Embedded Processors

01266212
CYBER PHYSICAL SYSTEM DESIGN

SEMESTER 1-2021

Original contents from
Edward A. Lee and Prabal Dutta, UC Berkeley, EECS 149/249A

Objective

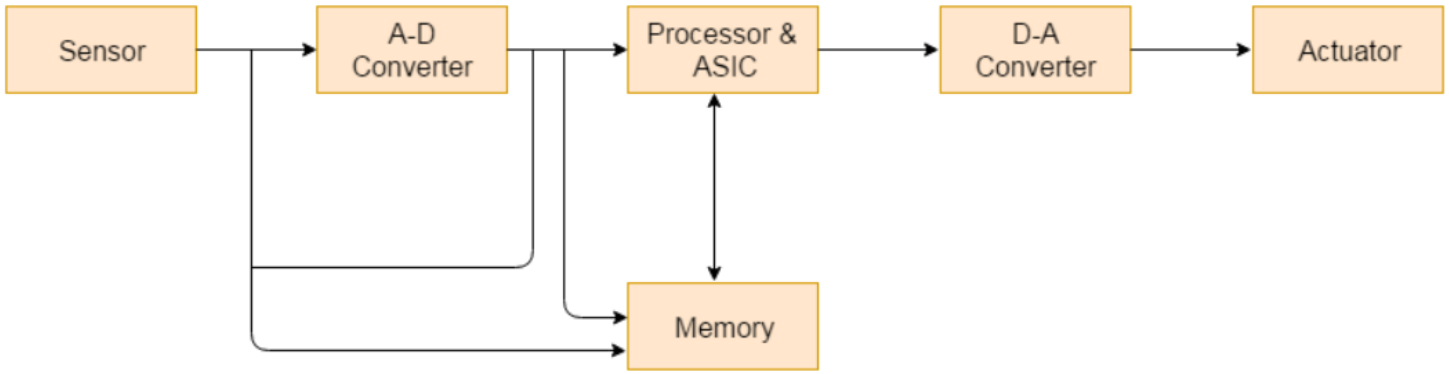
- To understand about Embedded System and Processors.
- To understand the options and to critically evaluate the properties of processors.

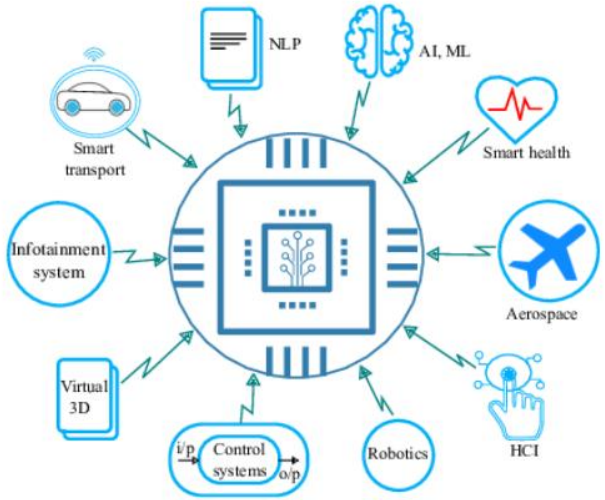
Topics

- Embedded Systems
- Computer Components
- Embedded Processors
- Parallelism

Embedded Systems

- A computer system with software that embedded into a larger product,
- Designed to perform dedicated function, either as an independent system or as a part of a large system.
- a **microcontroller** or **microprocessor**-based system.





Source: https://www.javatpoint.com/designing-of-an-embedded-system

Embedded Software

- A program to control devices that runs on an embedded computer and has time and memory constraints.
- Also known as firmware.
- Stored in ROM/Flash memory.

```
#include<reg51.h>                    /*prepocessor directive */

void main()
{

    unsigned int i;                  /*local variable*/

    P0=0x00;
    while(1)
     {
     P0=oxff;                        /*statements*/
     for(i=0;i<255;i++);
     P0=0x00;
     for(i=0;i<255;i++);
     }
```

Examples of Embedded System

- Automobiles: consist of 10~100 embedded systems designed to perform different tasks.
- Mobile phones:
- Industrial robots or machines:
- Medical equipment.
- Household application
- Etc.

Characteristics of an Embedded System

- ✓ **Designed for a specific task**. E.g. to detect the smoke in fire alarm system.
- ✓ **Tightly constrained** as need to embed it inside another product.
- ✓ **Reactive and Real time**: must continually react to changes in the system's physical environment and must compute certain results in real time without any delay. E.g. Car braking System.
- ✓ **Microcontroller or microprocessor based.**
- ✓ **Memory**: its software (firmware) is embedded in the memory (ROM).
- ✓ **No user interface but attached I/O**
- ✓ **Low power hardware**

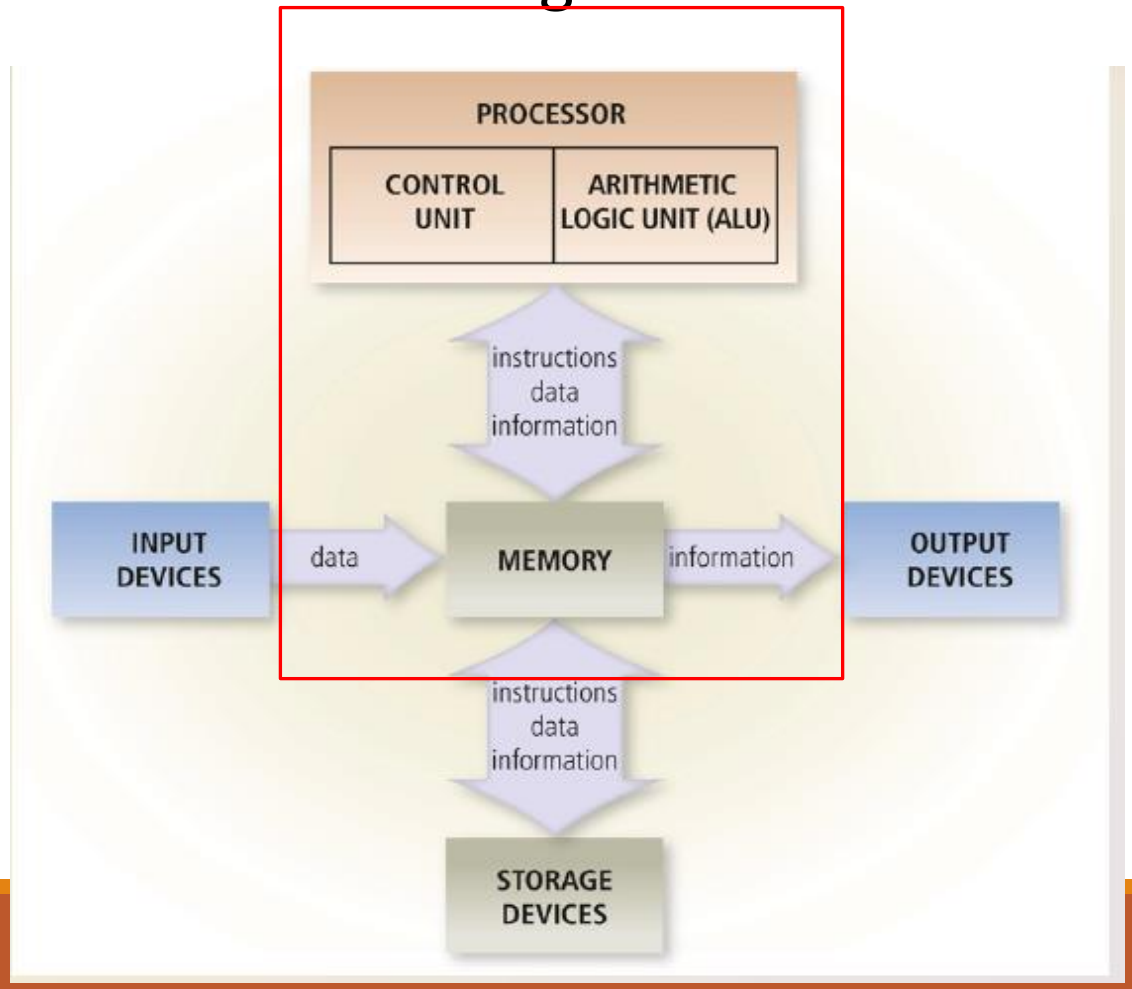
Computer Components

- At the most basic level, a computer is a device consisting of three pieces:

A processor to interpret and execute programs

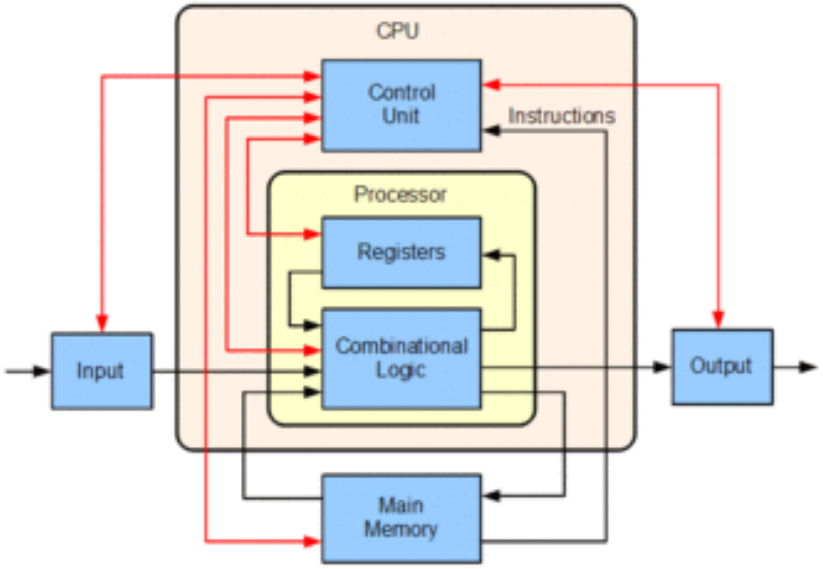
A memory to store both data and programs

A mechanism for transferring data to and from the outside world Or Input/Output mechanism.



Central Processing Unit (CPU)/Microprocessor

- CPU consists of a variety of circuitry and components (an ALU, register array, and a control unit) that are packaged together and connected directly to the motherboard.
- **ALU** performs arithmetical and logical operations on the data received.
- **Register** array consists of registers (word-size memory) using D flip-flop .
- **Control unit (CU)** controls the flow of data and instructions within the computer.



General Purpose Processor

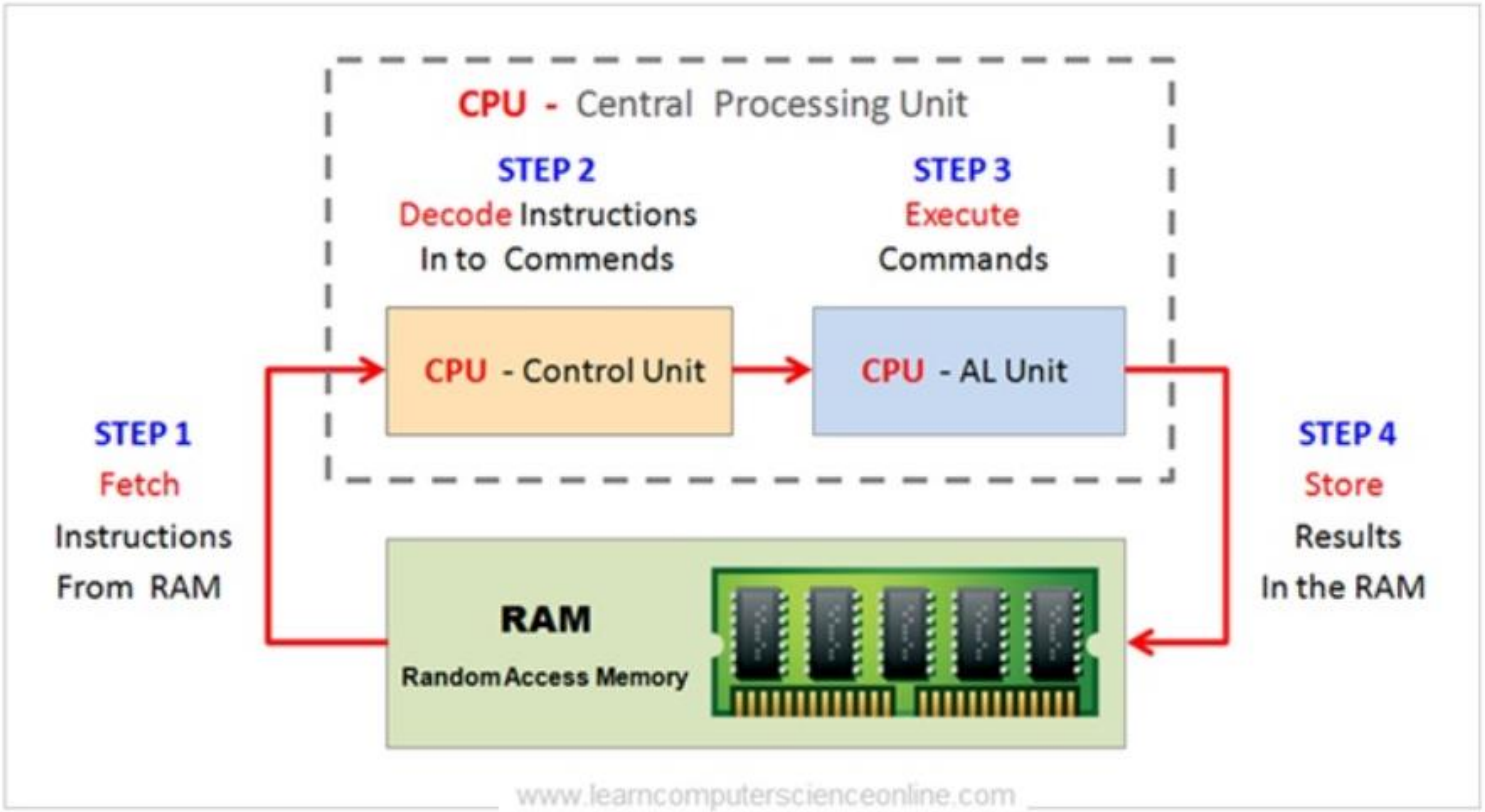
TYPE OF PROCESSOR	NAME	NUMBER OF CORES	CLOCK SPEED	TOTAL CACHE MEMORY		
				LEVEL 1	LEVEL 2	LEVEL 3
DESKTOP	Intel Core i7 AMD Phenom II	4 2–4	2.66–3.33 GHz 2.4–3.2 GHz	64 KB* 128 KB*	256 KB* 512 KB*	8 MB 4–6 MB
SERVER/ WORKSTATION	Intel Xeon (5500 series) AMD Opteron (3rd generation)	2 or 4 4 or 6	1.86–3.2 GHz 2.0–3.1 GHz	64 KB* 128 KB*	256 KB* 512 KB*	4–8 MB 6 MB
NOTEBOOK	Intel Core 2 Mobile AMD Turion X2 Mobile	1, 2, or 4 2	1.06–3.06 GHz 2.0–2.5 GHz	64 KB* 128 KB*	1–12 MB 1–2 MB*	none none
NETBOOK	Intel Atom AMD Athlon Neo	1–2 1	800 MHz–2 GHz 1.6 MHz	56 KB* 128 KB*	512 KB* 512 KB*	none none

* Per core

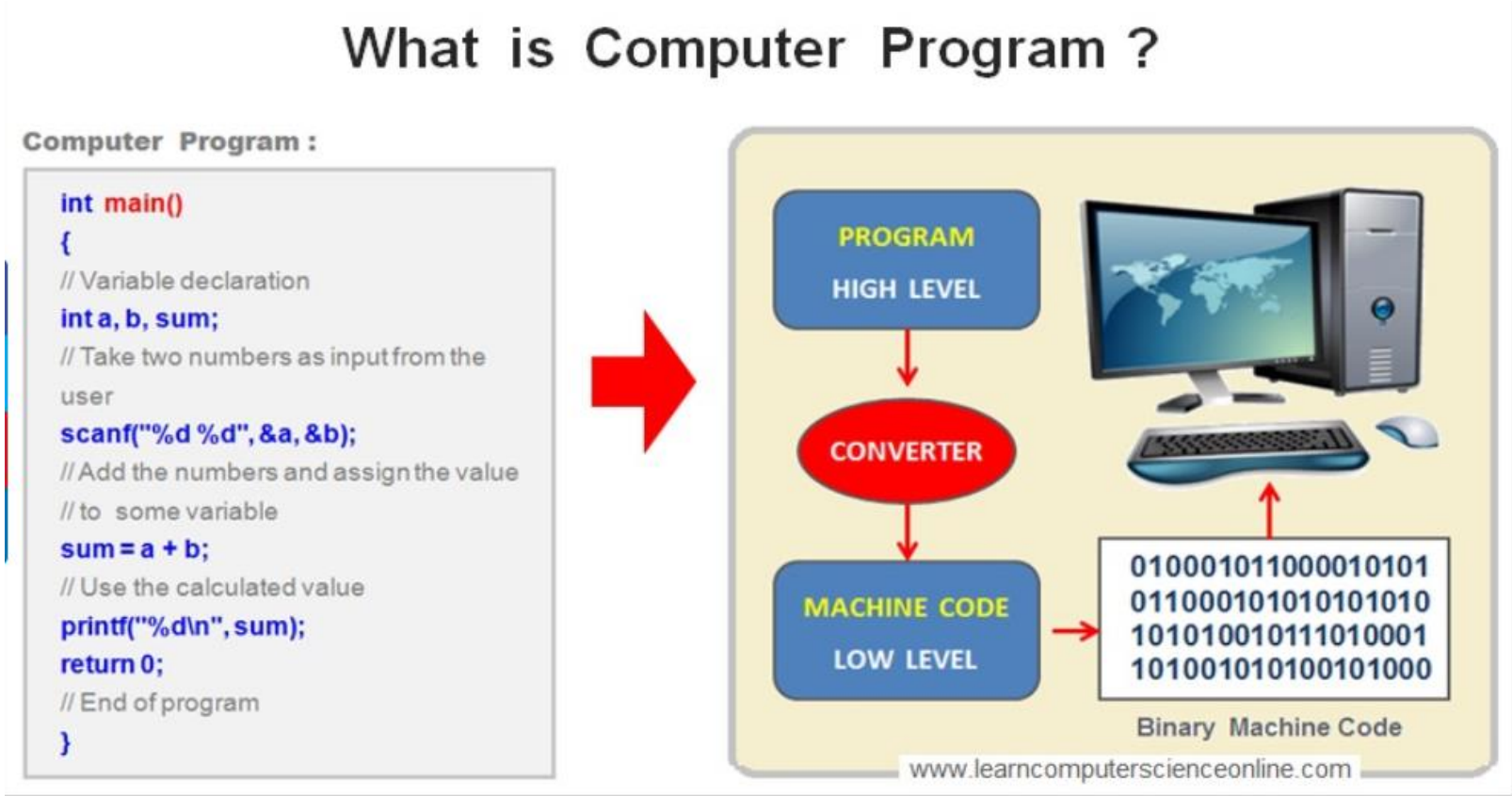
SOURCE: SOURCE: UNDERSTANDING COMPUTERS 13TH EDITION BY D. MORLEY AND C. S. PARKER

Machine Cycle

The main activity performed by the CPU to execute the program instructions stored in the memory together with data.

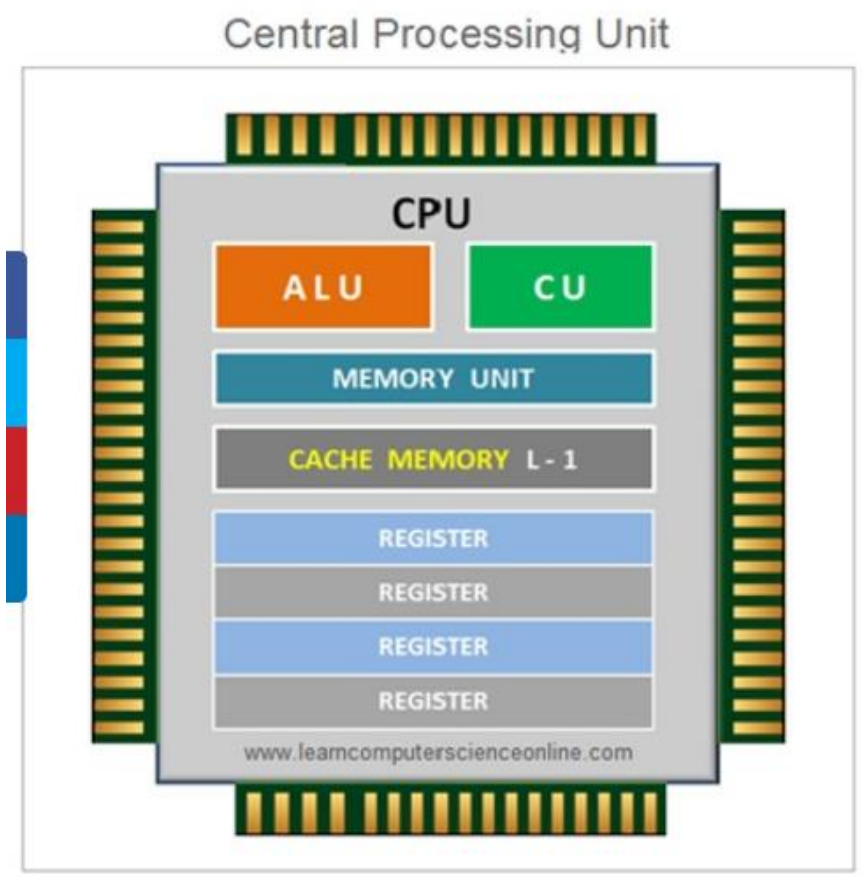


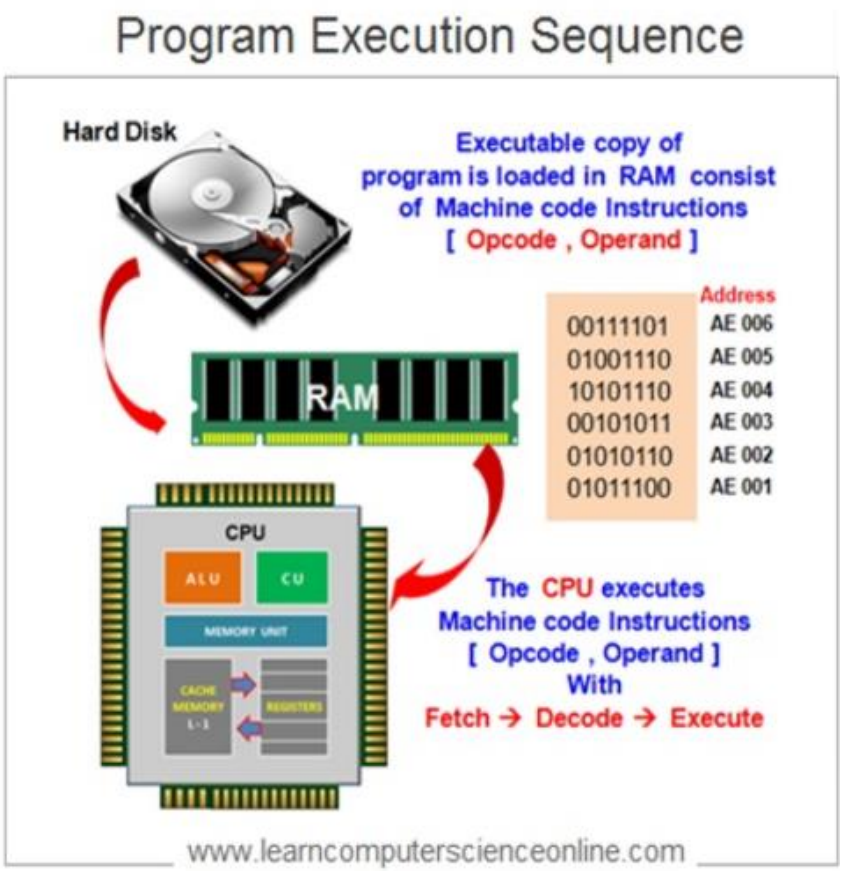
Program in Computer



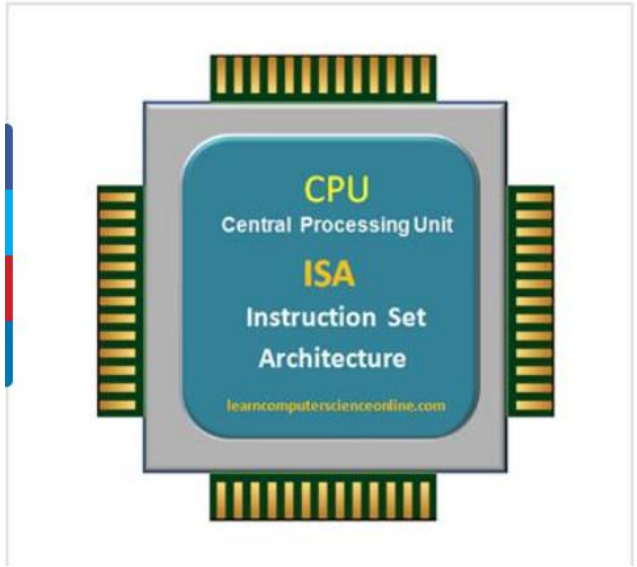
SOURCE: HTTPS://WWW.LEARNCOMPUTERSCIENCEONLINE.COM/WHAT-IS-MACHINE-CYCLE/

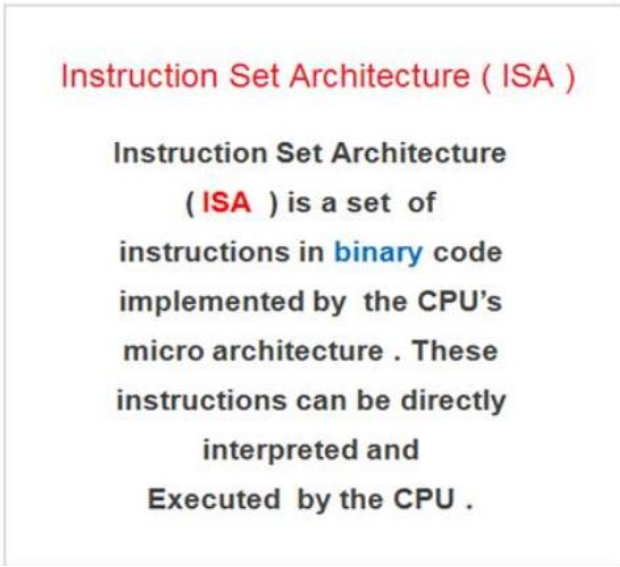
Executing A Program in Computer

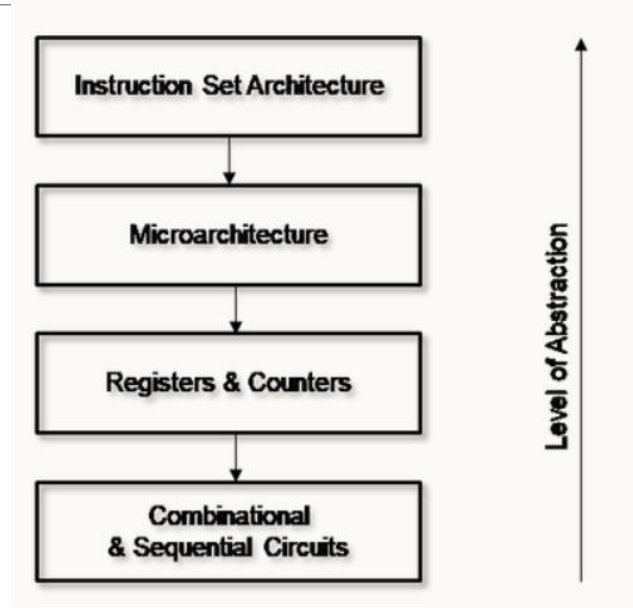




Instruction Set Architecture (ISA)





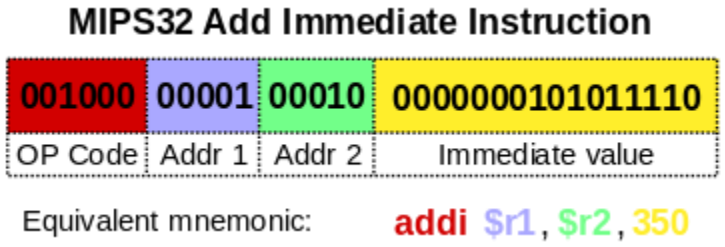


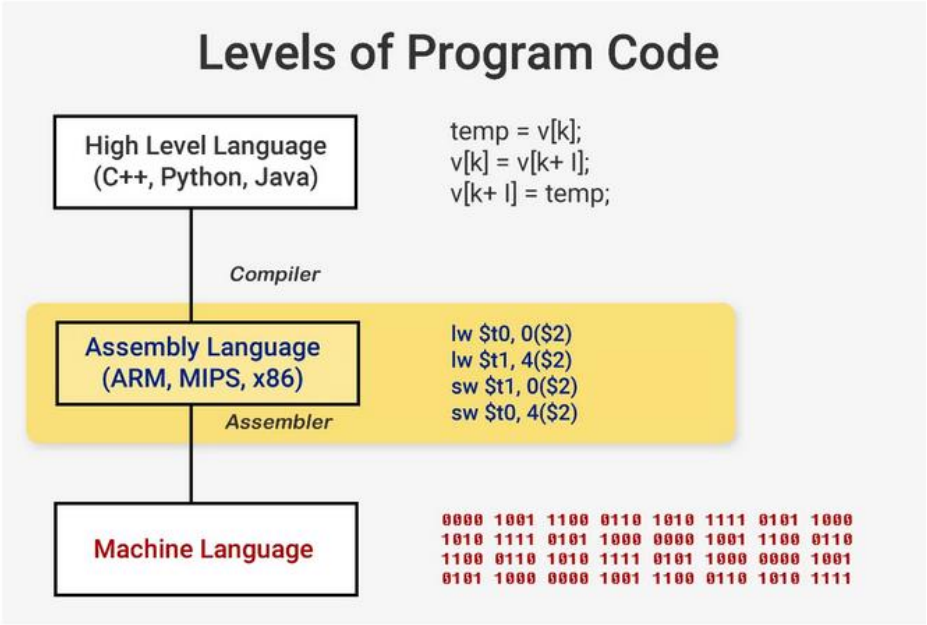
- An ISA specifies the format of its instructions and the primitive operations that the machine can perform.
- An ISA defines the supported data types, the registers, the hardware support for managing main memory, and the input/output model of a family of implementations of the ISA.

SOURCE: HTTPS://WWW.LEARNCOMPUTERSCIENCEONLINE.COM/WHAT-IS-MACHINE-CYCLE/

Instruction Set Architecture (ISA)

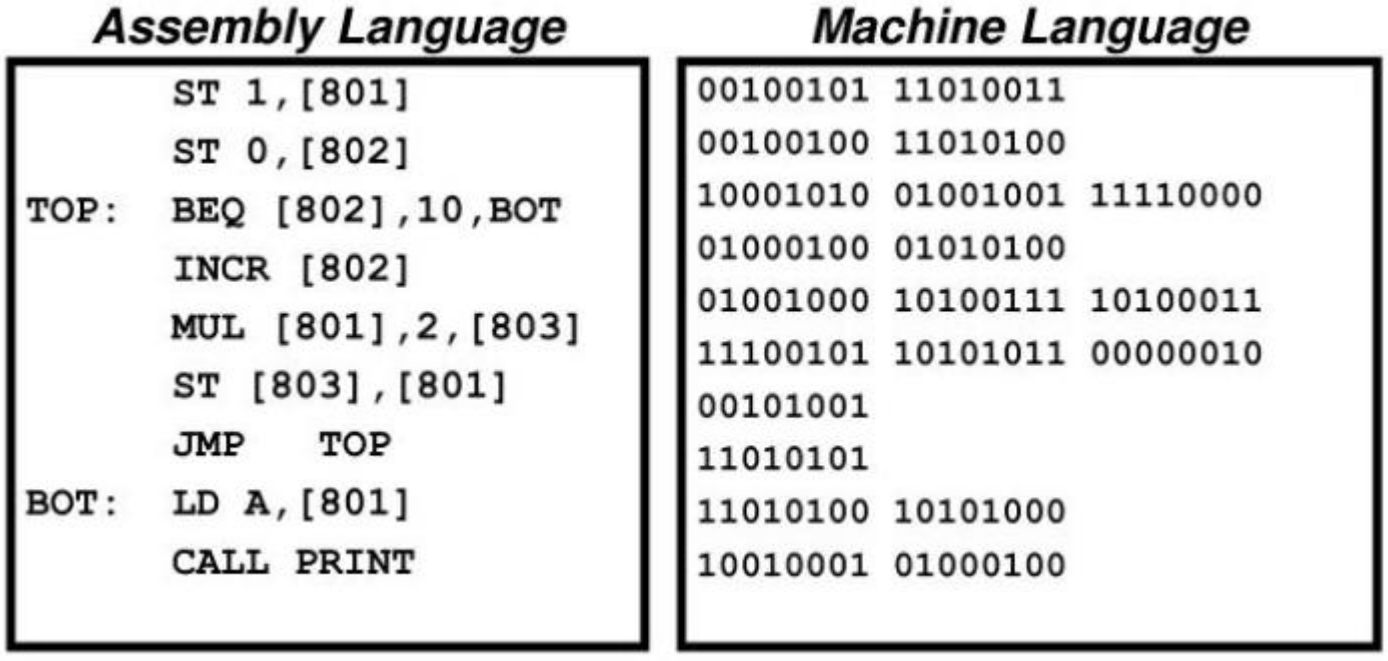
- ISA is an interface between HW and SW.
- Programs are compiled into a set of low-level instructions (ISA), Assembly language and Machine language.





Assembly Language

An **assembly language** is a type of low-level programming language that is intended to communicate directly with a computer's hardware. Unlike machine language, which consists of binary and hexadecimal characters, assembly languages are designed to be readable by humans



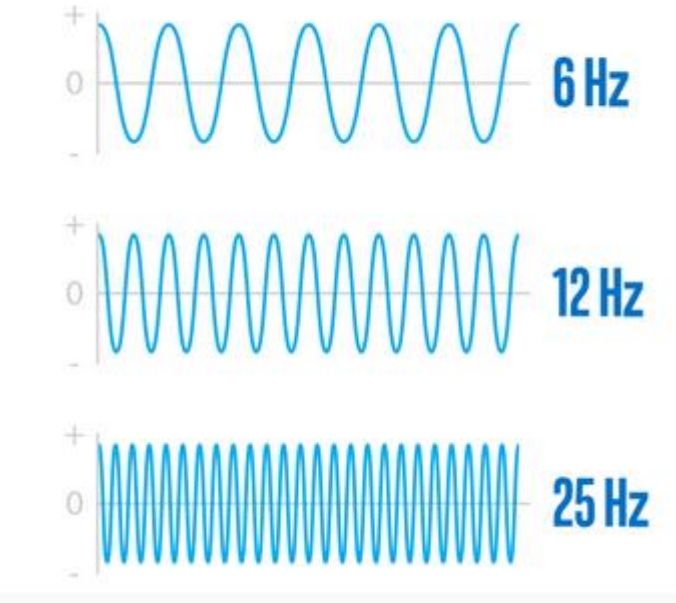
Processor Speed/Clock Speed in Processor

Processor speed in Hz is one of the most important parameters to consider about the performance.

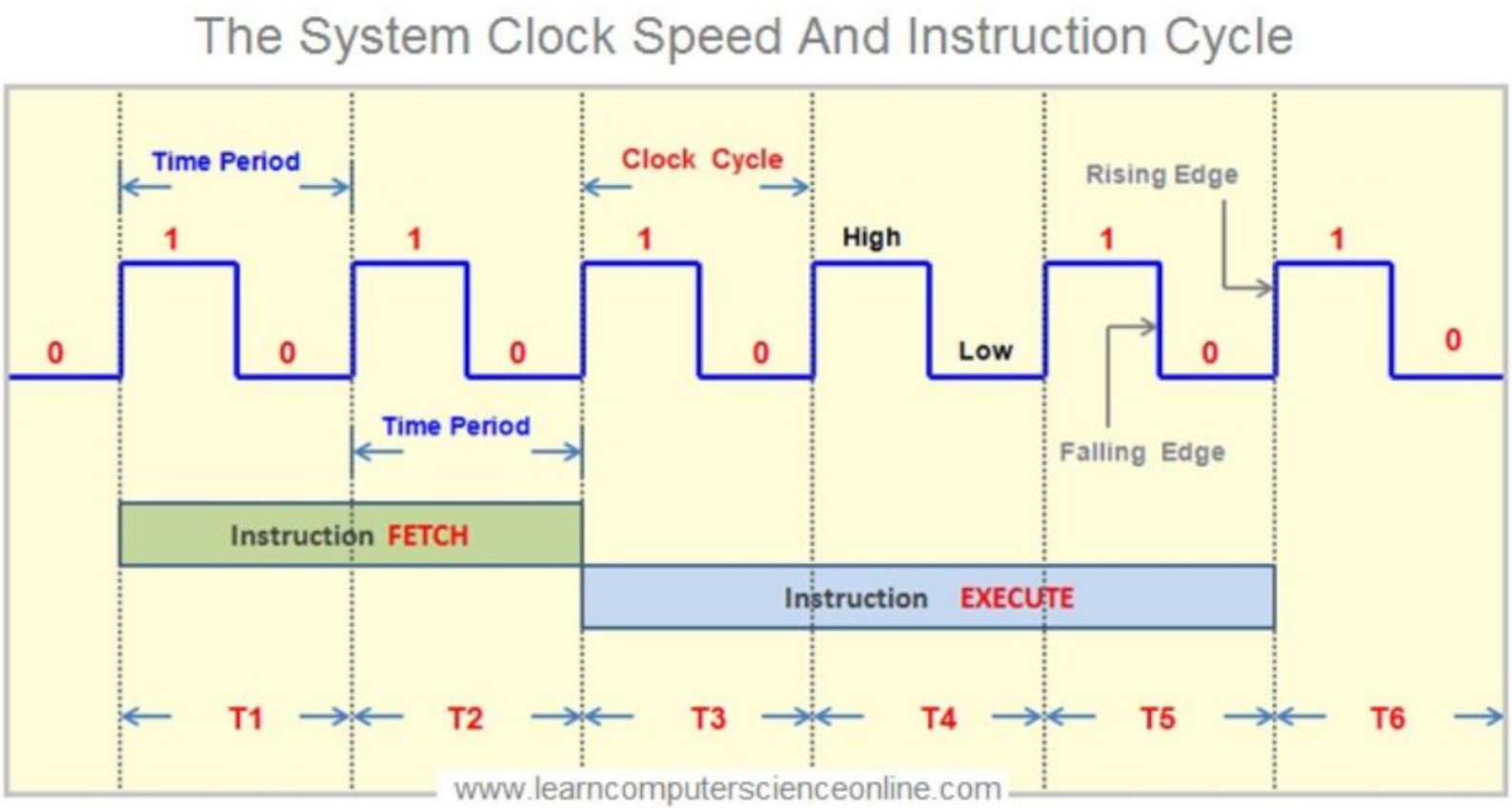
A higher clock speed means a faster processor but need to consider many factors.

E.g. A processor with a clock of 2 MHz executes 2 million cycles per second.

Instructions per cycle or Instructions per second will be depending on the clock speed.



Instruction Cycle and Clock Speed



CPU's clock speed: the clock cycle which is the amount of time between two pulses of an oscillator.

Instruction cycle: the process to execute one single program instruction.

Depending upon the type of instruction, a single instruction might need one or more machine cycles.

SOURCE: HTTPS://WWW.LEARNCOMPUTERSCIENCEONLINE.COM/WHAT-IS-MACHINE-CYCLE/

Processors for Embedded Systme

SOURCE: LEE, BERKELEY

Special-purpose Processor

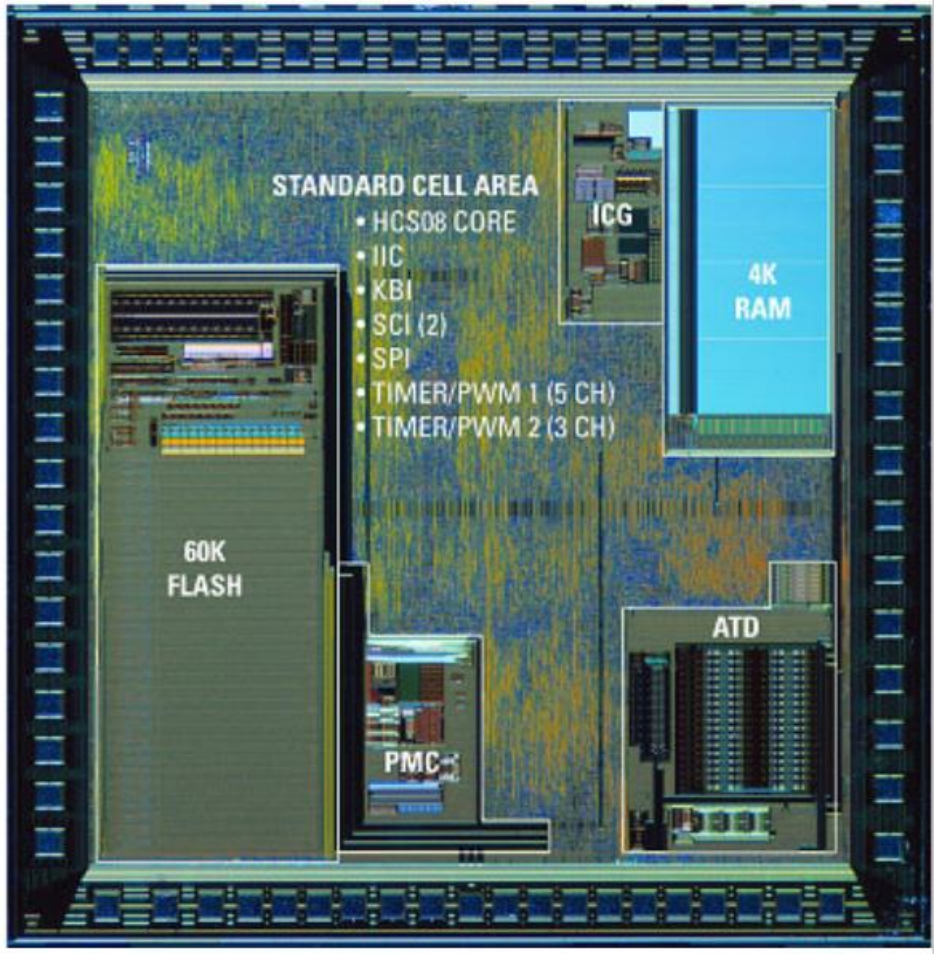
They range from very small, slow, inexpensive, low-power devices, to high-performance, special-purpose devices

- Microcontrollers integrate a microprocessor with peripheral devices in embedded systems.
- A digital signal processor (DSP) is specialized for signal processing.
- Graphics processing units (GPUs) are processors designed primarily for realtime rendering of images.
- Other specialized units exist for video processing and machine vision. (See: Hardware acceleration.)
- Systems on chip (SoCs) often integrate one or more microprocessor or microcontroller cores.

SOURCE: WIKIPEDIA

Microcontrollers (μC) or MCU

- A small computer on a single integrated circuit consisting of a relatively simple **CPU** combined with peripheral devices such as memories, I/O devices, and timers.
- By some accounts, more than half of all CPUs sold worldwide are microcontrollers.
- The difference between microcontrollers and general-purpose processors is getting indistinct.



SOURCE: LEE, BERKELEY

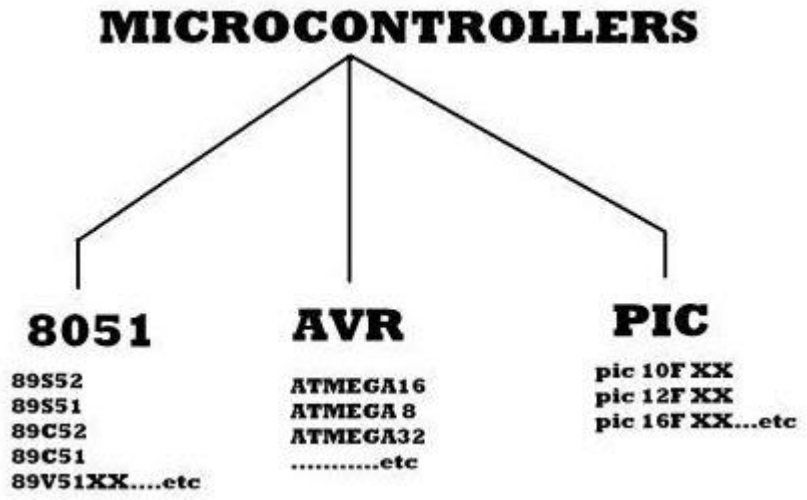
Microcontrollers (μC) or MCU

- An Embedded Computer System on a Chip
  - A CPU
  - Memory (Volatile and Non-Volatile)
  - Timers
  - I/O Devices
- Typically intended for limited energy usage
  - Low power when operating plus sleep modes
- Where might you use a microcontroller?

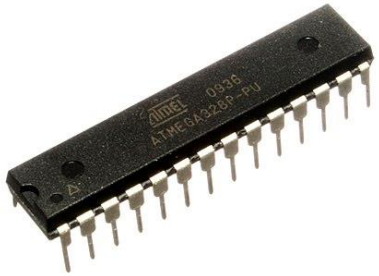
Examples MCU

The simplest microcontrollers operate on 8-bit/16-bits words and are suitable for applications that require small amounts of memory and simple logical functions (vs. performance-intensive arithmetic functions).

They may consume extremely small amounts of energy, and often include a sleep mode that reduces the power consumption to nanowatts.



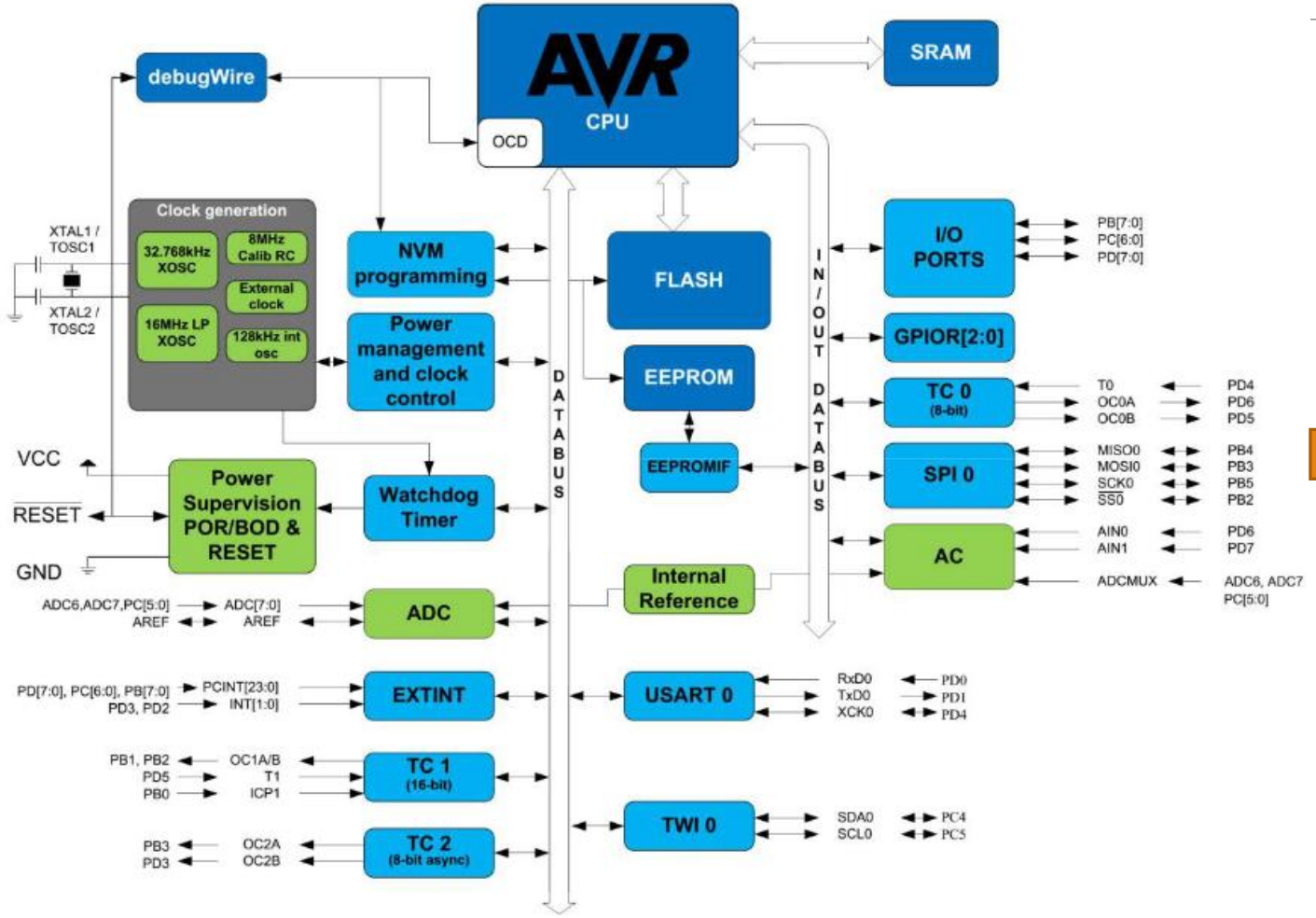
Atmel's AVR ATmega328 MCU

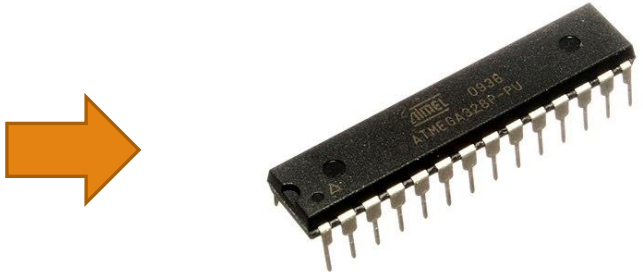


Parameter	Value
CPU type	8-bit AVR
Maximum CPU speed	20 MHz
Performance	20 MIPS at 20 MHz[3]
Flash memory	32 KB
SRAM	2 KB
EEPROM	1 KB
Package pin count	28 or 32
Capacitive touch sensing channels	16
Maximum I/O pins	23
External interrupts	2
USB interface	No

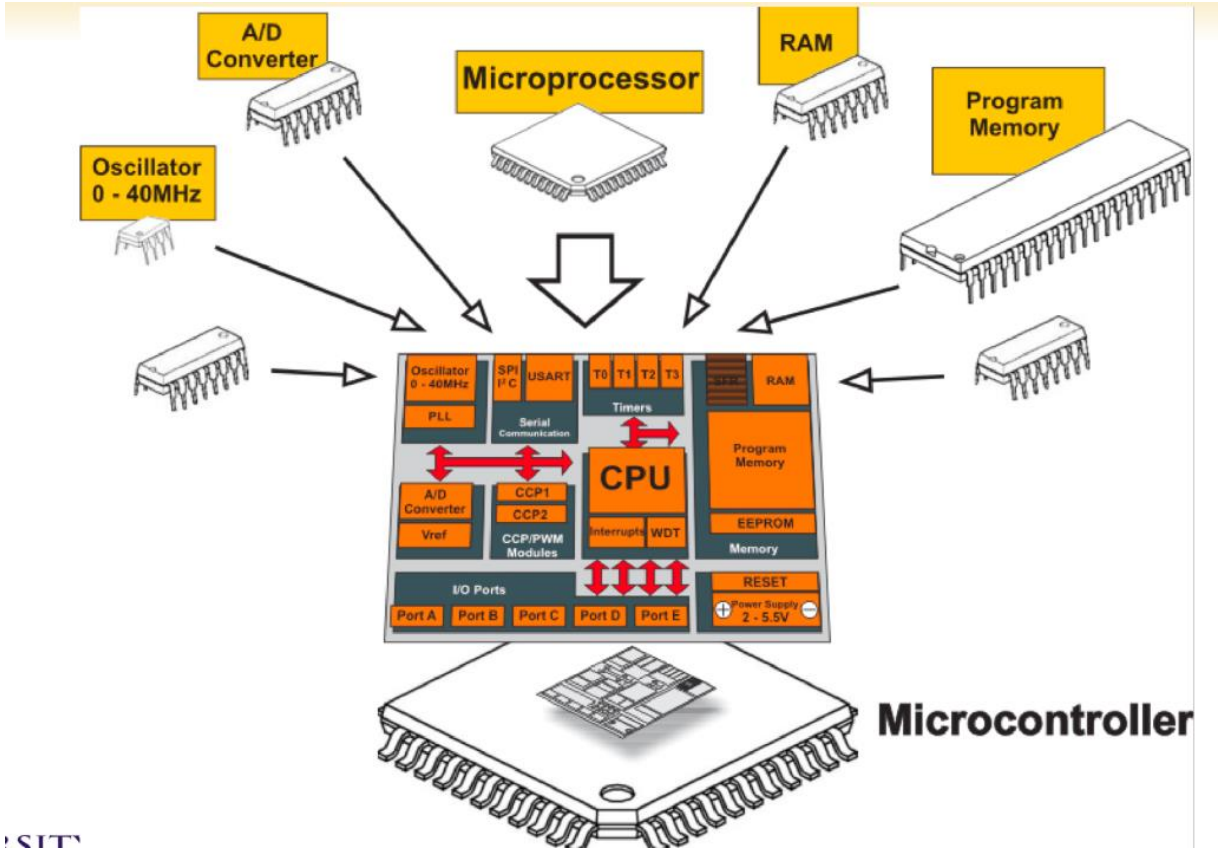
Have you seen or used it before? Where?

Atmel's AVR ATmega328 MCU





Microcontroller vs Microprocessor



Microprocessor	Microcontroller
It is used for big applications.	It is used to execute a single task within an application.
Microprocessor is the heart of computer system.	It is the heart of the embedded system.
It is just a processor. Memory and I/O components have to be having to be connected externally.	Microcontroller contains external processor along with internal memory and I/O components.
Since I/O and memory connected externally, the circuit becomes large.	Since I/O and memory present internally, the circuit is small.
Can't be used in compact systems and hence inefficient.	Can be used in compact systems and microcontroller is an efficient technique.
Cost of entire system increases.	Cost of entire system is low.
Power consumption is high.	Power consumption is low.
Most of the microprocessors do not have power saving modes.	Most of the microcontrollers have power saving mode.
Difficult to replace.	Easy to replace.
Mainly used in personal computers.	Used mainly in washing machine, MP3 players.

ARM Cortex Processors

ARM Cortex-A family:

Applications processors

Support OS and high-performance applications

Such as Smartphones, Smart TV

ARM Cortex-R family:

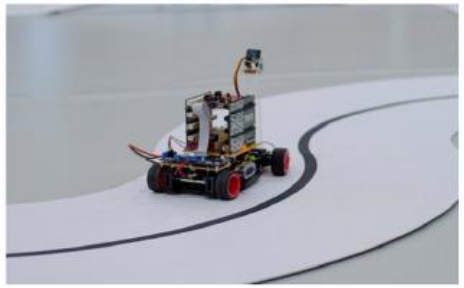
Real-time processors with high performance and high reliability

Support real-time processing and mission-critical control

ARM Cortex-M family:

Microcontroller

Cost-sensitive, support SoC



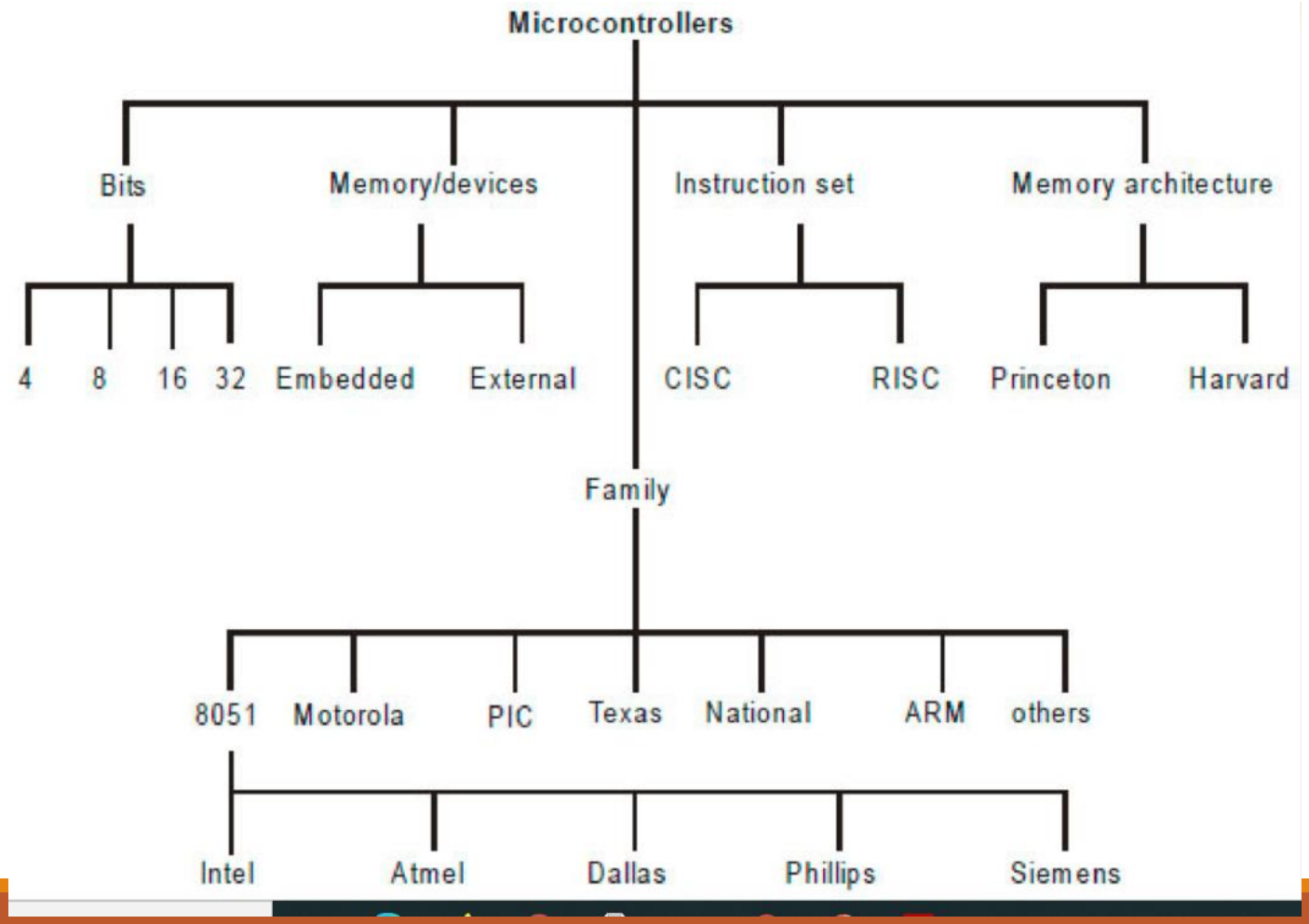
How to choose micro-processors/controllers?

Things that matter

- Peripherals (I/O)
- Concurrency & Timing
- Clock Rates
- Memory sizes (SRAM & flash)
- Package sizes

Parameter	Value
CPU type	8-bit AVR
Maximum CPU speed	20 MHz
Performance	20 MIPS at 20 MHz[3]
Flash memory	32 KB
SRAM	2 KB
EEPROM	1 KB
Package pin count	28 or 32
Capacitive touch sensing channels	16
Maximum I/O pins	23
External interrupts	2
USB interface	No

Types of Microcontrollers



SOURCE: LEE, BERKELEY

Digital Signal Processing (DSP) Processors

- Many embedded applications do quite a bit of signal processing.
- Processors designed specifically to support numerically intensive signal processing applications are called DSP processors, or DSPs (digital signal processors).
- Signal Processing Applications: interactive games; radar, sonar, and LIDAR (light detection and ranging) imaging systems; video analytics (the extraction of information from video, for example for surveillance); driver-assist systems for cars; medical electronics; and scientific instrumentation.
- They typically perform sophisticated mathematical operations on the data, including **filtering**, system identification, frequency analysis, machine learning, and feature extraction. These operations are mathematically intensive.

SOURCE: LEE, BERKELEY

Sample Rate

A signal is a collection of sampled measurements of the physical world, typically taken at a regular rate called the sample rate (how many sample per second).

Examples:

Motion Control Application with sensor: a few Hertz to a few hundred Hertz.

Audio signals: 8 kHz (telephone for voice signals) to 44.1 kHz (CD)

Ultrasonic application (medical image) and HIFI audio: at much higher rate.

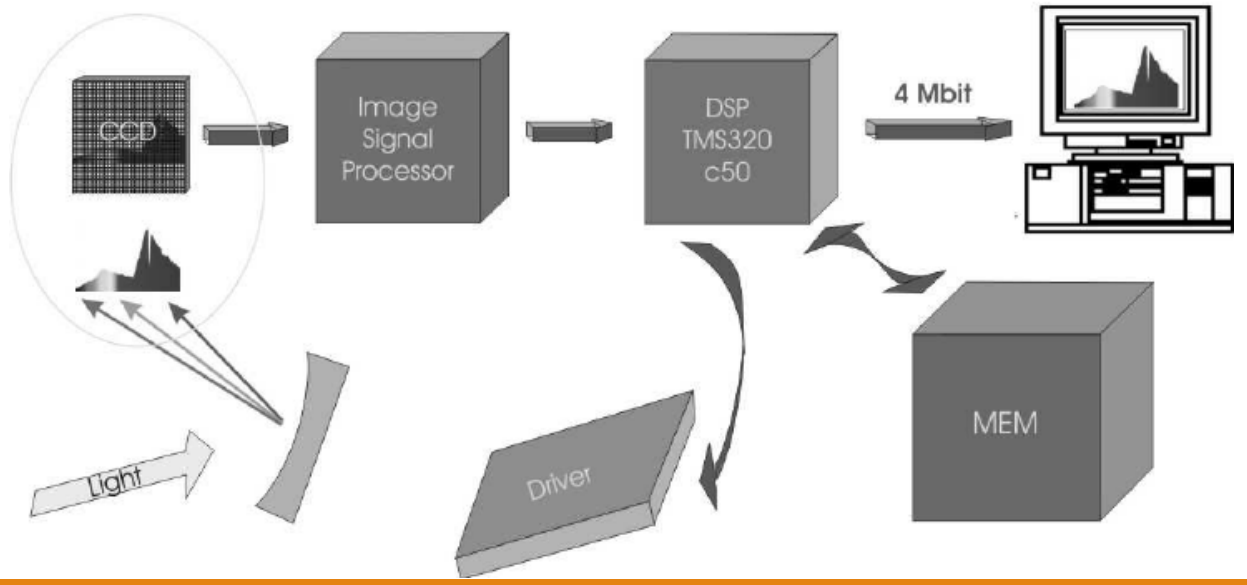
Sample rate 1 Hz $\leftrightarrow$ 1 sample every one second

Sample rate 10 Hz $\leftrightarrow$ 10 samples every second (1 sample every 0.1 second)

Finite impulse response (FIR) filtering

FIR filtering is very important in digital communications and signal (Audio and Video) processing.

Example: Moving average filter



Finite impulse response (FIR) filtering

FIR Formula

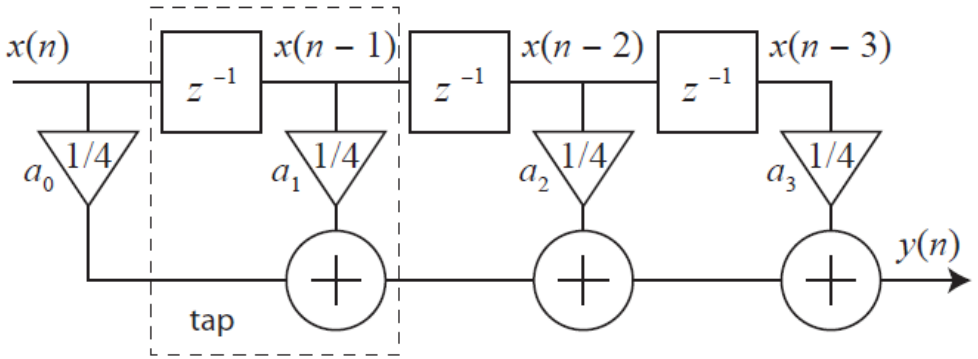
$$y(n) = \sum_{i=0}^{N-1} a_i x(n-i) \ ,$$

- N is the length of the FIR filter.
- ai are tap values (weight coef.)
- x(n) is the input sample

- A common Signal Processing with Multiply-Accumulate
- Consider a FIR filter with $N = 4$ and $a_0=a_1=a_2=a_3=1/4$

$$y(n) = (x(n) + x(n-1) + x(n-2) + x(n-3))/4$$

z^{-1} unit delay model



- # of operations: N = 4 Multiplication and N-1 = 3 Accumulation

Operations per second

- Suppose that an FIR filter is provided with samples at a rate of 1 MHz (one million samples per second), and that N = 32.

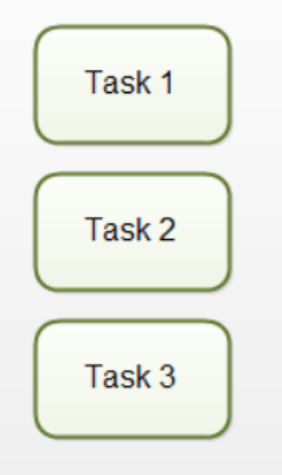
- Outputs must be computed at a rate of 1 MHz, and each output requires 32 multiplications and 31 additions.
- How many is arithmetic operations per second required by the processor?

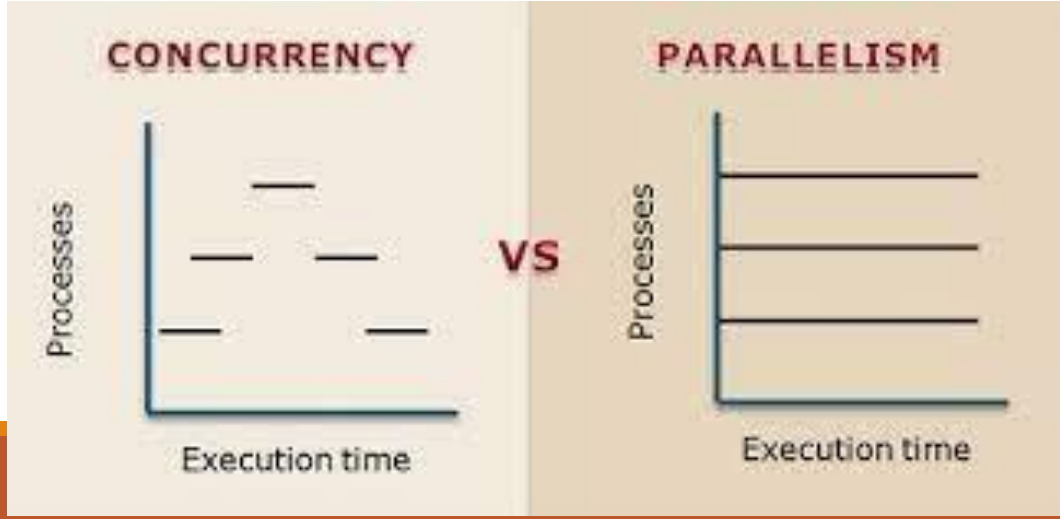
Graphics Processors or GPUs

- A graphics processing unit (GPU) is a specialized processor designed specially to perform the calculations required in graphics rendering.
- Most used for Gaming (earlier days)
- Common programming language: CUDA
- Modern GPUs support 3D graphics, shading, and digital video. Dominant providers of GPUs today are Intel, NVIDIA and AMD.
- GPUs are typically quite power hungry, and therefore today are not a good match for energy constrained embedded applications.

Parallelism vs Concurrency

- An embedded program often needs to monitor and react to multiple concurrent sources of stimulus, and simultaneously control multiple output devices that affect the physical world.
- Embedded programs are almost always concurrent programs,
- Tasks are said to be “concurrent” if they conceptually execute simultaneously
- Tasks are said to be “parallel” if they physically execute simultaneously on distinct hardware (such as on multicore machine/ multiprocessor/ servers on a server farm).





Imperative Language

- Non-concurrent programs specify a sequence of instructions to execute.
- Imperative (procedural) Language expresses a computation as a sequence of operations
- Example: C, C++, Java programming language

```
int total = 0;
int number1 = 5;
int number2 = 10;
int number3 = 15;
total = number1 + number2 + number3;
```

- How to write concurrent programs in imperative language? **Thread Library**

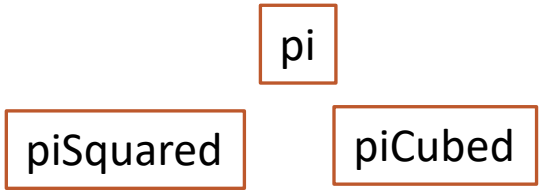
A thread library uses facilities provided not by C, but rather provided by the operating system and/or the hardware.

Program Dependency – Sequential Consistency

A compiler may analyze the dependencies between operations in a program and produce parallel code, if the target machine supports it. This analysis is called **dataflow analysis**.

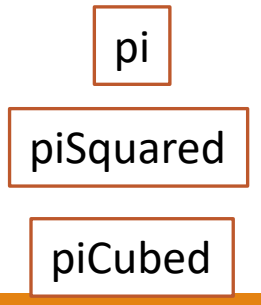
```
double pi, piSquared, piCubed;
pi = 3.14159;
piSquared = pi * pi ;
piCubed = pi * pi * pi;
```

No dependency




```
double pi, piSquared, piCubed;
pi = 3.14159;
piSquared = pi * pi ;
piCubed = piSquared * pi;
```

No independent



Parallelism

Parallelism in the hardware aims to improve performance for computation-intensive applications.

Temporal Parallelism – Pipelining

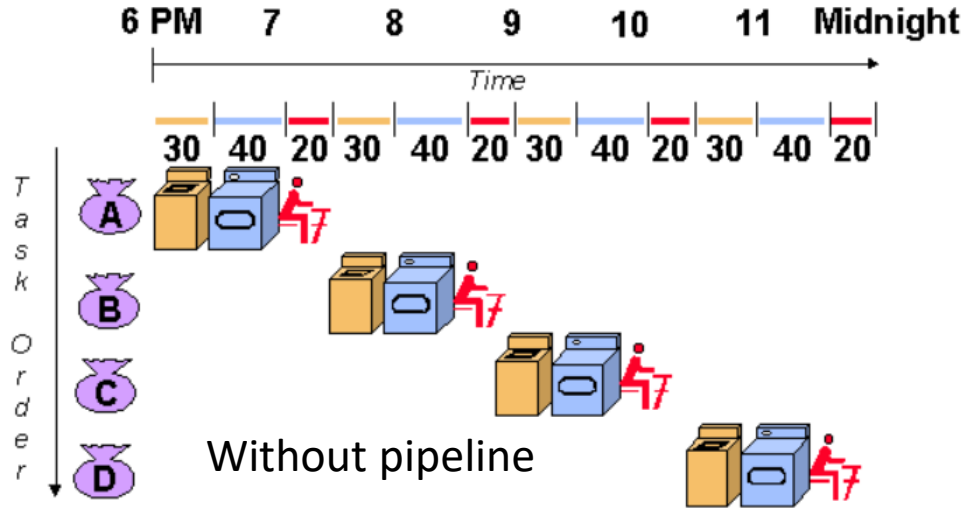
Spatial Parallelism – Superscalar, VLIW (very large instruction word), Multicore

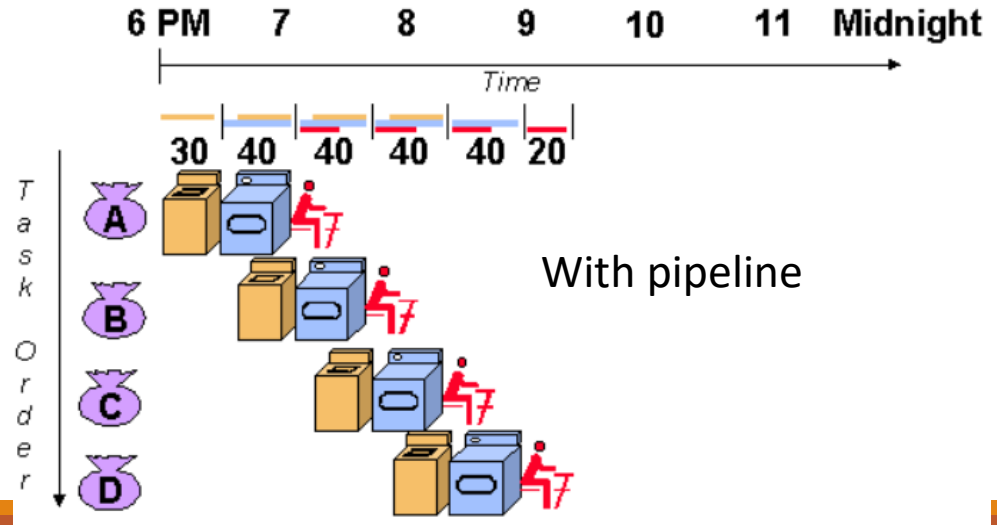
Temporal Parallelism – Pipelining

Pipelining is the process of accumulating instruction from the processor through a pipeline much like an assembly line.
It is used in a RISC machine (reduced instruction set computers).

laundry analogy

1. washing,
2. Drying,
3. folding

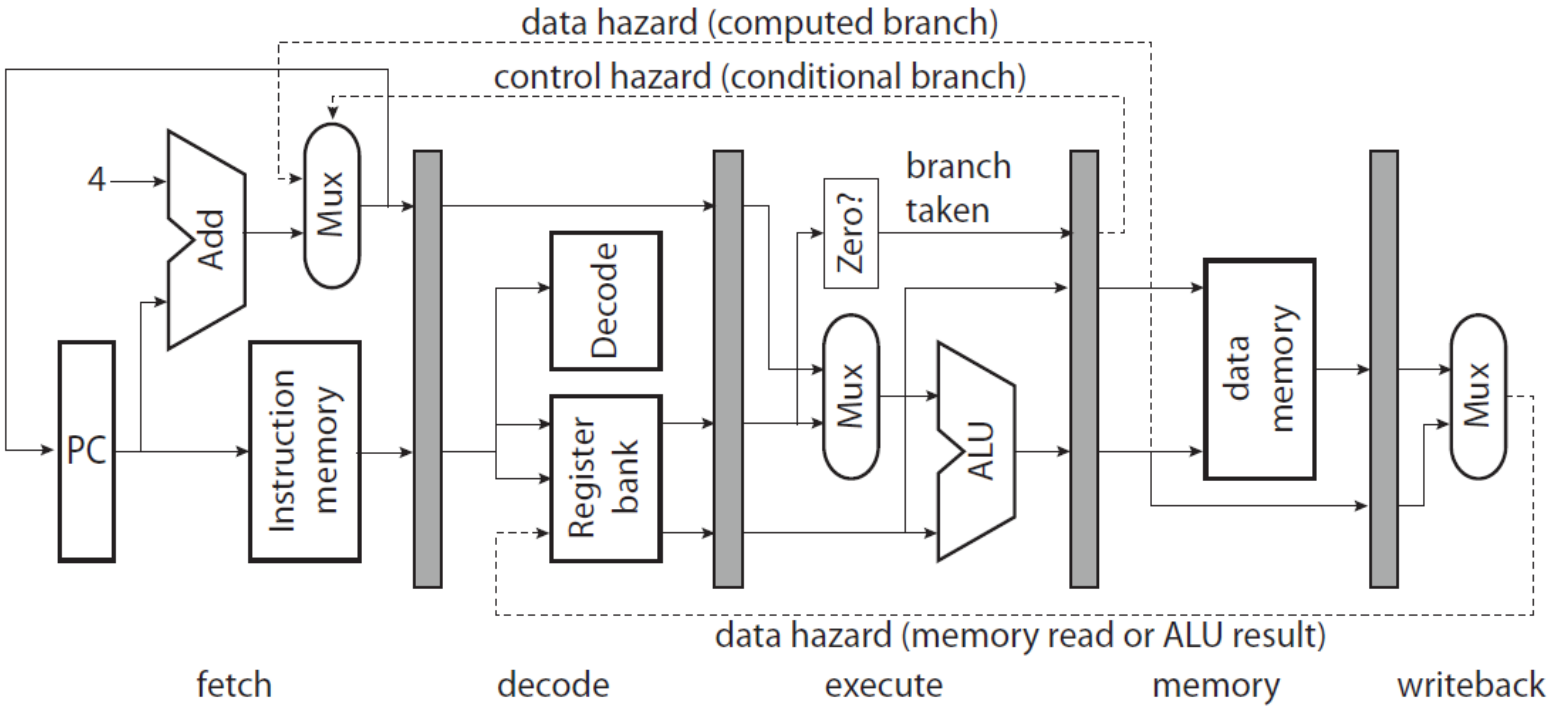




SOURCEHTTPS://CS.STANFORD.EDU/PEOPLE/EROBERTS/COURSES/SOCO/PROJECTS/RISC/PIPELINING/INDEX.HTML

Pipelining

- Pipelining is used in most modern processors.
- Separate into groups of components for each task (machine cycle)



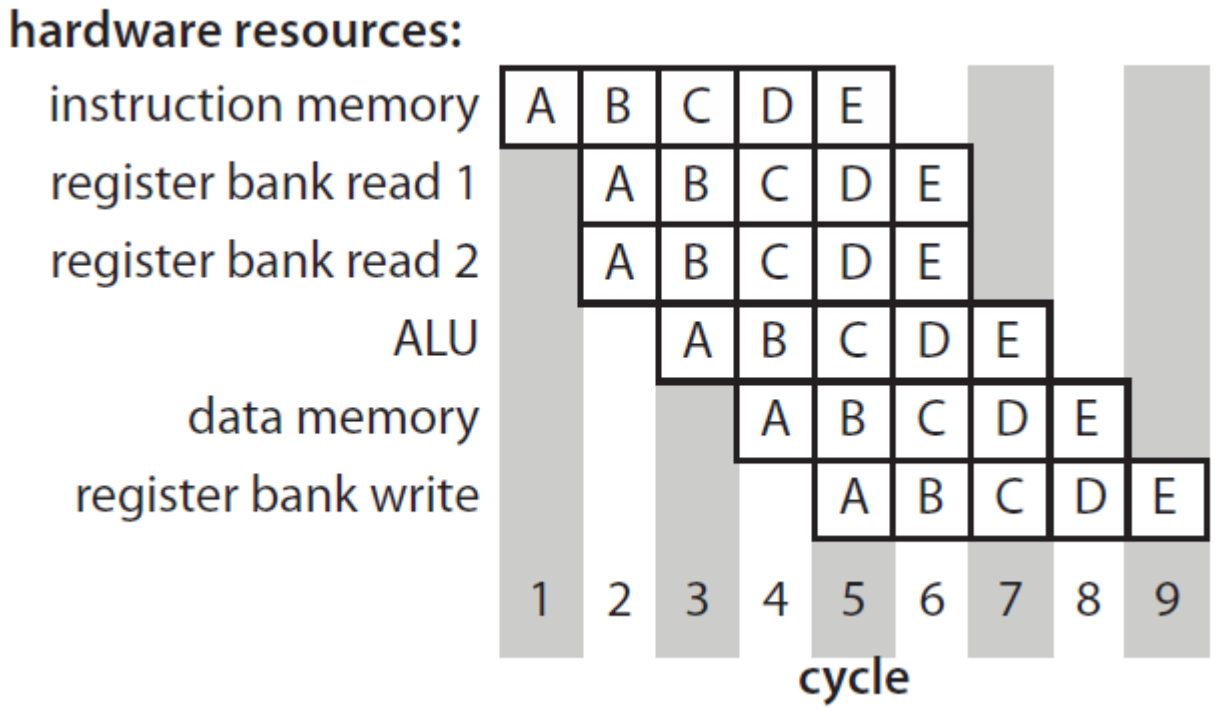
The shaded rectangles are **latches**, which are clocked at processor clock rate.
PC (Program counter) provides an address to the instruction memory.

5 Execution Stages

- **Instruction Fetch (IF):** Fetch instruction from memory pointed by PC, then increment PC
- **Instruction decode (ID):** Decode the instruction
- **Execution (EX):** ALU operates on the operands
- **Memory Access (MEM):** Read/Write to Memory.
- **Write-back cycle (WB):** Stores results in the register

Pipelining

- The portions of the pipeline between the latches operate in parallel.
- Five instructions {A, B, C, D, E} are being executed simultaneously at a different stage of execution.



Pipelining Hazard

- Data Hazard
- Control Hazard
- Out-of-order Execution
- Speculative Execution

Instruction-Level Parallelism

- RISC machine (reduced instruction set computers) a processor that uses simple instruction that can be executed within one clock cycle.
- RISC reduce the cycles per instruction at the cost of the number of instructions per program.
- CISC machine (complex instruction set computer) is a processor with complex (and typically, rather specialized) instructions.
- The CISC approach attempts to minimize the number of instructions per program but at the cost of increase in number of cycles per instruction.
- DSPs are typically CISC machines, and include instructions specifically supporting FIR filtering (and often other algorithms such as FFTs (fast Fourier transforms) and Viterbi decoding).

Superscalar

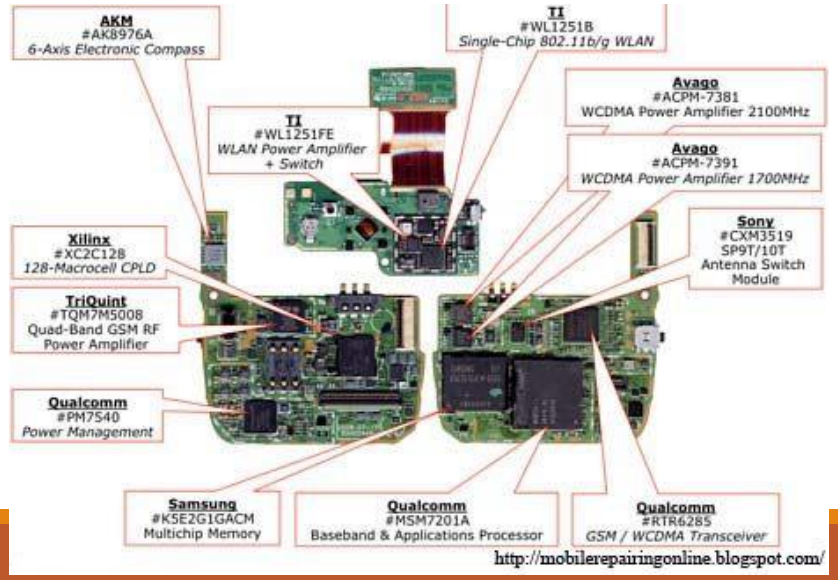
- Superscalar is a method of parallel computing used in many different execution units on the processer.
- The hardware can simultaneously dispatch multiple instructions to distinct hardware units when it detects that such simultaneous dispatch will not change the behavior of the program.
- Superscalar processors have a significant disadvantage for embedded systems, which is that execution times may be extremely difficult to predict, and in the context of multitasking (interrupts and threads), may not even be repeatable.

VLIW (Very Long Instruction Word)

- Processors intended for embedded applications often use VLIW architectures.
- Multiple independent instructions per cycle, packed into single large "instruction word" or "packet"

Multicore Architectures

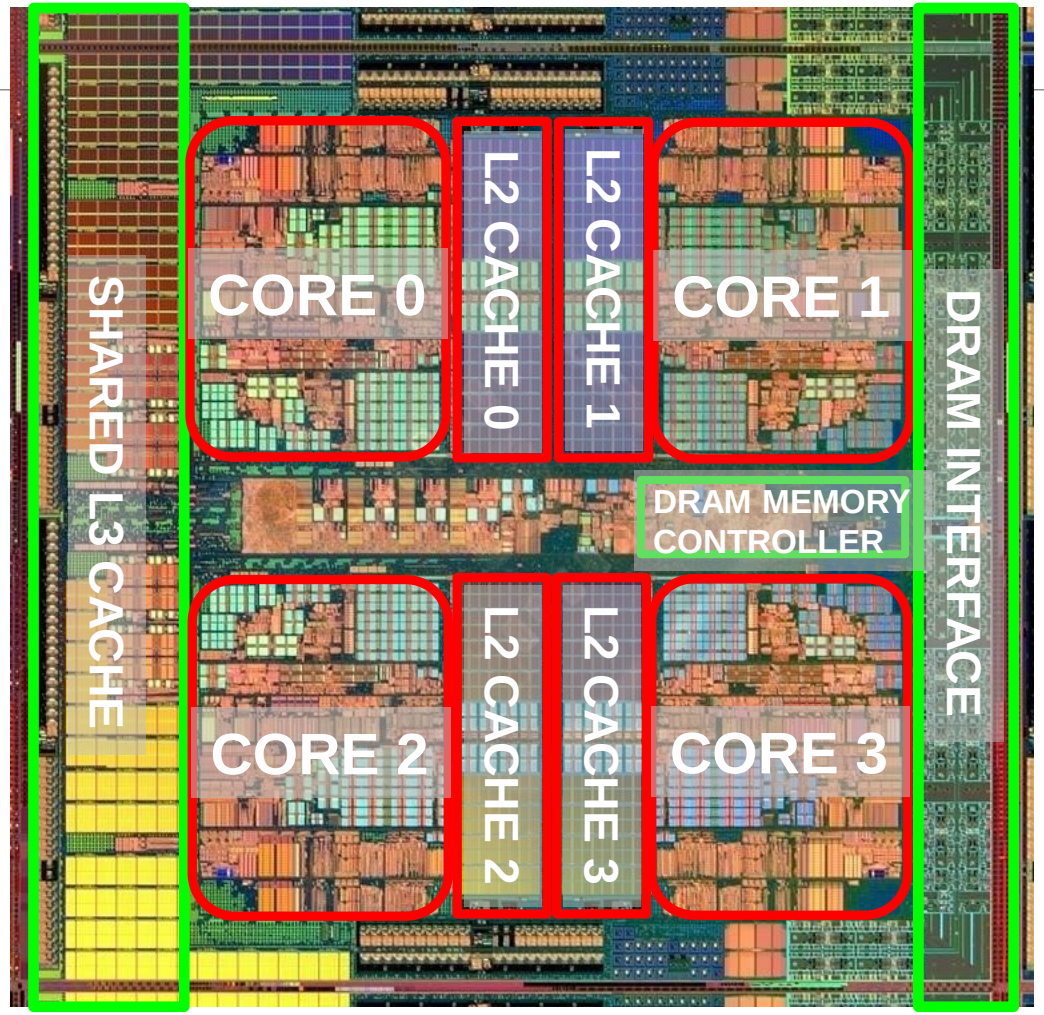
- A multicore machine is a combination of several processors on a single chip.
- For embedded applications, multicore architectures have a significant potential advantage over single-core architectures because real-time and safety-critical tasks can have a dedicated processor.
- This is the reason for the heterogeneous architectures used for cell phones, since the radio and speech processing functions are hard real-time functions with considerable computational load.

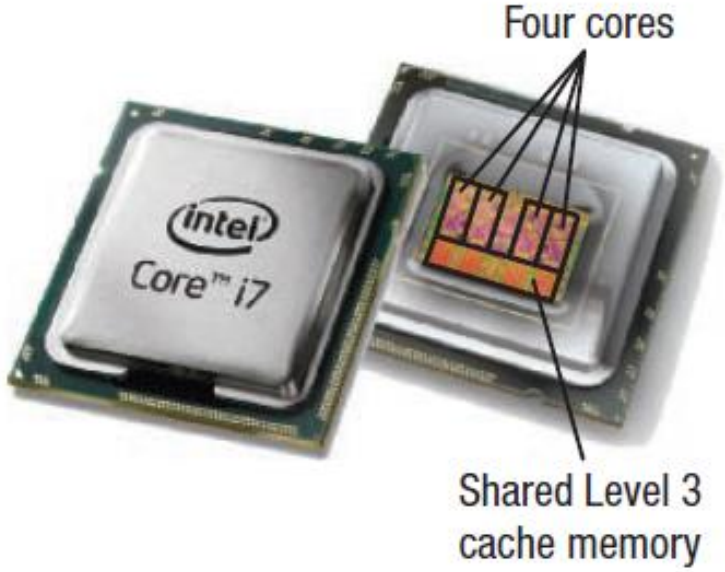


SOURCE: LEE, BERKELEY

An Example: Multi-Core Systems

Multi-Core Chip





*Die photo credit: AMD Barcelona

Reference

- Lee, Edward & Seshia, Sanjit. (2011). Introduction to Embedded Systems - A Cyber-Physical Systems Approach.
- Lecture Note Slides from EECS 149/249A: Introduction to Embedded Systems (UC Berkeley) by Prof. Prabal Dutta and Sanjit A. Seshia
- https://www.learncomputerscienceonline.com/what-is-machine-cycle/