AGENDA

- Embedded Systems
- µP vs µC
- Processor Architecture
- Memories
- I / O
- BUSES
- LAB

vivekn@cdac.in

Why Learn ES

– Lot of hype

–Everybody else is doing it

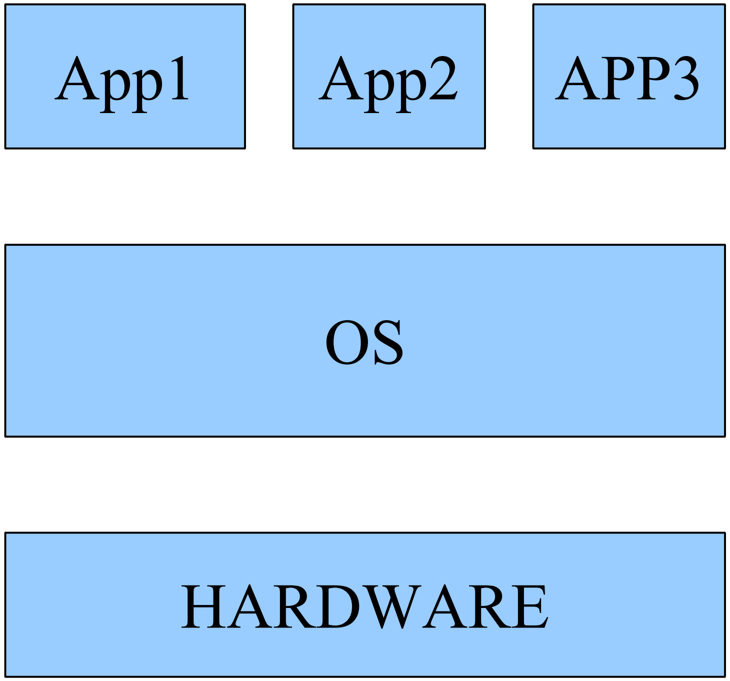
–My friends got a job in ES,hence....

–Desktop would die soon

–My brother in US told me to learn it ....

Embedded Systems Defined ?

- No formal definition
- Generally accepted to be a type of computer designed to solve a specific problem or task
- A combination of hardware and software, and perhaps additional mechanical or other parts, designed to perform a dedicated function.
- In some cases, embedded systems are part of a larger system or product

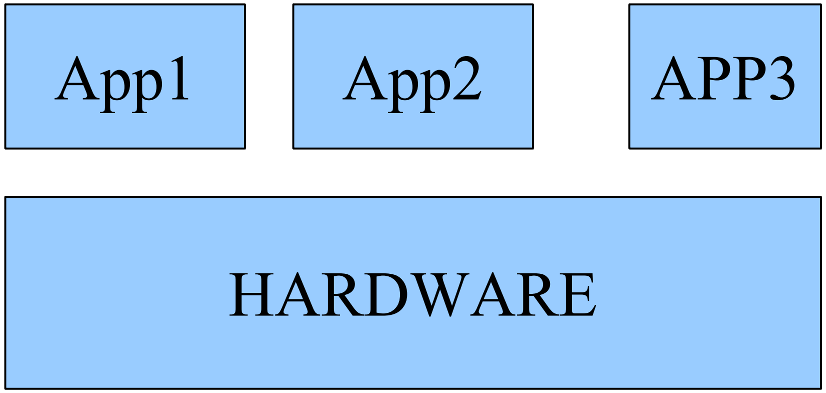


App1 App2 APP3

OS

HARDWARE

General purpose computer System



App1 App2 APP3

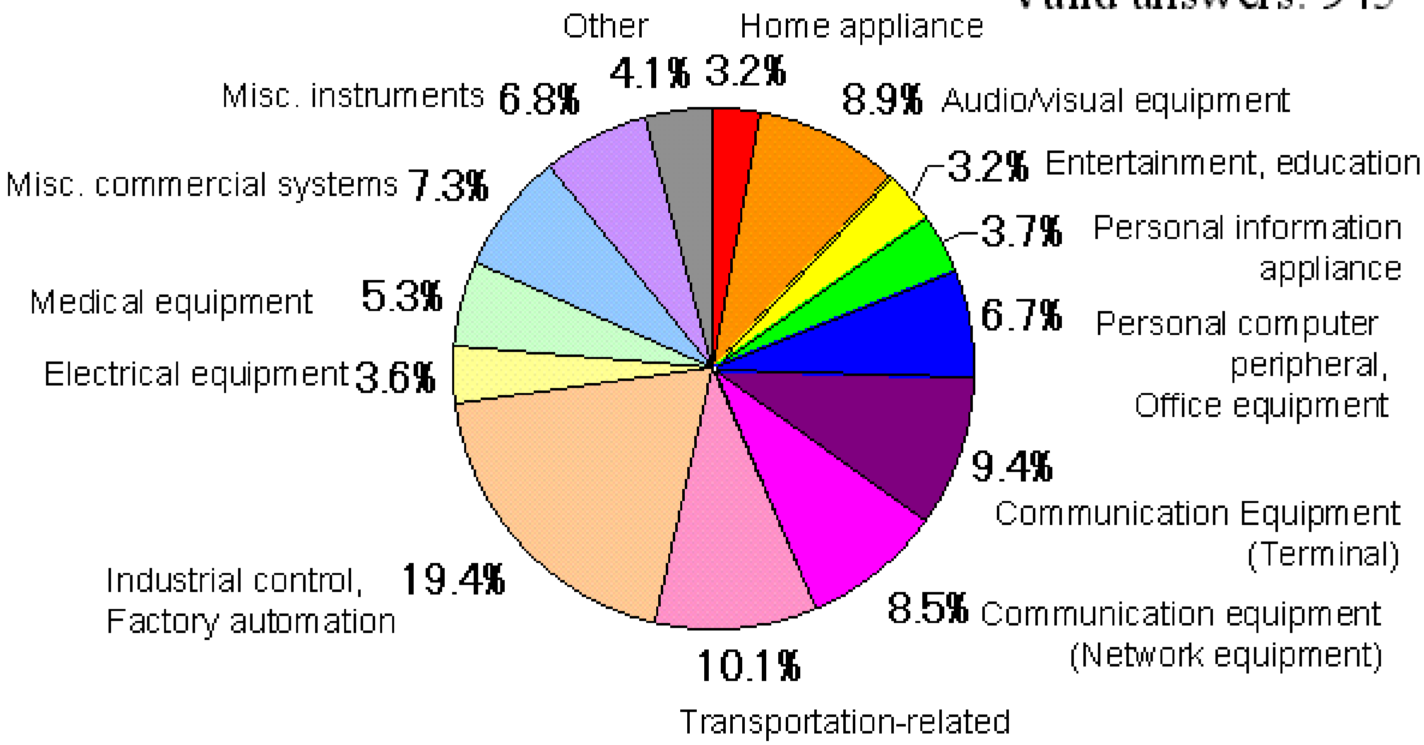
HARDWARE

Typical embedded system

Embedded Market

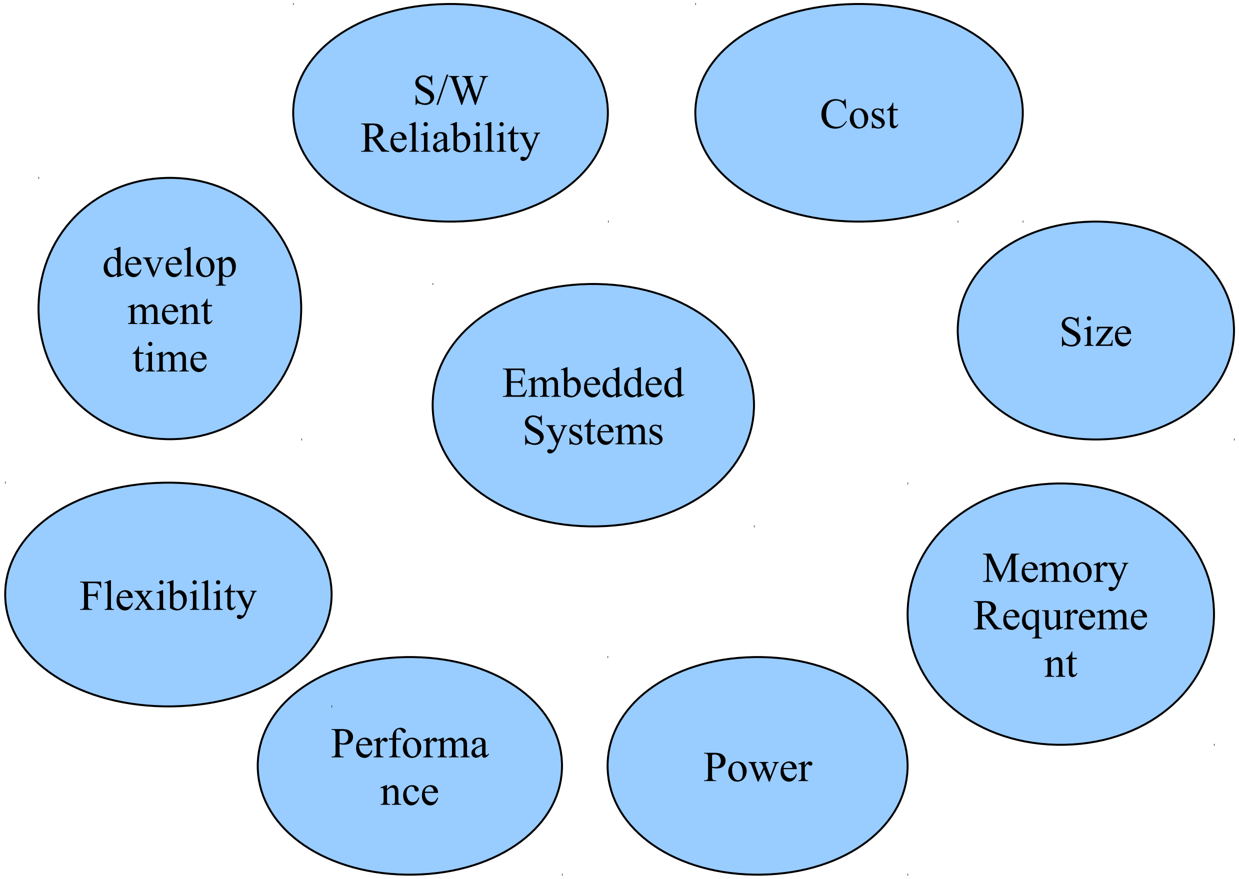
Application Fields of Recently Developed Embedded Systems

Valid answers: 945



ITRON Technical Committee

Design Consideration



General Framework

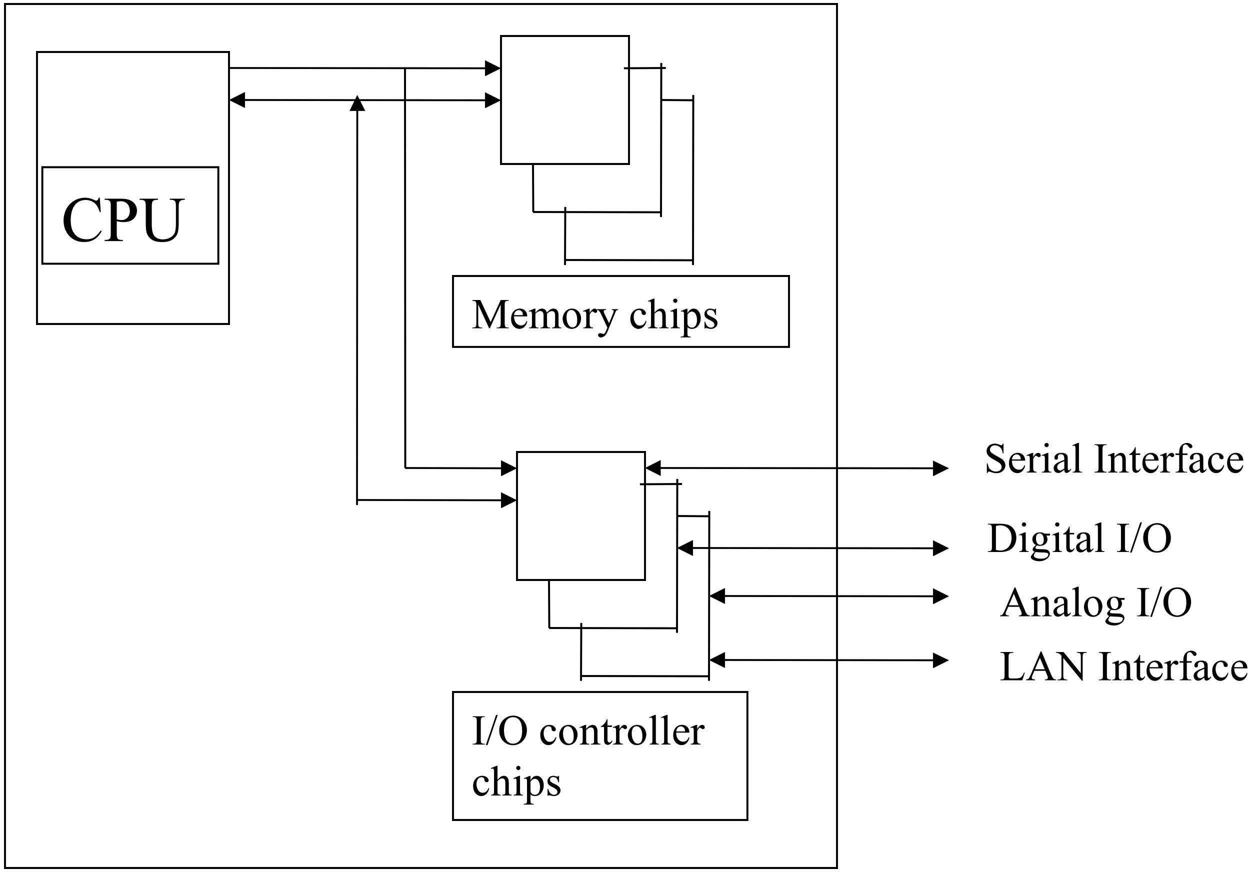
- Embedded systems typically use a processor combined with other hardware and software to solve a specific computing problem.

- The processors range from simple (by today’s standards) 8-bit microcontrollers to the worlds fastest and most sophisticated 64-bit microprocessors.

- Embedded system software ranges from a small executive to a large real-time operating system (RTOS) with a graphical user interface (GUI).

- Embedded systems range from large computers such as an air traffic control system to small computers.

Embedded System looks like . .



CPUs

- CPUs can be
  - Microprocessors or Micro controllers

- Any CPU can be studied by knowing following features of CPU
  - Clock speed, Address bus size, Data bus size, Register size
  - Register set, Instruction set, Address spaces, Endian type
  - Interrupt support, DMA support
  - Instruction and Data cache
  - Memory management
  - Protection features (user/supervisor modes)

Micro Processors

- Micro Processor is the Integration of a number of useful function in a single IC Package

- These functions are
  - Execute the stored set of instructions to carry out user defined task
  - Ability to access external memory chips for both read and write data from and to the memory

Microprocessor Vs Microcontroller

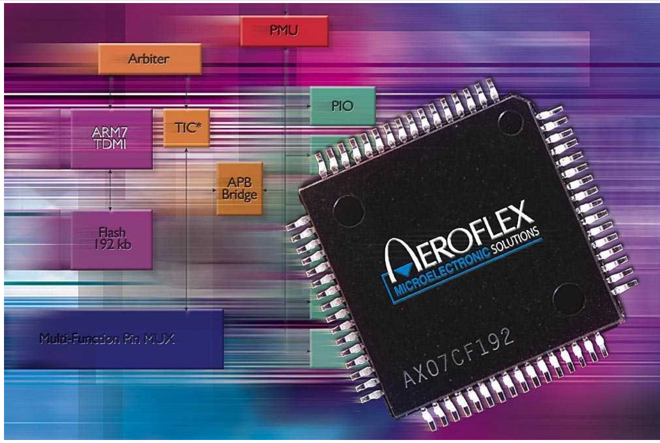
- Contains ALU, GP Registers, SP, PC, Clock timing circuit and interrupts

- Requires more H/W, increase in PCB size

- In addition in built ROM, RAM, IO devices, Timers

- Requires less H/W, reduces PCB size & increases reliability

Microcontroller Defined



- Very similar to a Microprocessor
- It typically includes a CPU, memory, and other peripherals.

Classification of Microcontrollers

- µc are classified into :
  - 8 bit µc e.g.: AVR 8515, Intel 8051, Motorola HC05
  - 16 bit µc e.g.: Siemens 80167, Intel 80C196
  - 32 bit µc e.g.:MCF5272, Power PC 8xxx
  - 64 bit µc e.g.: Texas 64xxx series

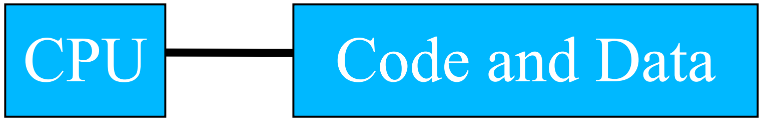
- The number of bits indicate the internal data bus of a µc. It shows how many bits of data the µc can process simultaneously.

Processor Architecture

Princeton

- Single Main Memory holding both program and Data

- Simple memory structure



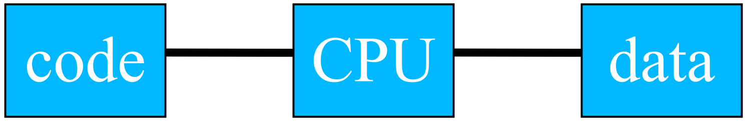
CPU

Code and Data

Harvard

- Contains 2 separate memory spaces- code & data

- Complex memory Structure



code

CPU

data

RISC vs CISC – Architecture

RISC	CISC
Fixed width instructions	Variable length instruction
Few formats of instructions	Several formats of instructions
Load/Store Architecture	Memory values can be used as operands in instructions
Large Register bank	Small Register Bank
Instructions are pipelinable	Cannot pipeline instructions

RISC vs CISC - Organization

RISC	CISC
Hardwired instruction decode	Microcode ROMS instruction decoder
Single cycle execution of instruction	Multi cycle execution on instruction

RISC vs CISC - Organization

RISC	CISC
Hardwired instruction decode	Microcode ROMS instruction decoder
Single cycle execution of instruction	Multi cycle execution on instruction

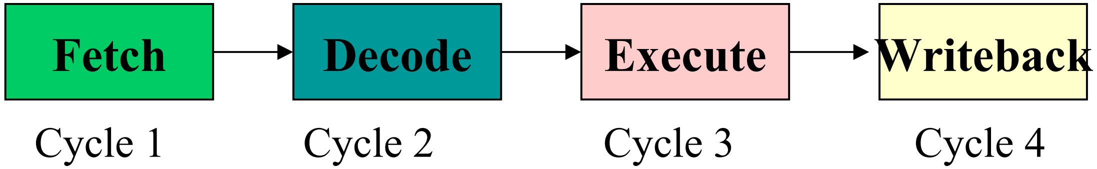
RISC CISC

Feature	Low Power RISC	PC/Desktop CISC
Power	A few hundreds of milliwatts	Many watts
Compute Speed	200-520 Mega Hz	2-5 Giga Hz
Memory Management		
Cost	Dollars	Tens to hundreds of Dollars
Environmental	High Temp, Low EM Emissions	Needs Fans, FCC/CE approval an issue

Life cycle of an instruction

- As the instruction moves through the processor it goes through the following stages.
  - Fetch : Instruction fetched from the address stored in the program counter.
  - Decode : Instruction decoded and registers read
  - Execute : in the ALU
  - Write back : Results written back to registers or memory.

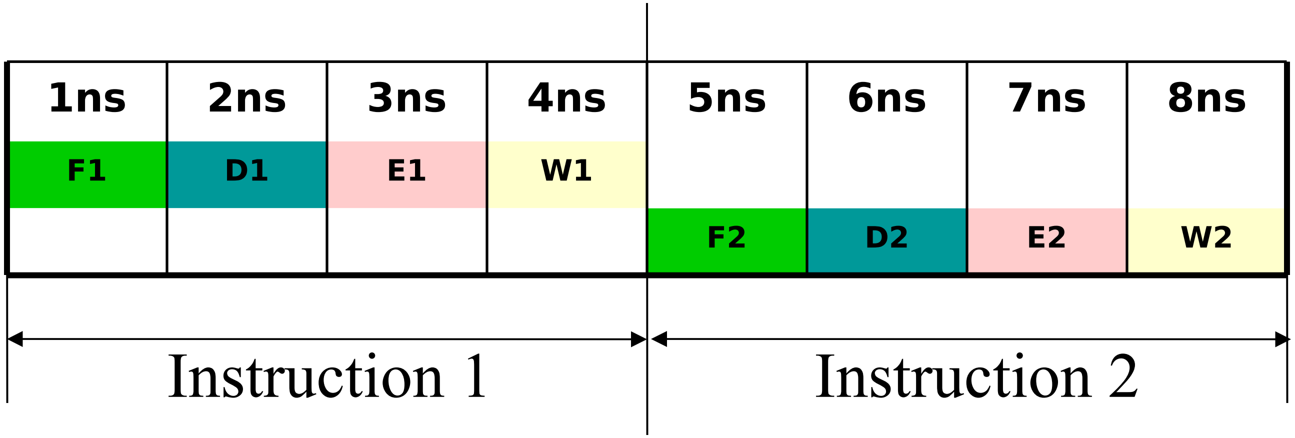
- Note : each stage takes one clock cycle
- The Instruction execution requires 4 clock cycles



Fetch → **Decode** → **Execute** → **Writeback**

Cycle 1 Cycle 2 Cycle 3 Cycle 4

Non Pipelined Example

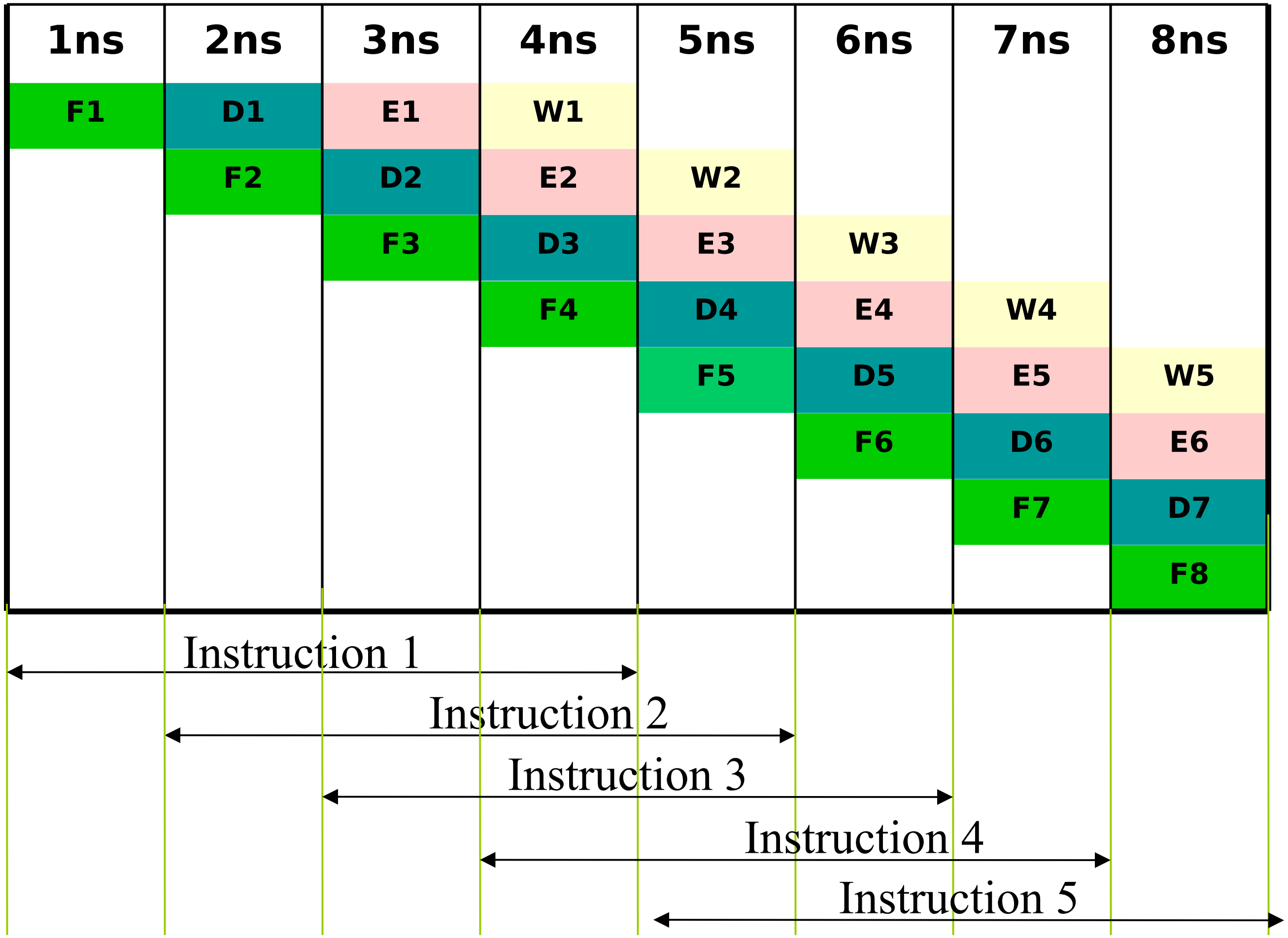


1ns	2ns	3ns	4ns	5ns	6ns	7ns	8ns
F1	D1	E1	W1				
				F2	D2	E2	W2

- 2 instructions take 4ns each
- A stage is used once every 4ns
- The resultant throughput is 4 cycles per instruction (CPI)

Pipeline Example

- Each stage is utilized at every clock cycle.
- 5 Instructions are executed in 8ns
- Resultant throughput is 1 instruction per cycle



1ns	2ns	3ns	4ns	5ns	6ns	7ns	8ns
F1	D1	E1	W1				
	F2	D2	E2	W2			
		F3	D3	E3	W3		
			F4	D4	E4	W4	
				F5	D5	E5	W5
					F6	D6	E6
						F7	D7
							F8

Big Endian

- Stores the most significant part first
- e.g.: Motorola

0x5050	45
0x5051	67

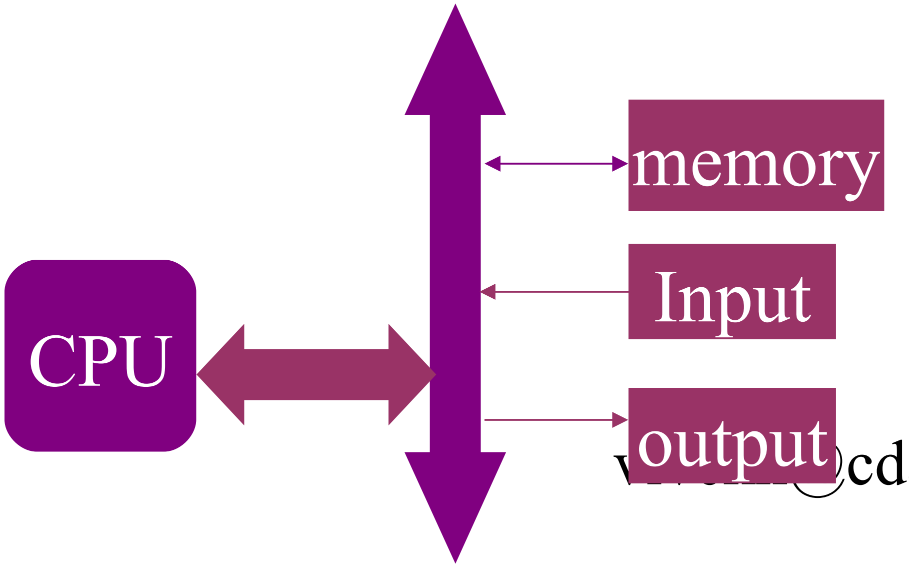
Little Endian

- Stores the least significant part first
- e.g.: Intel
- e.g.: consider data 0x4567 at location 0x5050

0x5050	67
0x5051	45

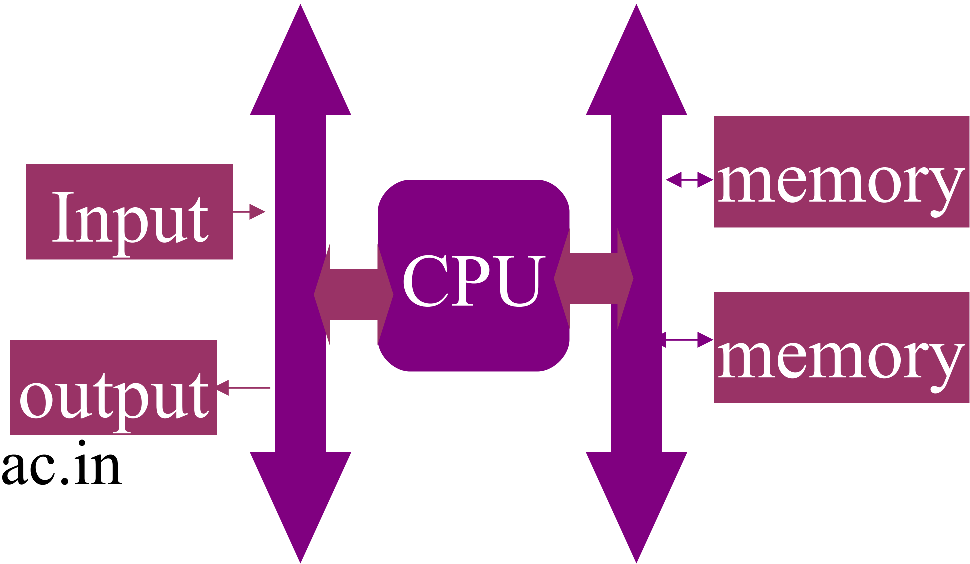
Memory Mapped IO

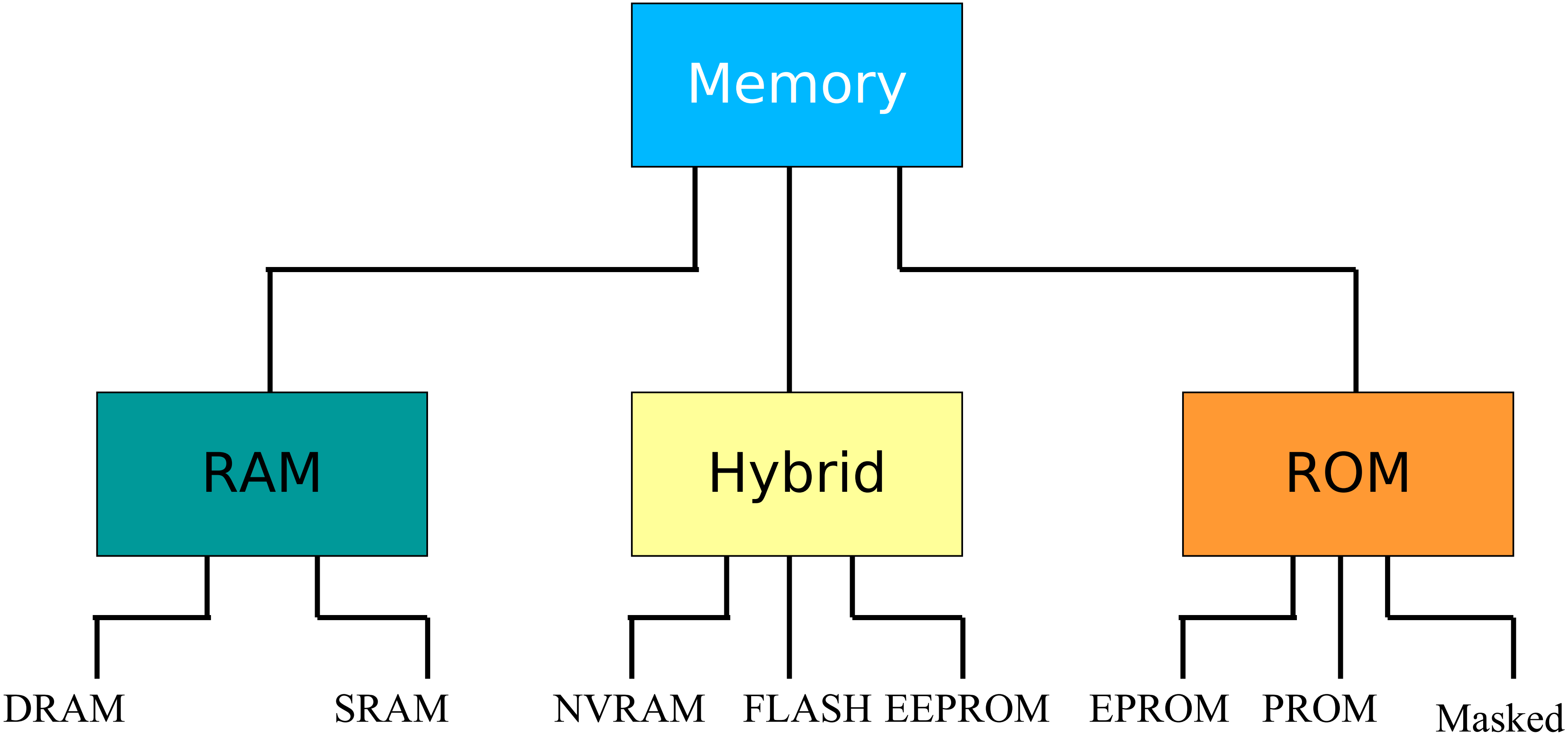
- IO devices are treated as like memory
- Memory related instructions should be used to access IO



IO Mapped IO

- IO devices are separately interfaced
- Separate instruction set available




Memory
RAM
Hybrid
ROM
DRAM
SRAM
NVRAM
FLASH
EEPROM
EPROM
PROM
Masked

Types of RAM

- Static RAM (SRAM) and Dynamic RAM (DRAM)
- **SRAM** retains its contents as long as electrical power is applied to the chip. If the power is turned off or lost temporarily, its contents will be lost forever.
- **DRAM** has an extremely short data lifetime - typically about four milliseconds. This is true even when power is applied constantly.
- When deciding which type of RAM to use, a system designer must consider *access time* and *cost*.
- Many embedded systems include both types.

Types of ROM

- **Masked ROM** - hardwired devices that contained a preprogrammed set of data or instructions. The primary advantage of a masked ROM is its low production cost.

- **PROM** (programmable ROM) - Known as one-time programmable (OTP) devices. The device programmer writes data to the device one word at a time by applying an electrical charge to the input pins of the chip.

- **EPROM** (erasable-and-programmable ROM) – Similar to *PROM*. However, EPROMs can be erased and reprogrammed. To erase an EPROM, simply expose the device to a strong source of ultraviolet light. An essential part of the software development and testing process.

Hybrid Memories

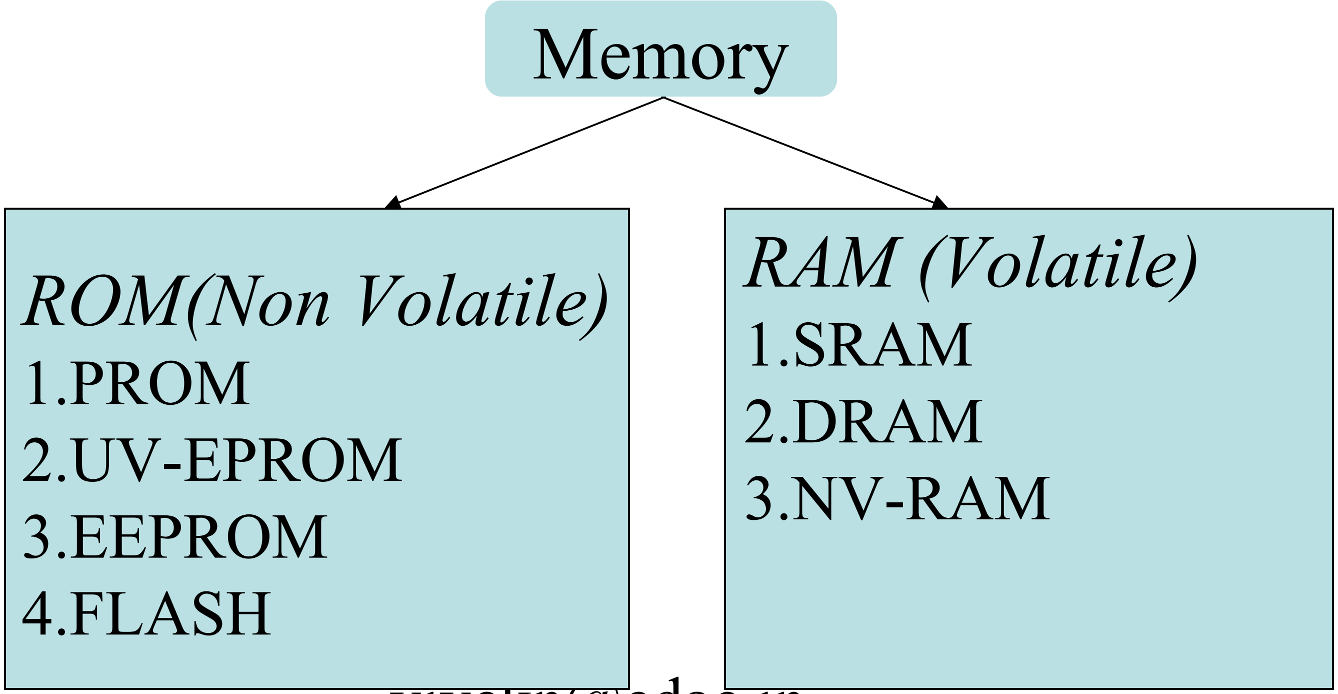
- **EEPROM**(electrically-erasable-and-programmable) - the erase operation is accomplished electrically. Any byte within an EEPROM may be erased and rewritten.
- **Flash memory** devices are
  - High density, low cost, nonvolatile, fast (to read, but not to write), and electrically reprogrammable.
  - These advantages are overwhelming and, as a direct result, the use of flash memory has increased dramatically in **embedded systems**.
  - Flash devices can only be erased one sector at a time, not byte-by-byte.

Hybrid Memories

- **NVRAM** (non-volatile RAM) - An NVRAM is usually just an SRAM with a battery backup.
  - An NVRAM is usually just an SRAM with a battery backup.
  - When the power is turned on, the NVRAM operates just like any other SRAM.
  - When the power is turned off, the NVRAM draws just enough power from the battery to retain its data.
  - NVRAM is fairly common in embedded systems, typically limited to the storage of a few hundred bytes of system-critical information

General components of a Microcontroller

- CPU – Heart of the controller. Composed of registers, ALU, instruction decoder and the control circuitry.
- On Chip Memory – Used to store information



Memory

ROM(Non Volatile)
1.PROM
2.UV-EPROM
3.EEPROM
4.FLASH

RAM (Volatile)
1.SRAM
2.DRAM
3.NV-RAM

General components of a Microcontroller

- I/O Ports
  - Used to interface with the peripheral devices and the controller.
  - Analog I/O and Digital I/O

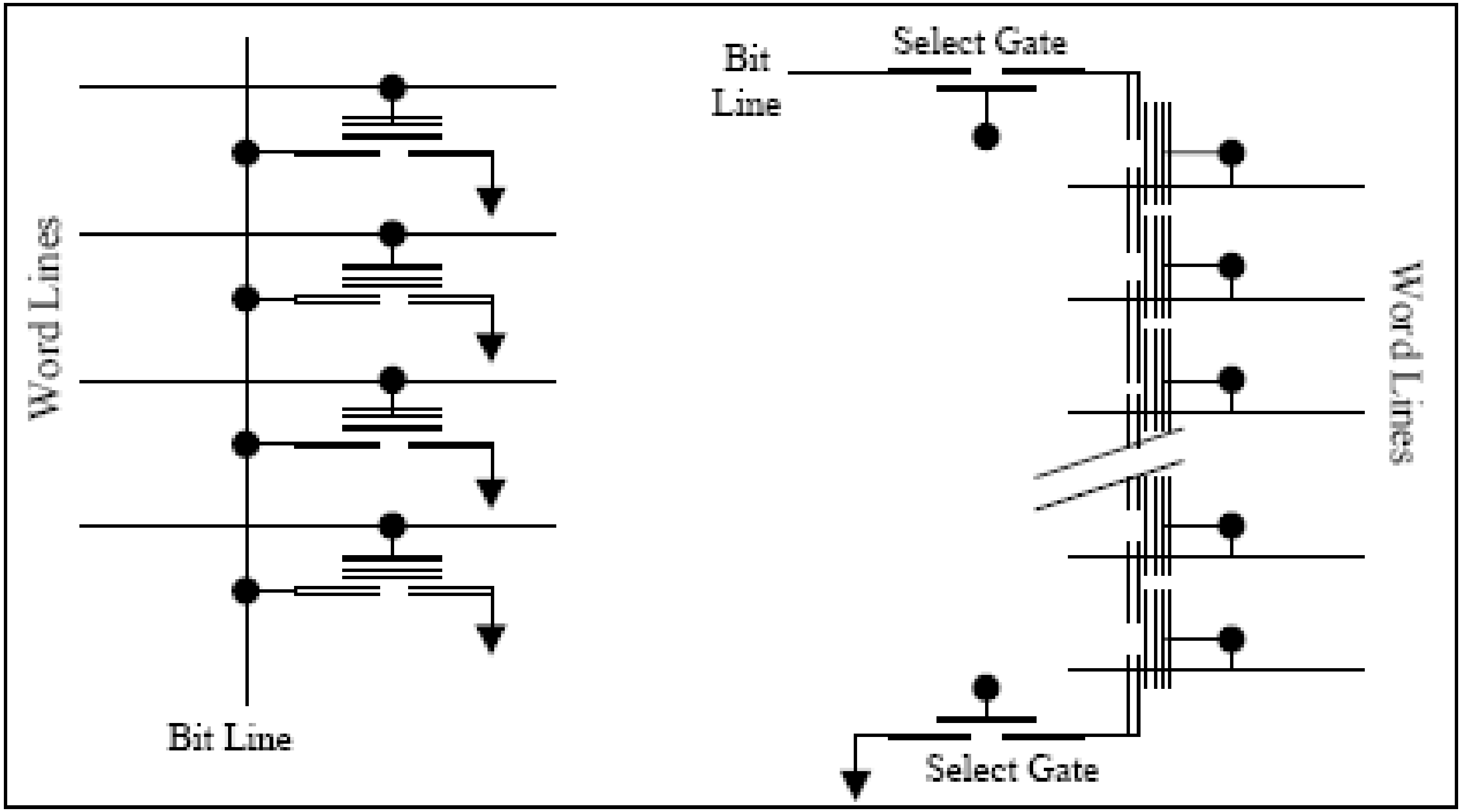
- Timer/Counter
  - For keeping Time and/or calculating the amount of time between events
  - For counting the events
  - Baud rate generation

FLASH MEMORY

- Flash is believed to be close to perfect memory
- Advantages
  - Fast read speeds
  - Long-term data retention at much lower cost
- Drawbacks
  - Doesn’t allow random bytes to be updated on the fly
  - Limitations on number of times it can be re-written

Types of Flash Memory

NOR NAND

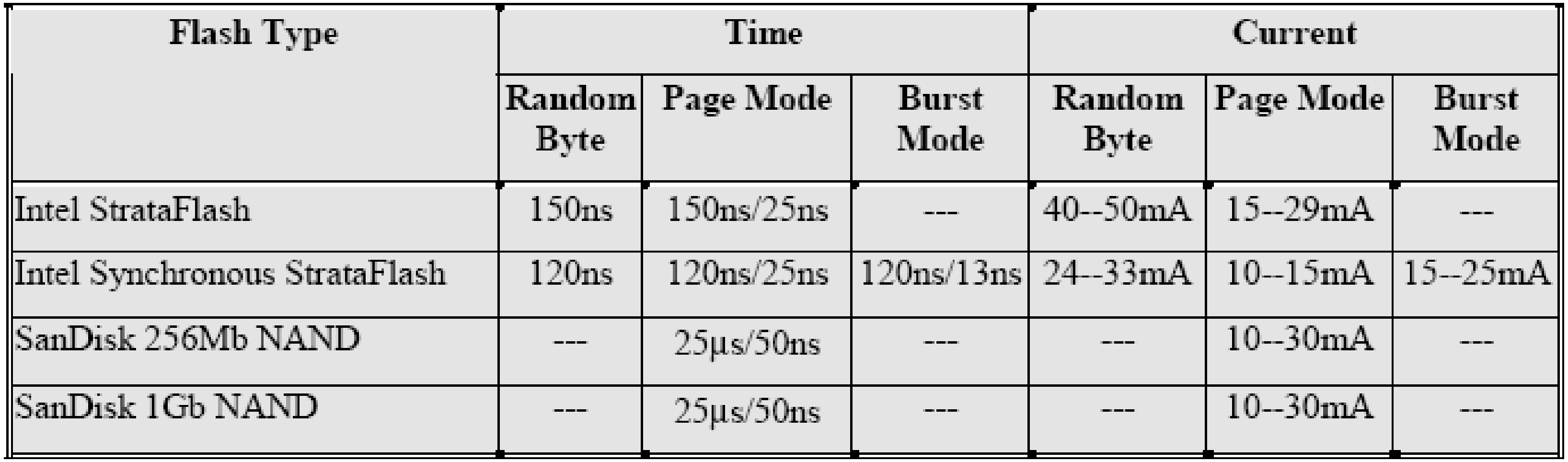


NOR vs NAND

- The parallel interconnection of the memory cells helps account for their fast random read accessibility.

- The arrangement of these cells is significantly more compact than in NOR.
- The select gates allow higher programming and erase voltages to be used without disturbing the charge stored in unselected cells.
- fast programming and erase times
- Poor random read performance

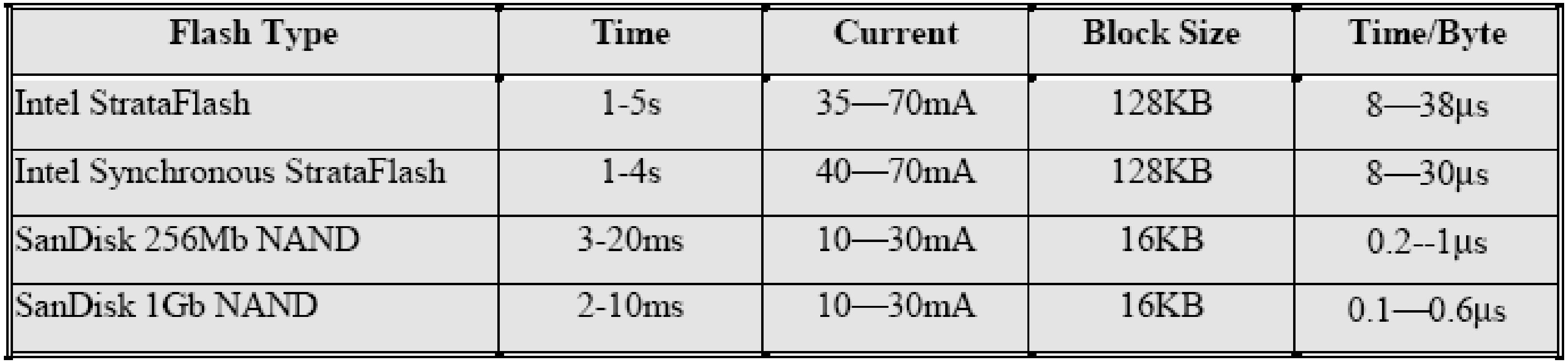
Flash memory read characteristics



Flash Type	Time			Current		
	Random Byte	Page Mode	Burst Mode	Random Byte	Page Mode	Burst Mode
Intel StrataFlash	150ns	150ns/25ns	---	40--50mA	15--29mA	---
Intel Synchronous StrataFlash	120ns	120ns/25ns	120ns/13ns	24--33mA	10--15mA	15--25mA
SanDisk 256Mb NAND	---	25μs/50ns	---	---	10--30mA	---
SanDisk 1Gb NAND	---	25μs/50ns	---	---	10--30mA	---

- Relatively long initial read delays
- More severe in NAND due to read of redundant region

Flash memory write characteristics



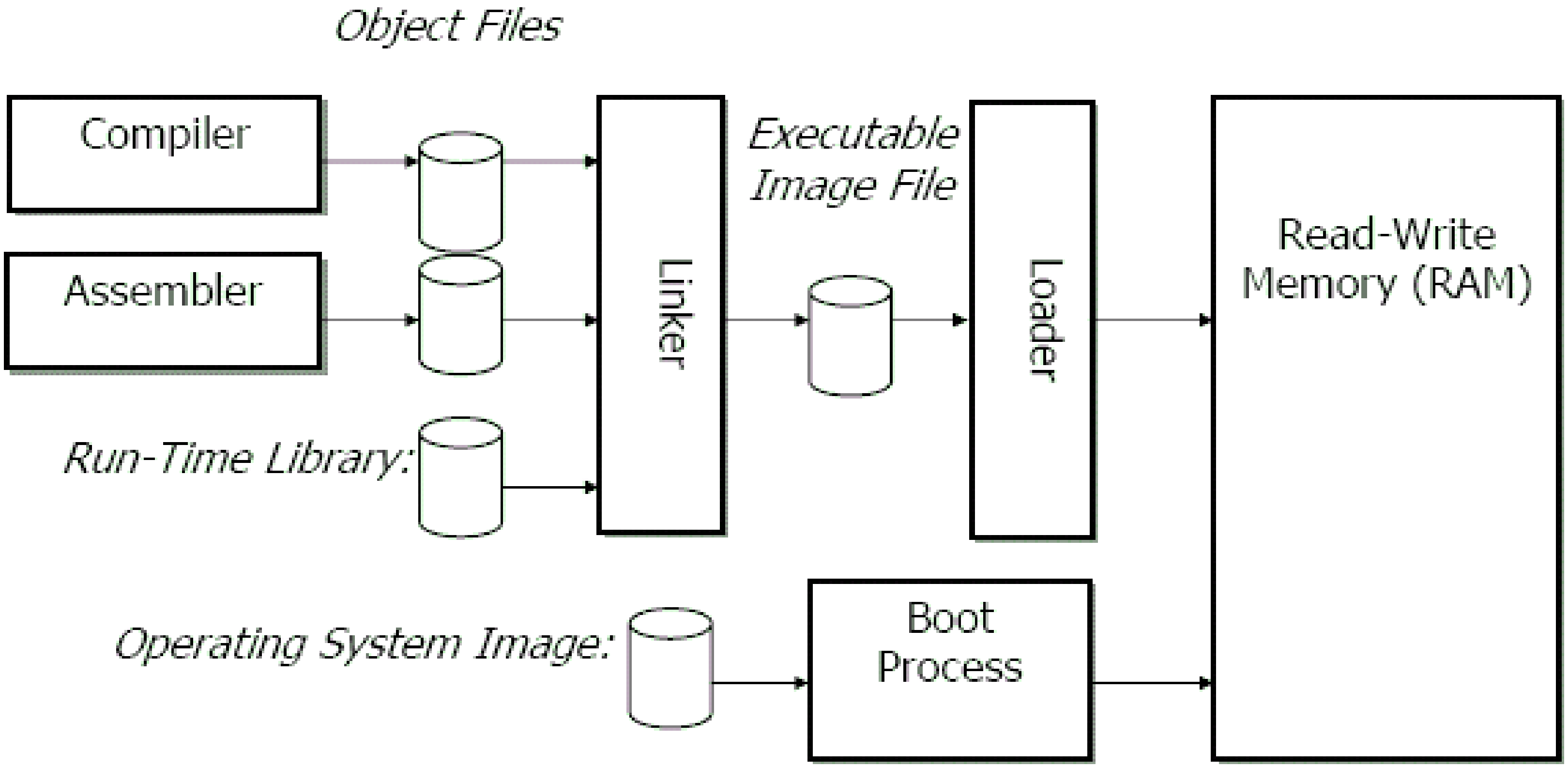
Flash Type	Time	Current	Block Size	Time/Byte
Intel StrataFlash	1-5s	35—70mA	128KB	8—38µs
Intel Synchronous StrataFlash	1-4s	40—70mA	128KB	8—30µs
SanDisk 256Mb NAND	3-20ms	10—30mA	16KB	0.2--1µs
SanDisk 1Gb NAND	2-10ms	10—30mA	16KB	0.1—0.6µs

- Fast write time for NAND memory

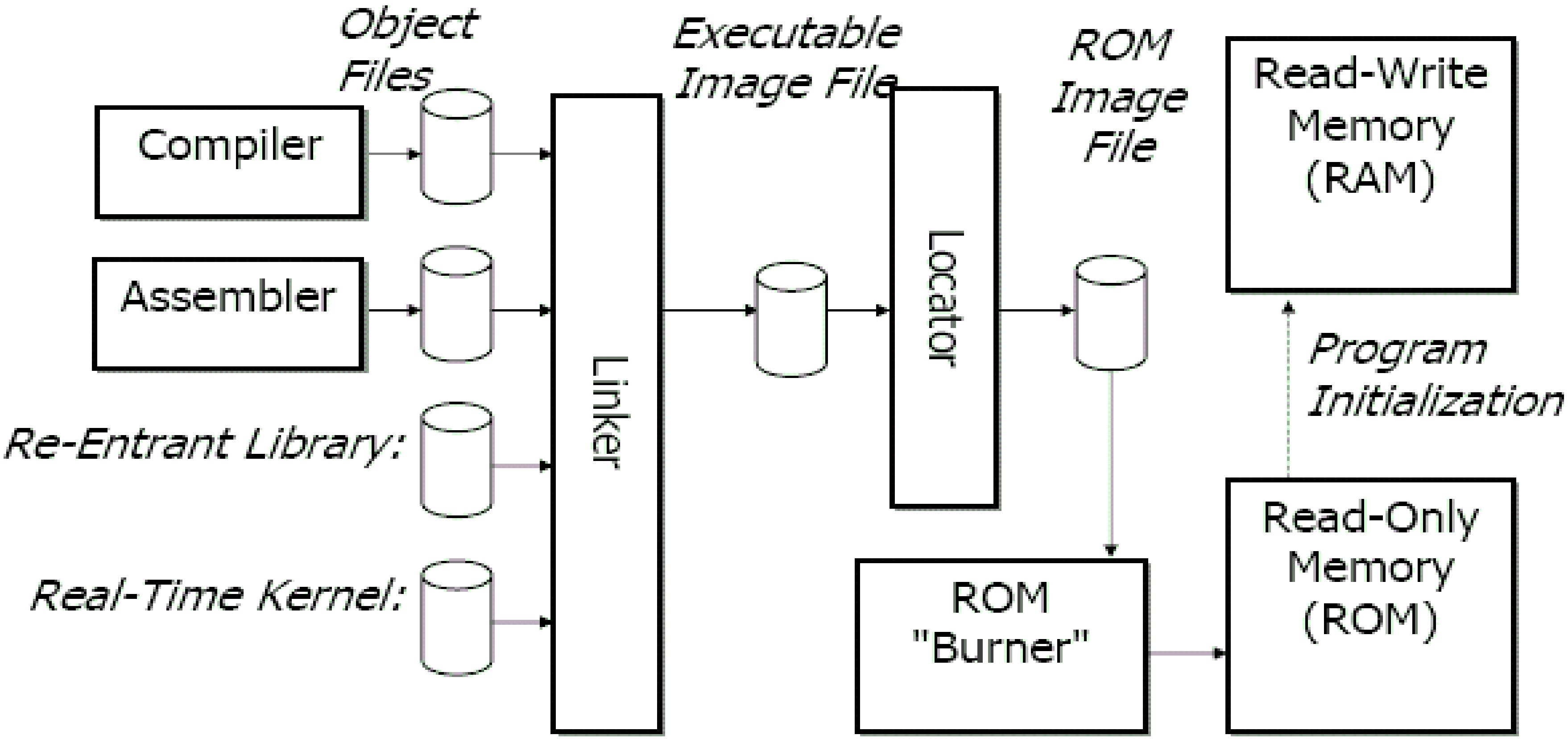
The winner is . . .

- NOR provides faster reads and allows random byte access
- NAND writes and erases data faster, costs less per Mb and uses less power than NOR.
- If the ROM (here flash) is rarely updated, then NOR is obvious choice
- If data is being logged to a flash disk then NAND is the right memory

The build and load process for desktop application programs.



The build and load process for embedded application programs.



Bus Mechanisms

- UART/ USART
- SPI – 4 Mbps
- I^{2}C – 1 to 2 Mbps
- CAN – few Mbps
- PCI – 32 / 64 – 133 Mhz
- USB
- Firewire
- Ethernet
- Parallel port (IEEE 1284)

How to choose a bus

- Bandwidth
- Presence of I/O controllers

DEVELOPMENT ENVIRONMENT

vivekn@cdac.in

Host - Target Development Environment

- The distinguishing feature of embedded software development is host-target development environment
- All the development tools like Editors, compilers and linkers are available on the host machine
- Typical host machines are Windows 95/98, NT and Unix workstations where the above development tools are available
- Application programs will be written on the host, get compiled, linked and get the executable file
- Target systems are the ones where compiled and linked code is executed
- Target systems being a microprocessor based boards does not offer any development environment themselves, so an host machine is required

Cross Compilers

- Another distinguishing feature of embedded software development is cross compilers

- Cross compilers are the ones, which runs on a machine based on one type of CPU and produces a machine instructions for a different kind of CPU

Downloading

- Downloading is the process of loading the executable image prepared on the host system on to a target board
- There are various methods to download the code to a target machine. They are:
  - Serial ports
  - EPROM/FLASH
  - Floppy disks
  - Ethernet
  - Across a common Bus

Debug Monitor

- Debug monitor is a software that resides in a ROM or EPROM of a target board during the development process
- When a target board is powered on, the debug monitor program runs and provides facilities for downloading and debugging the application program
- The debug features provided by the debug monitor are very low level ones
- Once application program is debugged and ready, then the debug monitor could be replaced by the application program, so that whenever system is powered on, application program runs directly
- In some cases both debug monitor and application program will reside in the ROM. First control goes to debug monitor which will pass control to the application program

ASSEMBLY

What’s Right With Assembly Language?

- Speed. ALP’s are generally the fastest programs around.
- Space. ALP’s are often the smallest.
- Knowledge. Your knowledge of assembly language will help you write better programs, even when using HLLs.

What’s Wrong With Assembly Language?

- Assembly is hard to learn & write.
- Assembly language programming is time consuming.
- Improved compiler technology has eliminated the need for assembly language.
- Assembly language is not portable.

LAB

- avr-gcc
- avr-as
- avr-ld

- EXPLORE . . .. .