prof. RAMESH PATIL
E&CE dept.
GNDEC, Bidar

Module - IV
Embedded systems.

* Definition

An embedded system is an electronic/electro-mechanical system designed to perform a specific function and is a combination of both hardware and firmware (software). Modern embedded systems are often based on micro controllers

* Embedded system V/s General computing systems.

General purpose computing system	Embedded system.
- A system which is a combination of generic hardware and a General purpose Operating System (GPOS) for executing a variety of applications.	- A system which is a combination of special purpose hardware and embedded OS for executing a specific set of applications.
- contains a General purpose Operating System (GPOS)	- may or may not contain an OS for functioning
- Applications are alterable by the user. It is also possible for the end user to re-install the OS and also add or remove user applications.	- The firmware (software) of the embedded system is pre-programmed and it is non alterable by the end user.
- performance is the key deciding factor in the selection of the system.	- Application-specific requirements are key deciding factors (Like power requirements, memory etc)
- Takes the support of hardware and OS to save the power	- Less / not at all tailored towards reduced operating power requirements.
- Response requirements are not time critical	- Response time requirement is highly critical.
- Execution behaviour is not deterministic	- Execution behaviour is deterministic for certain types of embedded systems. like 'Hard Real Time' systems.

* Classification of Embedded systems.

The classification of embedded systems are as follows:

1) Based on generation
2) Based on Complexity and performance requirements.
3) Based on deterministic behaviour.
4) Based on triggering.

1) Based on generation.

This classification based on the order in which the embedded systems evolved from the first version to where they are today.

* First Generation.

The early embedded systems were built around 8-bit microprocessors like 8085 and Z80 and 4-bit microcontrollers.

Ex: Digital telephone keypads, stepper motor controller etc.

* Second Generation.

These are embedded systems built around 16-bit microprocessor and 8 or 16-bit microcontrollers. The instruction set is much more complex compared to first generation.

Ex: Data Acquisition systems, SCADA systems etc.

* Third Generation.

These are embedded systems built around 32-bit processor and 16-bit microcontrollers. A new concepts of application and domain specific processors/controllers like Digital signal processor (DSP) and Application specific integrated circuits (ASICs) came in to the picture. The instruction set of processors become more complex & powerful.

Ex: Robotics, media, industrial process control, Nlwong etc.

* Fourth Generation.

The system on chips (SOC), reconfigurable processors & multicore processors brings high performance & miniaturisation in to the embedded system. use high performance, RTOS for their functioning.

Ex: smart phones, mobile internet devices (MIDs). etc.

2) Based on Complexity and performance.

* Small-scale Embedded systems.

A small-scale embedded systems are usually around low performance and low cost 8 or 16-bit microprocessor/microcontroller. A small scale embedded system may or may not contain OS for its functioning.

Ex: Electronic Toy.

* Medium-scale Embedded Systems.

Slightly complex in hardware and firmware. These are built around 16 or 32 bit microprocessor/microcontrollers. or DSP (digital signal processors).

* Large-Scale Embedded Systems/Complex Systems.

These are usually built around high performance 32 or 64-bit RISC processors or RSOC or multicore processor and programmable logic devices.

Ex: Decoding/Encoding of media, cryptographic function implementation etc.

* Major Application areas of Embedded Systems.

major application areas of embedded systems are as follows

1) Consumer electronics: Camcorders, Cameras, etc.

2) Household appliances; TV, DVD player, washing machines, Fridge, microwave oven, etc.

3) Home automation and Security systems:

Air conditioners, sprinklers, intruder detection alarm, CCTV cameras, fire alarms, etc.

4) Automotive industry:

Anti-lock braking systems (ABS), engine control, ignition systems, automatic navigation systems, etc.

5) Telecom.

Cellular phones, telephone switches, handset, multimedia applications, etc.

6) Computer peripherals
   Printers, scanners, Fax machines, etc.

7) Computer networking systems.
   Network routers, switches, hubs, firewalls, etc.

8) Health care.
   Different kinds of scanners, EEG, ECG machines, etc.

9) Measurement and Instrumentation.
   Digital multimeters, Digital CROs, logic analyzers, PLC, e

10) Banking and Retail.
   Automatic teller machines (ATM), currency counters
   Point of sales (POS).

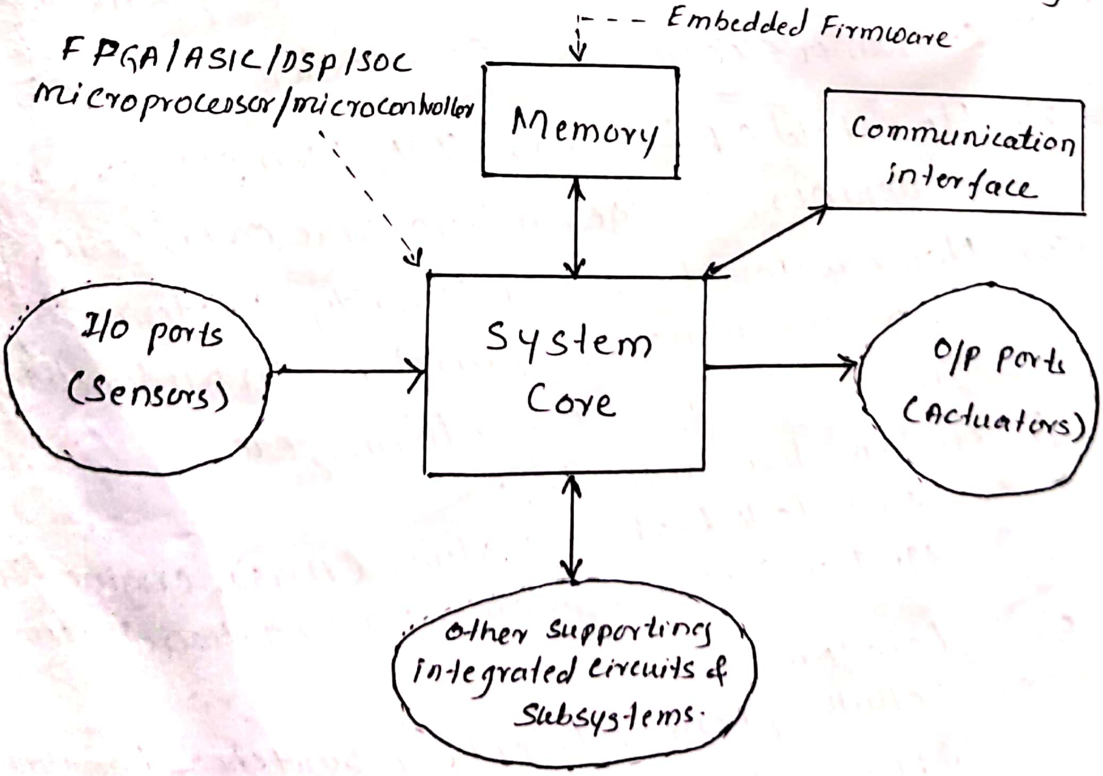
11) Card Readers.
   Bar Code, smart card readers, hand held devices, etc.

12) Wearable Devices.
   Health and fitness trackers, etc.

13) Cloud computing and Internet of Things (IOT).

* Elements of Embedded systems.

A typical embedded system is shown in fig.



A typical embedded system contains a single chip controller, which acts as the master brain of the system. A controller can be a microprocessor or a microcontroller or Field Programmable Gate Array (FPGA) or Digital signal processor (DSP) or an ASIC. Application specific integrated Circuit.

Embedded systems are basically designed to regulate a physical variable or to manipulate the state of some devices by sending some signals to the actuators or devices connected to the O/P ports of the system in response to the input signals provided by the end users or sensors which are connected to the input ports.

Keyboards, push button switches, etc are examples for input devices whereas LEDs, LCD, buzzers etc are examples for output devices.

The memory of the system is responsible for holding the Control algorithm and other important configuration details. An embedded system without code implemented memory has all the peripherals but is not capable of making decisions depending on the situational as well as real world changes. memory for implementing the code may be present on the processor or may be implemented as a seperate chip interfacing the processor.

* Core of the Embedded System.

The core of the embedded system falls into any of the following categories:

1) General purpose and Domain specific processors.
   a) microprocessors
   b) micro controllers.
   c) Digital signal processors (DSP)
2) Application Specific Integrated Circuits (ASICs)
3) Programmable Logic Devices (PLDs)
4) Commercial off-the-self components (COTS).

In any embedded system we will find that it is built arround any of the core units mentioned above.

* Microprocessor vs Microcontroller.

Microprocessor	Microcontroller
- A silicon chip representing a central processing unit (CPU), which is capable of performing arithmatic as well as logical operations according to a predefined set of instructions	- A microcontroller is highly integrated chip that contains a CPU, RAM, Rom for program storage, timer and intrupt control units and dedicated I/O ports.
- It is a dependent unit. It requires combination of other chips like timers, program and data memory chips intrupt controller, etc. for functioning.	- It is self-contained unit and it does not require intrupt controller, timer, UART, etc. for its functioning.
- most of the time general purpose in design and operation	- mostly application-oriented or domain specific.
- Doesnot contain a built-in I/O port.	- most of the processors contain built-in I/O ports.
- Targeted for high end market where performance is important	- Targeted for embedded market where performance is not so critical.
- Limited power saving options compared to microcontrollers	- Includes lot of power saving features.
Ex: Intel 8086 microprocessor	Ex: Intel 8051 microcontroller.

* RISC Vs. CISC processors / controllers.

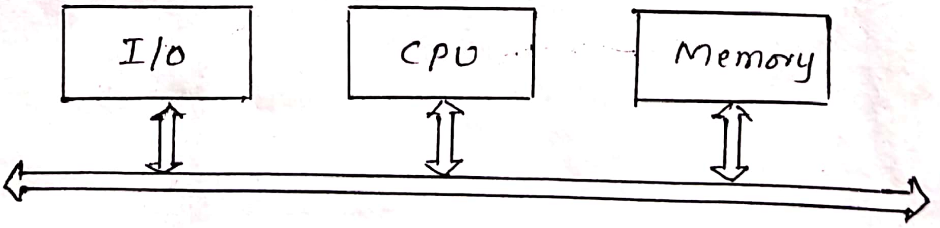
The term RISC stands for Reduced Instruction set Computing. As name implies, all RISC processors / controllers possess lesser number of instructions.

CISC stands for Complex Instruction set Computing. From the definition itself it is clear that the instruction set is complex and instructions are high in number.

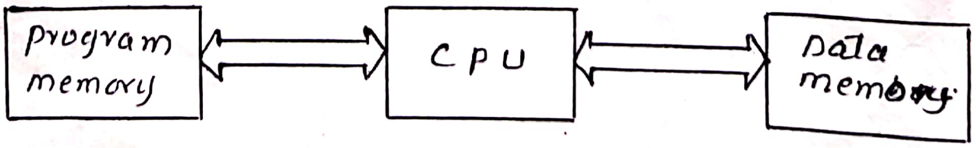
RISC	CISC
- Lesser number of instructions.	- Greater number of instructions.
- instruction pipelining and increased execution speed.	- Generally no instruction pipelining feature
- orthogonal instruction set	- Non orthogonal instruction set
- Operations are performed on registers only, the only memory operations are load and store	- operations are performed on registers or memory depending on the instruction.
- A large number of registers are available	- Limited number of general purpose registers.
- programmer needs to write more code, fixed length instruction	- variable length instruction
- less silicon usage and pin count	- more silicon usage.
- with Hardvard Architecture	- can be Hardvard or Von-Neumann Architecture.

* Hardvard Vs. Von-Neumann processor/controller Architecture

microprocessor/controllers based on the Von-Neumann architecture shares a single common bus for fetching both instructions and data. Also know an princeton architecture.



microprocessor/controllers based on the Harvard architecture will have seperate data bus and instruction bus. as shown below.



Harvard Architecture	Von-Neumann Architecture
- seperate buses for instruction and data fetching	- single shared bus for instruction and data fetching.
- Easier to pipeline, so high performance can be achieved	- Low performance compared to Hardard architecture
- Comporatively high cost	- Cheaper.
- No memory alignment problem.	- Allows self modifying codes
- Since data memory and program memory are stored physically in different Locations, no changes for accidental Corruption of program memory.	- since data and program memory are stored physically in the same chip, chances for accidental corruption of program memory.

E&CE dept.
GNDEC, Bidar.

Sensors and Interfacing.

Sensor is a device that provide us with a means of generating signals that can be used as inputs to electronic circuits.

* Instrumentation and control system.

Fig.1 shows the arrangement of an instrumentation system.

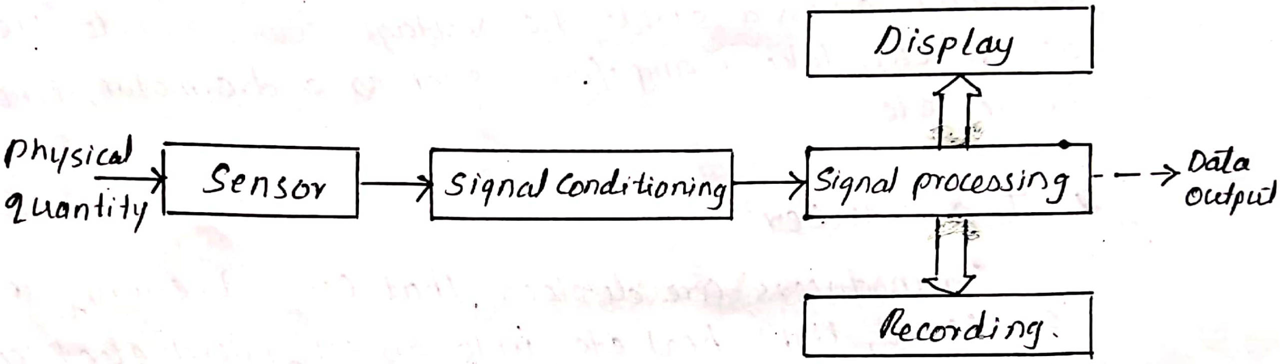


Fig.1. An Instrumentation system.

The physical quantity (eg. temperature) to be measured acts upon a sensor that produces an electrical output signal. This signal is passed to the signal conditioning since the o/p produced by sensor may contain unwanted signal (i.e noise), so further signal conditioning is required. O/p of signal conditioning is applied for further signal processing, display or for recording.

Fig. 2. Shows the arrangement of a control system.

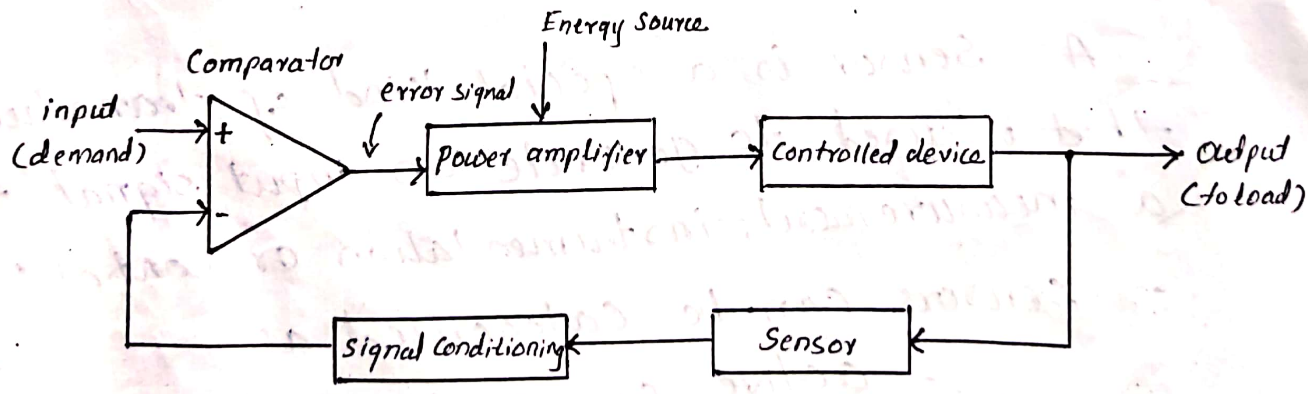


Fig. 2. A. Control system.

This uses negative feedback in order to regulate and stab[illegible] the output. It thus becomes possible to set the input or deman[illegible] 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 discrepancy is detected the input to the power amplifier is adjusted accordingly. This signal is called error signal.

The input is often derived from a single potentiometer connected across a stable d.c. voltage source while the controlled device can take many forms such as a d.c. motor, linear actuator, heater etc.

* Transducers:

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

Transducers may be used both as inputs to electronic circuits and outputs from them.

Two types of transducers.

1) Input transducers
   Ex: microphone, thermocouple, rotary potentiometer etc.
2) output transducers
   Ex: Loud speakers, heating element, rotary potentiometer etc.

* Sensors:

A Sensor is a special kind of transducer that is used to generate an input signal to a measurement, instrumentation or control signal

- sensors can be categorized as
  - active or passive

An active sensor generates a current or voltage output on its own.

A passive sensor requires a source of current or voltage for its operation.

- Sensors can be classified as
  - digital or analog sensor

The output of a digital sensor can exist in only two discrete state i,e 'ON' or 'OFF', 'LOW' or 'HIGH' or 'logic 1' or 'logic 0'.

The output of an analog sensor can take any one of an infinite number of voltage or current. It is a contineous signal.

Sensor is a transducer which converts energy from physical form to electrical form. Sensors acts as an input device.

Example : 1) physical quantity : sound
input transducer : microphone

2) physical quantity : Temperature
input transducer : Thermocouple

3) physical quantity : Angular position
input transducer : Rotary potentiometer