Industrial Embedded Systems
- Design for Harsh Environment -

Dr. Alexander Walsch
alexander.walsch@ge.com

IN2244

WS 2012/13

Technical University Munich (TUM)

Motivation

- What is critical infrastructure?

Motivation

- What is critical infrastructure?
  (from wikipedia) a term used by governments to describe assets that are essential for the functioning of a society and economy. Most commonly associated with the term are facilities for:
  - electricity generation, transmission and distribution
  - oil and gas production, transport and distribution
  - telecommunication
  - water and food supply
  - heating (e.g. natural gas, fuel oil, district heating)
  - public health (hospitals, ambulances)
  - transportation systems (fuel supply, railway network, airports, harbors)
  - financial services (banking, clearing)
  - security services (police, military).

Source: GE O&G

Motivation II

- What does it relate to this lecture?
  Critical infrastructure relies on computer systems, sometimes deeply embedded (depending on the hierarchical control level and the business strategy) – a hidden technology:
  No-one cares if they do their job. A disaster if they fail.
- It is not only important what we do but also how or how well we do
  - process (especially in regulated environment)
  - performance (there are a few but one counts more than others): *reliability*
- Why will it be even more important in the future?
  - More and more electric systems (electrification)
  - Autonomous systems (no human in the loop)

What will you (hopefully) learn about?

- Reliability in embedded systems design
- Functional safety
- Design patterns for hardware
- Design patterns for software
- How some design documents could look like
- V+V terms and its applications

We will not focus on cost, market strategy, etc. We will have a technical view.

Some Books

- D. Patterson, J. Hennessy, Computer Organization and Design: The Hardware/Software Interface
- A. Tanenbaum, Operating Systems Design and Implementation
- D. Smith, Reliability, Maintainability and Risk
- A. Korff, Modellierung von eingebetteten Systemen mit UML und SysML (I believe only in German)
- L. Hatton, Safer C

Lecture Organization

- When and What

22.10.	Introduction
05.11.	Requirements, reliability
19.11.	Safety
03.12.	Example PMU
17.12.	Hardware Architecture
14.01.	Software Architecture
28.01.	Example PMU + V&V
Tbd	Exam (oral or written)

- Homepage:
  http://www6.in.tum.de/Main/TeachingWs2012IndEmbSystems

Part I: Basic Knowledge

Process

Probability

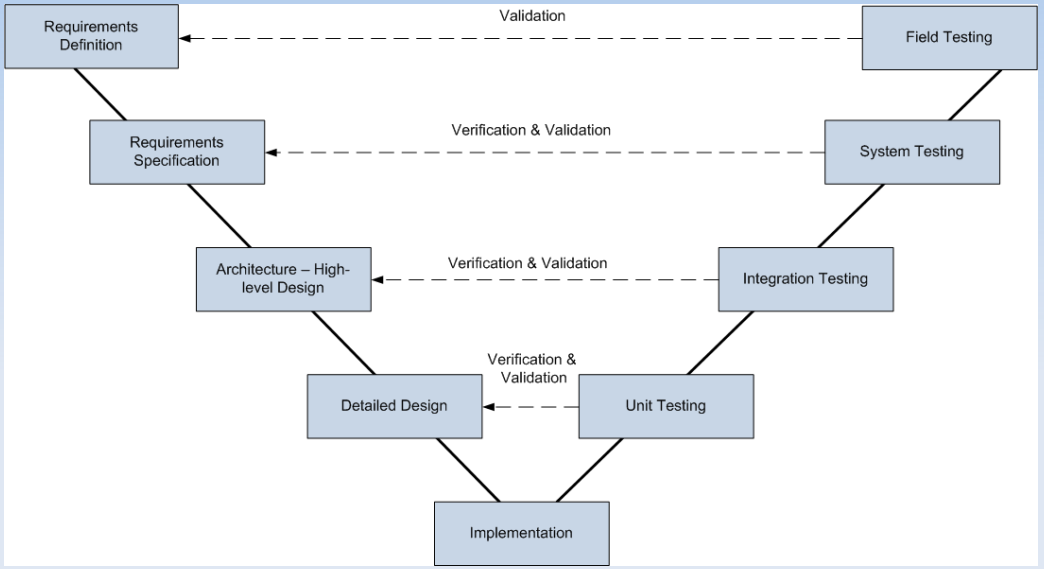
Systems

EE

CS

The V-model
– a design process -

- Modified V-model on the right side (covers field test, y = refinenement, x = time)
- A requirements driven approach
- Design gets more detailed as it proceeds in time
- Tests are defined at the time of design (testability - left wing of V)
- Tests are executed at the V+V stage (pass/fail - right wing of V)



- Iterative process that defines phases for design and testing
- Used for system, HW, SW
- Other processes: waterfall, spiral, agile

Probability
- an example -

A patient is admitted to the hospital and a potentially life-saving drug is administered. The following dialog takes place between the nurse and a concerned relative.

RELATIVE: Nurse, what is the probability that the drug will work?
NURSE: I hope it works, we'll know tomorrow.
RELATIVE: Yes, but what is the probability that it will?
NURSE: Each case is different, we have to wait.
RELATIVE: But let's see, out of a hundred patients that are treated under similar conditions, how many times would you expect it to work?
NURSE (somewhat annoyed): I told you, every person is different, for some it works, for some it doesn't.
RELATIVE (insisting): Then tell me, if you had to bet whether it will work or not, which side of the bet would you take?
NURSE (cheering up for a moment): I'd bet it will work.
RELATIVE (somewhat relieved): OK, now, would you be willing to lose two dollars if it doesn't work, and gain one dollar if it does?
NURSE (exasperated): What a sick thought! You are wasting my time!

Source: Bertsekas, Tsitsiklis Introduction to Probability

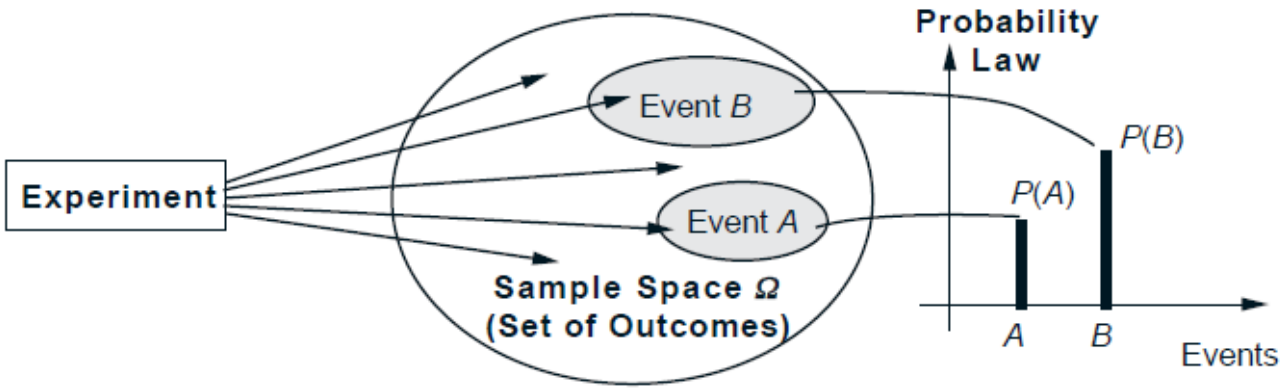
Probability II

- Uncertain (statistical) vs. deterministic outcome
- Think about a percentage (e.g. 1 out of a million) as in terms of frequency of occurrence
- Probabilistic models express scenarios with uncertain outcome and help us to assign a probability to certain sets of outcome (e.g. the probability that a certain component (or a set of components) will fail after a certain time).

Probability III
- probabilistic model -

Elements of a Probabilistic Model

- The **sample space** Ω, which is the set of all possible outcomes of an experiment.
- The **probability law**, which assigns to a set A of possible outcomes (also called an **event**) a nonnegative number $\mathbf{P}(A)$ (called the **probability** of A) that encodes our knowledge or belief about the collective "likelihood" of the elements of A. The probability law must satisfy certain properties to be introduced shortly.



Source: Bertsekas, Tsitsiklis Introduction to Probability

Probability IV
- probability axioms and probability law properties -

Probability Axioms

1. **(Nonnegativity)** $\mathbf{P}(A) \geq 0$, for every event A.
2. **(Additivity)** If A and B are two disjoint events, then the probability of their union satisfies

$$\mathbf{P}(A \cup B) = \mathbf{P}(A) + \mathbf{P}(B).$$

   Furthermore, if the sample space has an infinite number of elements and $A_1, A_2, \ldots$ is a sequence of disjoint events, then the probability of their union satisfies

$$\mathbf{P}(A_1 \cup A_2 \cup \cdots) = \mathbf{P}(A_1) + \mathbf{P}(A_2) + \cdots$$

3. **(Normalization)** The probability of the entire sample space Ω is equal to 1, that is, $\mathbf{P}(\Omega) = 1$.

Some Properties of Probability Laws

Consider a probability law, and let A, B, and C be events.

(a) If $A \subset B$, then $\mathbf{P}(A) \leq \mathbf{P}(B)$.

(b) $\mathbf{P}(A \cup B) = \mathbf{P}(A) + \mathbf{P}(B) - \mathbf{P}(A \cap B)$.

(c) $\mathbf{P}(A \cup B) \leq \mathbf{P}(A) + \mathbf{P}(B)$.

(d) $\mathbf{P}(A \cup B \cup C) = \mathbf{P}(A) + \mathbf{P}(A^c \cap B) + \mathbf{P}(A^c \cap B^c \cap C)$.

Source: Bertsekas, Tsitsiklis Introduction to Probability

Probability V
- conditional probability -

Properties of Conditional Probability

- The conditional probability of an event A, given an event B with $\mathbf{P}(B) > 0$, is defined by

$$\mathbf{P}(A \mid B) = \frac{\mathbf{P}(A \cap B)}{\mathbf{P}(B)},$$

  and specifies a new (conditional) probability law on the same sample space Ω. In particular, all known properties of probability laws remain valid for conditional probability laws.

- Conditional probabilities can also be viewed as a probability law on a new universe B, because all of the conditional probability is concentrated on B.

- In the case where the possible outcomes are finitely many and equally likely, we have

$$\mathbf{P}(A \mid B) = \frac{\text{number of elements of } A \cap B}{\text{number of elements of } B}.$$

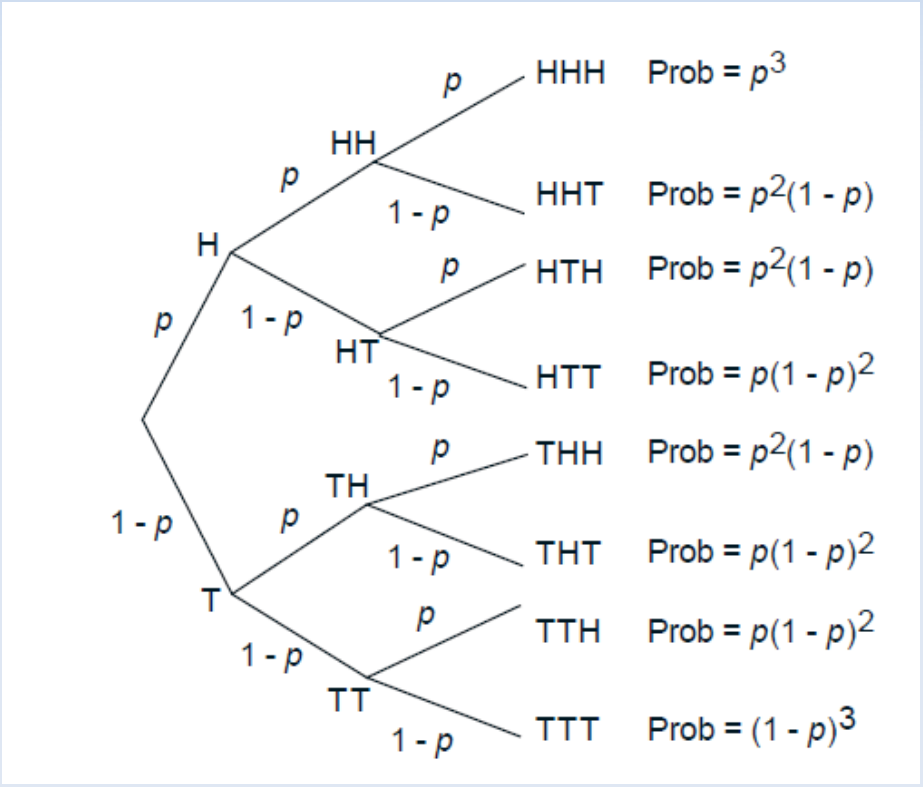
$$\mathbf{P}(A \cap B) = \mathbf{P}(A)\mathbf{P}(B)$$

If A and B are independent

Source: Bertsekas, Tsitsiklis Introduction to Probability

Probability VI
- the Bernoulli experiment -

- An experiment involves a sequence of independent but identical stages -> independent trials
- In the case of two possible outcomes (heads or tails)



Source: Bertsekas, Tsitsiklis
Introduction to Probability

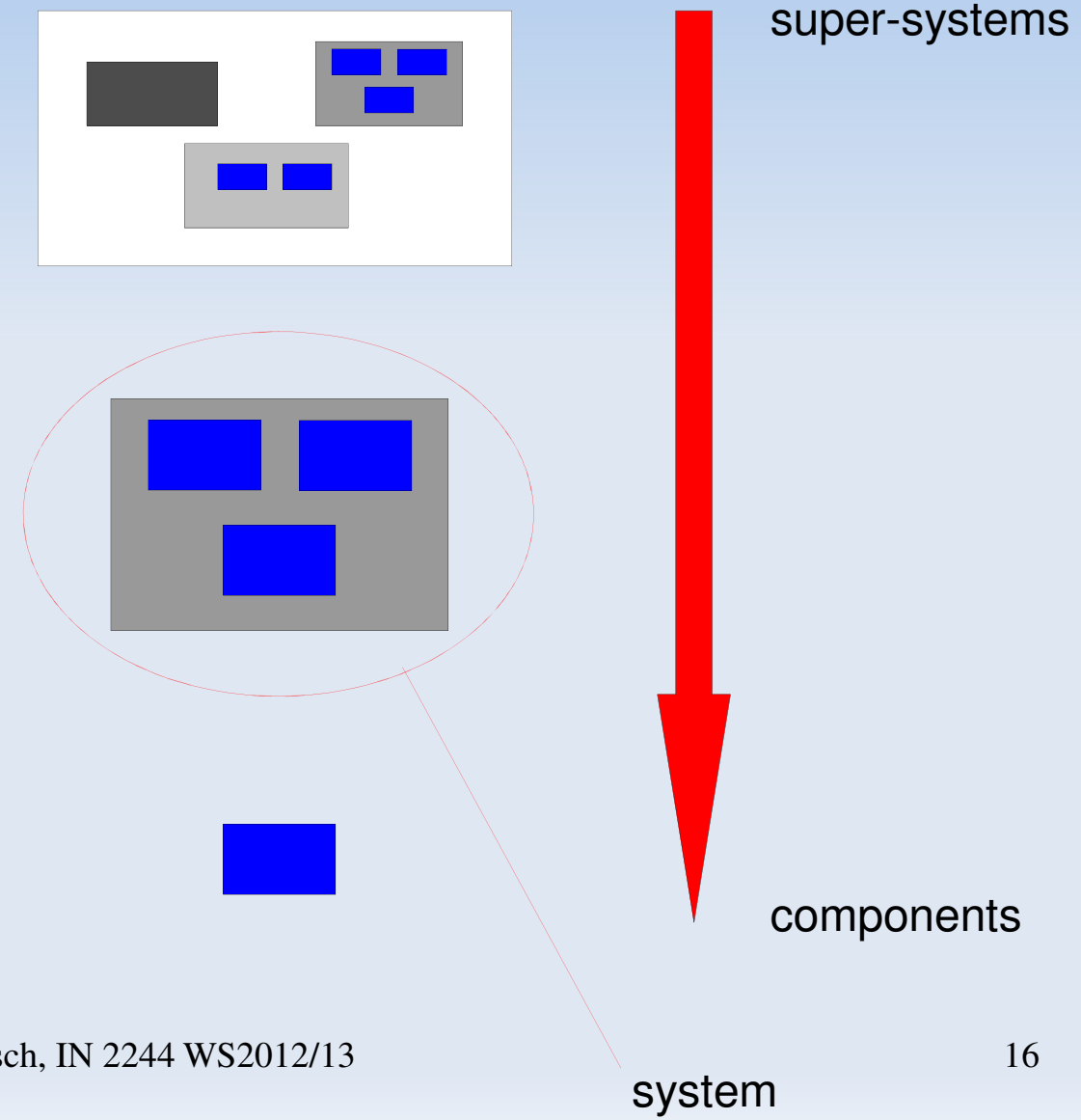
$$p(k) = \binom{n}{k} p^k (1-p)^{n-k},$$

where

$$\binom{n}{k} = \text{number of distinct } n\text{-toss sequences that contain } k \text{ heads.}$$

Systems I
- what is a system? -

- System border
- System components
- System function



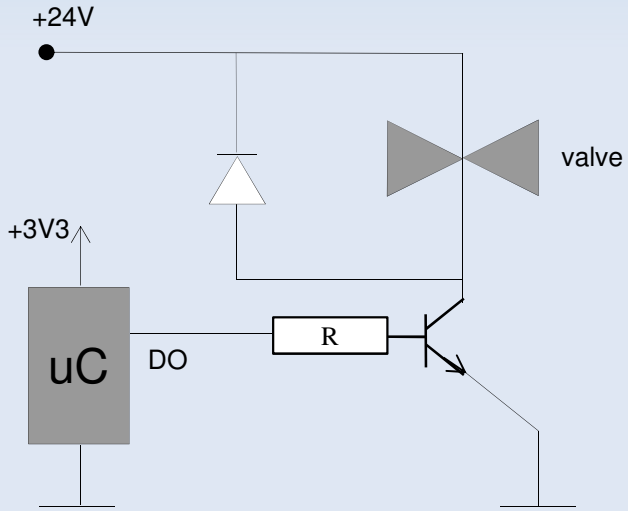
System II
- classification -

- Control systems – open loop/closed loop
- Monitoring systems – monitoring and diagnostics
- Protection systems – safety functions
- Combination of the above

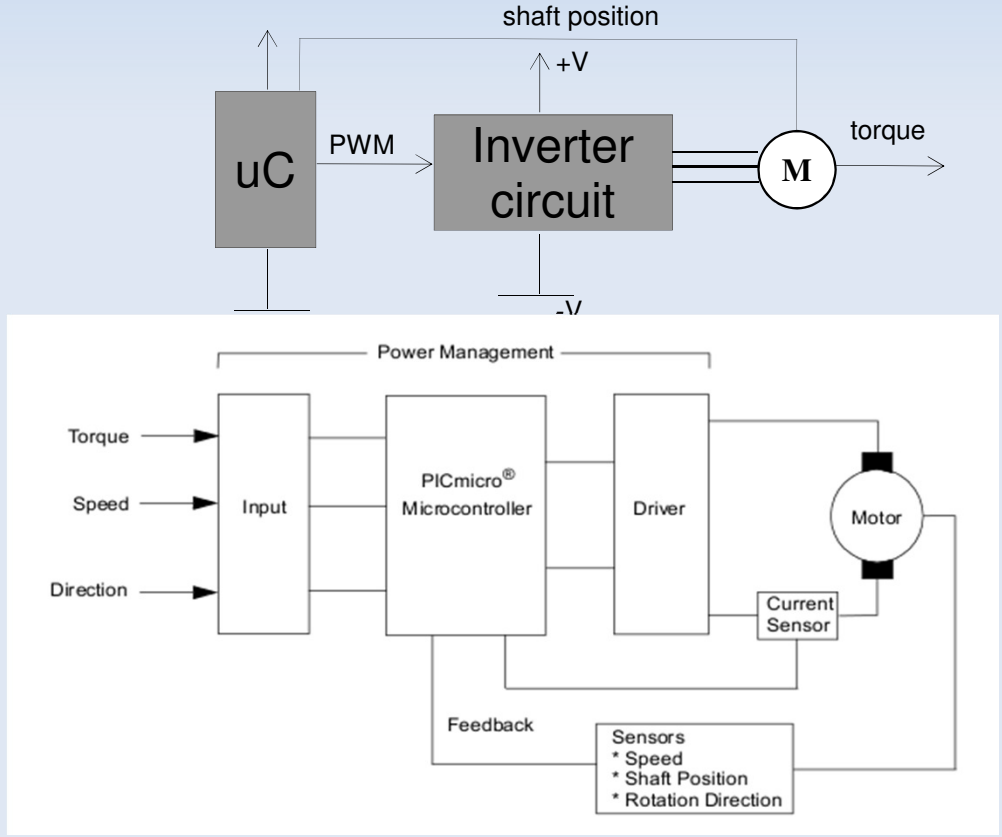
Systems II
- control systems -

- Control systems – open loop and closed loop

open loop: solenoid valve



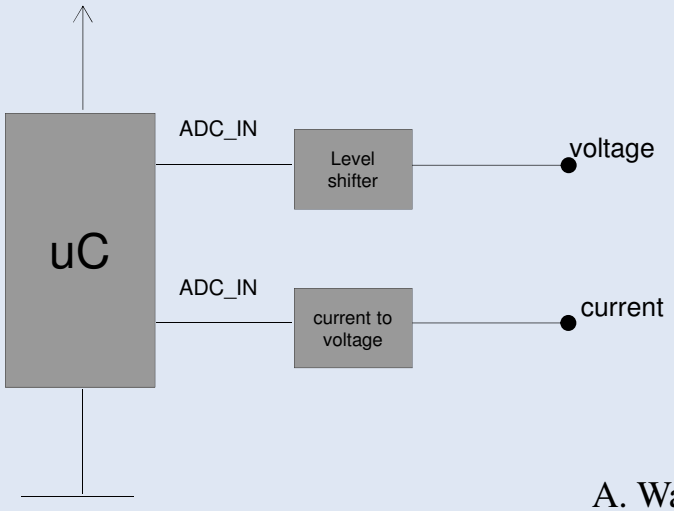
closed loop: BLDC motor



Source: Microchip AN894

Systems III
- monitoring systems -

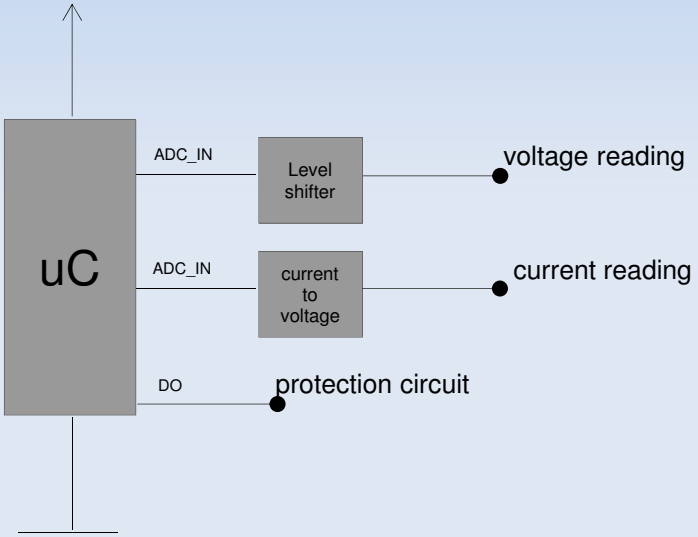
- Monitoring systems – monitoring and diagnostics
- Monitoring systems are used to collect system data (on-board, off-board) for online or offline diagnostics (RM&D, CBM).
- analysis in case of remote systems (”flight recorder”).



- Temperature
- Humidity
- Pressure
- Currents
- Voltages

Systems IV
- protection systems -

- Protection systems – safety functions



Source: VW

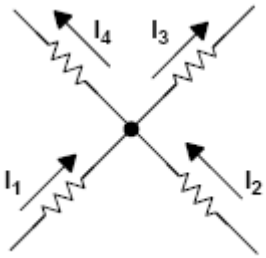
- Protection systems have the sole purpose to put the system into a safe state upon fault detection (fail safe). The safe state needs to be maintained until a clearance operation has been carried out (e.g. safety chains).

Systems V
- environmental constraints -

- Important environmental constraints:
  - Temperature (e.g. -40 to +85 °C ambient)
  - Electromagnetic compatibility – EMC (conductive, inductive, capacitive, radiative coupling)
  - Shock (e.g. 2000 g)
  - Vibration (displacement, velocity, acceleration)
  - More exotic once (rad hard etc.)

Electrical Engineering I
- analog electronics -

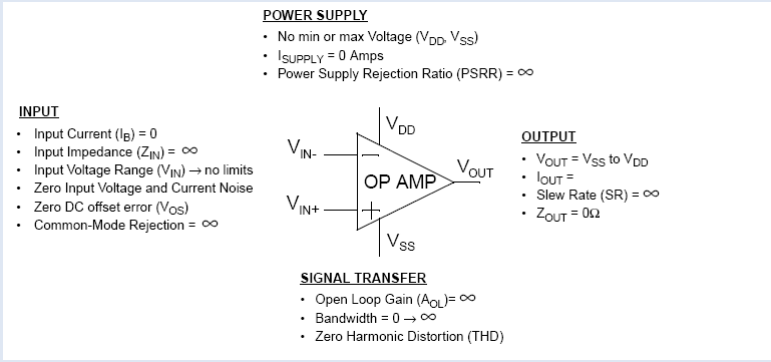
- An analog frontend to a digital computer must be able to provide elements for signal conditioning or math.
- A single component, the operational amplifier, is able of doing that depending on its configuration.
- An (ideal) op amp is an amplifier circuit with differential input and (infinite) large gain.
- Op amp is always used with a component network (R, C) and negative feedback
- Voltage calculations based on Kirchhoff's law, superposition, and „virtual mass" (U_d=0)



$$\sum V_{SOURCES} = \sum V_{DROPS}$$

$$\sum I_{IN} = \sum I_{OUT}$$

$$I_1 + I_2 = I_3 + I_4$$

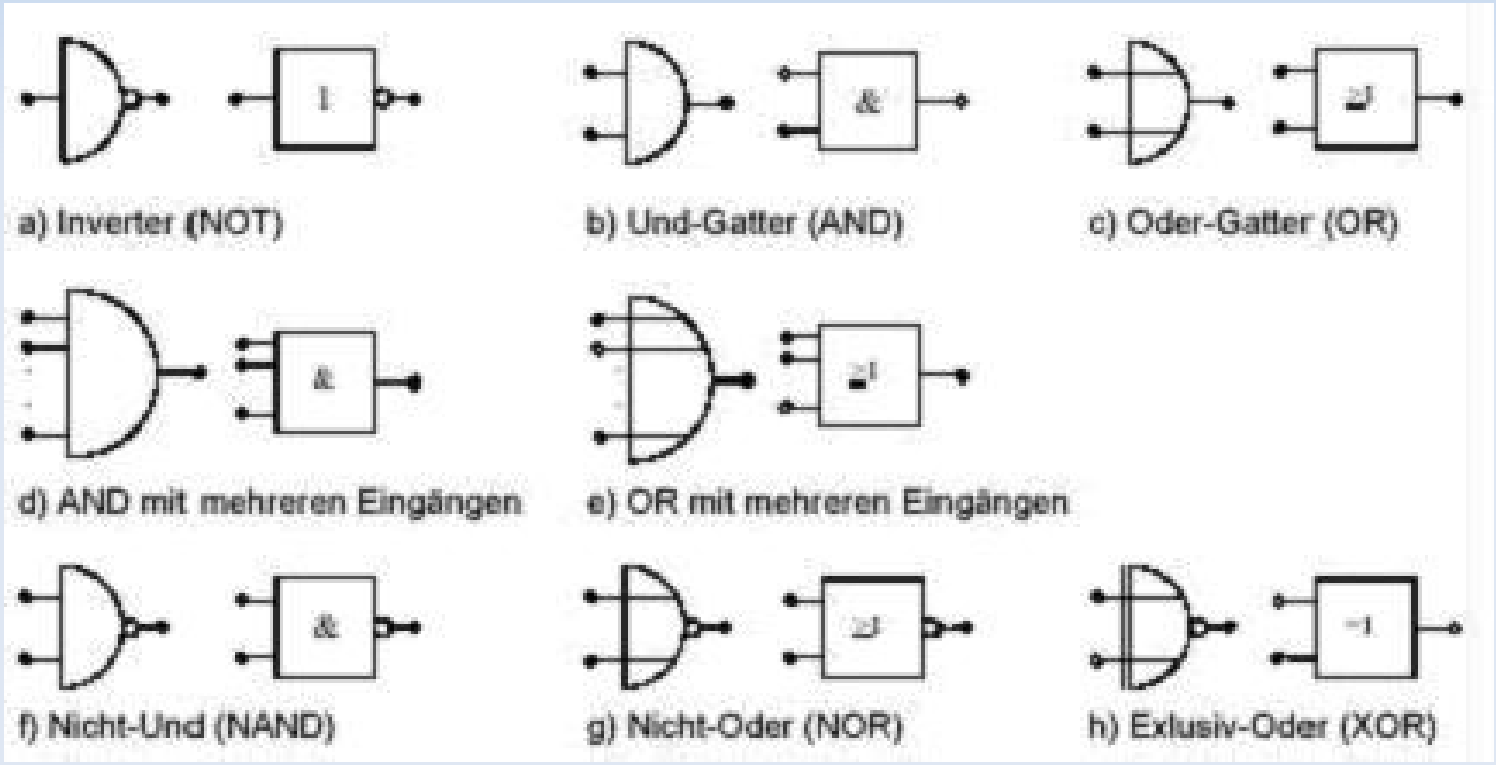


Source:
TI, Op Amps for Everyone

Source:
Microchip, AN722

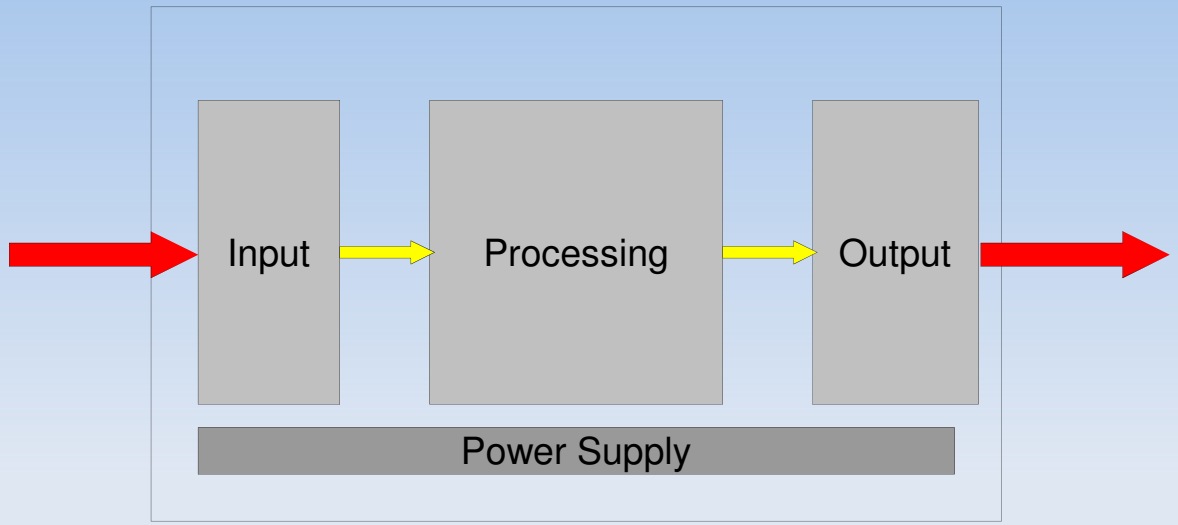
Electrical Engineering II
- digital electronics -

- Combinatorical circuits – the output is a boolean function of the input



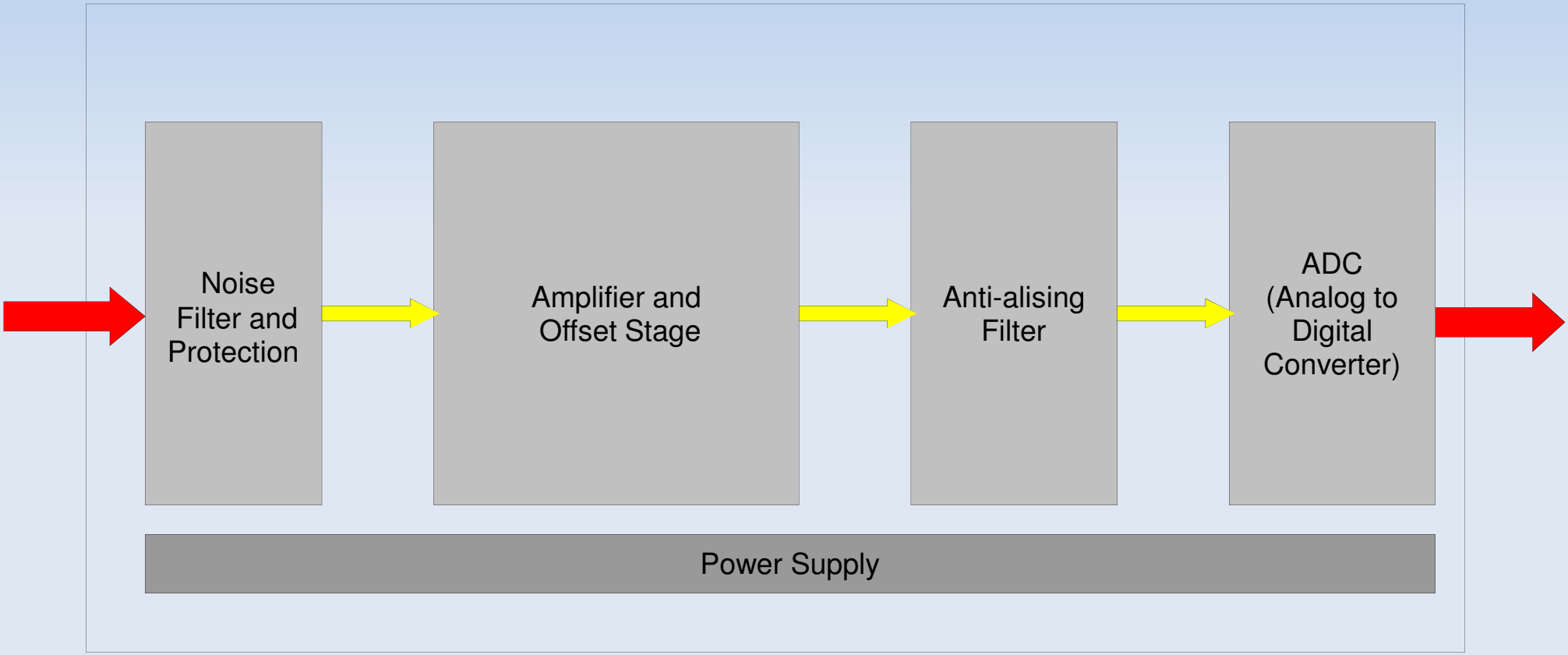
Source: http://www.virtualuniversity.ch/

Computer Science I
- digital computer -



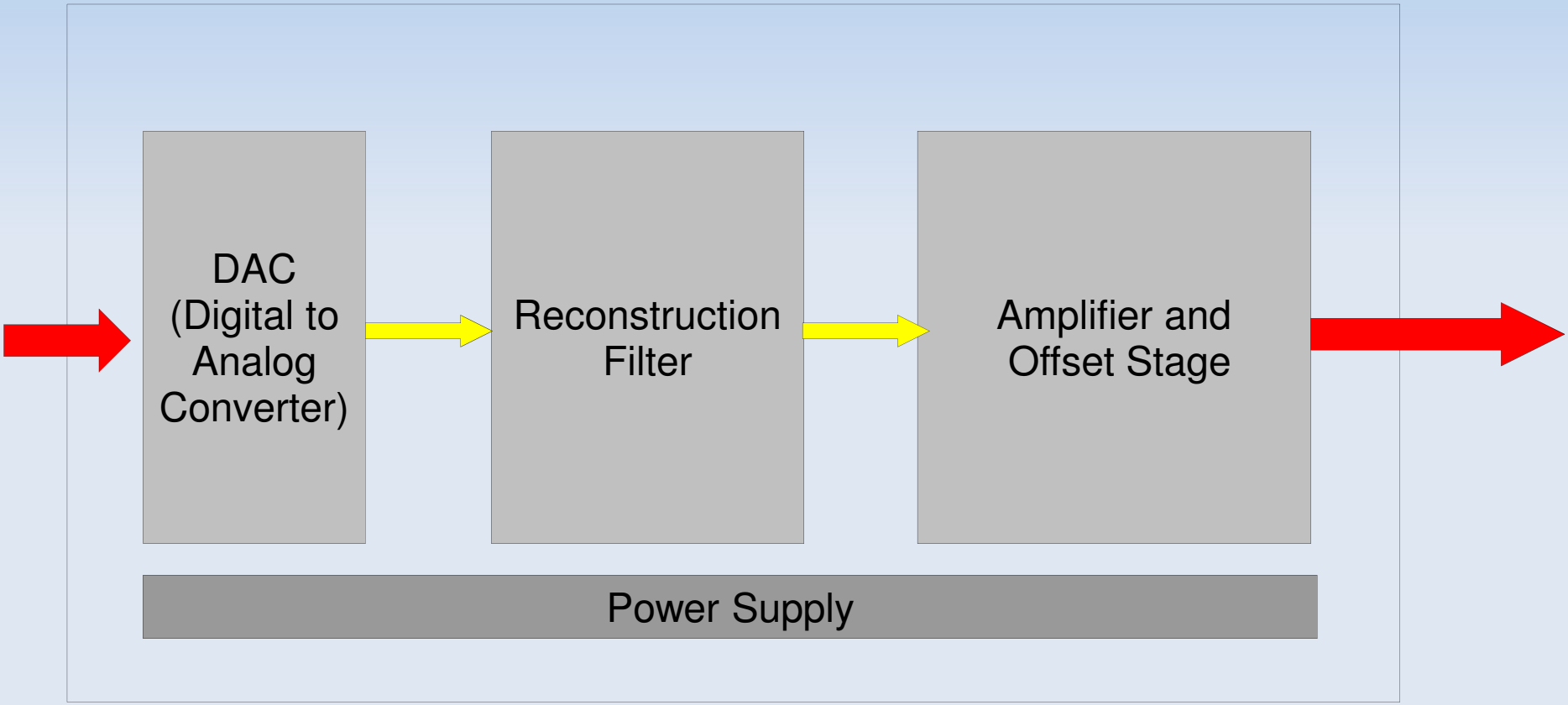
- The digital computer computes a result
  - Computer architecture (e.g. von Neumann) independent of calculation
  - The functionality is translated into an algorithm (= step by step recipe)
  - Real-time performance depends on I/O, instruction set, clock speed
- The digital computer is easily reconfigurable (software, stored program computer).
- The digital computer has (in theory) unlimited precision.

Computer Science II
– analog input -

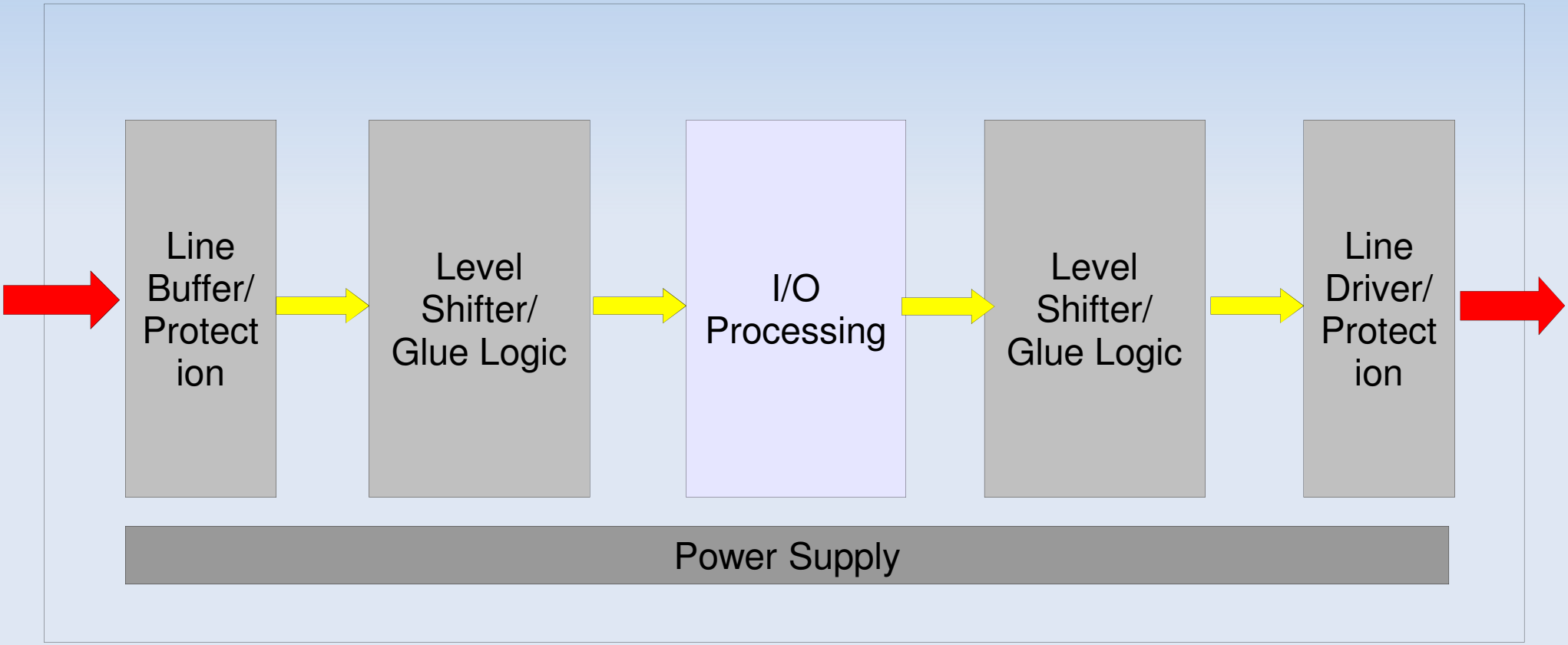


- Nyquist frequency $f_N = 0.5 \times f_S$; f_N is the highest frequency component that must be present at the ADC input → proper reconstruction

Computer Science III
– analog output -



Computer Science IV
- digital input/output -



- Protective circuits can contain both transient, over/under-voltage protection and galvanic isolation.

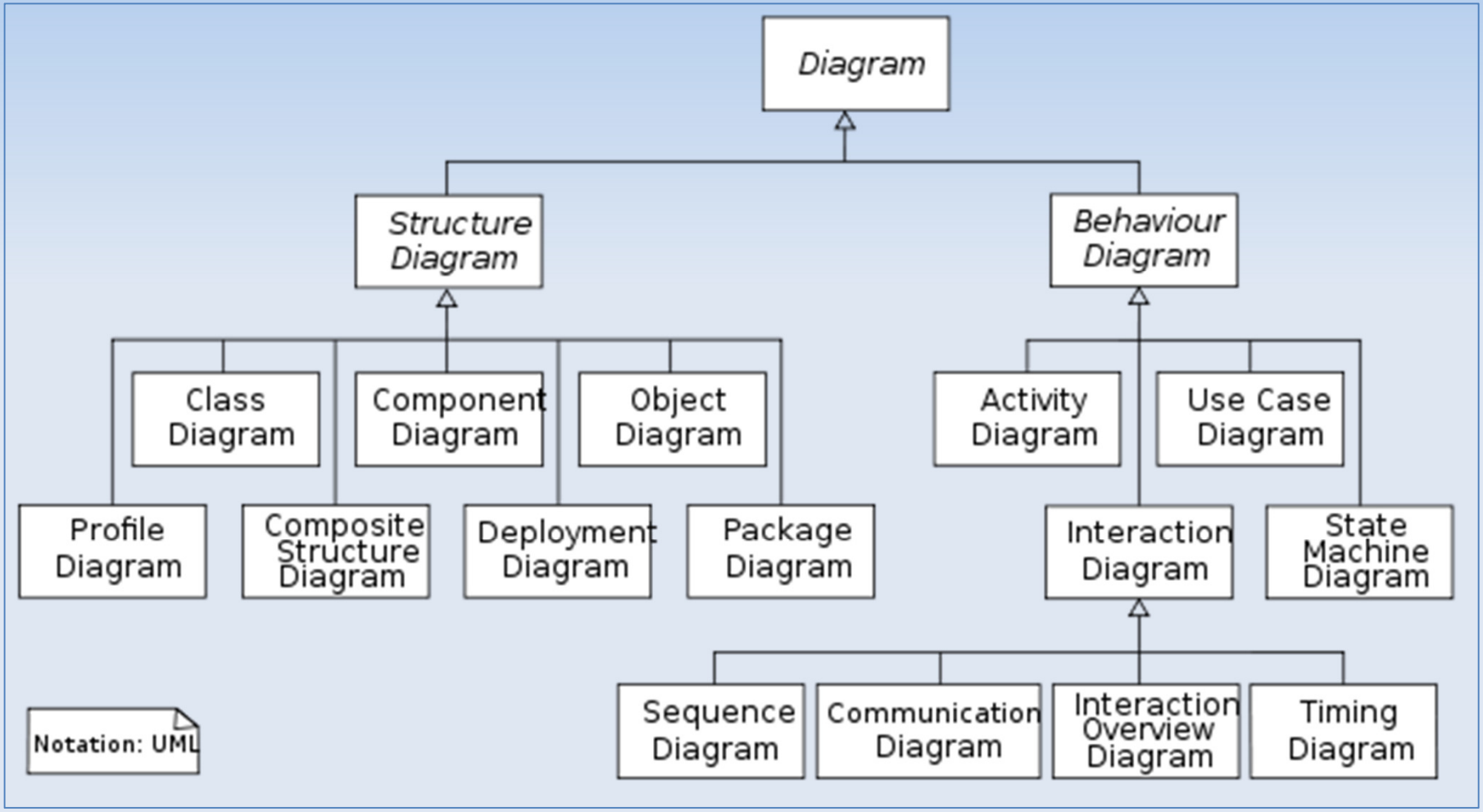
Computer Science V
- definitions -

- Embedded computer systems are inside another device used for running one predetermined application or collection of software (according to this a computer without software is not an embedded computer system).
- Real-time performance requirement means that a function of the embedded computer system has an absolute maximum execution time.
- Hard real-time means that severe damage to assets or loss of life results on violation of the performance requirement
- Soft real-time means that the average time for a function is constrained and loss of performance (or quality) results on violation.

Computer Science VI
- Some More Definitions -

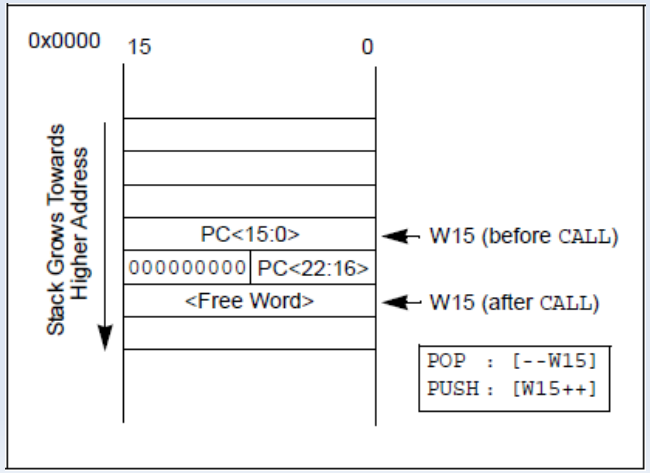
- Cyber-physical systems are networked embedded systems with a connections amongst each other or to the outside world.
- RTOS-based embedded systems run a real-time operating system. A real-time operating system provides support to meet real-time performance requirements.
- Bare-metal embedded systems are not running an RTOS. All requirements are realized without an abstraction layer running on the bare metal.
- Small footprint systems refer to limitions with respect to size, power, memory, etc.

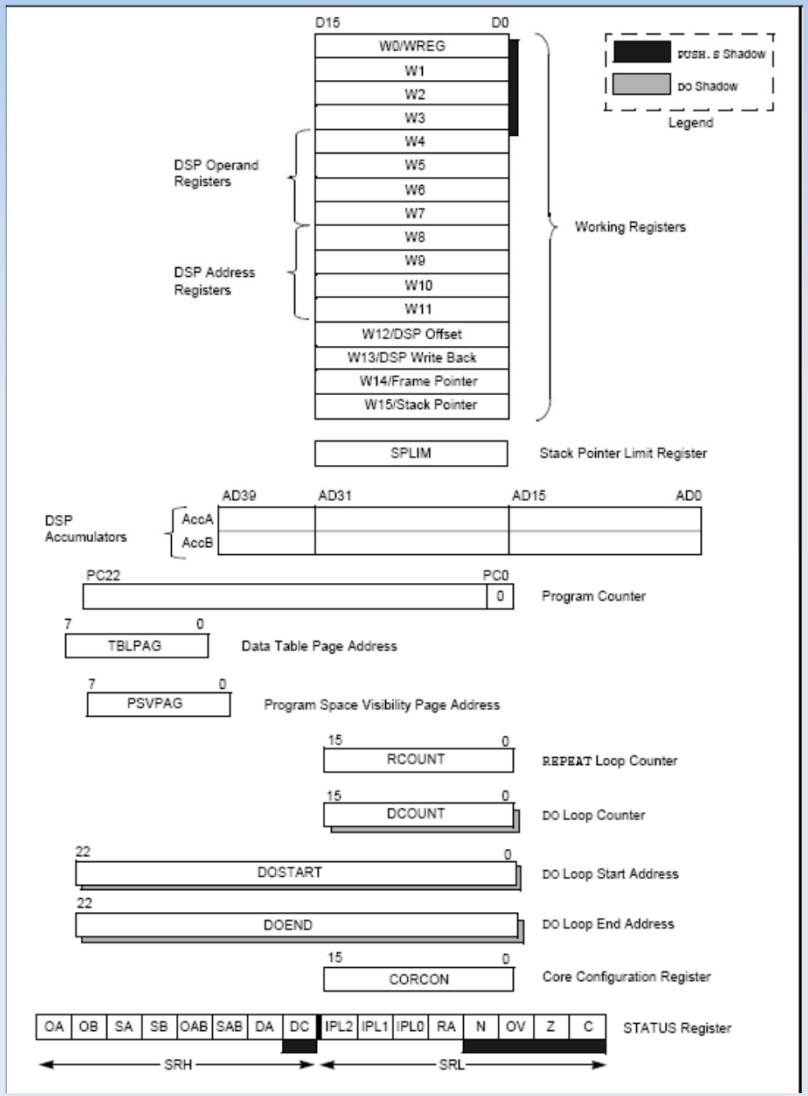
Computer Science VII
- UML-



Computer Science VIII
- programmer's model -

- A programmer's model is an abstract or conceptual view of the structure and operation of a computing system. A microprocessor's programming model shows its register file (e.g. data, address and special registers like stack pointer (here W15)).





Source: microchip.com

Example System - PMU

- PMU: Pressure Measurement Unit
- Will be used as an example virtual technology development
- Measures pressure, temperature compensation, inexpensive, CAN interface
- We will work on the design throughout the lecture

- COTS single board small form factor
- Networking
- Small footprint
- Vendor specific design tools
- Safety-related

Questions ?