and the General purpose computer are at two extremes. The embedded system is designed to perform a specific task whereas as per definition the general purpose computer is meant for general use. It can be used for playing games, watching movies, creating software, work on documents or spreadsheets etc. Following are certain specific points that differentiates between embedded systems and general purpose computers:

Criteria	General Purpose Computer	Embedded system
Contents	It is combination of generic hardware and a general purpose OS for executing a variety of applications.	It is combination of special purpose hardware and embedded OS for executing specific set of applications
Operating System	It contains general purpose operating system	It may or may not contain operating system.
Alterations	Applications are alterable by the user.	Applications are non-alterable by the user.
Key factor	Performance" is key factor.	Application specific requirements are key factors.
Power Consumption	More	Less
Response Time	Not Critical	Critical for some applications

CLASSIFICATION OF EMBEDDED SYSTEMS

The classification of embedded system is based on following criteria's:

- On generation
- On complexity & performance
- On deterministic behavior
- On triggering

➤ **On generation:**

1. First generation (1G):

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

2. Second generation (2G):

- Built around 16-bit μp & 8-bit μc .
- They are more complex & powerful than 1G μp & μc .

- Examples: SCADA systems

3. Third generation (3G):

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

4. Fourth generation:

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

➤ **On complexity & performance:**

1. Small-scale:

- Simple in application need
- Performance not time-critical.
- Built around low performance & low cost 8 or 16 bit $\mu p/\mu c$. Example: an electronic toy

2. Medium-scale:

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

3. Large-scale:

- Highly complex hardware & firmware.
- Built around 32 or 64 bit RISC $\mu p/\mu c$ or PLDs or Multicore-Processors.
- Response is time-critical.
- Examples: Mission critical applications.

➤ **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.

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

APPLICATION OF EMBEDDED SYSTEM

The application areas and the products in the embedded domain are countless.

1. Consumer Electronics: Camcorders, Cameras.
2. Household appliances: Washing machine, Refrigerator.
3. Automotive industry: Anti-lock breaking system(ABS), engine control.
4. Home automation & security systems: Air conditioners, sprinklers, fire alarms.
5. Telecom: Cellular phones, telephone switches.
6. Computer peripherals: Printers, scanners.
7. Computer networking systems: Network routers and switches.
8. Healthcare: EEG, ECG machines.
9. Banking & Retail: Automatic teller machines, point of sales.
10. Card Readers: Barcode, smart card readers.

PURPOSE OF EMBEDDED SYSTEM

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 System with data collection/storage/representation of data.
- 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 video 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

- A system with control functionality contains both sensors and actuators. Sensors are connected to the input port for capturing the changes in environmental variable and the 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

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:

1. General purpose and Domain Specific Processors Microprocessors Microcontrollers Digital Signal Processors.
2. Application Specific Integrated Circuits. (ASIC)
3. Programmable logic devices(PLD's)
4. Commercial off-the-shelf components (COTs)

GENERAL PURPOSE AND DOMAIN SPECIFIC PROCESSOR.

- 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.

Microprocessors

- A microprocessor is a silicon chip representing a central processing unit.
- A microprocessor is a dependent unit and it requires the combination of other hardware like memory, timer unit, and interrupts controller, etc. for proper functioning.

Developers of microprocessors.

- Intel – Intel 4004 – November 1971(4-bit).
- Intel – Intel 4040. o Intel – Intel 8008 – April 1972.
- Intel – Intel 8080 – April 1974(8-bit).
- Motorola – Motorola 6800.
- Intel – Intel 8085 – 1976.
- Zilog - Z80 – July 1976.

Architectures used for processor design are Harvard or VonNeumann.

Harvard architecture	Von-Neumann architecture
<input type="checkbox"/> It has separate buses for instruction as well as data fetching.	<input type="checkbox"/> It shares single common bus for instruction and data fetching.
<input type="checkbox"/> Easier to pipeline, so high performance can be achieved.	<input type="checkbox"/> Low performance as compared to Harvard architecture.
<input type="checkbox"/> Comparatively high cost.	<input type="checkbox"/> It is cheaper.
<input type="checkbox"/> Since data memory and program memory are stored physically in different locations, no chances exist for accidental corruption of program memory.	<input type="checkbox"/> Accidental corruption of program memory may occur if data memory and program memory are stored physically in the same chip.

- RISC and CISC are the two common Instruction Set Architectures (ISA) available for processor design.

RISC	CISC
<input type="checkbox"/> Reduced Instruction Set Computing	<input type="checkbox"/> Complex Instruction Set Computing
<input type="checkbox"/> It contains lesser number of instructions.	<input type="checkbox"/> It contains greater number of instructions.
<input type="checkbox"/> Instruction pipelining and increased execution speed.	<input type="checkbox"/> Instruction pipelining feature does not exist.
<input type="checkbox"/> Orthogonal instruction set(allows each instruction to operate on any register and use any addressing mode.	<input type="checkbox"/> Non-orthogonal set(all instructions are not allowed to operate on any register and use any addressing mode.
<input type="checkbox"/> Operations are performed on registers only, only memory operations are load and store.	<input type="checkbox"/> Operations are performed either on registers or memory depending on instruction.
<input type="checkbox"/> A larger number of registers are available.	<input type="checkbox"/> The number of general purpose registers are very limited.
<input type="checkbox"/> Programmer needs to write more code to execute a task since instructions are simpler ones.	<input type="checkbox"/> Instructions are like macros in C language. A programmer can achieve the desired functionality with a single instruction which in turn provides the effect of using more simpler single instruction in RISC.
<input type="checkbox"/> It is single, fixed length instruction.	<input type="checkbox"/> It is variable length instruction.

Microcontrollers

- A microcontroller is a highly integrated chip that contains a CPU, scratch pad RAM, special and general purpose register arrays, on chip ROM/FLASH memory for program storage, timer and interrupt control units and dedicated I/O ports.
- Texas Instrument's TMS 1000 is considered as the world's first microcontroller.
- Some embedded system applications require only 8 bit controllers whereas some requiring superior performance and computational needs demand 16/32 bit controllers.
- The instruction set of a microcontroller can be RISC or CISC.
- Microcontrollers are designed for either general purpose application requirement or domain specific application requirement.

Digital Signal Processors

- DSP are powerful special purpose 8/16/32 bit microprocessors 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.
- This is because of the architectural difference between DSP and general purpose microprocessors.
- DSPs implement algorithms in hardware which speeds up the execution whereas general purpose processors implement the algorithm in software and the speed of execution depends primarily on the clock for the processors.
- DSP includes following key units:
 - i. Program memory: It is a memory for storing the program required by DSP to process the data.
 - ii. Data memory: It is a working memory for storing temporary variables and data/signal to be processed.
 - iii. 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.
 - iv. 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. SOP(Sum of Products) calculation, convolution, FFT(Fast Fourier Transform), DFT(Discrete Fourier Transform), etc are some of the operations performed by DSP.

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.
- The developers of such chips may not be interested in revealing the internal detail of it .

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)

1) A Commercial off the Shelf product is one which is used 'asis'.

2) The COTS components itself may be develop around a general purpose or domain specific processor or an ASICs or a PLDs.

3) 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

4) 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

SENSORS & ACTUATORS**Sensor**

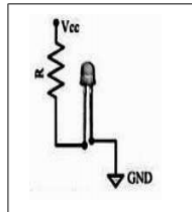
- A Sensor is used for taking Input
- It is a transducer that converts energy from one form to another for any measurement or control purpose Ex. A Temperature sensor

Actuator

Actuator is used for output. It is a transducer that may be either mechanical or electrical which converts signals to corresponding physical actions.

LED (Light Emitting Diode)

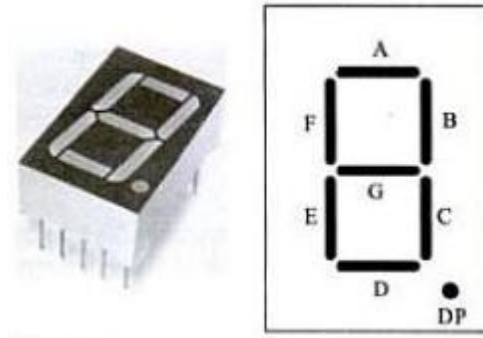
LED is a p-n junction diode and contains a CATHODE and ANODE For functioning the anode is connected to +ve end of power supply and cathode is connected to -ve end of power supply. The maximum current flowing through the LED is limited by connecting a RESISTOR in series between the power supply and LED as shown in the figure below



There are two ways to interface an LED to a microprocessor/microcontroller:

1. The Anode of LED is connected to the port pin and cathode to Ground : In this approach the port pin sources the current to the LED when it is at logic high (ie. 1).
2. The Cathode of LED is connected to the port pin and Anode to Vcc : In this approach the port pin sinks the current to the LED when it is at logic high (ie. 1). Here the port pin sinks the current and the LED is turned ON when the port pin is at Logic low (ie. 0)

7-segment display:



A **seven-segment display (SSD)**, or **seven-segment indicator**, is a form of electronic display device for displaying decimal numerals that is an alternative to the more complex dot matrix displays. Seven-segment displays are widely used in digital clocks, electronic meters, basic calculators, and other electronic devices that display numerical information.

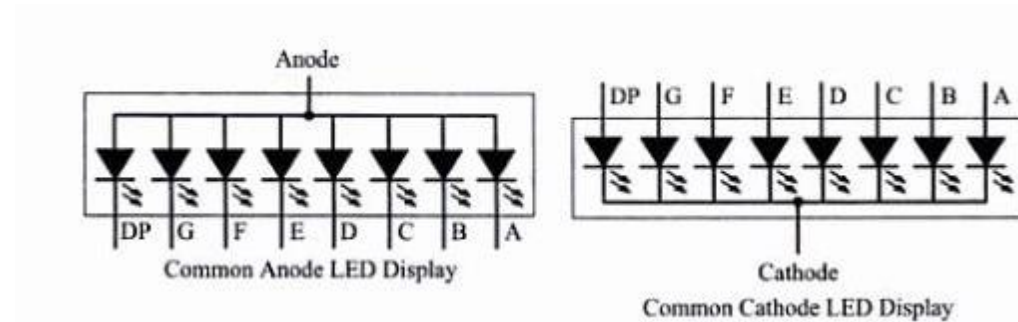
The seven elements of the display can be lit in different combinations to represent the Arabic numerals. Often the seven segments are arranged in an *oblique* (slanted) arrangement, which aids readability. In most applications, the seven segments are of nearly uniform shape and size (usually elongated hexagons, though trapezoids and rectangles can also be used), though in the case of adding machines, the vertical segments are longer and more oddly shaped at the ends in an effort to further enhance readability.

The numerals 6 and 9 may be represented by two different glyphs on seven-segment displays, with or without a 'tail'.^{[2][3]} The numeral 7 also has two versions, with or without segment F.^[4]

The seven segments are arranged as a rectangle of two vertical segments on each side with one horizontal segment on the top, middle, and bottom. Additionally, the seventh segment bisects the rectangle horizontally. There are also fourteen-segment displays and sixteen-segment displays (for full alphanumeric); however, these have mostly been replaced by dot matrix

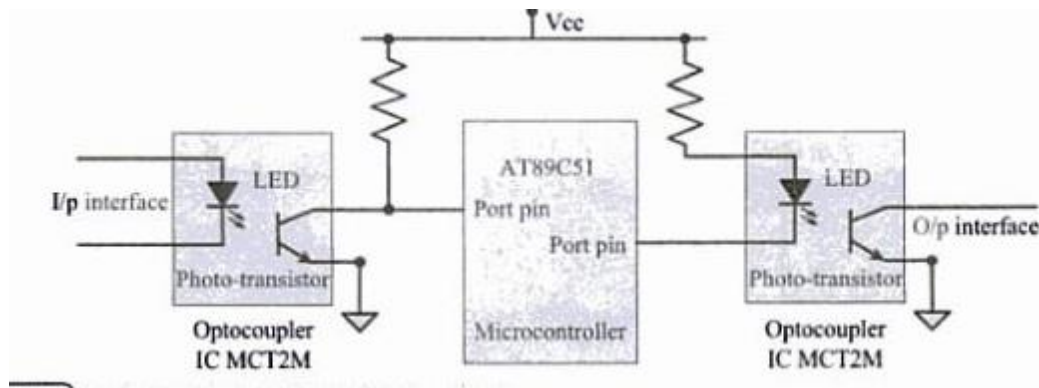
displays. Twenty-two segment displays capable of displaying the full ASCII character set^[5] were briefly available in the early 1980s, but did not prove popular.

The segments of a 7-segment display are referred to by the letters A to G, where the optional decimal point (an "eighth segment", referred to as DP) is used for the display of non-integer numbers.



Optical coupler:

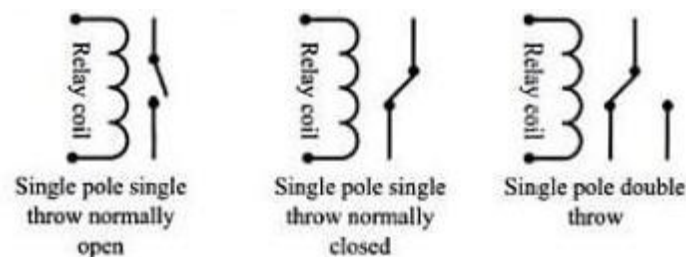
An optical coupler, also called opto-isolator, optocoupler, opto coupler, photocoupler or optical isolator, is a passive optical component that can combine or split transmission data (optical power) from optical fibers. It is an electronic device which is designed to transfer electrical signals by using light waves in order to provide coupling with electrical isolation between its input and output. The main purpose of an optocoupler is to prevent rapidly changing voltages or high voltages on one side of a circuit from distorting transmissions or damaging components on the other side of the circuit. An optocoupler contains a light source often near an LED which converts electrical input signal into light, a closed optical channel and a photosensor, which detects incoming light and either modulates electric current flowing from an external power supply or generates electric energy directly. The sensor can either be a photoresistor, a silicon-controlled rectifier, a photodiode, a phototransistor or a triac.



Applications for Optocouplers:

Photoresistor-based opto-isolators are the slowest type of optocouplers, but also the most linear isolators and are used in the audio and music industry. Most opto-isolators available use bipolar silicon phototransistor sensors and reach medium data transfer speed, which is enough for applications like electroencephalography. High speed opto-isolators are used in computing and communications applications. Other industrial applications include photocopiers, industrial automation, professional light measurement instruments and auto-exposure meters.

Relay :

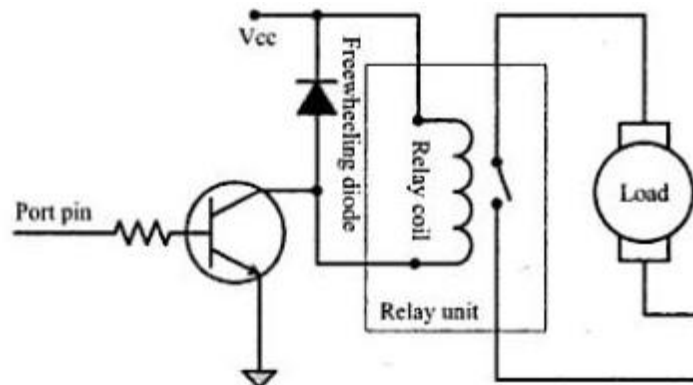


A relay is an electrically operated switch. Many relays use an electromagnet to mechanically operate a switch, but other operating principles are also used, such as solid-state relays. Relays are used where it is necessary to control a circuit by a separate low-power signal, or where several circuits must be controlled by one signal. The first relays were used in long distance telegraph circuits as amplifiers: they repeated the signal coming in from one circuit and re-transmitted it on another circuit. Relays were used extensively in telephone exchanges and early computers to perform logical operations. A type of relay that can handle the high power required

to directly control an electric motor or other loads is called a contactor. Solid-state relays control power circuits with no moving parts, instead using a semiconductor device to perform switching. Relays with calibrated operating characteristics and sometimes multiple operating coils are used to protect electrical circuits from overload or faults; in modern electric power systems these functions are performed by digital instruments still called "protective relays".

Magnetic latching relays require one pulse of coil power to move their contacts in one direction, and another, redirected pulse to move them back. Repeated pulses from the same input have no effect. Magnetic latching relays are useful in applications where interrupted power should not be able to transition the contacts.

Magnetic latching relays can have either single or dual coils. On a single coil device, the relay will operate in one direction when power is applied with one polarity, and will reset when the polarity is reversed. On a dual coil device, when polarized voltage is applied to the reset coil the contacts will transition. AC controlled magnetic latch relays have single coils that employ steering diodes to differentiate between operate and reset commands.



Buzzer :

A **buzzer** or **beeper** is an audio signalling device, which may be mechanical, electromechanical, or piezoelectric (*piezo* for short). Typical uses of buzzers and beepers include alarm devices, timers, and confirmation of user input such as a mouse click or keystroke.

Types of Buzzers

There are several different kinds of buzzers. At Future Electronics we stock many of the most common types categorized by Type, Sound Level, Frequency, Rated Voltage, Dimension and Packaging Type. The parametric filters on our website can help refine your search results depending on the required specifications.

The most common sizes for Sound Level are 80 dB, 85 dB, 90 dB and 95 dB. We also carry buzzers with Sound Level up to 105 dB. There are several types available including Electro-Acoustic, Electromagnetic, Electromechanic, Magnetic and Piezo, among others.

Applications for Buzzers:

Typical uses of buzzers include:

- Alarm devices
- Timers
- Confirmation of user input (ex: mouse click or keystroke)
- Electronic metronomes
- Annunciator panels
- Game shows
- Sporting events
- Household appliances

Push button switch:

A **push-button** (also spelled **pushbutton**) or simply **button** is a simple switch mechanism for controlling some aspect of a machine or a process. Buttons are typically made out of hard material, usually plastic or metal.^[1] The surface is usually flat or shaped to accommodate the human finger or hand, so as to be easily depressed or pushed. Buttons are most often biased switches, although many un-biased buttons (due to their physical nature) still require a spring to return to their un-pushed state. Terms for the "pushing" of a button include **pressing**, **depressing**, **mashing**, **hitting**, and **punching**. The "push-button" has been utilized in calculators, push-button telephones, kitchen appliances, and various other mechanical and electronic devices, home and commercial.

In industrial and commercial applications, push buttons can be connected together by a mechanical linkage so that the act of pushing one button causes the other button to be released. In this way, a stop button can "force" a start button to be released. This method of linkage is used in simple manual operations in which the machine or process has no electrical circuits for control.

Red pushbuttons can also have large heads (called mushroom heads) for easy operation and to facilitate the stopping of a machine. These pushbuttons are called emergency stop buttons and for increased safety are mandated by the electrical code in many jurisdictions. This large mushroom shape can also be found in buttons for use with operators who need to wear gloves for their work and could not actuate a regular flush-mounted push button.

Communication Interface (onboard and external types):

For any embedded system, the communication interfaces can broadly classified into:

1. Onboard Communication Interfaces :These are used for internal communication of the embedded system i.e: communication between different components present on the system.

Common examples of onboard interfaces are:

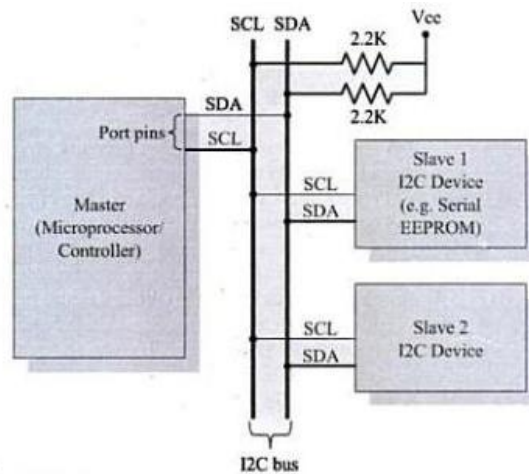
- Inter Integrated Circuit (I2C)
- Serial Peripheral Interface (SPI)
- Universal Asynchronous Receiver Transmitter (UART)
- 1-Wire Interface
- Parallel Interface
- **Inter Integrated Circuit (I2C)**

I²C was originally developed in 1982 by Philips for various Philips chips. The original spec allowed for only 100kHz communications, and provided only for 7-bit addresses, limiting the number of devices on the bus to 112 (there are several reserved addresses, which will never be used for valid I²C addresses). In 1992, the first public specification was published, adding a 400kHz fast-mode as well as an expanded 10-bit address space. Much of the time (for instance, in the ATmega328 device on many Arduino-compatible boards) , device support for I²C ends at this

point. There are three additional modes specified: fast-mode plus, at 1MHz; high-speed mode, at 3.4MHz; and ultra-fast mode, at 5MHz.

Each I²C bus consists of two signals: SCL and SDA. SCL is the clock signal, and SDA is the data signal. The clock signal is always generated by the current bus master; some slave devices may force the clock low at times to delay the master sending more data (or to require more time to prepare data before the master attempts to clock it out). This is called “clock stretching” and is described on the protocol page.

Unlike UART or SPI connections, the I²C bus drivers are “open drain”, meaning that they can pull the corresponding signal line low, but cannot drive it high. Thus, there can be no bus contention where one device is trying to drive the line high while another tries to pull it low, eliminating the potential for damage to the drivers or excessive power dissipation in the system. Each signal line has a pull-up resistor on it, to restore the signal to high when no device is



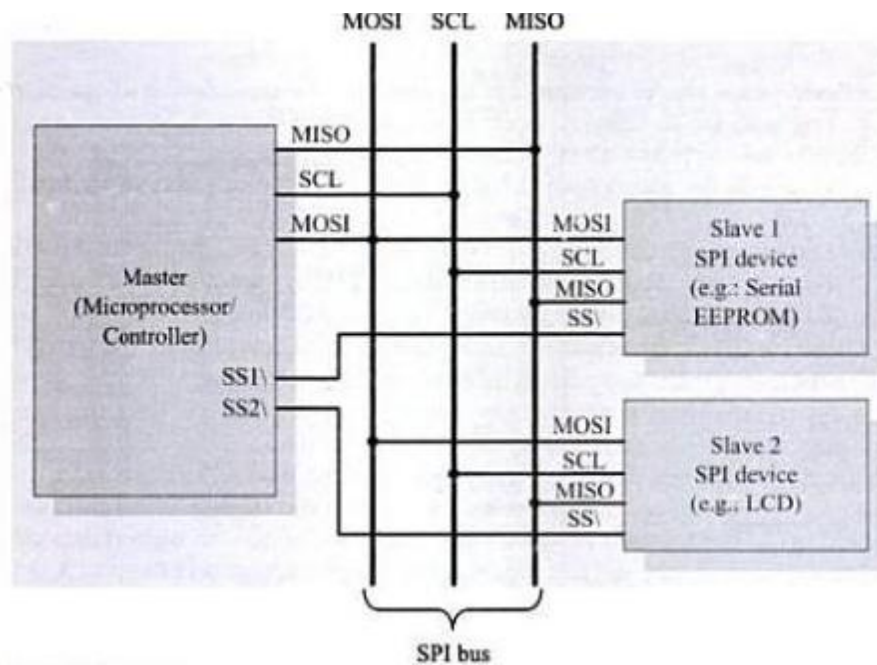
asserting it low.

Serial Data Line (SDA)

The **Serial Data Line (SDA)** is the data line (of course!). All the data transfer among the devices takes place through this line.

Serial Clock Line (SCL)

The **Serial Clock Line (SCL)** is the serial clock (obviously!). I²C is a synchronous protocol, and hence, SCL is used to synchronize all the devices and the data transfer together. We'll learn how it works a little later in this post.

SPI BUS :

Serial Peripheral Interface, or **SPI**, is a very common communication protocol used for two-way communication between two devices. A standard SPI bus consists of 4 signals, **Master Out Slave In (MOSI)**, **Master In Slave Out (MISO)**, the clock (**SCK**), and **Slave Select (SS)**. Unlike an asynchronous serial interface, SPI is not symmetric. An SPI bus has one **master** and one or more **slaves**. The master can talk to any slave on the bus, but each slave can only talk to the master. Each slave on the bus must have its own unique slave select signal. The master uses the slave select signals to *select* which *slave* it will be talking to. Since SPI also includes a clock signal, both devices don't need to agree on a data rate beforehand. The only requirement is that the clock is lower than the maximum frequency for all devices involved.

Each SPI transfer is **full-duplex**, meaning that data is sent from the master to the slave and from the slave to the master at the same time. There is no way for a slave to opt-out of sending data when the master makes a transfer, however, devices will send dummy bytes (usually all 1's or all 0's) when communication should be one way. If the master is reading data in for a slave, the slave will know to ignore the data being sent by the master.

Devices that use SPI typically will send/receive multiple bytes each time the **SS** signal goes low. This way the **SS** signal acts as a way to frame a transmission. For example, if you had a flash memory that had an SPI bus and you want to read some data, the **SS** signal would go low, the

master would send the command to read memory at a certain address, and as long as the master kept **SS** low and toggling **SCK** the flash memory would keep sending out data. Once **SS** returned high the flash memory knows to end the read command.

Since the **MISO** signal can be connected to multiple devices, each device will only drive the line when its **SS** signal is low. This is shown by the grey area.

Advantages of SPI:

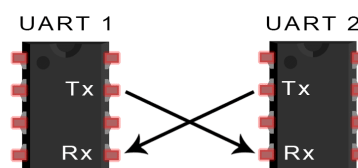
- It's faster than asynchronous serial
- The receive hardware can be a simple shift register
- It supports multiple slaves

Disadvantages of SPI:

- It requires more signal lines (wires) than other communications methods
- The communications must be well-defined in advance (you can't send random amounts of data whenever you want)
- The master must control all communications (slaves can't talk directly to each other)
- It usually requires separate SS lines to each slave, which can be problematic if numerous slaves are needed.

UART

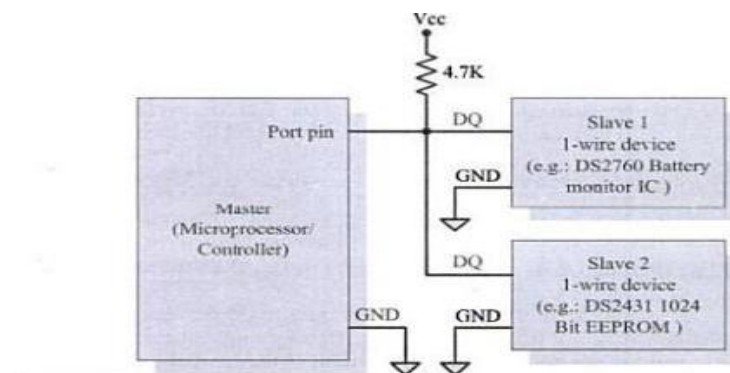
In UART communication, two UARTs communicate directly with each other. The transmitting UART converts parallel data from a controlling device like a CPU into serial form, transmits it in serial to the receiving UART, which then converts the serial data back into parallel data for the receiving device. Only two wires are needed to transmit data between two UARTs. Data flows from the Tx pin of the transmitting UART to the Rx pin of the receiving UART:



UARTs transmit data *asynchronously*, which means there is no clock signal to synchronize the output of bits from the transmitting UART to the sampling of bits by the receiving UART. Instead of a clock signal, the transmitting UART adds start and stop bits to the data packet being transferred. These bits define the beginning and end of the data packet so the receiving UART knows when to start reading the bits.

When the receiving UART detects a start bit, it starts to read the incoming bits at a specific frequency known as the *baud rate*. Baud rate is a measure of the speed of data transfer, expressed in bits per second (bps). Both UARTs must operate at about the same baud rate. The baud rate between the transmitting and receiving UARTs can only differ by about 10% before the timing of bits gets too far off.

1-wire interface:



A 1994 application note explained that the only serial-port interface options for 1-Wire devices were microcontroller port pins, UARTs, and UART-based COM ports. Since that time special driver chips have been developed for direct connection to a UART, PC bus, or USB port. Meanwhile, the number of 1-Wire devices also grew to a long list. These various developments made it necessary to update the earlier documentation. Instead of merging the specifics of all relevant information into a single document, this new document refers the reader to other application notes whenever possible.

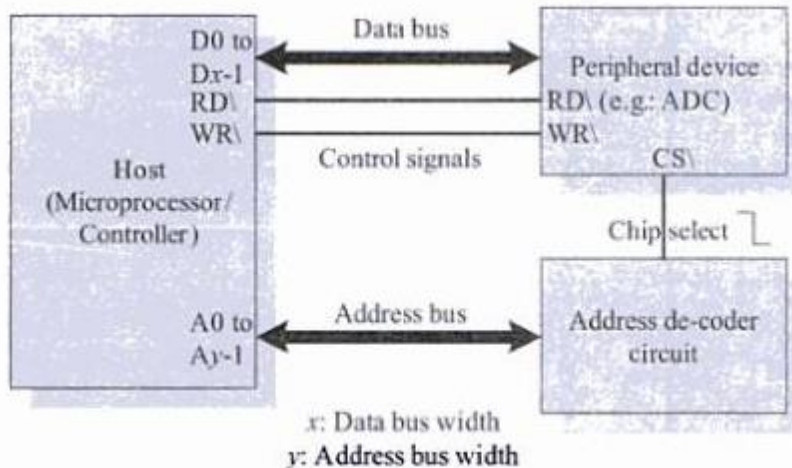
The first 1-Wire devices, the DS199x series, were produced in SRAM technology. Next the nonvolatile EPROM technology became available, and the DS198x and DS250x series devices were released. These EPROM devices need a 12V programming pulse and are not erasable. The next leap forward was EEPROM technology, which allows programming and erasing at 5V or

less. EEPROM technology is found in DS197x, DS243x and DS28Exx series devices. To ensure proper power, EEPROM devices may need a master that supports "strong pullup", a feature that temporarily bypasses the 1-Wire pullup resistor with a low-impedance path. The extra power is needed for write cycles and, in case of the DS1977, also for reading. Besides EEPROM devices, the strong pullup also powers 1-Wire temperature sensors and special functions such as a SHA-1 engine, which is found in secure 1-Wire devices. Temperature logger *iButtons*® use SRAM technology and, therefore, do not have any special, external power requirements.

General Information:

1-Wire is the only voltage-based digital system that works with two contacts, data and ground, for half-duplex bidirectional communication. A 1-Wire system consists of a single 1-Wire master and one or more 1-Wire slaves. The 1-Wire concept relies both on a master that initiates digital communication, and on self-timed 1-Wire slave devices that synchronize to the master's signal. The timing logic of master and slave must measure and generate digital pulses of various widths. When idle, a high-impedance path between the 1-Wire bus and the operating voltage puts the 1-Wire bus in the logic-high state. Each device on the bus must be able to pull the 1-Wire bus low at the appropriate time by using an open-drain output (wired AND). If a transaction needs to be suspended for any reason, the bus must be left in the idle state so the transaction can resume.

Parallel port:



A **parallel port** is a type of interface found on computers (personal and otherwise) for connecting peripherals. The name refers to the way the data is sent; parallel ports send multiple bits of data at once, in parallel communication, as opposed to serial interfaces that send bits one at a time. To do this, parallel ports require multiple data lines in their cables and port connectors, and tend to be larger than contemporary serial ports which only require one data line.

There are many types of parallel ports, but the term has become most closely associated with the **printer port** or Centronics port found on most personal computers from the 1970s through the 2000s. It was an industry *de facto* standard for many years, and was finally standardized as IEEE 1284 in the late 1990s, which defined the Enhanced Parallel Port (EPP) and Extended Capability Port (ECP) bi-directional versions. Today, the parallel port interface is virtually non-existent because of the rise of Universal Serial Bus (USB) devices, along with network printing using Ethernet and Wi-Fi connected printers.

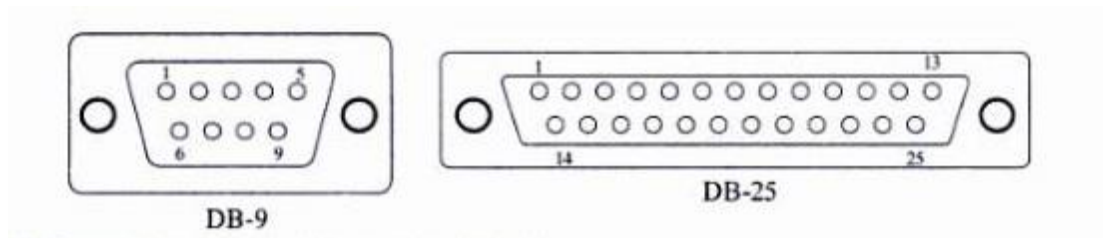
The parallel port interface was originally known as the **Parallel Printer Adapter** on IBM PC-compatible computers. It was primarily designed to operate printers that used IBM's eight-bit extended ASCII character set to print text, but could also be used to adapt other peripherals. Graphical printers, along with a host of other devices, have been designed to communicate with the system.

External communication interface:

In telecommunications, **RS-232, Recommended Standard 232**^[1] is a standard introduced in 1960^[2] for serial communication transmission of data. It formally defines the signals connecting between a *DTE (data terminal equipment)* such as a computer terminal, and a *DCE (data circuit-terminating equipment or data communication equipment)*, such as a modem. The RS-232 standard had been commonly used in computer serial ports. The standard defines the electrical characteristics and timing of signals, the meaning of signals, and the physical size and pinout of connectors. The current version of the standard is *TIA-232-F Interface Between Data Terminal Equipment and Data Circuit-Terminating Equipment Employing Serial Binary Data Interchange*, issued in 1997.

An RS-232 serial port was once a standard feature of a personal computer, used for connections to modems, printers, mice, data storage, uninterruptible power supplies, and other peripheral devices. RS-232, when compared to later interfaces such as RS-422, RS-485 and Ethernet, has

lower transmission speed, short maximum cable length, large voltage swing, large standard connectors, no multipoint capability and limited multidrop capability. In modern personal computers, USB has displaced RS-232 from most of its peripheral interface roles. Many computers no longer come equipped with RS-232 ports (although some motherboards come equipped with a COM port header that allows the user to install a bracket with a DE-9 port) and must use either an external USB-to-RS-232 converter or an internal expansion card with one or more serial ports to connect to RS-232 peripherals. Nevertheless, thanks to their simplicity and past ubiquity, RS-232 interfaces are still used—particularly in industrial machines, networking equipment, and scientific instruments where a short-range, point-to-point, low-speed wired data connection is adequate.



RS-232 logic and voltage levels

Data circuits	Control circuits	Voltage
0 (space)	Asserted	+3 to +15 V
1 (mark)	Deasserted	-15 to -3 V

USB:

USB, short for Universal Serial Bus, is a standard type of connection for many different kinds of devices. Generally, USB refers to the types of cables and connectors used to connect these many types of external devices to computers.

More About USB

The Universal Serial Bus standard has been extremely successful. USB ports and cables are used to connect hardware such as printers, scanners, keyboards, mice, flash drives, external hard drives, joysticks, cameras, and more to computers of all kinds, including desktops, tablets, laptops, netbooks, etc.

In fact, USB has become so common that you'll find the connection available on nearly any computer-like device such as video game consoles, home audio/visual equipment, and even in many automobiles.

Many portable devices, like smartphones, ebook readers, and small tablets, use USB primarily for charging. USB charging has become so common that it's now easy to find replacement electrical outlets at home improvement stores with USB ports built in, negating the need for a USB power adapter.

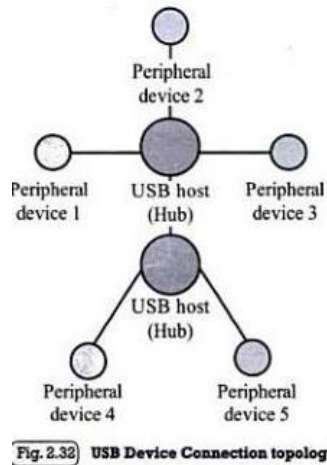
USB Versions

There have been three major USB standards, 3.1 being the newest:

- USB 3.1: Called *Superspeed+*, USB 3.1 compliant devices are able to transfer data at 10 Gbps (10,240 Mbps).
- USB 3.0: Called *SuperSpeed USB*, USB 3.0 compliant hardware can reach a maximum transmission rate of 5 Gbps (5,120 Mbps).
- USB 2.0: Called *High-Speed USB*, USB 2.0 compliant devices can reach a maximum transmission rate of 480 Mbps.
- USB 1.1: Called *Full Speed USB*, USB 1.1 devices can reach a maximum transmission rate of 12 Mbps.

Most USB devices and cables today adhere to USB 2.0, and a growing number to USB 3.0.

Important: The parts of a USB-connected system, including the host (like a computer), the cable, and the device, can all support different USB standards so long as they are physically compatible. However, all parts must support the same standard if you want it to achieve the maximum data rate possible.



IEEE1394:

IEEE 1394, High Performance Serial Bus, is an electronics standard for connecting devices to your personal computer. IEEE 1394 provides a single plug-and-socket connection on which up to 63 devices can be attached with data transfer speeds up to 400 Mbps (megabit s per second). The standard describes a serial bus or pathway between one or more peripheral devices and your computer's microprocessor . Many peripheral devices now come equipped to meet IEEE 1394. Two popular implementations of IEEE 1394 are Apple's FireWire and Sony's i.LINK . IEEE 1394 implementations provide:

- A simple common plug-in serial connector on the back of your computer and on many different types of peripheral devices
- A thin serial cable rather than the thicker parallel cable you now use to your printer, for example
- A very high-speed rate of data transfer that will accommodate multimedia applications (100 and 200 megabits per second today; with much higher rates later)
- Hot-plug and plug and play capability without disrupting your computer
- The ability to chain devices together in a number of different ways without terminators or complicated set-up requirements

Working

There are two levels of interface in IEEE 1394, one for the backplane bus within the computer and another for the point-to-point interface between device and computer on the serial cable. A simple bridge connects the two environments. The backplane bus supports 12.5, 25, or 50 megabits per second data transfer. The cable interface supports 100, 200, or 400 megabits per second. Each of these interfaces can handle any of the possible data rates and change from one to another as needed.

IrDA

- IrDA (Infrared Data Association)
- Bluetooth 2.4 GHz
- 802.11 WLAN and 802.11b WiFi
- ZigBee 900 MHz
- Used in mobile phones, digital cameras, keyboard, mouse, printers to communicate to laptop computer and for data and pictures download and synchronization.
- Used for control TV, air-conditioning, LCD projector, VCD devices from a distance
- Use infrared (IR) after suitable modulation of the data bits.
- Communicates over a line of sight Phototransistor receiver for infrared rays

IrDA protocol suite

- Supports data transfer rates of up to 4 Mbps
- Supports bi-directional serial communication over viewing angle between $\pm 15^\circ$ and distance of nearly 1 m At 5 m, the IR transfer data can be up to data transfer rates of 75 kbps
- Should be no obstructions or wall in between the source and receiver

Bluetooth

Bluetooth is a wireless technology standard for exchanging data over short distances (using short-wavelength UHF radio waves in the ISM band from 2.4 to 2.485 GHz) from fixed and mobile devices, and building personal area networks(PANs). Invented by telecom vendor Ericsson in 1994, it was originally conceived as a wireless alternative to RS-232 data cables.

Bluetooth is managed by the Bluetooth Special Interest Group (SIG), which has more than 30,000 member companies in the areas of telecommunication, computing, networking, and consumer electronics.^[5] The IEEE standardized Bluetooth as IEEE 802.15.1, but no longer maintains the standard. The Bluetooth SIG oversees development of the specification, manages the qualification program, and protects the trademarks.^[6] A manufacturer must meet Bluetooth SIG standards to market it as a Bluetooth device.

Bluetooth operates at frequencies between 2402 and 2480 MHz, or 2400 and 2483.5 MHz including guard bands 2 MHz wide at the bottom end and 3.5 MHz wide at the top.^[15] This is in the globally unlicensed (but not unregulated) industrial, scientific and medical (ISM) 2.4 GHz short-range radio frequency band. Bluetooth uses a radio technology called frequency-hopping spread spectrum. Bluetooth divides transmitted data into packets, and transmits each packet on one of 79 designated Bluetooth channels. Each channel has a bandwidth of 1 MHz. It usually performs 800 hops per second, with Adaptive Frequency-Hopping (AFH) enabled. Bluetooth Low Energy uses 2 MHz spacing, which accommodates 40 channels.

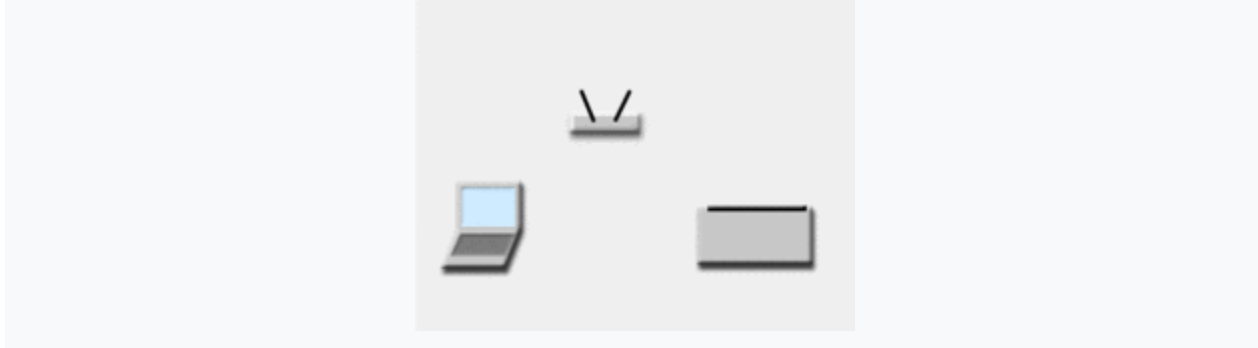
Originally, Gaussian frequency-shift keying (GFSK) modulation was the only modulation scheme available. Since the introduction of Bluetooth 2.0+EDR, $\pi/4$ -DQPSK(differential quadrature phase shift keying) and 8DPSK modulation may also be used between compatible devices. Devices functioning with GFSK are said to be operating in basic rate (BR) mode where an instantaneous bit rate of 1 Mbit/s is possible. The term Enhanced Data Rate (EDR) is used to describe $\pi/4$ -DPSK and 8DPSK schemes, each giving 2 and 3 Mbit/s respectively. The combination of these (BR and EDR) modes in Bluetooth radio technology is classified as a "BR/EDR radio".

Wifi :

Wi-Fi is a technology for wireless local area networking with devices based on the IEEE 802.11 standards. *Wi-Fi* is a trademark of the Wi-Fi Alliance, which restricts the use of the term *Wi-Fi Certified* to products that successfully complete interoperability certification testing.

Devices that can use Wi-Fi technology include personal computers, video-game consoles, phones and tablets, digital cameras, smart TVs, digital audio players and modern printers. Wi-Fi compatible devices can connect to the Internet via a WLAN and a wireless access point. Such an access point (or hotspot) has a range of about 20 meters (66 feet) indoors and a

greater range outdoors. Hotspot coverage can be as small as a single room with walls that block radio waves, or as large as many square kilometres achieved by using multiple overlapping access points.



Wi-Fi most commonly uses the 2.4 gigahertz (12 cm) UHF and 5.8 gigahertz (5 cm) SHF ISM radio bands. Anyone within range with a wireless modem can attempt to access the network; because of this, Wi-Fi is more vulnerable to attack (called eavesdropping) than wired networks. Wi-Fi Protected Access is a family of technologies created to protect information moving across Wi-Fi networks and includes solutions for personal and enterprise networks. Security features of Wi-Fi Protected Access constantly evolve to include stronger protections and new security practices as the security landscape change.

Zigbee :

Zigbee is an IEEE 802.15.4-based specification for a suite of high-level communication protocols used to create personal area networks with small, low-power digital radios, such as for home automation, medical device data collection, and other low-power low-bandwidth needs, designed for small scale projects which need wireless connection. Hence, Zigbee is a low-power, low data rate, and close proximity (i.e., personal area) wireless ad hoc network.

The technology defined by the Zigbee specification is intended to be simpler and less expensive than other wireless personal area networks (WPANs), such as Bluetooth or more general wireless networking such as Wi-Fi. Applications include wireless light switches, home energy monitors, traffic management systems, and other consumer and industrial equipment that requires short-range low-rate wireless data transfer.

Its low power consumption limits transmission distances to 10–100 meters line-of-sight, depending on power output and environmental characteristics. Zigbee devices can transmit data

over long distances by passing data through a mesh network of intermediate devices to reach more distant ones. Zigbee is typically used in low data rate applications that require long battery life and secure networking (Zigbee networks are secured by 128 bit symmetric encryption keys.) Zigbee has a defined rate of 250 kbit/s, best suited for intermittent data transmissions from a sensor or input device.

Zigbee was conceived in 1998, standardized in 2003, and revised in 2006. The name refers to the waggle dance of honey bees after their return to the beehive.

Typical application areas include:

- Home Entertainment and Control—Home automation such as in QIVICON,^[11] smart lighting,^[12] advanced temperature control, safety and security, movies and music
- Wireless sensor networks—Starting with individual sensors like Telosb/Tmote and Iris from Memsic
- Industrial control
- Embedded sensing
- Medical data collection
- Smoke and intruder warning
- Building automation
- Remote wireless microphone configuration, in Shure Wireless Microphone Systems^[13]

General Packet Radio Service:

GPRS is a packet oriented mobile data service on the 2G and 3G cellular communication system's global system for mobile communications (GSM). GPRS was originally standardized by European Telecommunications Standards Institute (ETSI) in response to the earlier CDPD and i-mode packet-switched cellular technologies. It is now maintained by the 3rd Generation Partnership Project (3GPP).

GPRS usage is typically charged based on volume of data transferred, contrasting with circuit switched data, which is usually billed per minute of connection time. Sometimes billing time is broken down to every third of a minute. Usage above the bundle cap is charged per megabyte, speed limited, or disallowed.

GPRS is a best-effort service, implying variable throughput and latency that depend on the number of other users sharing the service concurrently, as opposed to circuit switching, where a certain quality of service (QoS) is guaranteed during the connection. In 2G systems, GPRS provides data rates of 56–114 kbit/second.¹ 2G cellular technology combined with GPRS is sometimes described as 2.5G, that is, a technology between the second (2G) and third (3G) generations of mobile telephony. It provides moderate-speed data transfer, by using unused time division multiple access (TDMA) channels in, for example, the GSM system. GPRS is integrated into GSM Release 97 and newer releases.

MEMORIES

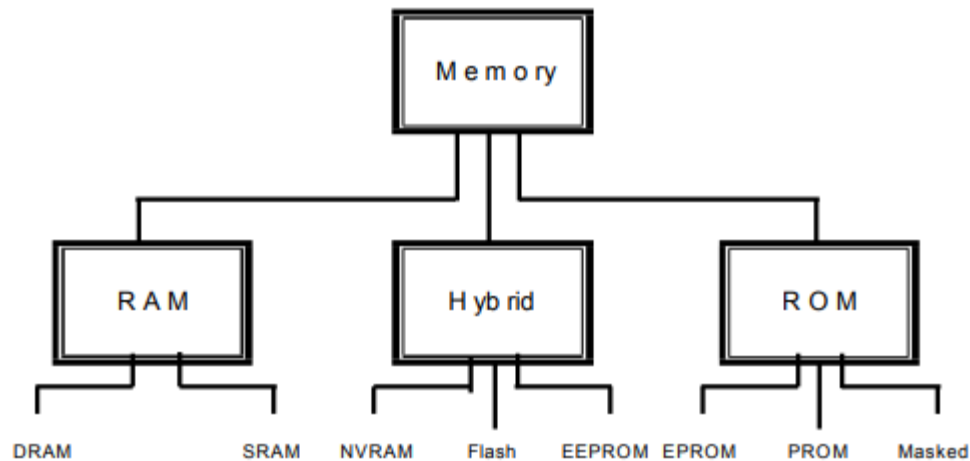
There are different types of memories available to be used in computers as well as embedded system. This chapter guides the reader through the different types of memories that are available and can be used and tries to explain their differences in simple words.

TYPES OF MEMORY

There are three main types of memories, they are

- a) **RAM** (Random Access Memory) It is read write memory.
 - Data at any memory location can be read or written.
 - It is volatile memory, i.e. retains the contents as long as electricity is supplied.
- Data access to RAM is very fast
- b) **ROM** (Read Only Memory) It is read only memory.
 - Data at any memory location can be only read.
 - It is non-volatile memory, i.e. the contents are retained even after electricity is switched off and available after it is switched on. Data access to ROM is slow compared to RAM.
- c) **HYBRID** It is combination of RAM as well as ROM
 - It has certain features of RAM and some of ROM
 - Like RAM the contents to hybrid memory can be read• and written Like ROM the contents of hybrid memory are non volatile

- The following figure gives a classification of different types of memory



TYPES OF RAM

There are 2 important memory device in the RAM family.

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

SRAM (Static RAM)

- It retains the content as long as the power is applied to the chip.
- If the power is turned off then its contents will be lost forever.

DRAM (Dynamic RAM)

- DRAM has extremely short Data lifetime(usually less than a quarter of second).

This is true even when power is applied constantly.

- b) A DRAM controller is used to make DRAM behave more like SRAM.
- c) The DRAM controller periodically refreshes the data stored in the DRAM. By refreshing the data several times a second, the DRAM controller keeps the contents of memory alive for a long time.

TYPES OF ROM

There are three types of ROM described as follows:

Masked ROM

- a. These are hardwired memory devices found on system. b. It contains pre-programmed set of instruction and data and it cannot be modified or appended in any way.
- b. (it is just like an Audio CD that contains songs pre-written on it and does not allow to write any other data)
- c. The main advantage of masked ROM is low cost of production.

PROM (PROGRAMMABLE ROM)

- a) This memory device comes in an un-programmed state i.e. at the time of purchased it is in an un-programmed state and it allows the user to write his/her own program or code into this ROM.
- b) In the un-programmed state the data is entirely made up of 1's. c) PROMs are also known as one-time-programmable (OTP) device because any data can be written on it only once. If the data on the chip has some error and needs to be modified this memory chip has to be discarded and the modified data has to be written to another new PROM.

EPROM (ERASABLE-AND-PROGRAMABLE ROM)

- a) It is same as PROM and is programmed in same manner as a PROM.
- b) It can be erased and reprogrammed repeatedly as the name suggests.
- c) The erase operation in case of an EPROM is performed by exposing the chip to a source of ultraviolet light.
- d) The reprogramming ability makes EPROM as essential part of software development and testing process.

TYPES OF HYBRID MEMORY

There are three types of Hybrid memory devices: EEPROMs

- a. EEPROMs stand for Electrically Erasable and Programmable ROM.
- b. It is same as EPROM, but the erase operation is performed electrically.
- c. Any byte in EEPROM can be erased and rewritten as desired

Flash

- a. Flash memory is the most recent advancement in memory technology.
- b. Flash memory devices are high density, low cost, nonvolatile, fast (to read, but not to write), and electrically reprogrammable.
- c. Flash is much more popular than EEPROM and is rapidly displacing many of the ROM devices.
- d. Flash devices can be erased only one sector at a time, not byte by byte.

NVRAM

- a. NVRAM is usually just a SRAM with battery backup.
- b. When power is turned on, the NVRAM operates just like any other SRAM but when power is off, the NVRAM draws enough electrical power from the battery to retain its content.
- c. NVRAM is fairly common in embedded systems.
- d. It is more expensive than SRAM.

DIRECT MEMORY ACCESS (DMA)

DMA is a technique for transferring blocks of data directly between two hardware devices. In the absence of DMA the processor must read the data from one device and write it to the other one byte or word at a time. DMA Absence Disadvantage: If the amount of data to be transferred is large or frequency of transfer is high the rest of the software might never get a chance to run.

DMA Presence Advantage: The DMA Controller performs entire transfer with little help from the Processor. Working of DMA The Processor provides the DMA Controller with source and destination address & total number of bytes of the block of data which needs transfer. After copying each byte each address is incremented & remaining bytes are reduced by one. When number of bytes reaches zeros the block transfer ends & DMA Controller sends an Interrupt to Processor.

EMBEDDED FIRMWARE

Embedded firmware is the flash memory chip that stores specialized software running in a chip in an embedded device to control its functions.

Firmware in embedded systems fills the same purpose as a ROM but can be updated more easily for better adaptability to conditions or interconnecting with additional equipment.

Hardware makers use embedded firmware to control the functions of various hardware devices and systems much like a computer's operating system controls the function of software applications. Embedded firmware exists in everything from appliances so simple you might not imagine they had computer control, like toasters, to complex tracking systems in missiles. The toaster would likely never need updating but the tracking system sometimes does. As the complexity of a device increases, it often makes sense to use firmware in case of design errors that an update might correct.

Embedded firmware is used to control the limited, set functions of hardware devices and systems of greater complexity but still gives more appliance-like usage instead of a series of terminal commands. Embedded firmware functions are activated by external controls or external actions of the hardware. Embedded firmware and ROM-based embedded software often have communication links to other devices for functionality or to address the need for the device to be adjusted, calibrated or diagnosed or to output log files. It is also through these connections that someone might attempt embedded device hacking.

Embedded software varies in complexity as much the devices it is used to control. Although *embedded software* and *embedded firmware* are sometimes used synonymously, they

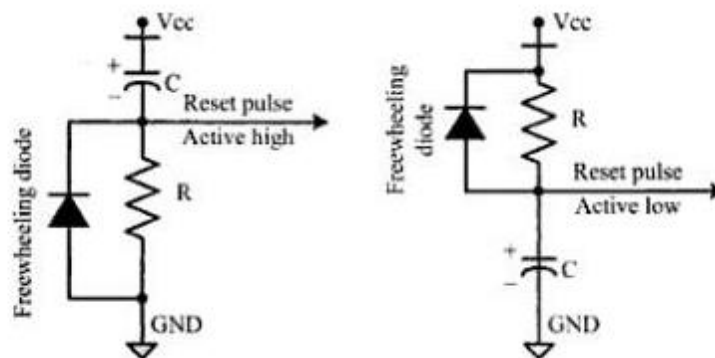
are not exactly the same thing. For example, embedded software may run on ROM chips. Also, embedded software is often the only computer code running on a piece of hardware while firmware can also refer to the chip that houses a computer's EFI or BIOS, which hands over control to an OS that in turn launches and controls programs.

Other components :

Reset circuit:

Microprocessors are complex, state-driven devices that must start up in a consistent way to function properly. You can establish proper processor operation by supplying a reset input that is normally asserted until the system is ready to execute the boot-up firmware. When the reset signal is deasserted, some subset of the processor's registers (depending on the specific chip) will be initialized to default values and the processor will start executing from fixed location (also specific to the chip). It's crucial to design this reset circuit properly to avoid system lockup, erratic processor operation, and possible corruption of your nonvolatile memory.

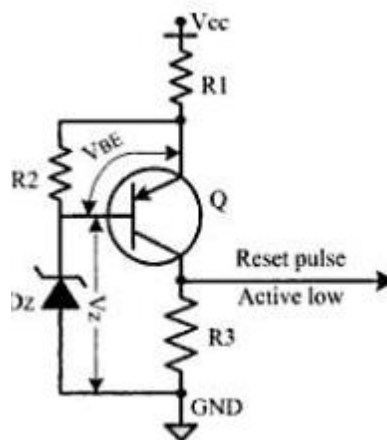
This is all complex enough that many companies now offer integrated circuit reset devices, commonly referred to as "reset supervisors." Good design practice suggests using these reset supervisors for most embedded systems because designing discrete reset circuitry is beyond the expertise of many embedded systems engineers. My personal experience has led me to rely on reset supervisors exclusively and ignore the various RC, transistor, and diode networks that are scattered throughout data books and shown in "example" circuits.



Brownout Protection

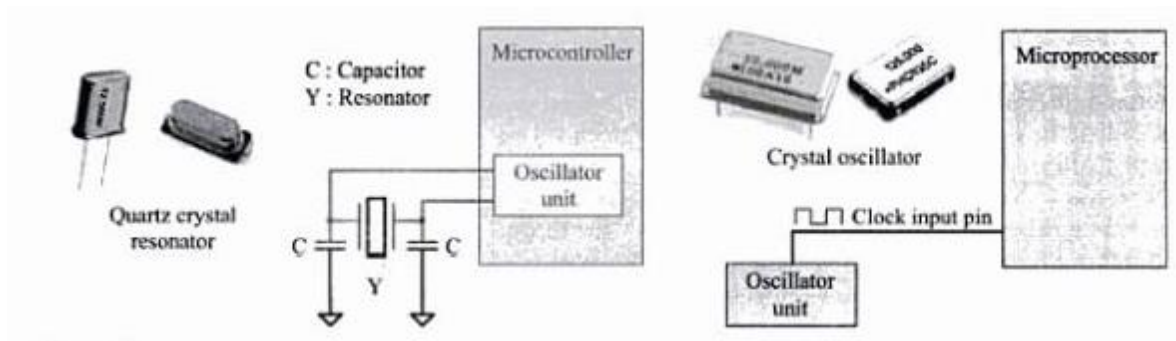
Brownout protection is built in them but when connecting a controller to an industry sensor and controlling devices (which are extremely costly) it's better we know what is a brownout and how it is detected in a microcontroller because many devices in low to medium scale industry may not be as immune to brownout as our controller. The brownout can cause one of the three things for a dc supply system. These things in turn can damage the connected embedded systems.

1. An unregulated direct current supply will produce a lower output voltage for electronic circuits. The output ripple voltage will decrease in line with the usually reduced load current.
2. A linear direct current regulated supply will maintain the output voltage unless the brownout is severe and the input voltage drops below the dropout voltage for the regulator, at which point the output voltage will fall and high levels of ripple from the rectifier/reservoir capacitor will appear on the output.
3. A switched-mode power supply which has a regulated output will be affected. As the input voltage falls, the current draw will increase to maintain the same output voltage and current, until such a point that the power supply malfunctions.



Oscillator circuit :

The majority of clock sources for microcontrollers can be grouped into two types: those based on mechanical resonant devices, such as crystals and ceramic resonators, and those based on electrical phase-shift circuits such as RC (resistor, capacitor) oscillators. Silicon oscillators are typically a fully integrated version of the RC oscillator with the added benefits of current sources, matched resistors and capacitors, and temperature-compensation circuits for increased stability.



These modules contain all oscillator circuit components and provide a clock signal as a low-impedance square-wave output. Operation is guaranteed over a range of conditions. Crystal oscillator modules and fully integrated silicon oscillators are most common. Crystal oscillator modules provide accuracy similar to discrete component circuits using crystals. Silicon oscillators are more precise than discrete component RC oscillator circuits, and many provide comparable accuracy to ceramic resonator-based oscillators.

RTC

A **real-time clock (RTC)** is a computer clock (most often in the form of an integrated circuit) that keeps track of the current time.

Although the term often refers to the devices in personal computers, servers and embedded systems, RTCs are present in almost any electronic device which needs to keep accurate time. A common RTC used in single-board computers is the DS1307.

Although keeping time can be done without an RTC,^[1] using one has benefits:

- Low power consumption^[2] (important when running from alternate power)
- Frees the main system for time-critical tasks
- Sometimes more accurate than other methods

RTCs are widely used in many different devices which need accurate time keeping.

- Real-time clocks normally have batteries attached to them that have very long life.
- Therefore, the batteries last a very long time, several years. The battery keeps the RTC operating, even when there is no power to the microcontroller that is connected up to. So even if the microcontroller powers off, the RTC can keep operating due to its battery. Therefore, it can always keep track of the current time and have accurate time, ongoing.

An RTC maintains its clock by counting the cycles of an oscillator – usually an external 32.768kHz crystal oscillator circuit, an internal capacitor based oscillator, or even an embedded quartz crystal. Some can detect transitions and count the periodicity of an input that may be connected.

This can enable an RTC to sense the 50/60Hz ripple on a mains power supply, or detect and accumulate transitions coming from a GPS unit epoch tick. An RTC that does this operates like a phase locked loop (PLL), shifting its internal clock reference to ‘lock’ it onto the external signal. If the RTC loses its external reference, it can detect this event (as its PLL goes out of lock) and free run from its internal oscillator.

A watchdog timer (WDT):

WDT is a hardware timer that automatically generates a system reset if the main program neglects to periodically service it. It is often used to automatically reset an embedded device that hangs because of a software or hardware fault. Some systems may also refer to it as a computer operating properly (COP) timer. Many microcontrollers including the embedded processor have watchdog timer hardware.

The main program typically has a loop that it constantly goes through performing various functions. The watchdog timer is loaded with an initial value greater than the worst case time delay through the main program loop. Each time it goes through the main loop the code resets the watchdog timer (sometimes called “kicking” or “feeding” the dog). If a fault occurs and the main program does not get back to reset the timer before it counts down, an interrupt is generated to reset the processor. Used in this way, the watchdog timer can detect a fault on an unattended embedded device and attempt corrective action with a reset. Typically after reset, a register can also be read to determine if the watchdog timer generated the reset or if it was a normal reset. On the mbed this register is called the Reset Source Identification Register (RSID).

