Department of Electronics & Communication Engg.

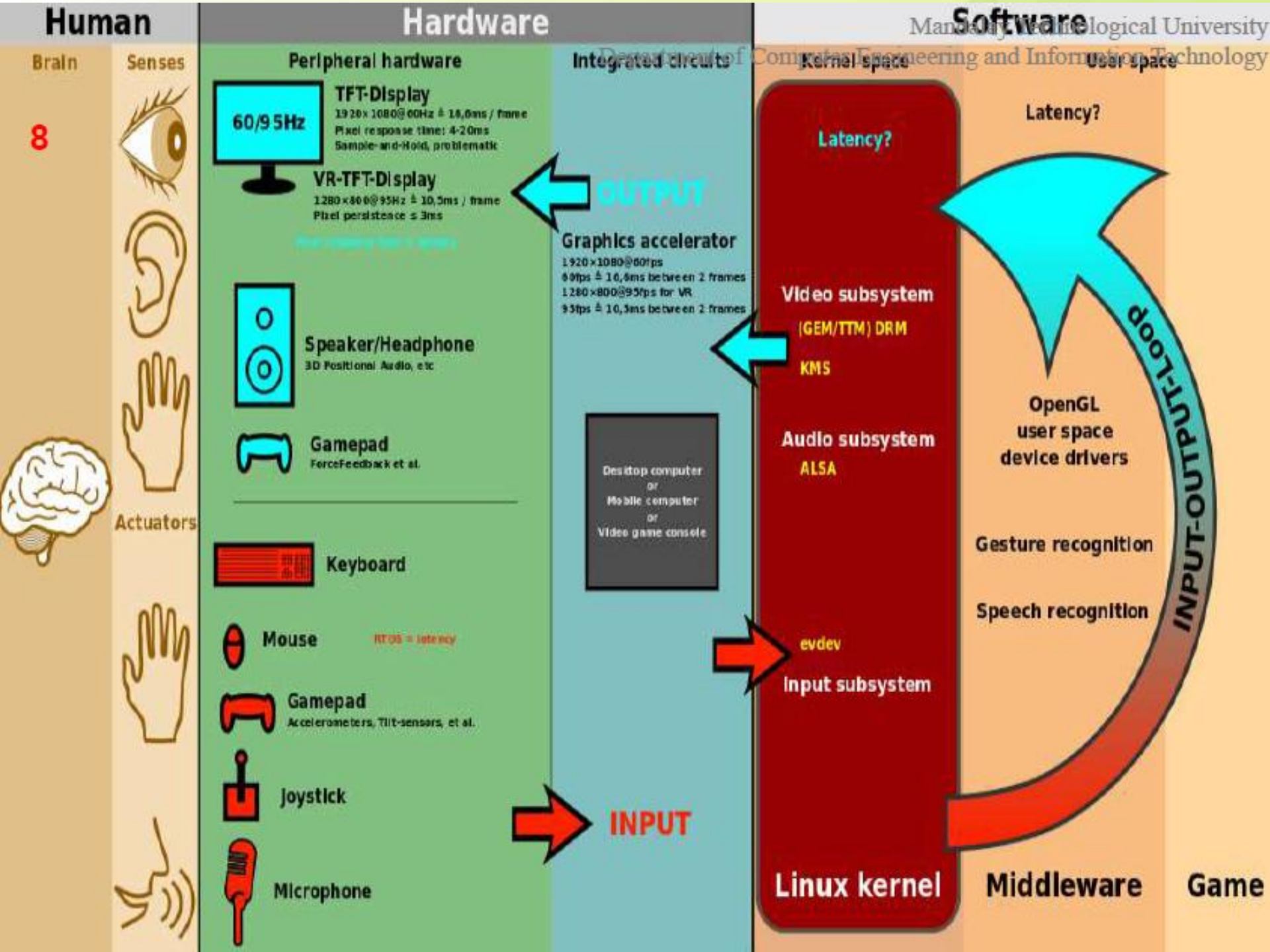
Course : ARM Microcontroller & ES-15EC62. . **Sem.: 6th (2017-18)**

Course Coordinator:

Prof. Sachin S Patil

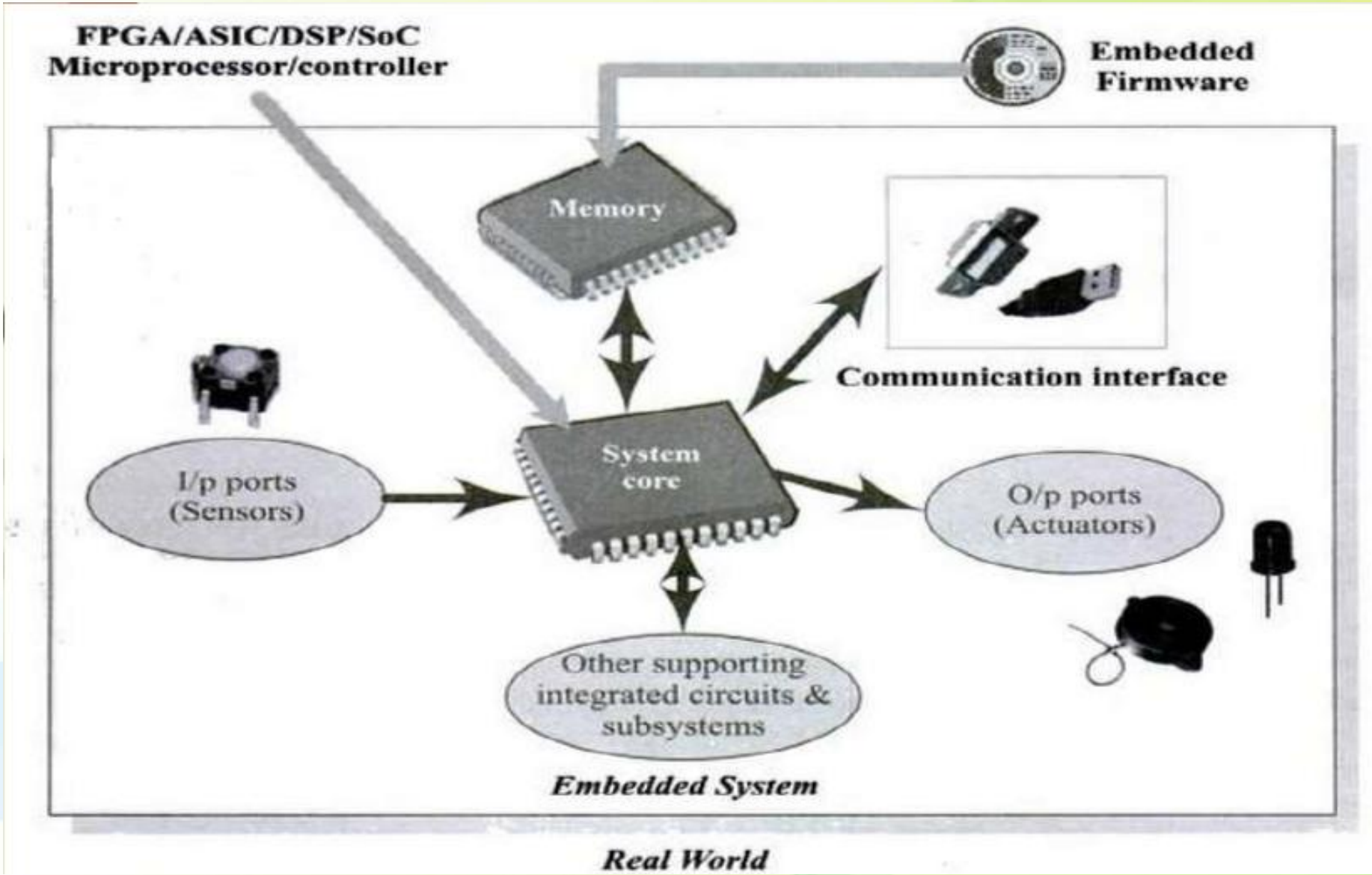
Typical Embedded System

 A typical embedded system contains a single chip controller, which acts as the master brain of the system.
 The controller can be a Microprocessor or a microcontroller or a Field programmable Gate Array (FPGA) device or a Digital Signal Processor (DSP) or an Application Specific Integrated Circuit (ASIC)/ Application Specific Standard Product (ASSP).
 Embedded hardware/software systems are basically designed to **regulate a physical variable or to manipulate the state of some devices by sending** some control 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 common user interface **input devices whereas LEDs, liquid crystal displays,** piezoelectric buzzers, etc. are examples for common user interface **output devices for a typical embedded system.**
- **For example, if the embedded system is designed for any handheld** application, such as a mobile handset application, then the system should contain user interfaces like a keyboard for performing input operations and display unit for providing users the status of various activities in progress.

Elements of an Embedded System



Core of Embedded Systems

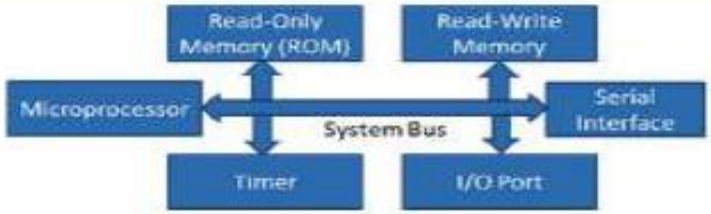
 The core of the embedded system falls into any of the followingcategories:

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

Merits, Drawbacks and Application Areas of Microcontrollers and Microprocessors

- **Microcontrollers are designed to perform specific tasks. However, Microprocessors are designed to perform unspecific tasks like developing** software, games, website, photo editing, creating documents, etc.
- Depending on the input, some processing for **microcontroller needs to be done** and output is **defined. However, the relationship between input and output for microprocessor is not defined.**
- Since the applications of **microcontroller are very specific, they need small resources like RAM, ROM, I/O ports etc. and hence can be embedded on a single chip. Microprocessors need high amount of resources like RAM,** ROM, I/O ports etc.
- The clock speed of Microprocessor is **quite high as compared to the** microcontroller. Whereas the **microcontrollers operate from a few MHz (from** 30 to 50 MHz), today's **microprocessor operate above 1 GHz as they perform complex tasks**

 Microprocessor cannot be used stand alone. They need other peripherals like RAM, ROM, buffer, I/O ports etc and hence a system designed around a microprocessor is quite costly.
 Application areas of microcontroller: Mobile phones, CD/DVD players, Washing machines, Cameras, Security alarms, microwave oven, etc.
 Application areas of microprocessor: Calculators, Accounting Systems, Games Machine, Complex Industrial Controllers, Data Acquisition Systems, Military applications, Communication systems, etc.

Microprocessor	Micro Controller
	
Microprocessor is heart of Computer system.	Micro Controller is a heart of embedded system.
It is just a processor. Memory and I/O components have to be connected externally	Micro controller has external processor along with internal memory and i/O components
Since memory and I/O has to be connected externally, the circuit becomes large.	Since memory and I/O are present internally, the circuit is small.
Cannot be used in compact systems and hence inefficient	Can be used in compact systems and hence it is an efficient technique
Cost of the entire system increases	Cost of the entire system is low
Due to external components, the entire power consumption is high. Hence it is not suitable to used with devices running on stored power like batteries.	Since external components are low, total power consumption is less and can be used with devices running on stored power like batteries.
Most of the microprocessors do not have power saving features.	Most of the micro controllers have power saving modes like idle mode and power saving mode. This helps to reduce power consumption even further.
Since memory and I/O components are all external, each instruction will need external operation, hence it is relatively slower.	Since components are internal, most of the operations are internal instruction, hence speed is fast.
Microprocessor have less number of registers, hence more operations are memory based.	Micro controller have more number of registers, hence the programs are easier to write.
Microprocessors are based on von Neumann model/architecture where program and data are stored in same memory module	Micro controllers are based on Harvard architecture where program memory and Data memory are separate
Mainly used in personal computers	Used mainly in washing machine, MP3 players

Digital Signal Processors

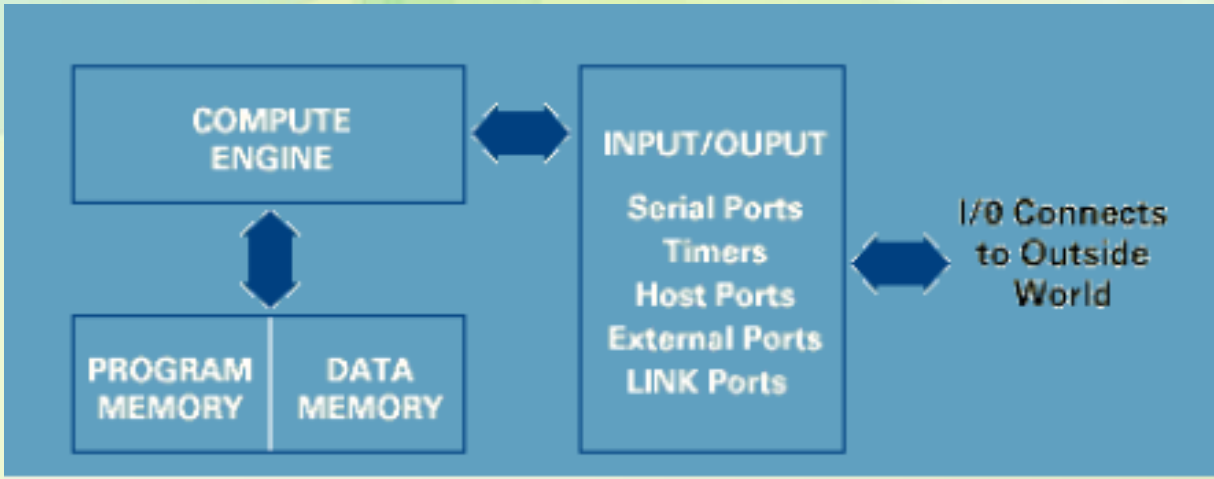
- DSPs are **powerful special purpose 8/16/32 bit microprocessors designed specifically** to meet the computational demands and power constraints of today's embedded audio, video, and communications **applications.**
- Digital signal processors are **2 to 3 times faster than the general purpose** microprocessors in signal processing applications.
- A typical digital signal processor incorporates the following key units:

i. Program Memory : Memory for storing the program required by DSP to process the data.

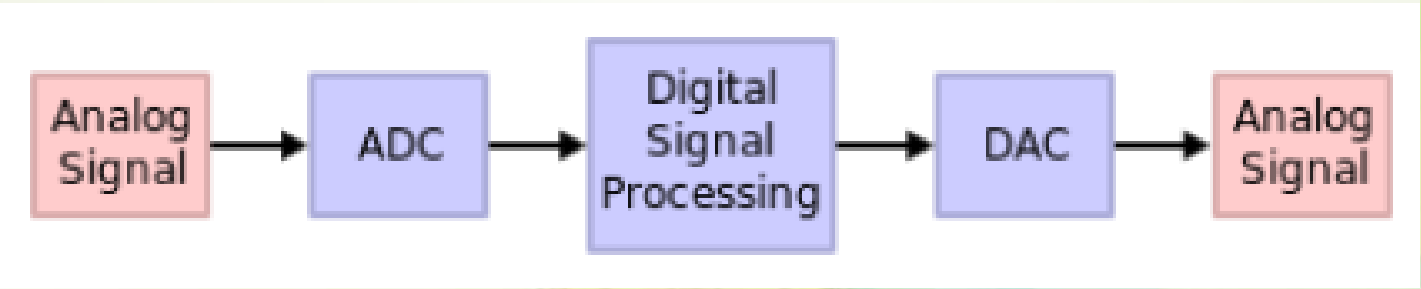
ii. Data Memory : Working memory for storing temporary variables/ information and data/signal to be processed.

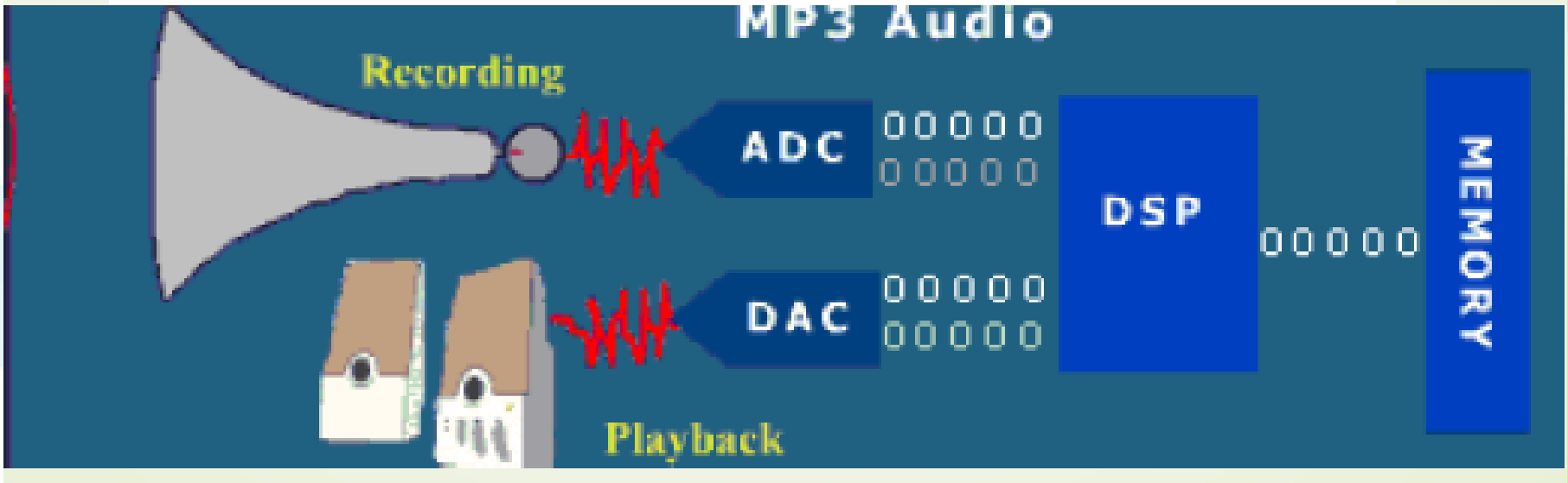
iii. Computational Engine : Performs the signal/math processing, accessing the program from the Program Memory and the data from the Data Memory.

iv. I/O Unit : Acts as an interface between the outside world and DSP. It is responsible for capturing signals to be processed and delivering the processed signals.



- **Application areas : Audio video signal processing, telecommunication and** multimedia applications.
- DSP employs **a large amount of real-time calculations, Sum of products (SOP) calculation, convolution, fast Fourier transform (FFT), discrete Fourier transform (DFT), etc. are some of the operations performed by digital signal** processors.

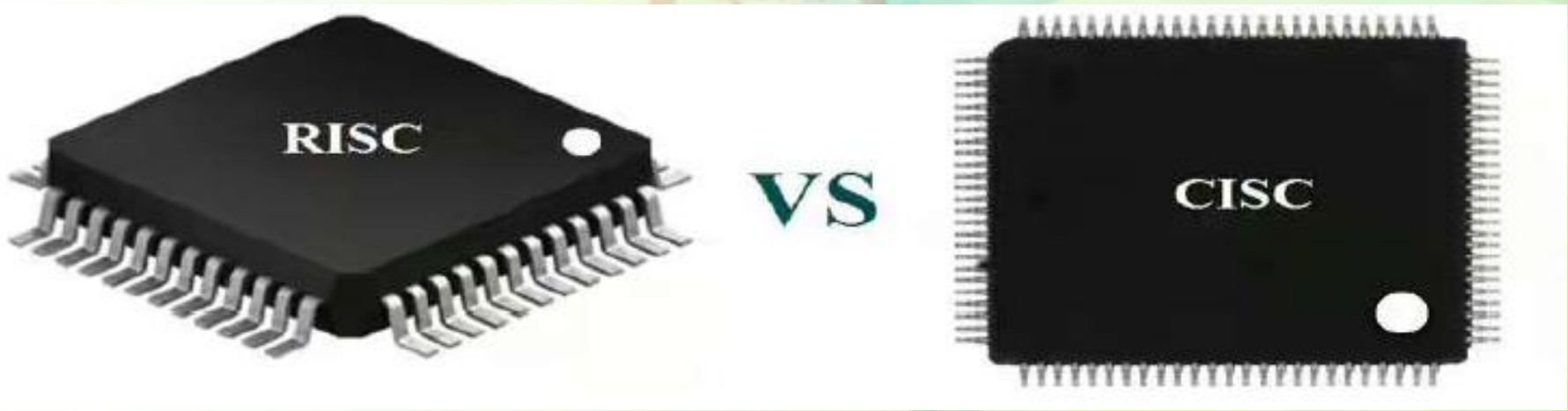




RISC vs CISC Processors/Controllers

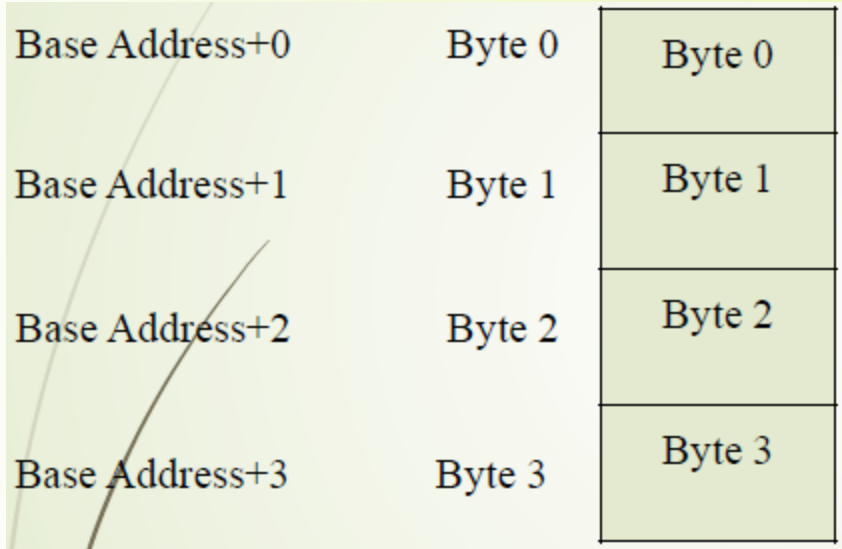
RISC	CISC
Lesser number of instructions	**Greater number of Instructions**
Instruction pipelining and increased execution speed	Generally **no instruction pipelining feature**
Orthogonal instruction set	**Non-orthogonal** instruction set
Operations are performed **on registers only**, the only memory operations are load and store.	Operations are performed **on registers or memory** depending on the instruction.
A large number of registers are available.	**Limited** number of general purpose registers.
Programmer **needs to write more code to execute a task** since **the instructions are simpler ones**.	Instructions are like macros in **C language**. A programmer can achieve the desired functionality with a single instruction which in turn provides the effect of using **more simpler single instructions** in RISC.

RISC	CISC
Single, fixed length instructions	**Variable length instructions**
Less silicon usage and pin count	**More** silicon usage since more additional decoder logic is required to implement the complex instruction decoding.
With **Harvard Architecture**	Can be **Harvard or Von-Neumann Architecture**



Big-Endian vs. Little-Endian Processors/Controllers

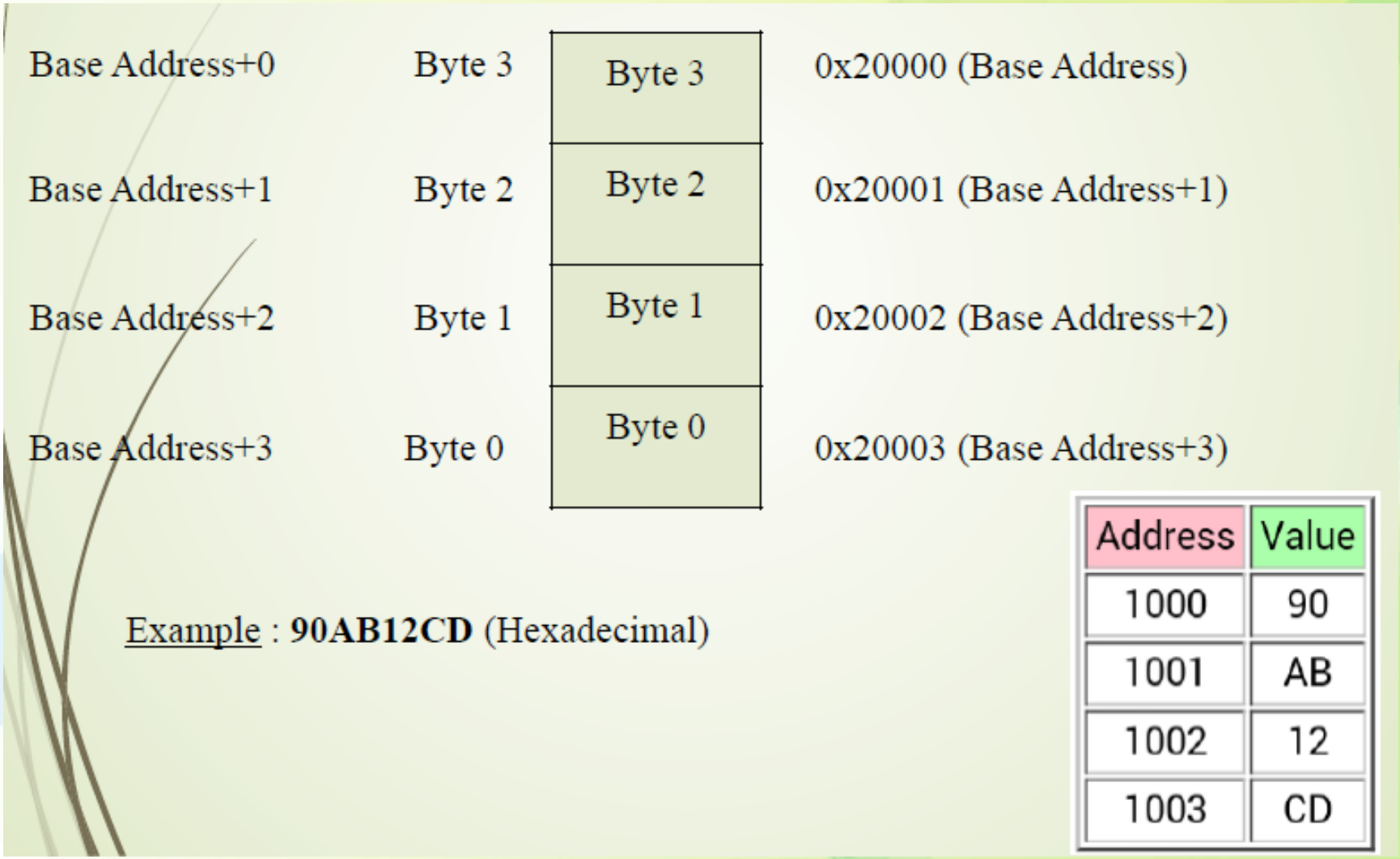
- **Endianness specifies the order in which a sequence of bytes are stored** in computer memory.
- **Little-endian is an order in which the "little end"/ the lower-order byte** of the data (least significant value in the sequence) is stored in memory at the lowest address. (The little end comes first.)
- For example, a 4 byte long integer **Byte3, Byte2, Byte1, Byte0 will be** stored in the memory as shown below: 20 Mandalay Technological University Department of Computer Engineering and Information Technology



Example : **90AB12CD** (Hexadecimal)

Address	Value
1000	CD
1001	12
1002	AB
1003	90

- **Big-endian is an order in which the "big end" / the higher-order byte of** the data (most significant value in sequence) is stored in memory at the lowest address. (The big end comes first.)
- For example, a 4 byte long integer **Byte3, Byte2, Byte1, Byte0 will be** stored in the memory as shown below:



Address	Value
1000	90
1001	AB
1002	12
1003	CD

Queries ...?