Introduction

ET2223 Microprocessors, Microcontrollers, and Embedded Systems

What is a system?

- A system is a way of working, organizing or doing one or many tasks according to a fixed plan, program or set of rules.
- A system is also an arrangement in which all its units assemble and work together according to the plan or program.

System Examples: Watch

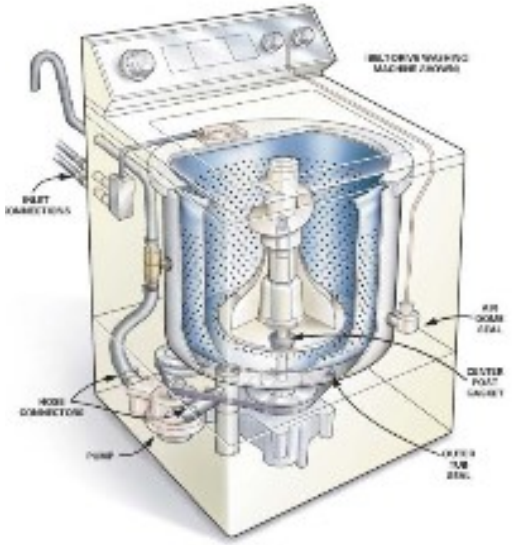
- It is a time display SYSTEM
- Parts: Hardware, Needles, Battery, Dial, Chassis and Strap

- Rules:
  - All needles move clockwise only
  - A thin needle rotates every second
  - A long needle rotates every minute
  - A short needle rotates every hour
  - All needles return to the original position after 12 hours

System Examples: Washing Machine

- It is an automatic clothes washing SYSTEM
- Parts: Status display panel, Switches & Dials, Motor, Power supply & control unit, Inner water level sensor and solenoid valve.

- Rules:
  - Wash by spinning
  - Rinse
  - Drying
  - Wash over by blinking
  - Each step display the process stage
  - In case interruption, execute only the remaining

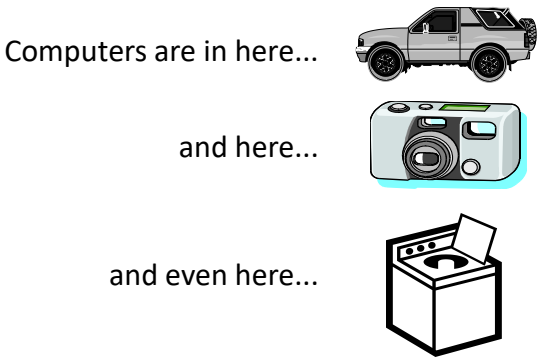


Embedded Systems Overview

- Computing systems are everywhere
- Most of us think of computers as
  - PC's
  - Laptops
  - Mainframes
  - Servers
- But there's another type of computing system that's more common

Embedded Systems Overview

- Embedded computing systems
  - Computing systems embedded within electronic devices
  - Hard to define.
  - Nearly any computing system other than a desktop computer
  - Billions of units produced yearly, versus millions of desktop units
  - Perhaps 50 per household and per automobile



Embedded Systems Definition

- An Embedded System is one that has computer hardware with software embedded in it as one of its important components.

- Its software embeds in ROM (Read Only Memory). It does not need secondary memories as in a personal computer.

Where in our daily life do we use embedded systems?

Embedded Systems Applications

- Consumer Products
  - TV, stereo, remote control, mobile phone, refrigerator, microwave, washing machine
- Automobiles
  - engine management, trip computer, cruise control, immobilizer, car alarm,
  - airbag, ABS, ESP
- Building Systems
  - elevator, heater, air conditioning, lighting, key card entries, locks, alarm systems
- Agriculture
  - feeding systems, milking systems
- Space
  - satellite systems
- Medical Systems
  - pace maker, patient monitoring systems, injection systems, intensive care units
- Office Equipment
  - printer, copier, fax
- Tools
  - multimeter, oscilloscope, line tester, GPS
- Banking
  - ATMs, statement printers
- Transportation
  - Planes/Trains/[Automobiles] and Boats
  - radar, traffic lights, signaling systems

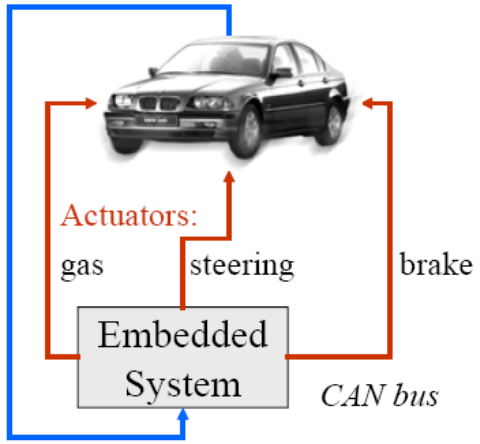
Example: Automobiles

Autonomous cars:

- Electronic gas
- Electronic brake
- Electronic steering

See: *The Daimler Story*

Bräunl 2004



Sensors: Stereo-cameras, speedometer, accelerometers, signalling

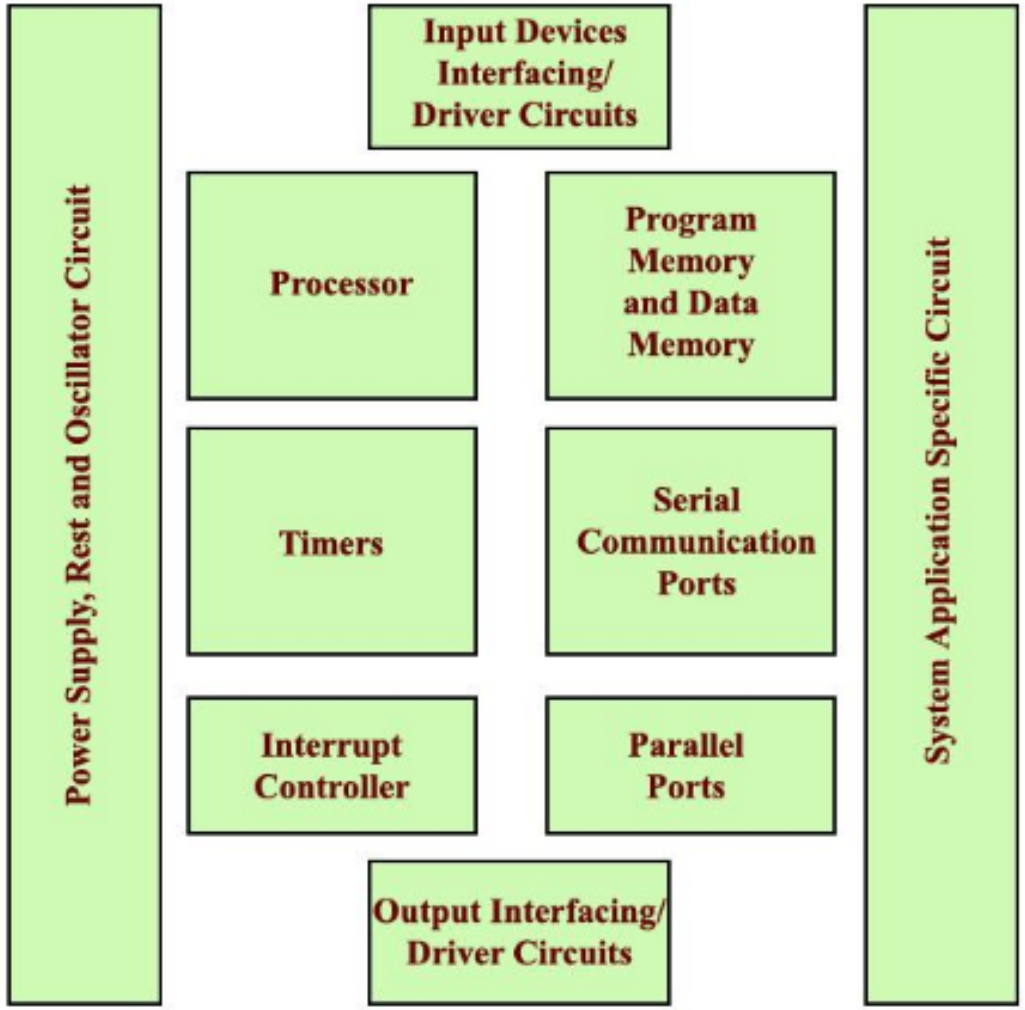
2002: Opel Vectra has over 40 sensors (25 types)

Bräunl 2004

Components of Embedded Systems

- It has Hardware
  - Processor, Timers, Interrupt controller, I/O Devices, Memories, Ports, etc.
- It has main Application Software
  - Which may perform concurrently the series of tasks or multiple tasks.
- It has Real Time Operating System (RTOS)
  - RTOS defines the way the system work. Which supervise the application software. It sets the rules during the execution of the application program. A small scale embedded system may not need an RTOS.

Embedded System Hardware

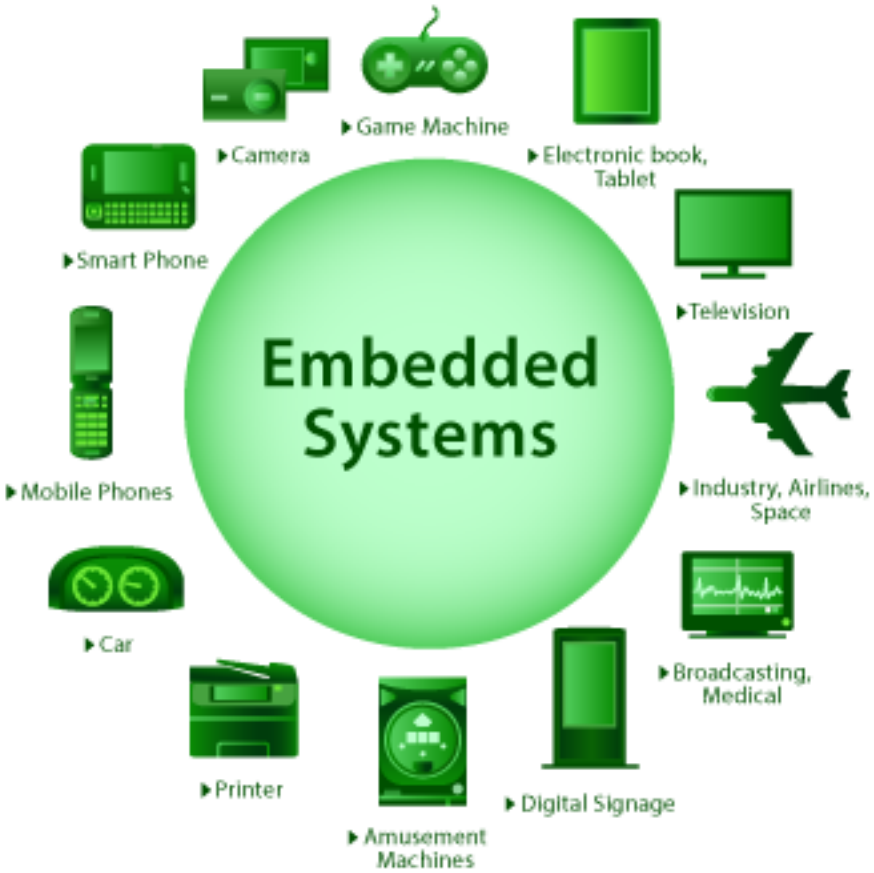


Embedded System Constraints

- An embedded system is software designed to keep in view three constraints:
  - Available system memory
  - Available processor speed
  - The need to limit the power dissipation
- When running the system continuously in cycles of wait for events, run, stop and wakeup.

What makes embedded systems different?

- Real-time operation
- Size
- Cost
- Time
- Reliability
- Safety
- Energy
- Security



Embedded System Classifications

1. Small Scale Embedded System
2. Medium Scale Embedded System
3. Sophisticated Embedded System

Small Scale Embedded System

- Single 8 bit or 16bit Microcontroller
- Little hardware and software complexity
- May be battery operated
- Need to limit power dissipation when system is running continuously
- Usually “C” is used for developing these system
- Programming tools: Editor, Assembler and Cross Assembler

Medium Scale Embedded System

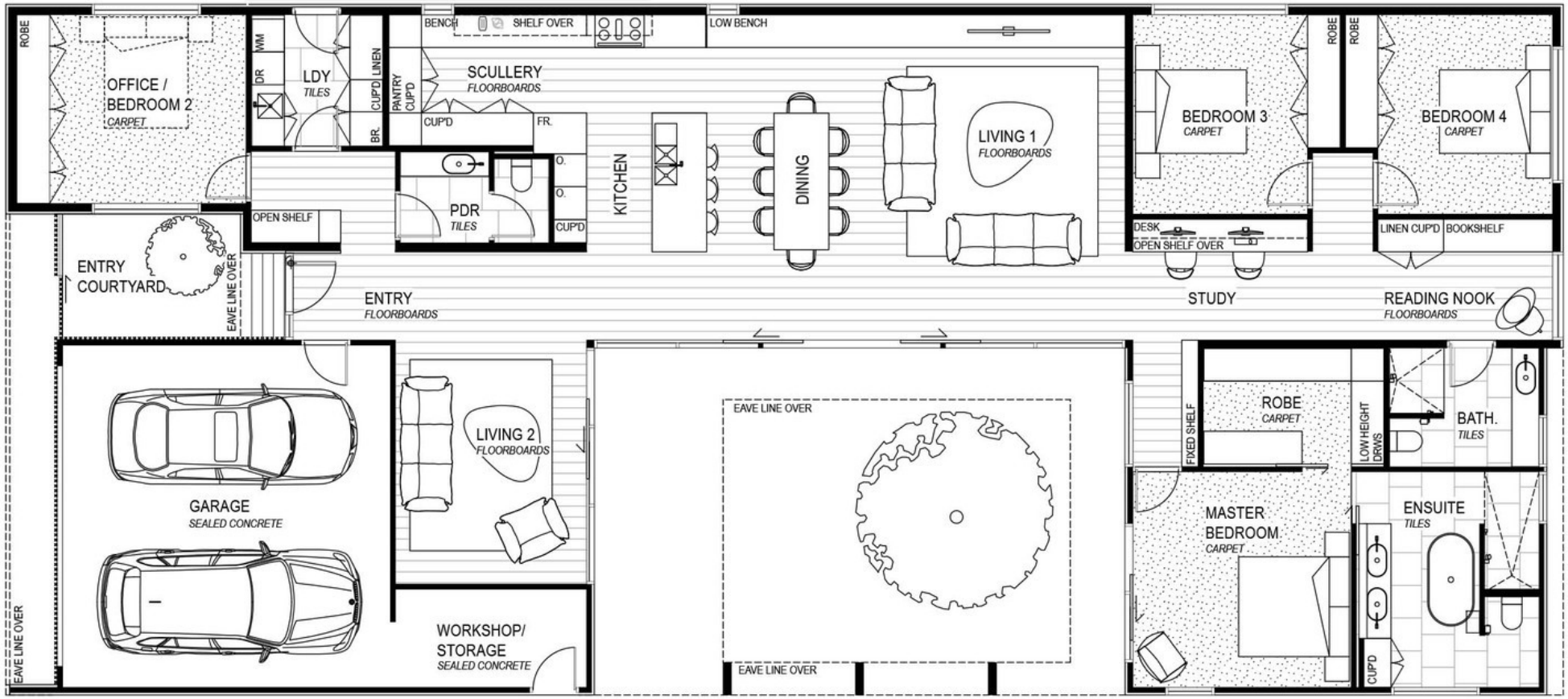
- Single or few 16 or 32 bit microcontrollers or Digital Signal Processors (DSP) or Reduced Instructions Set Computers (RISC).
- Both hardware and software complexity.
- Programming tools: RTOS, Source code Engineering Tool, Simulator, Debugger and Integrated Development Environment (IDE).

Sophisticated Embedded System

- Enormous hardware and software complexity, which may need scalable processor or configurable processor and programming logic arrays.
- Constrained by the processing speed available in their hardware units.
- Programming Tools: For these systems may not be readily available at a reasonable cost or may not be available at all. A compiler or retargetable compiler might have to be developed for this.

Computer architecture

What is computer architecture?



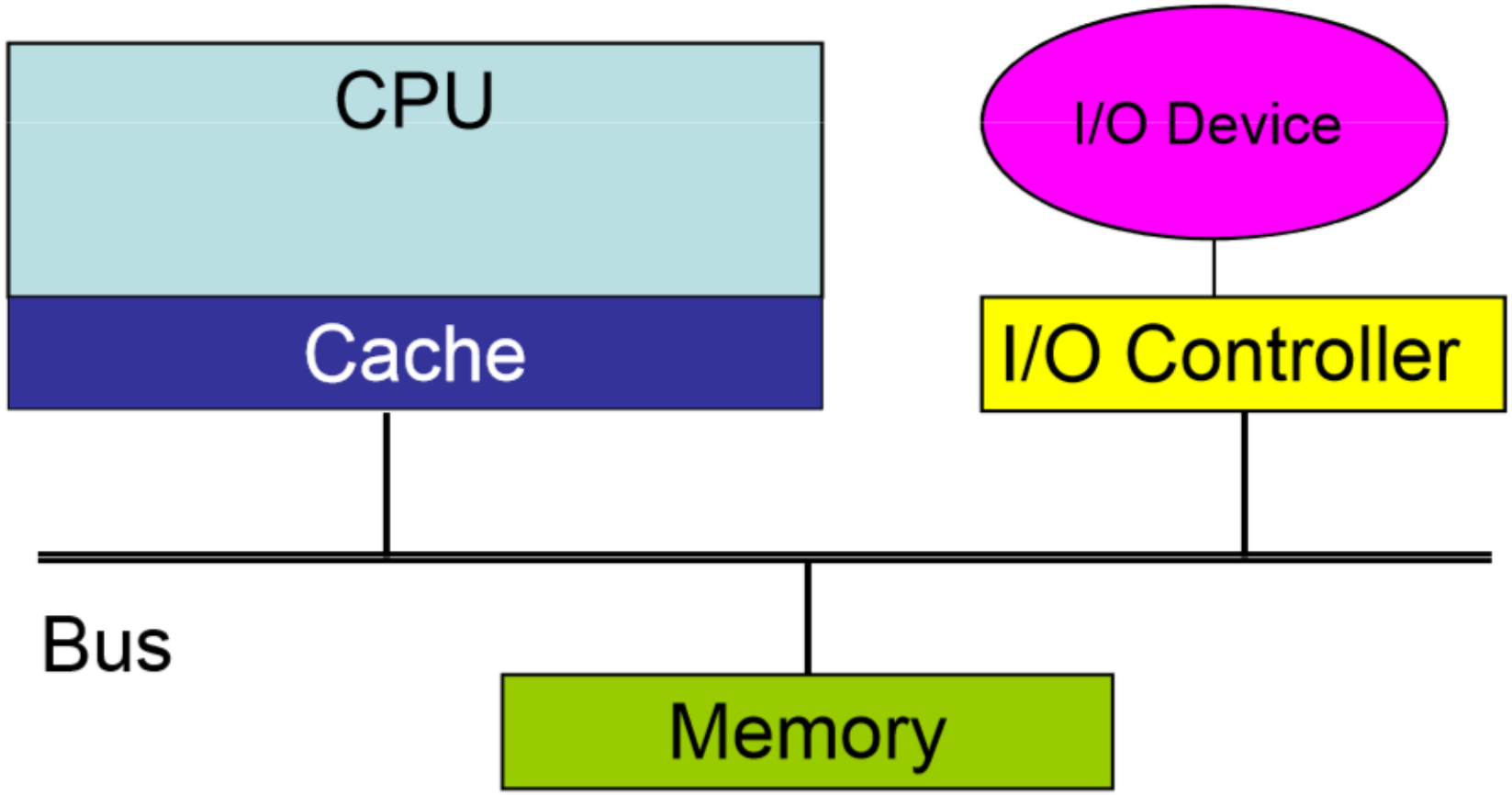
Architecture and organization

- Architecture is the design of the system visible to the assembly level programmer.
  - What instructions
  - How many registers
  - Memory addressing scheme
- Organization is how the architecture is implemented.
  - How much cache memory
  - Microcode or direct hardware
  - Implementation technology

Same architecture, different organization

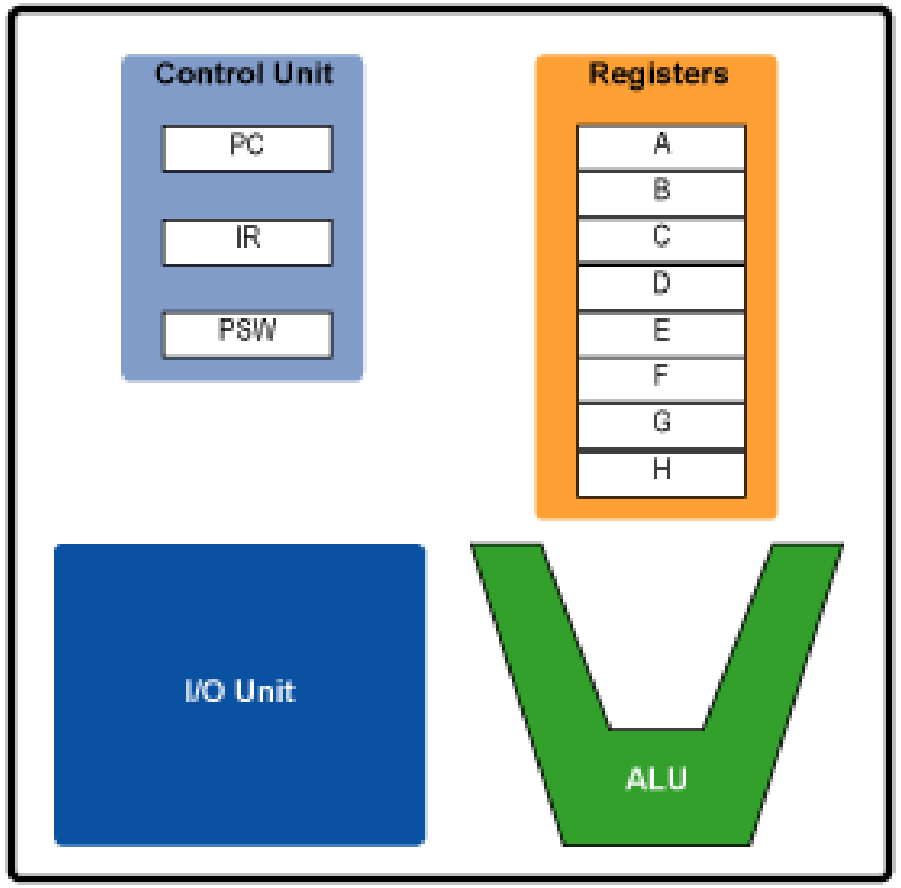
- Almost every program that can run on a Core i3 can run on a Core i5.
- All computers in the Intel Core series have the same architecture.
- Each version of the Intel Core has a different organization or implementation, speed, and price.

Basic computer components



Central Processing Unit

- Contains the control logic that initiates most activities in the computer.
- The Arithmetic Logic Units perform the math and logic calculations.
- Registers contain temporary data values.
- Program Counter contains the address of the next instruction to execute.

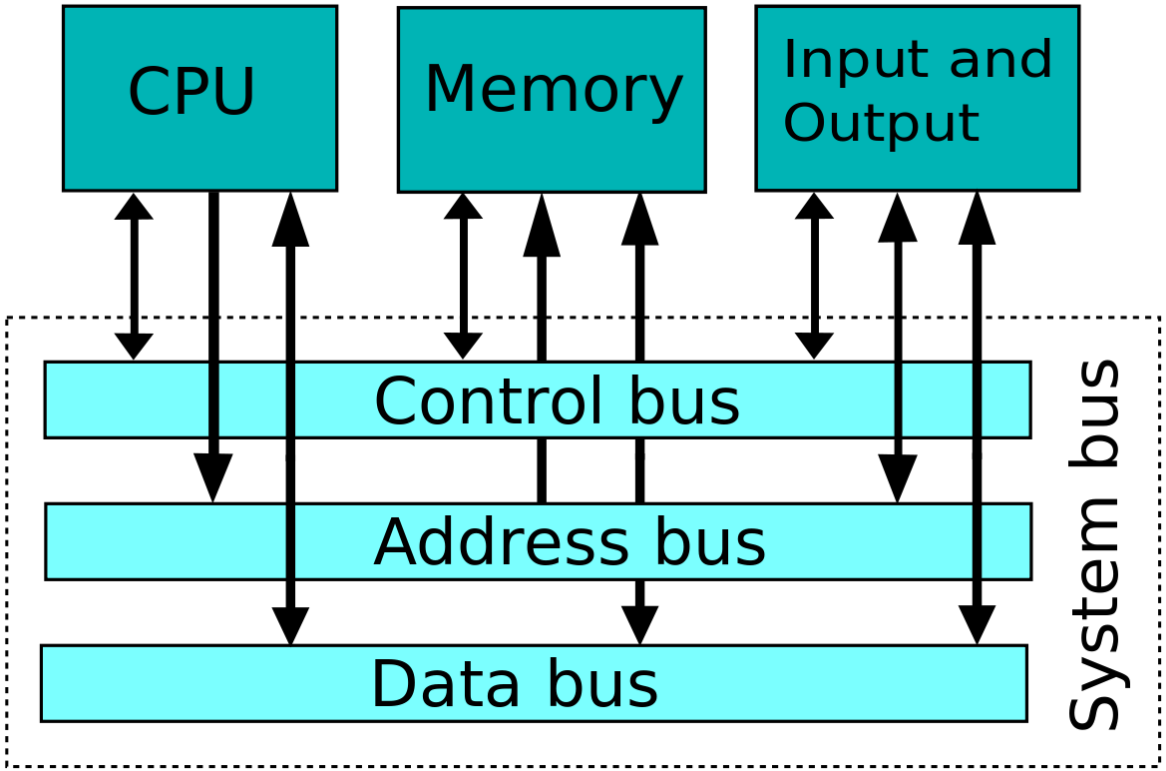


Registers

- The CPU has registers to temporarily hold data being acted upon.
- Different architectures have different number of registers.
- Some registers are available for the user programs to use directly.
- Some registers are used indirectly (such as the program counter).
- Some registers are used only by the operating system (i.e. program status reg)

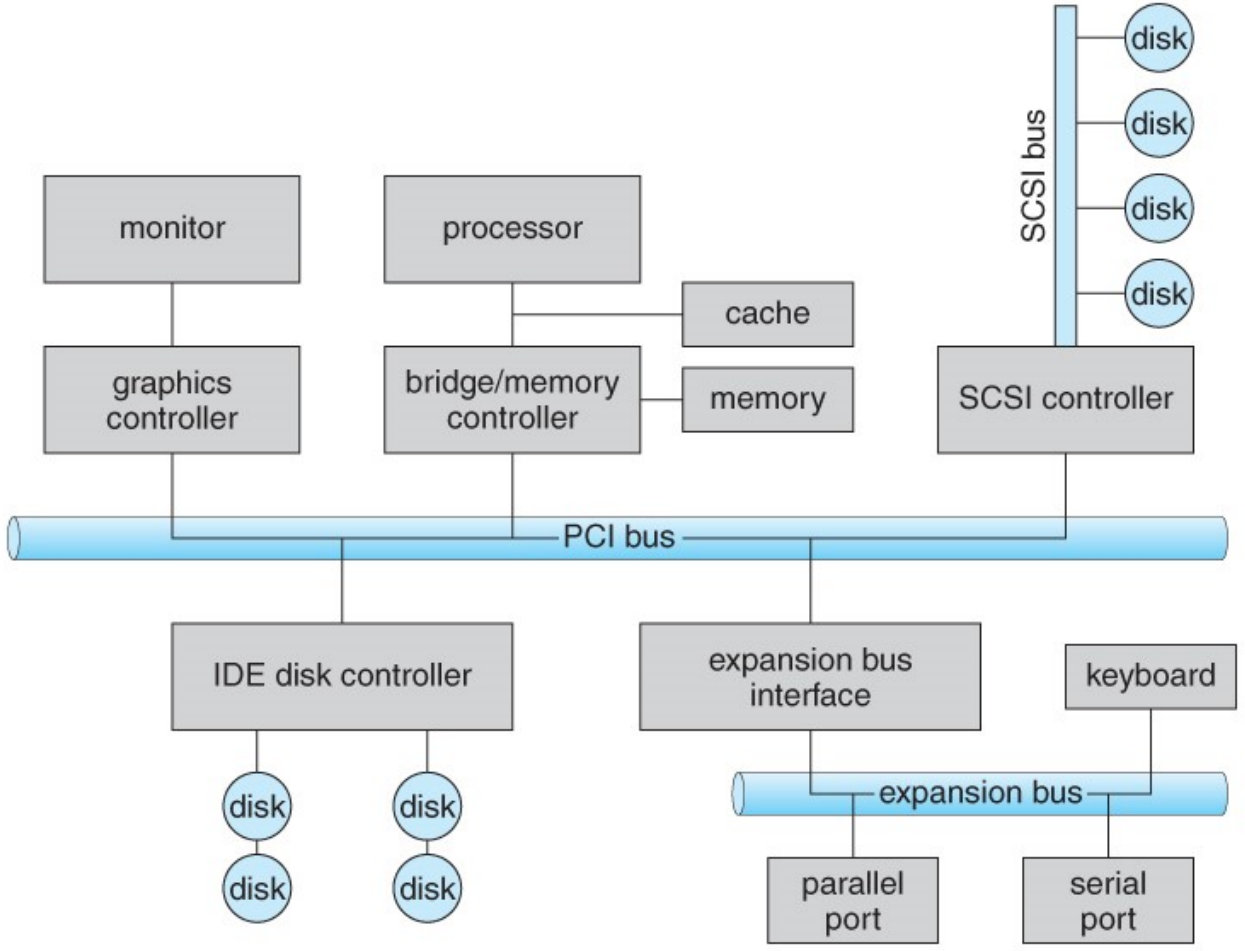
Bus

- The bus is a set of parallel wires that connect the CPU, memory and I/O controllers.
- It has logic (the chipset) to determine who can use the bus at any given instant.
- The width of the bus determines the maximum memory configuration



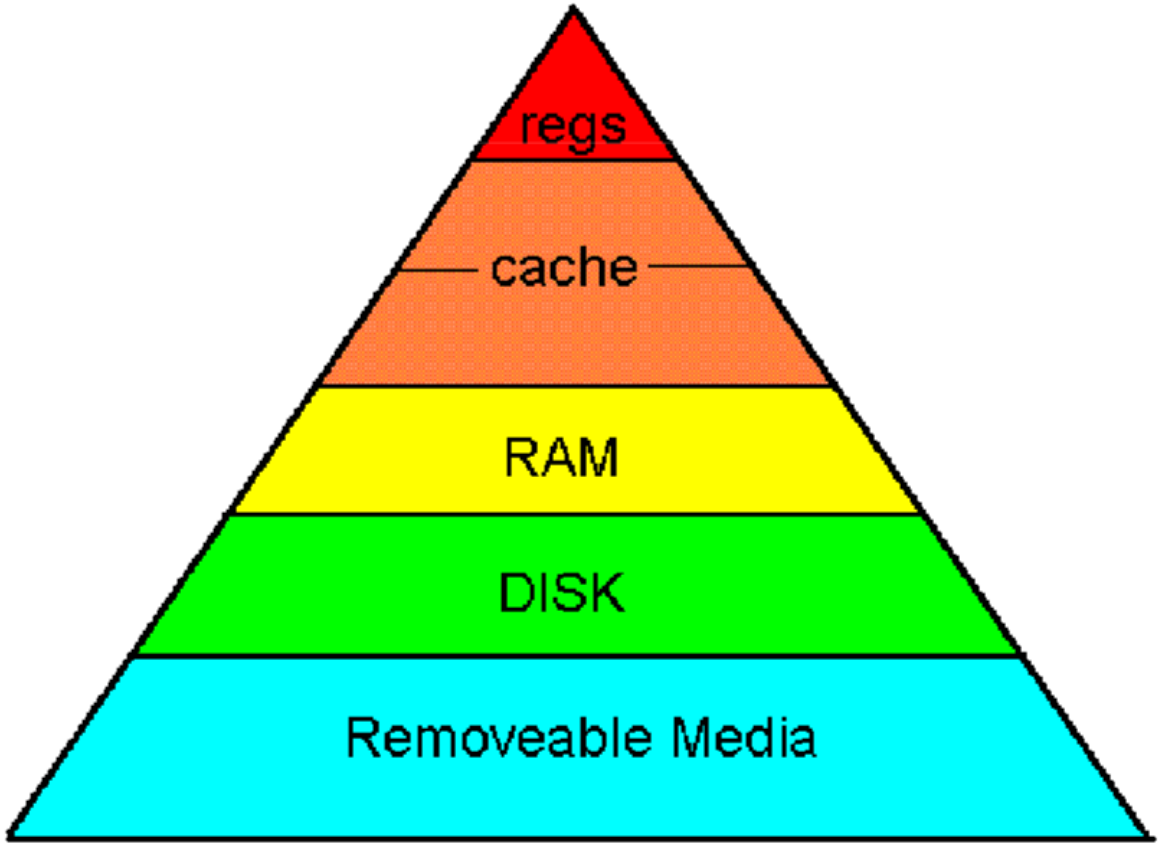
I/O controllers

- Direct the flow of data to and from I/O devices.
- CPU sends a request to the I/O controller to initiate I/O.
- I/O controllers run independently and in parallel with the CPU
- I/O controllers may interrupt the CPU upon completion of request or error.



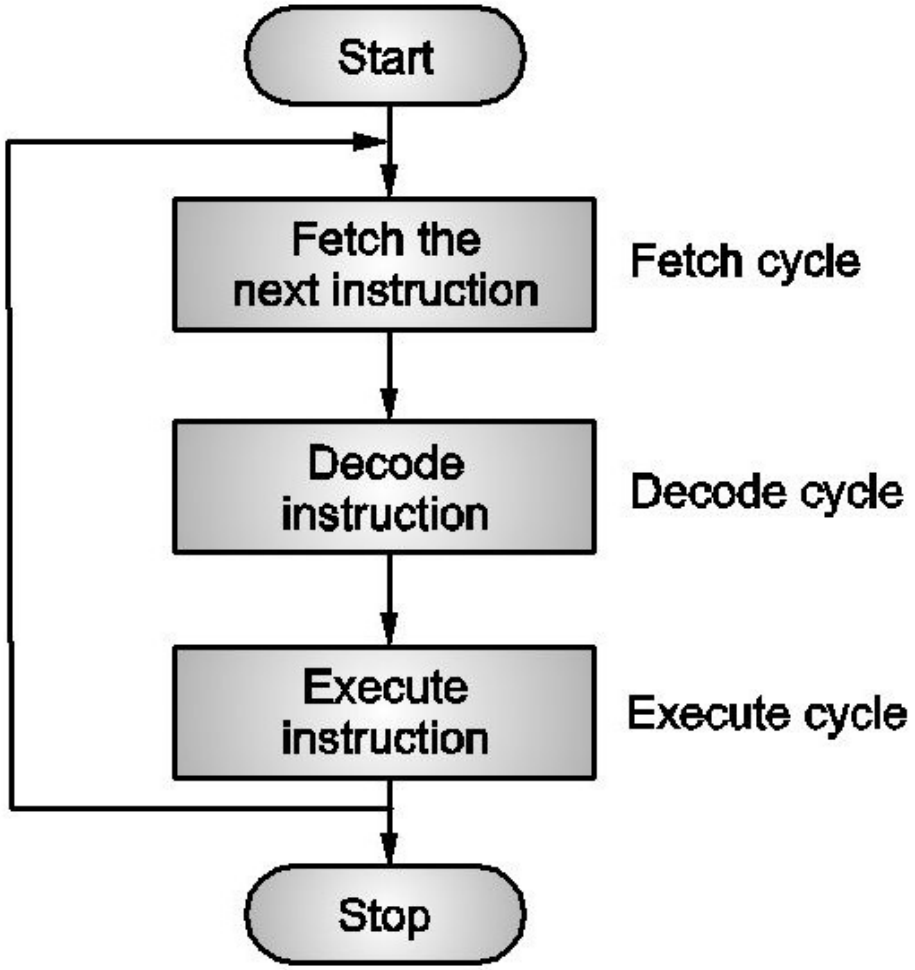
Memory hierarchy

- The internal memory is Random Access Memory (RAM).
- Both data and program instructions are kept in RAM.
- Instructions must be in RAM to be executed.



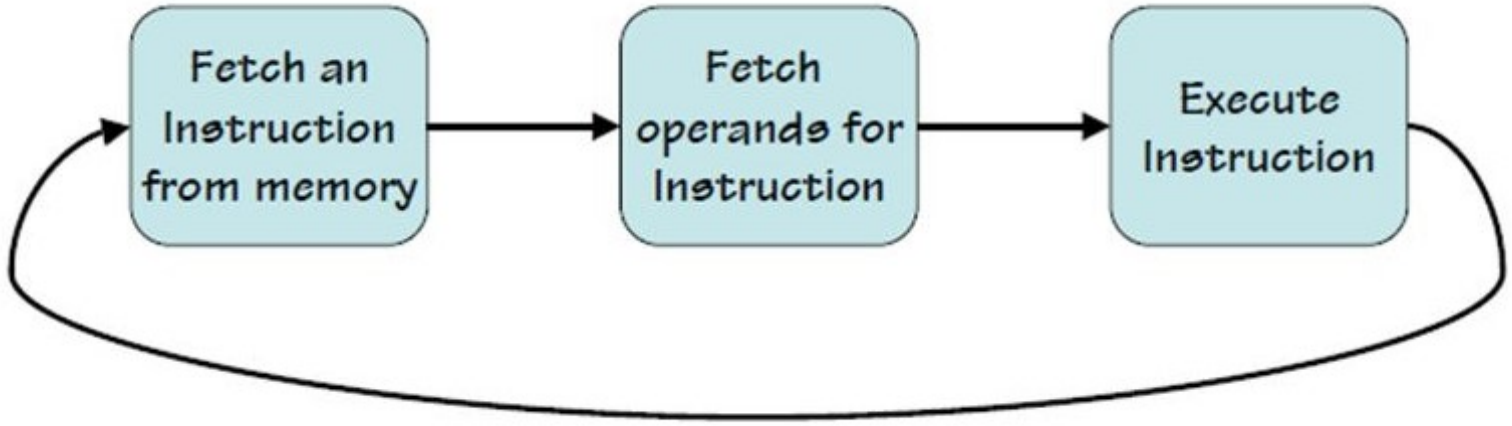
Instruction cycle

- Fetching the instruction from memory and executing the instruction
  1. Fetch the instruction from the memory address in the Program Counter register
  2. Increment the Program Counter
  3. Decode the type of instruction
  4. Fetch the operands
  5. Execute the instruction
  6. Store the results



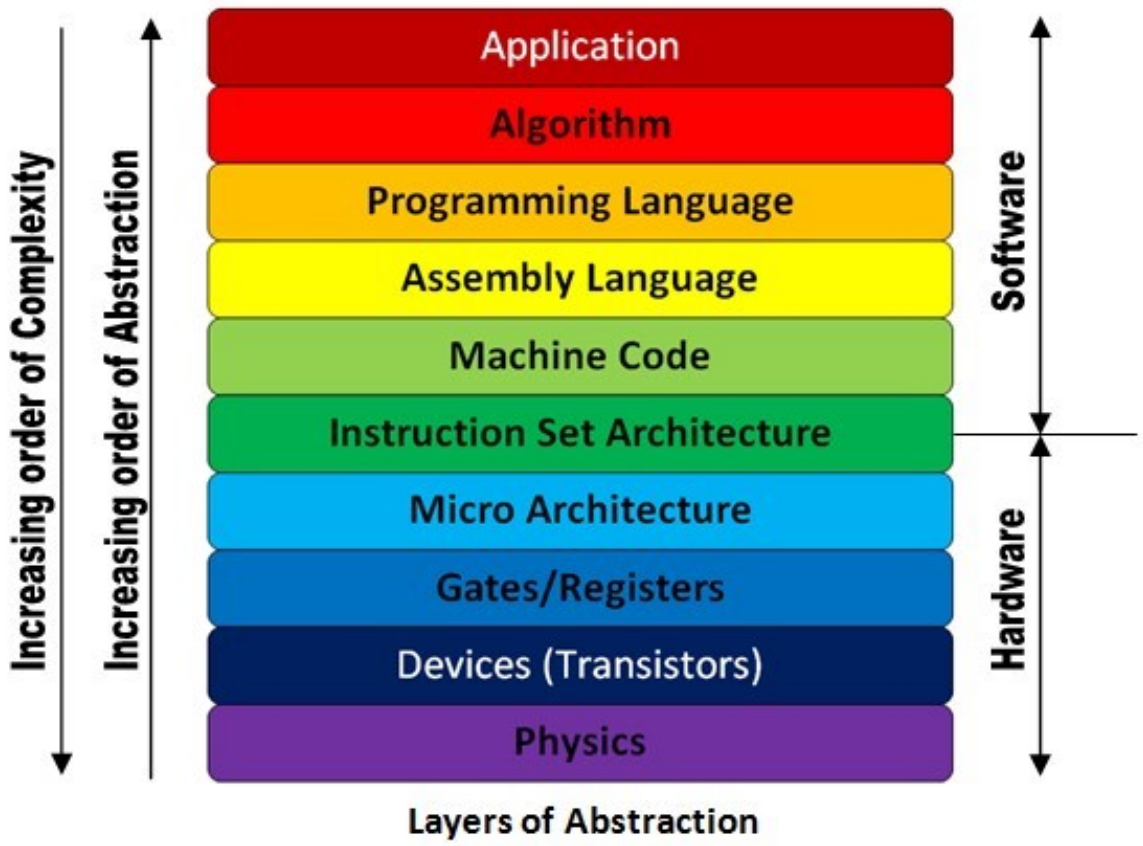
Simple model of execution

- Instruction sequence is determined by a simple conceptual control point.
- Each instruction is completed before the next instruction starts.
- One instruction is executed at a time.



Layers

- You can consider computer operation at many different levels.
  - Applications
  - Middleware
  - High level languages
  - Machine Language
  - Microcode
  - Logic circuits
  - Gates
  - Transistors
  - Silicon structures



Microprocessors & Microcontrollers

Processor

- A Processor is the heart of the Embedded System.
- For an embedded system designer knowledge of microprocessor and microcontroller is a must.

Microprocessor

- A microprocessor is a single chip semi conductor device also which is a computer on chip, but not a complete computer.
- Its CPU contains an ALU, a program counter, a stack pointer, some working register, a clock timing circuit and interrupt circuit on a single chip.
- To make complete micro computer, one must add memory usually ROM and RAM, memory decoder, an oscillator and a number of serial and parallel ports.

Microcontroller

- A microcontroller is a functional computer system-on-a-chip. It contains a processor, memory, and programmable input/output peripherals.
- Microcontrollers include an integrated CPU, memory (a small amount of RAM, program memory, or both) and peripherals capable of input and output.

Various Microcontrollers

- INTEL
  - 8031,8032,8051,8052,8751,8752
- PIC
  - 8-bit PIC16, PIC18,
  - 16-bit DSPIC33 / PIC24,
  - PIC16C7x
- Motorola
  - MC68HC11

Microprocessor vs. Microcontroller

MICROPROCESSOR	MICROCONTROLLER
The functional blocks are ALU, registers, timing & control units	It includes functional blocks of microprocessors & in addition has timer, parallel i/o, RAM, EPROM, ADC & DAC
Bit handling instruction is less, One or two type only	Many type of bit handling instruction
Rapid movements of code and data between external memory & MP	Rapid movements of code and data within MC
It is used for designing general purpose digital computer systems	They are used for designing application specific dedicated systems

Embedded Processor

- Special microprocessors & microcontrollers often called, Embedded processors.
- An embedded processor is used when fast processing fast context-switching & atomic ALU operations are needed.
- Examples : ARM 7, INTEL i960, AMD 29050.

Other Hardware

- Power Source
- Clock Oscillator
- Real Time Clock (RTC)
- Reset Circuit, Power-up Reset and watchdog timer Reset
- Memory
- I/O Ports, I/O Buses
- Interrupt Handler
- DAC and ADC
- LCD and LED Display
- Keypad/Keyboard


...AND THAT'S WHAT WE'LL BE COVERING THIS TERM. ANY QUESTIONS?
YEAH, CAN WE GET AN EXTENSION ON THE HOMEWORK?
AN EXTENSION? IT'S THE FIRST DAY OF CLASS!
UH, SORRY.
CAN WE GET AN EXTENSION ON THE EXTENSION THEN?
LET ME GUESS: PRE-MED, RIGHT?
JORGE CHAM © 2008
WWW.PHDCOMICS.COM