Embedded Systems- An Overview

Prof.S.S.S.P.Rao
Dept. of Computer Science & Engg.
I.I.T.Bombay
ssspr@cse.iitb.ac.in
http://www.cse.iitb.ac.in/~ssspr

TOPICS

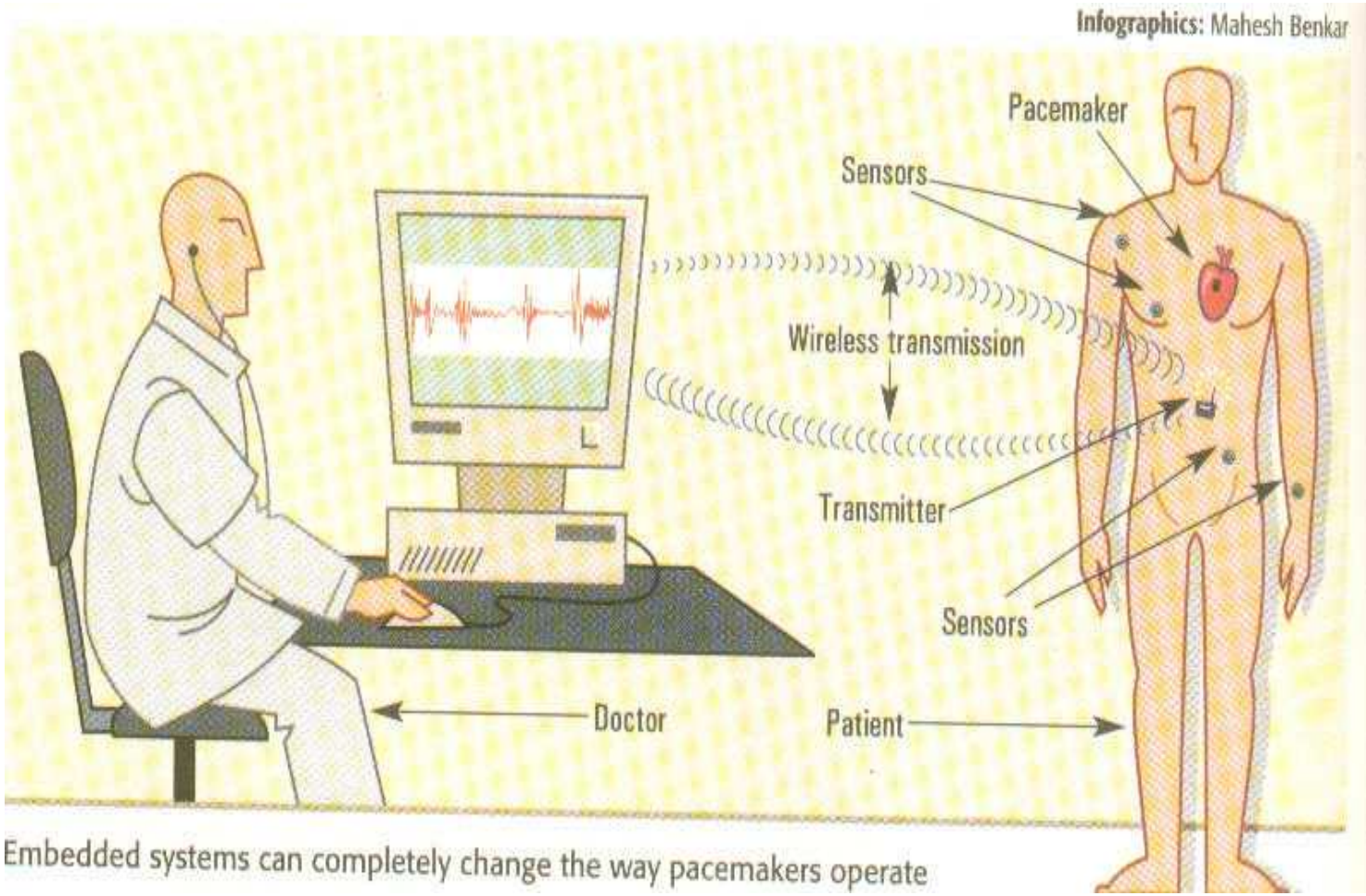
- **Introduction**
- **Microcontrollers**
- **Embedded Systems design issues**
- **Conclusions**

A Doctor is configuring a cardiac Pacemaker inside his patient's chest while sitting 2 kms away.

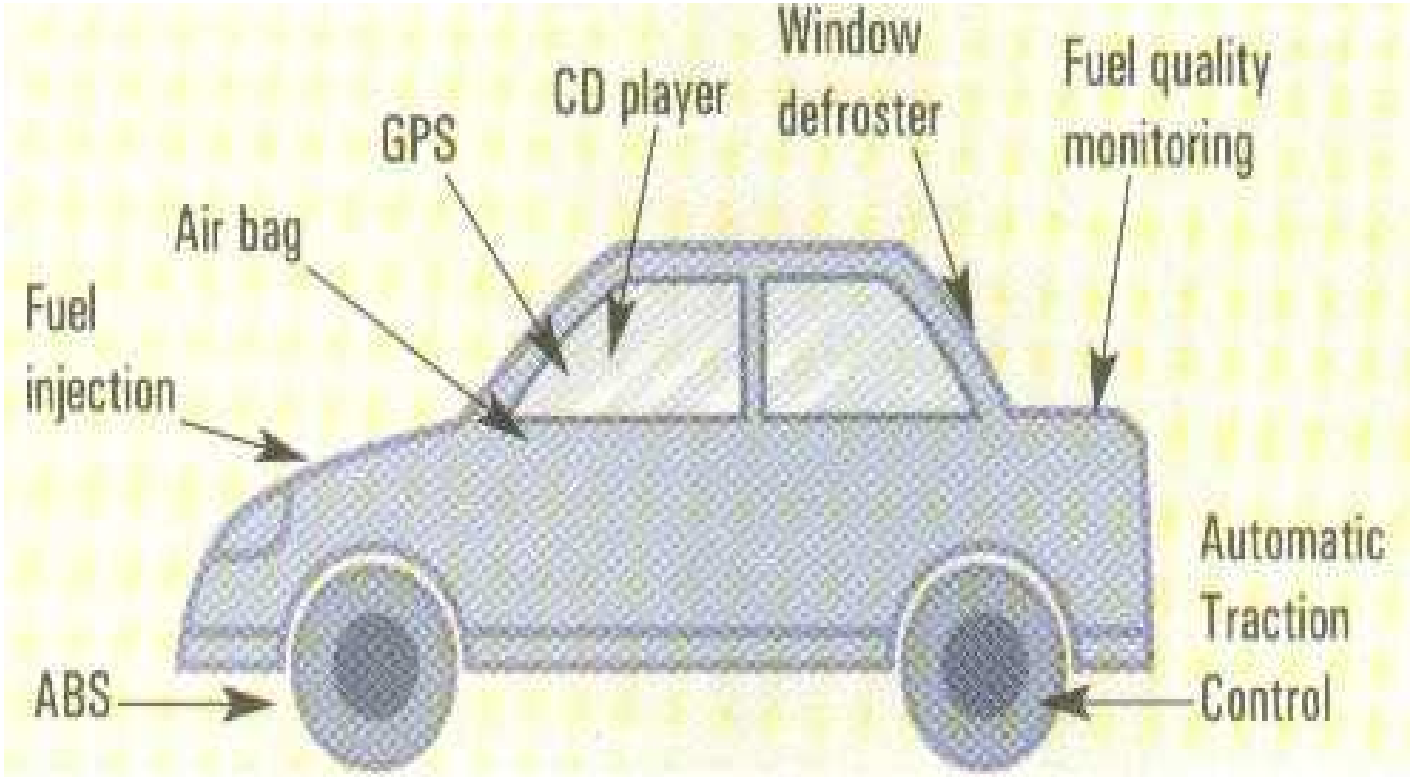
Another person is travelling in a driverless car that takes him from Mumbai to New Delhi using his inbuilt navigation programme.

Looks impossible and sounds like fairy tale!!!!

No not really….



Embedded systems can completely change the way pacemakers operate



Embedded systems in a car

- **Advances in Technology have taken place at such a speed that these fictitious scenario are likely to be translated into reality very soon in a couple of years.**
- **Real Time Operating System (RTOs) and Embedded system are the major technologies that will play a major role in making the above fairy tales come true.**

Introduction to Embedded Systems

What are embedded systems?

- Computer (Programmable part) surrounded by other subsystems,sensors and actuators
- computer a (small) part of a larger system.
- The computer is called a micro-controller

- Embedded Systems or Electronics systems that include an application Specific Integrated Circuit or a Microcontroller to perform a specific dedicated application.

- Embedded System is pre-programmed to do a specific function while a general purpose system could be used to run any program of your choice. Further, the Embedded Processor Is only one component of the electronic system of which it is the part. It is cooperating with the rest of the components to achieve the overall function.

Why Sudden interest in Embedded sytems?

Possible Reasons:

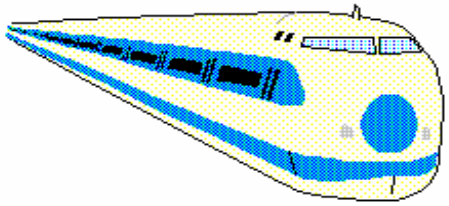
- Processors have shrunk in size with increased performance
- Power consumption has drastically reduced.
- Cost of processors have come down to affordable level.
- There is a greater awareness now that rather than a totally hardwired electronic system, incorporation of a programmable processor in a circuit makes the design more robust with the reduction in the design time cycle.

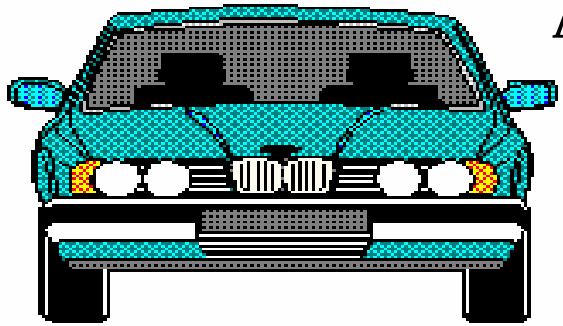
- The concept of a development environment where you can prototype the system and do a simulation/emulation also reduces the design cycle and total development time.
- The latest model of the Ford car has more than 21 microcontrollers performing functions such as anti-lock breaking system, fuel management, air-condition management, etc.

- Development of standard run time platforms like Java, which enabled their use in Myriad ways that were unimaginable in the past.
- Emergence of several integrated software environments which simplified the development of the applications.
- Coming together of Embedded systems and the Internet which make possible networking of several Embedded systems to operate as a part of large system across networks - be it a LAN, WAN or the internet.

Embedded Systems are everywhere

- **wristwatches, washing machines, microwave ovens,**
- **elevators, mobile telephones,**
- **printers, FAX machines, Telephone exchanges,**
- **automobiles, aircrafts**
- **An average home in developed countries**
  - has one general purpose desktop PC
  - but has a dozen of embedded systems.
- **More prevalent in industrial sectors**
  - half a dozen embedded computers in modern automobiles
  - chemical and nuclear power plants

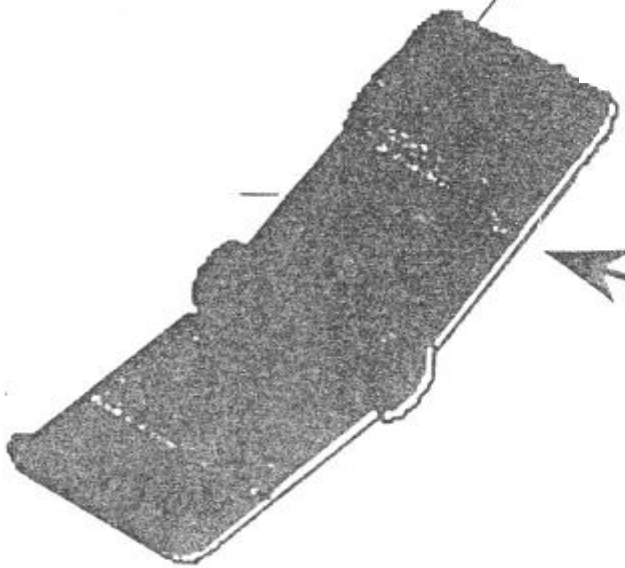


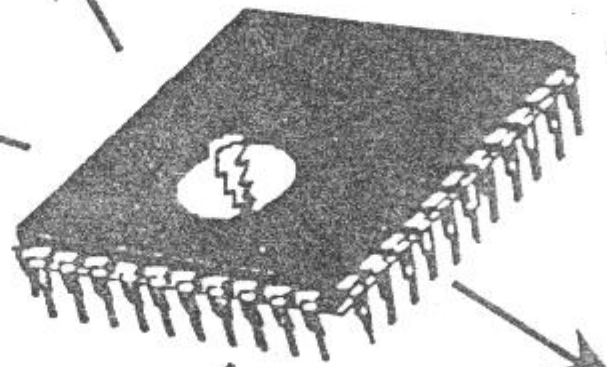


Auto- motive

Communications

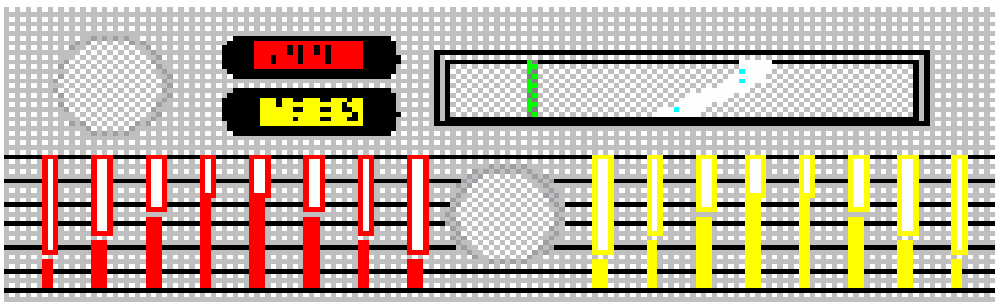
Medical

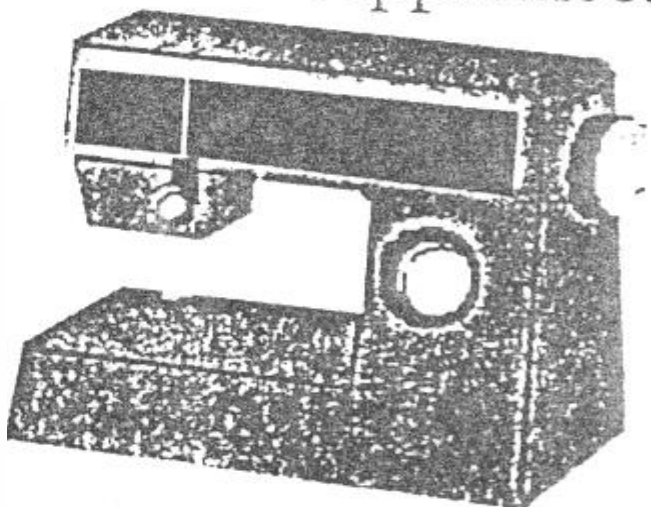




Consumer electronics

Appliances





Examples of Microcomputer applications

Consumer	Function Performed by Microcomputer
Washing Machine	Controls the water & spin cycles
Remote controls	Accepts key touches and sends infrared (IR) pulses to base systems
Exercise equipment	Measures speed, distance calories, heart rate, logs workouts
Clocks and watches	Maintains the time, alarm and display
Games and toys	Entertains the user, joystick input, video output
Audio/video	Interacts with the operator & enhances performance

Communication:

Telephone answering machines	Plays outgoing message, saves and organizes messages
Telephone system	Interactive switching and information retrieval
Cellular phones and pagers	Key pad and inputs, sound I/O, and communicates with central station
ATM machines	Provides both security and banking convenience

Automotive

Automatic braking	Optimizes stopping on slippery surfaces
Noise cancellation	Improves sound quality by removing background noise
Theft deterrent devices	Keyless entry, alarm systems
Electronic ignition	Controls sparks and fuel injectors
Power windows & seats	Remembers preferred settings for each driver
Instrumentation	Collects & provides the driver with necessary information

Military

Smart weapons	Recognizes friendly targets
Missile guidance systems	Directs ordinance at the desired target
Global positioning systems	Determines where you are on the planet

Industrial:

Setback thermostats	Adjusts day/night thresholds, thus saving energy
Traffice control systems	Senses car positions and controls traffic lights
Robot systems	Input from sensors, controls the motors
Bar code readers and writers	Input from readers, output to writers for inventory control and shipping
Automatic sprinklers	Used in farming to control the wetness of the soil

Medical:

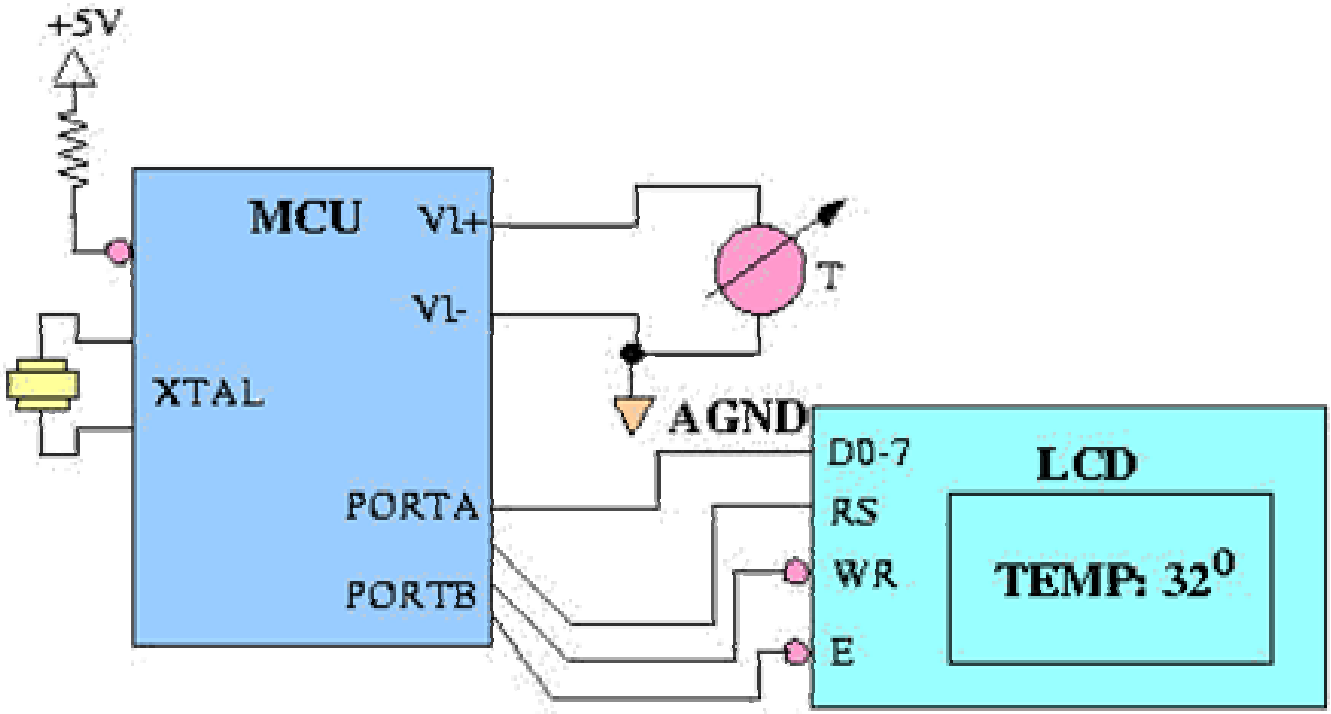
Apnea monitors	Detects breathing & alarms if the baby stops breathing
Cardiac monitors	Measures heart functions
Renal monitors	Measures kidney functions
Drug delivery	Administers proper doses
Cancer treatments	Controls doses of radiation, drugs, or heat
Prosthetic devices	Increases mobility for the handicapped
Dialysis machines	Performs functions normally done by the kidney
Pacemaker	Helps the heart beat regularly

Controller's function is

- to monitor parameters of physical process of its surrounding system
- to control these processes whenever needed.

Example:

- a simple thermostat controller
  - periodically reads the temperature of the chamber
  - Displays the reading
  - and switches on or off the cooling system, as required.



A Microcontroller-Based System

Another important example:

- a pacemaker
  - constantly monitors the heart and
  - paces the heart when heart beats are missed

What goes into an Embedded System

An Embedded system is a device controlled by instructions stored on a chip. These devices are usually controlled by a microprocessor that executes the instructions stored on a Read Only Memory (ROM) chip.

One of the most popular Real Time Operating Systems (RTOS) is in use today is QNX (pronounced `queue nicks'). It is used for everything from medical instrumentation and monitoring nuclear reactors, to traffic

lights and industrial process control. In fact, it is so widely used that we use devices having QNX several times a day without being aware of it.

QNX makes use of a micro kernel as opposed to OSs such as Windows and UNIX, so system level functions such as device drivers are not part of the system. The kernel contains a minimum number of features for implementing basic system calls. These include message passing along with other interprocess communication,

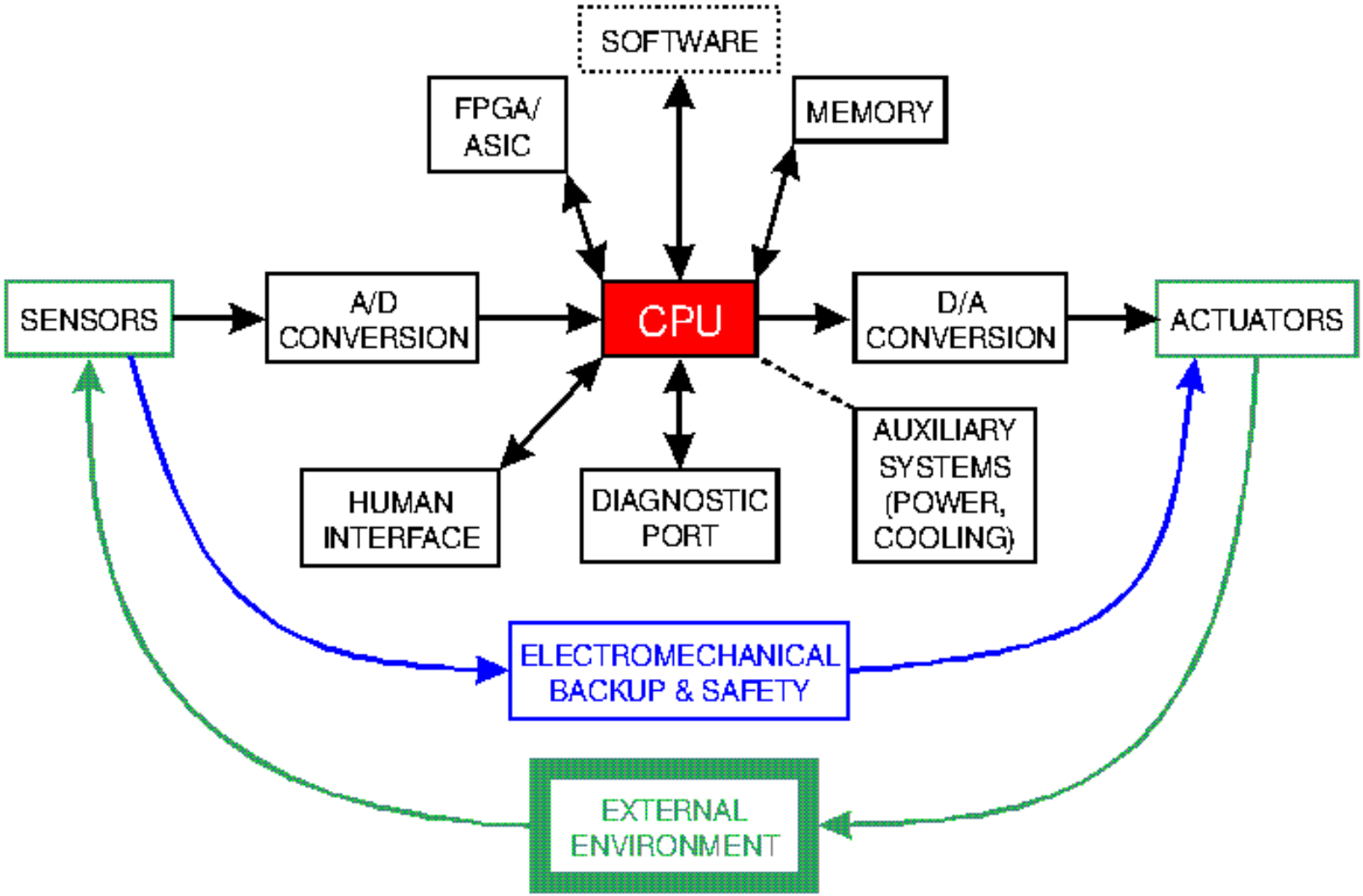
An embedded system has

- a digital signal processor,
- a variety of I/O devices connected to
- sensors and actuators.

Controllers and DSP

- are programmable parts,
  - customizable for different application by writing software.

Typical Embedded System Organization

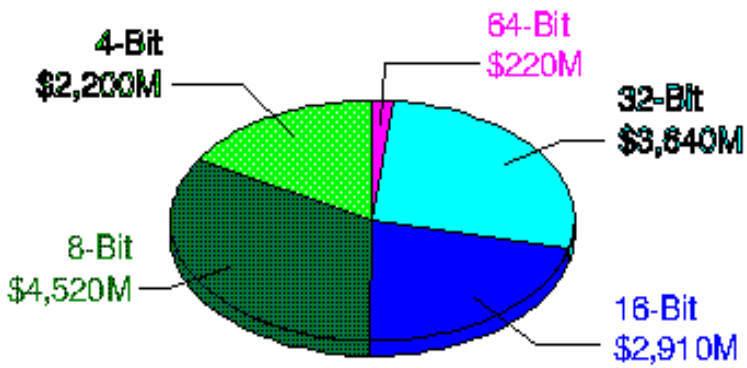


Microcontrollers

- A Microcontoller is a device where CPU and limited associated resources such as memory, I/O are integration on the same single chip. Because of this integrated on a single chip, the reliability of a micro-controller is far superior to an equivalent system designed using CPU, memory, I/O interface chip on a PCB. But it has limitation in terms of program memory, data memory and I/O interfaces. Manufacturers of I/O controllers (Intel, Motorola, Atmel, Scenix, SGS Thomson, Hitachi, Zilog etc).

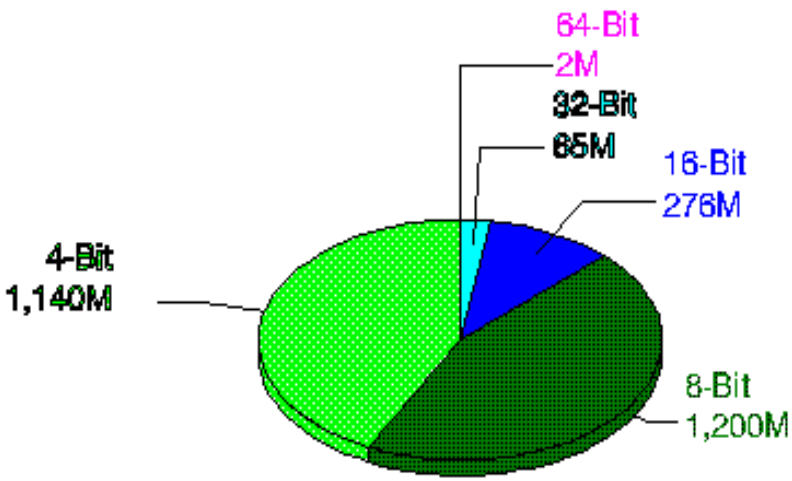
- Provide facilities for expansion of resource requirement for memory. But it is always advisable to use the Microcontroller in its single chip mode of operation rather than in the expanded mode. These Microcontrollers are available in sizes of 4 bit, 8 bit, 16 bit and 32 bit from various manufacturers.

Embedded CPUs Dominate Market



$13,490M Total

1994 Worldwide
Microcontroller Revenue
($Million U.S.)



2,683M Total

1994 Worldwide
Microcontroller Units
(Million Devices)

Approximated from EE Times, March 20, 1995
Source: The Information Architects

Microcontrollers (Embedded Systems)

- CPU + Interface
- CPU + ROM + RAM + Interfaces + Timers + A/D, D/A Converter
- For prototyping ROM is replaced by EPROM. ROM is essentially used for programming code.
- CPU + EPROM + RAM + Interfaces + Timers + A/D, D/A Converters
- CPU + FLASH ROM + RAM + Interfaces + Timers + A/D, D/A Converters

- Examples : 8048, 8748, 8051, 8751, i960, 6801, 68701, 68HC11, etc.
- On chip ROM, RAM (Code and data memory), interfaces are very limited. (Examples - ROM, EPROM, 4 KB, RAM 256 B)
- Reliability + Suitable for specific application without additional external hardware.
- Also provision is available to expand memory outside the chip.

SOME EXAMPLES

- 4 Bit Microcontroller
  - TMS1000 (Texas)
  - NEC uPD7500
  - COPS 400 (National)
  - HD44795 (Hitachi) (LCDIII)
- 8 bit Microcontrollers
  - MC6801/68701
  - 6805
  - 68HC11
  - MCS-51 (8051/8751)(Intel
  - TMS 370 (Texas)

- 16 bit Microcontroller
  - 80186/80188
  - MCS-96 (80C196)
  - Motorola MC 68332 (16/32)
- 32 bit Microcontroller
  - 80960 (Intel)
  - LR 33000 (LSI Logic)
  - AMD 29050
  - NS32000
  - Intel Strong Arm SA1100, SA1110

- 16 bit Microcontroller
  - 80186/80188
  - MCS-96 (80C196)
  - Motorola MC 68332 (16/32)
- 32 bit Microcontroller
  - 80960 (Intel)
  - LR 33000 (LSI Logic)
  - AMD 29050
  - NS32000
  - Intel Strong Arm, SA1110, SA1111, SA1100, SA1101, SA110

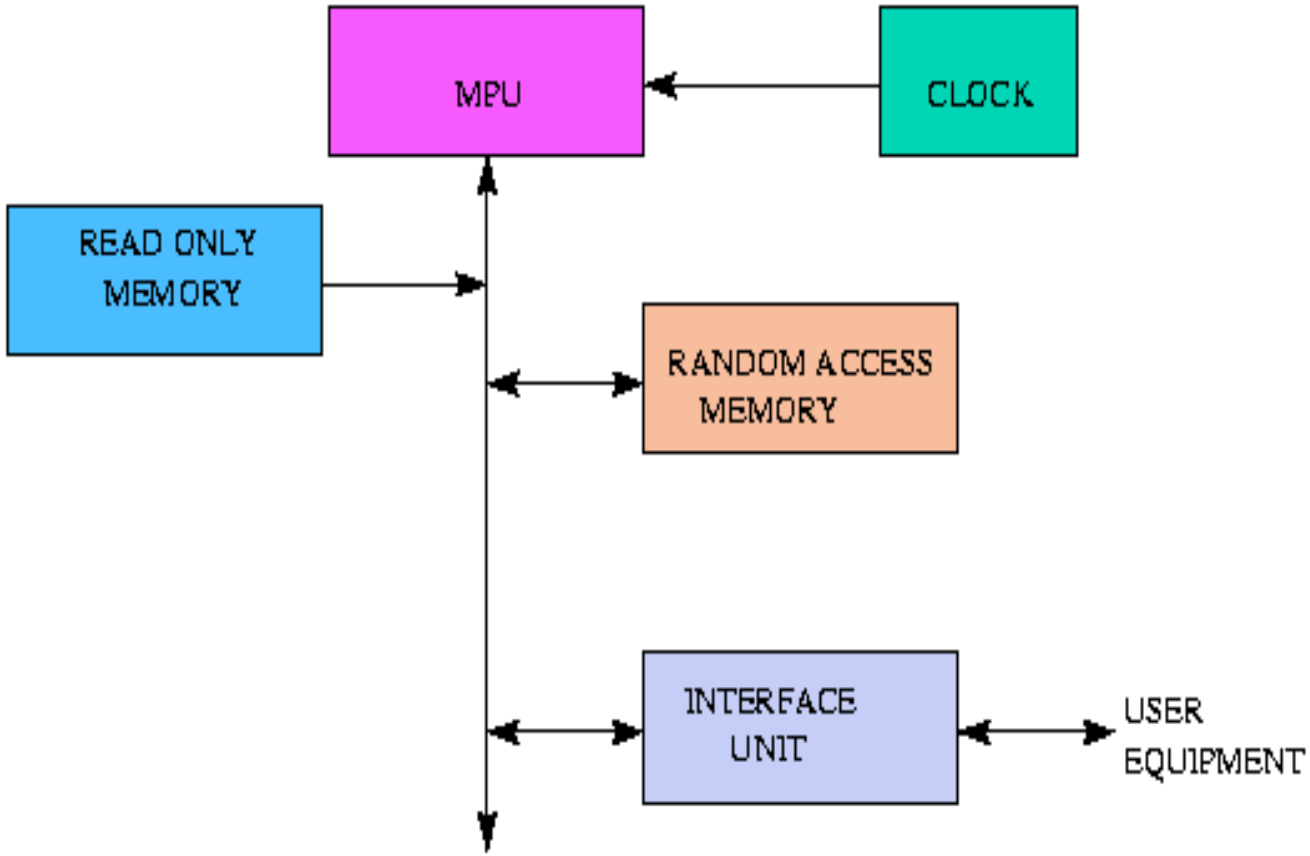
Capabilities of Representative 8-bit Microcontrollers from 2 Families

Chip	RAM	ROM/ EPROM EAROM/ EEPROM	CLO-CK us	I/O Ports	A/D	Tim-ers
M68HC 11A0	256	--	0.476	4x8 1x6	4/8	9
M68HC 11A1	256	512 EEPROM	0.476	4x8 1x6	4/8	9
M68HC 11A2		2048 EEPROM	0.476	4x8 1x6	4/8	9
M68HC 11A8	256	8k ROM, 512 EEPROM	0.476	4x8 1x6	4/8	9
M68HC 11E0	512	--	0.476	4x8 1x6	8	9

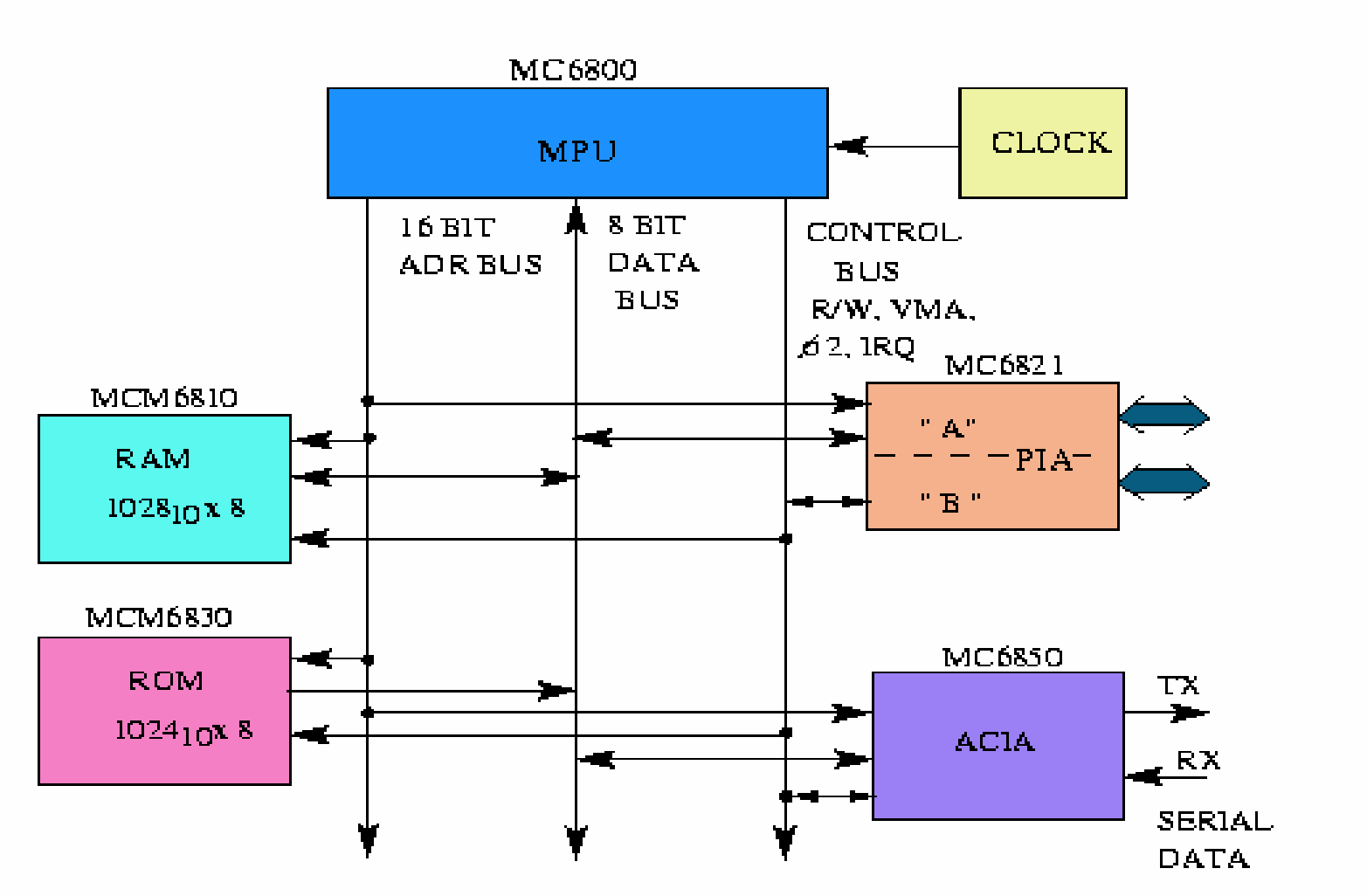
M68HC11E1	512	512 EEPROM	0.476	4x8 1x6	8	9
M68HC11E2	256	2048 (EE)	0.476	4x8 1x6	4/8	9
M68HC11AE9	512	12k (ROM), 512 (EE)	0.476	4x8 1x6	8	9
M68HC11D3	192	4096 (ROM)	0.476	4x8 1x6	8	9
M68HC11F1	1024	512 EEPROM	0.476	4x8 1x6	8	9
Intel 8021	64	1024 (ROM)	2.5	2x8 1x4	-	2
Intel 8022	64	2048 (ROM)	2.5	3x8	-	2
Intel 8035	64	--	2.5	3x8	-	2
Intel 8039	128	--	1.4	3x8	-	2

Intel 8041	64	1024 (ROM)	2.5	3x8	-	2
Intel 8048	64	1024 (ROM)	2.5	3x8	-	2
Intel 8049	64	2048 (ROM)	1.4	3x8	-	2
Intel 8748	64	1024 (ROM)	2.5	3x8	-	2
Intel 8031	128	--	1	4x8	-	2
Intel 8051	128	4096 (ROM)	1	4x8	-	2
Intel 8751	128	4096 (EEROM)	1	4x8	-	2

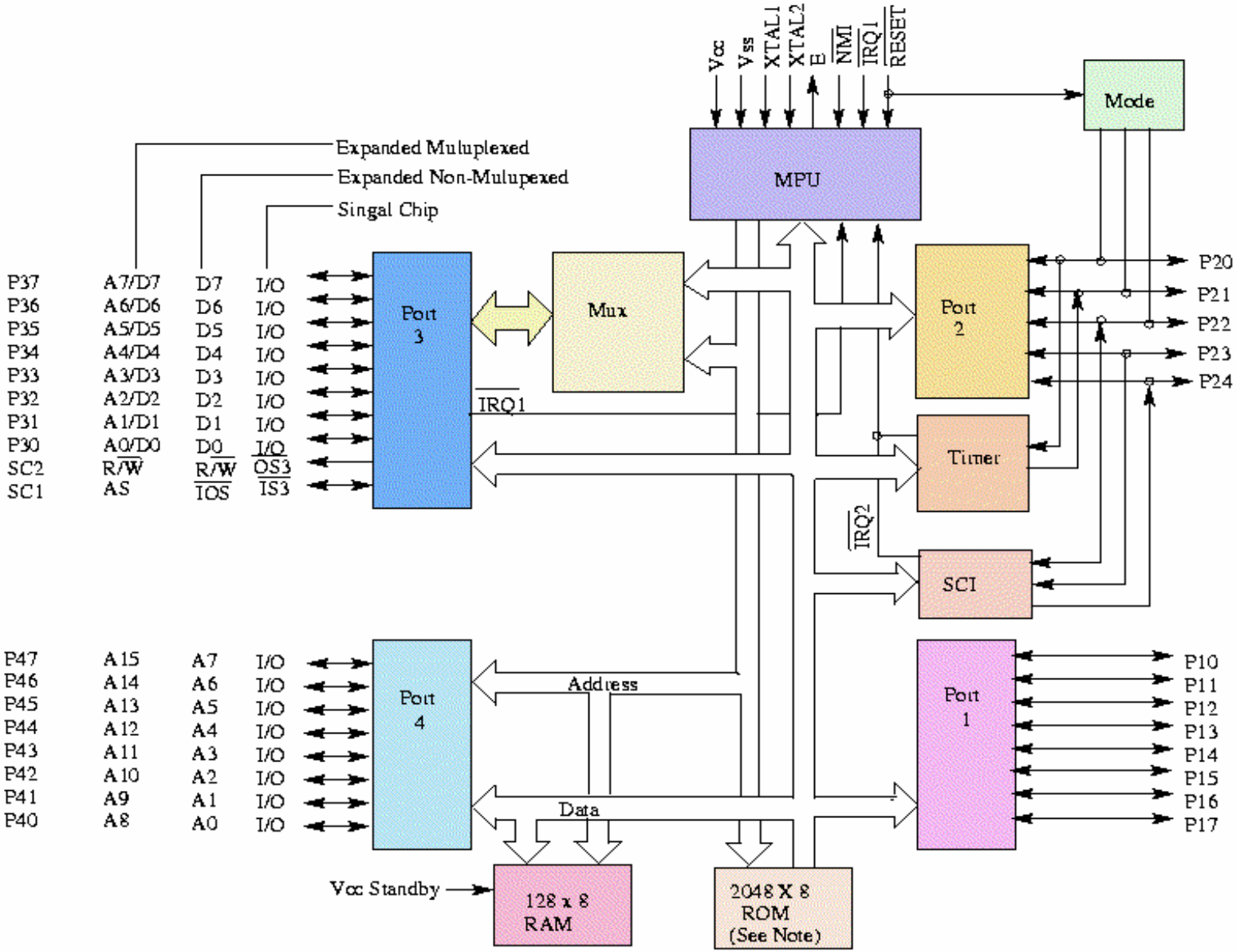
Microcomputer Based Systems



Microcomputer Based Systems



M6801 Microcomputer Family Block Diagram



Embedded System Design

Design Challenges

Embedded Systems are quite complex:

1 Embedded system have different constraints then general

- Cost may matter more than speed
- Long life cycle may dominate design decisions
- Reliability/safety may constrain design choices

2 Correct functioning is crucial

- safety-critical applications
- damage to life, economy can result

3 Real-time systems

- not only right output but at right time
- imagine a delay of few minutes in pacemaket system

4 They are concurrent systems

- System and environment run concurrently
- multi-functional

5 They are Reactive Systems*

- Once started run forever
- Termination is a bad behaviour
- Compare conventional computing (transformational systems)

* Reactive Systems are systems that have continuous interaction with their environment. Hardware and OS are typical examples. In general, all embedded systems are reactive.

These systems can be contrasted with conventional software systems like data processing applications. The latter are called transformational systems. These systems take input from the environment, transform these inputs and terminate giving the outputs.

Their interaction with the environment is limited:

once at the time of taking inputs and once at the end when the outputs are generated.

6 Stringent resource constraints

- compact systems
  - Simple processors
  - limited memory
- quick response
- good throughput
- low power

Embedded Computer Design Required

- Real time/Reactive operation
- Small size/low weight
- Low power, limited cooling
- Safe and reliable
- Moderate to extreme cost sensitivity

Real Time/Reactive Operation

- Real time: correctness of result is a function of time it is delivered
  - Not necessarily "real fast" -- consistency may be more important than raw speed
  - Worst case performance often limits design
- Reactive: computation rate is in response to external events
  - Periodic events can be scheduled statically
  - Aperiodic events must be statistically predicted, and (to avoid overdesign) dynamically scheduled when possible

Embedded System Design

- involves HW and SW Design
  - Software for flexibility
  - Hardware for speed
- Co-design of Software and Hardware
- nonstandard HW - ASIC
- System Partitioning - difficult step
  - Choice of programmable parts and ASICs

- Communication between ASIC and SW
- Timing is crucial
- Common languages for HW and SW
  - Hardware C, SpecC, SpecCharts, Statecharts