1. Mechatronic systems

What is mechatronics?

- Tomizuka: The best practice of SYNTHESIS for engineers in various domains
- It is a “philosophy” of design (development)
- Each design decision should take into account all aspects of the device being designed

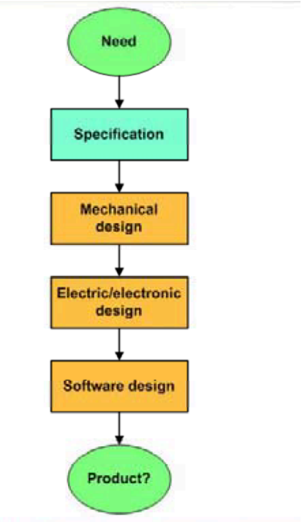
Design process

- Inverse problem
- Often ill-posed
- Usually it is easier to estimate parameters of a designed device than to design a device with given parameters
    - Design decisions are based on heuristics (experience)
    - Design process is iterative (design, check, redesign, check,...)

Design of mechatronic system

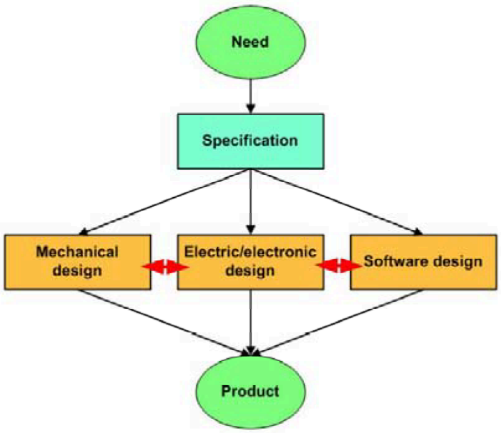
- Mechatronic system is usually to complex to be designed by a single person
- Large group of experts in various domains does not automatically guarantee a success

Traditional approach — sequential design



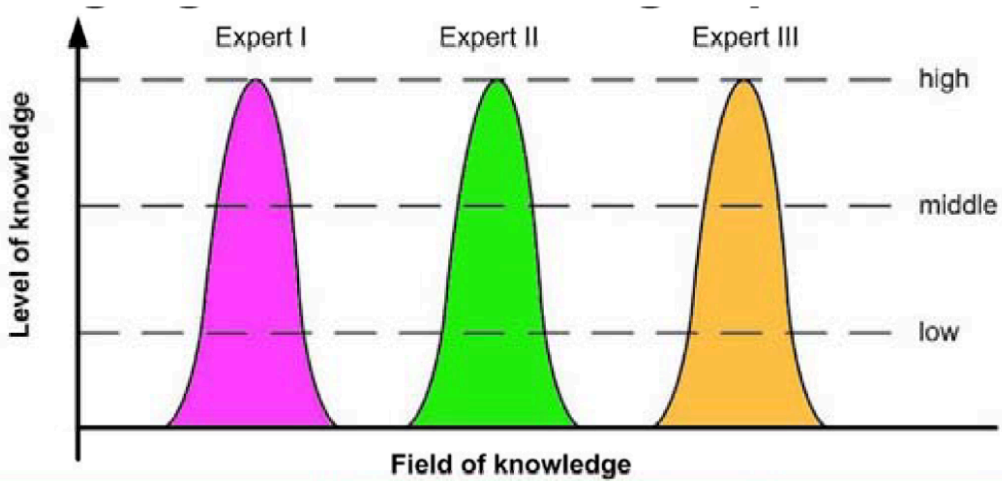
Mechatronic approach — concurrent design

- Teamwork
    - Design experts
    - Mnfg. experts
    - Marketing exp.
    - Management
    - …
- Needed: verification method of components compatibility
    - Virtual prototyping



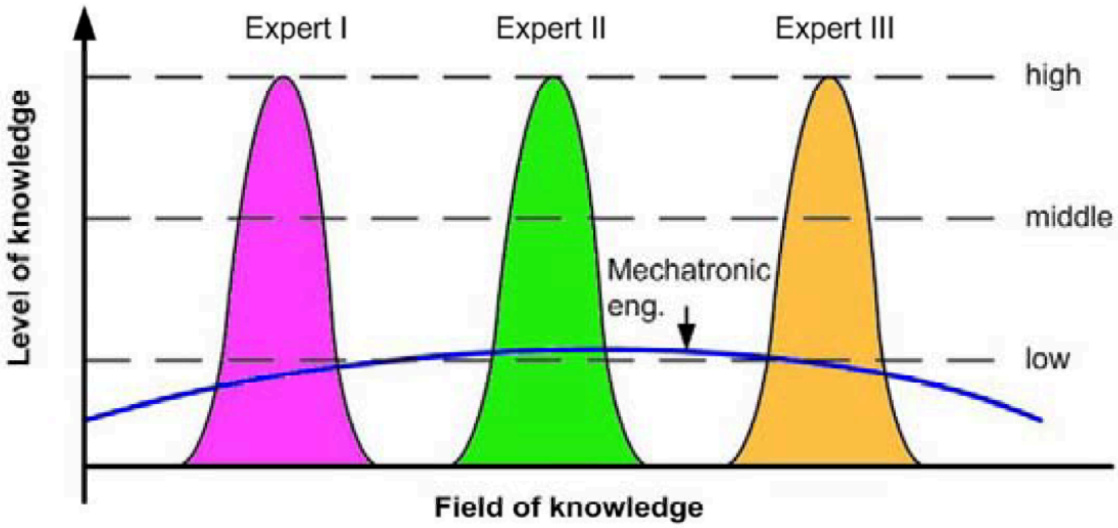
Communication in a team

- Engineer must be not only expert in his domain, but must be able to communicate with others
- Otherwise each talk in his own hermetic language — understanding impossible



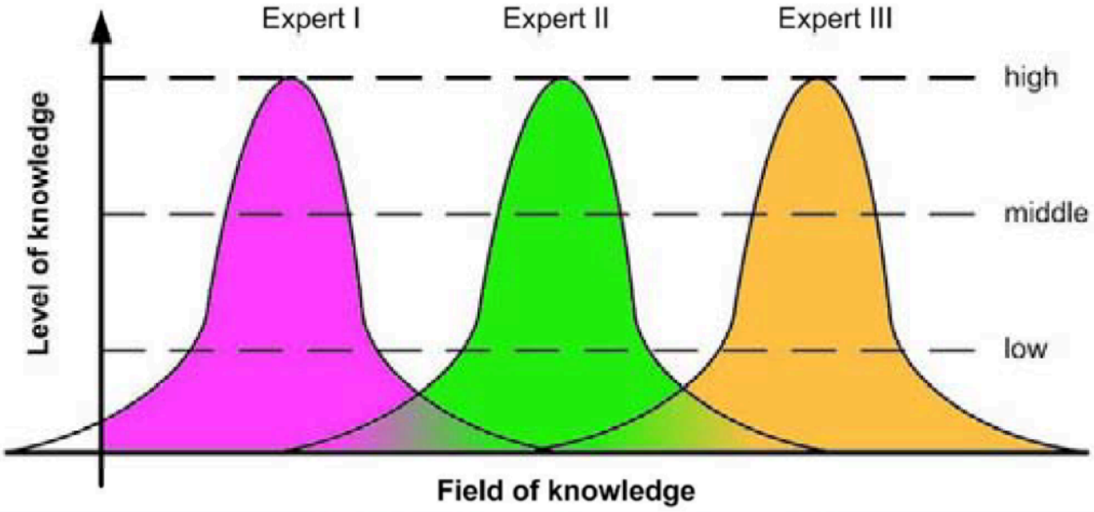
Solution I

- Include into the team mechatronic engineers of wide but shallow knowledge
- Most of mechatronic courses at universities based on this idea
- Result: tendency to replace in teams “narrow” experts with deep knowledge by larger number of “general” mechatronic engineers
- Savings expected
- Projects failed at the first problem requiring advanced knowledge and experience in a particular domain



Solution II

Education of specialists in traditional fields (mechanics, electronics, control,...) expanded by foundations of other domains and an interdisciplinary communication



Mechatronic education in Japan

- In a big company each design engineer is assigned to a large number of different projects one after another as a full member
    - Gains knowledge and experience in all domains of mechatronics
    - Eventually becomes a team leader
- Two conditions must be satisfied:
    - The company big enough to have different projects
    - The design engineer works long enough for the company to gain experience needed in this company (possible only in Japan)

Learning of mechatronics

Mechatronics cannot be learned on a theoretical way but by participating, in realization of complete projects ending with implementation, which is expensive

Benefits of concurrent mechatronic design

- Shortening of the design phase
  - No need to wait for others
- Simplification and acceleration of implementation
  - Careful verification (virtual prototyping) — often the first prototype fulfills requirements and can be sold
- Possibility of flexible realization of system's various function
  - When all components are designed at the same time, realization of functions can be moved easier between domains

Example of flexibility — breaks in a car

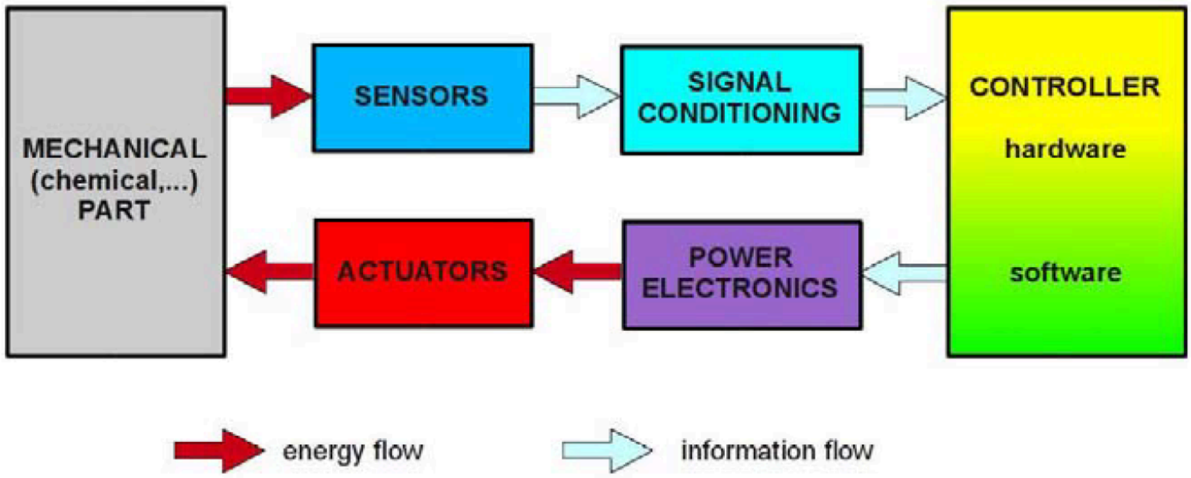
- In a Classical hydraulic brake, the intention of a driver, expressed by pressing the pedal, is transmitted to the brake shoes on a mechanical (hydraulic) way
- In modern, ABS breaks it is transmitted electrically, by a sensor in the pedal and electric motor driving the shoes
  - Application of a µP with a wheel movement sensor allows for improvement of the function by avoiding wheel blocking and shortening of the brake system reaction time

Trends in mechatronic devices

- Decreasing of price/adding new functionalities
- Domination of software in realization of functions and providing quality
- Substitution of mechanical components by electric and electronic ones
- Demand for individualized devices
- Increasing role of HMI as the main differentiating factor
- More rigorous environment regulations
- Increasing number of microsystems as components

2. Structure of mechatronic systems

Typical structure of a mechatronic system



The "main" part of the system

- Mechanism
  - Robot manipulator
- Chemical
  - Process (fertilizers, petrochemical,...)
  - Fuel cells
- Complex
  - Combustion engine
  - Jet engine
  - Power plant turbine (turbogenerator)

Sensors

- Provide information about the state of the "main" part of the system
- Converts energy into information
- Sensors output can be used as:
  - Feedback signal for control
  - Input signal for monitoring/diagnostic systems (eg. machine health)
  - Input signals in human-machine -> Sensing intentions of an operator

Signal conditioning

- Conversion to a form readable by the controller eg. AD conversion
- Filtering
  - Noise
  - Unneeded information
  - Antialiasing
- Amplification
- Electrical matching

- Voltage levels
- Impedance

Actuators

- Supply energy in appropriate form for the “main” part of the system
- Sometimes indirectly (eg. a valve)
- Can be used in HMI
- Converts energy from:
  - Electrical
  - Chemical
  - Mechanical
- Into:
  - Mechanical

Power electronics

- Translates control signals for actuators
- Usually supply energy for actuators
  - In a controlled way
  - Sometimes indirectly (eg. ignition in a combustion engine)
- Converts information into energy
  - Directly into electrical energy

Controller

- Executes algorithms(processes information):
  - Signal processing
    - Control
    - Monitoring (state estimation)
    - Presentation (display)
  - Predefined (unconditional) actions
    - Open-loop control

Controller

- Software
  - Implementation of algorithms
- Hardware
  - For executing software -> Then microprocessor(s) needed
  - For implementing parts of or entire algorithms -> When entire algorithm then software (and microprocessor) not needed

Controller hardware

- Industrial computer
  - Modular
  - Operating system (OS)
    - Real-time (RTOS)
- Embedded controller
  - Microprocessor, microcontroller, DSP
  - FPGA, ASIC
- Fast prototyping hardware
  - Flexible, powerful (computationally)
  - Expensive
  - Different from target hardware

Trends

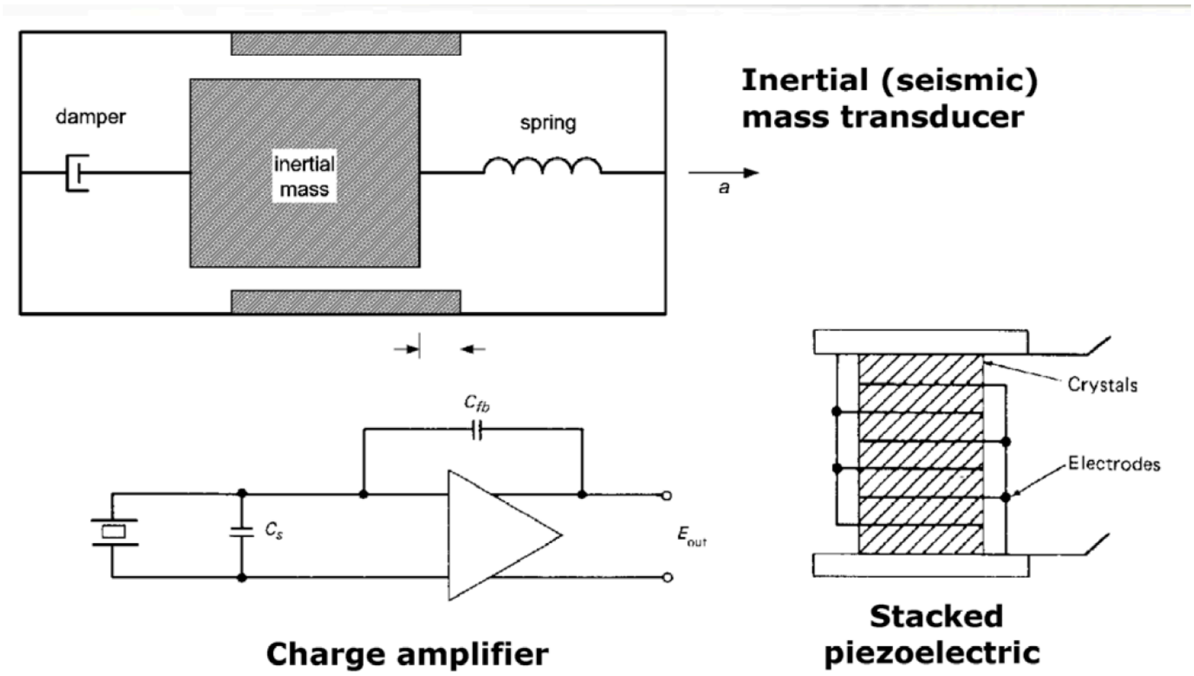
- Substitute energy transmission by information transmission
  - Preferably information in digital form -> “smart” sensors, “smart” actuators
- Supply and convert energy as close as possible to the “main” part of the system -> eg. direct drives
- Move functions of the system from hardware to software
- Cheaper
  - More flexible (customization)
  - More and more feasible

3. Examples of mechatronic systems — inertial sensors

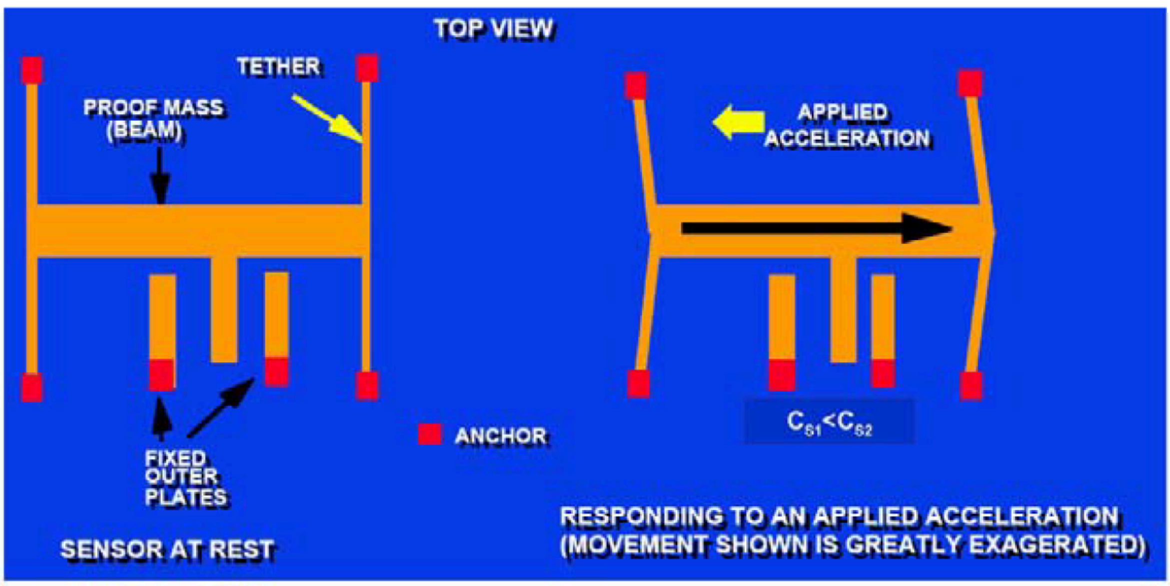
Accelerometer

1. Convert acceleration into directly measurable quantity
2. Convert this quantity into electrical quantity or parameter
3. Process the output - measure the electrical quantity or parameter

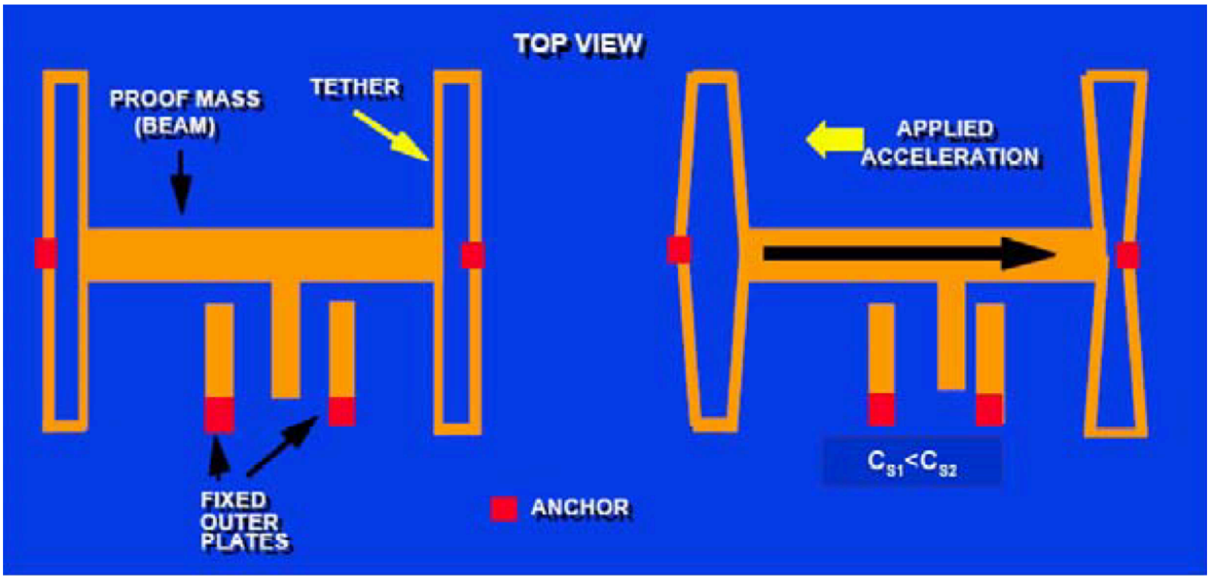
Piezoelectric accelerometer



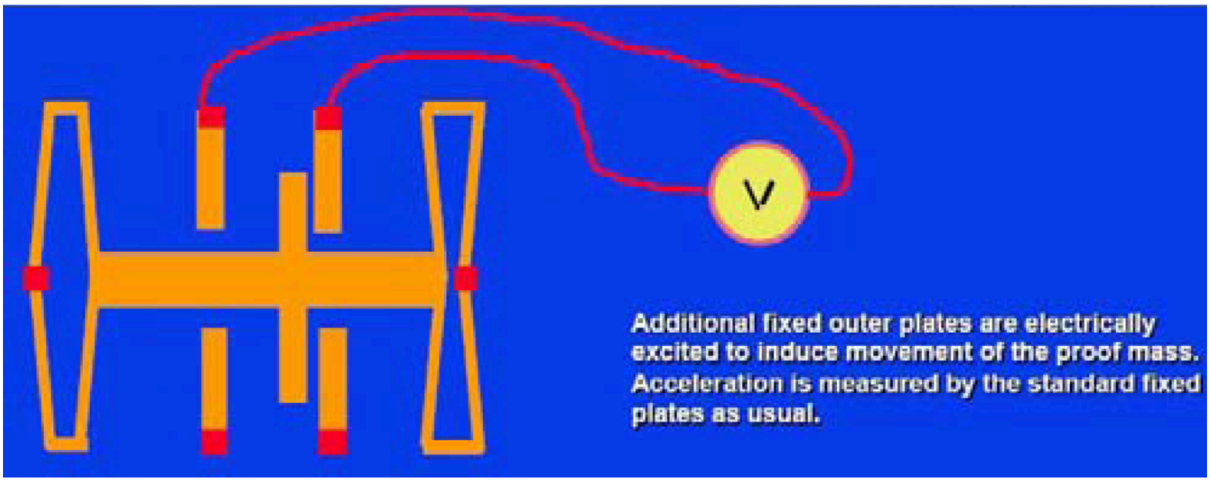
MEMS capacitive accelerometer



MEMS Improved version



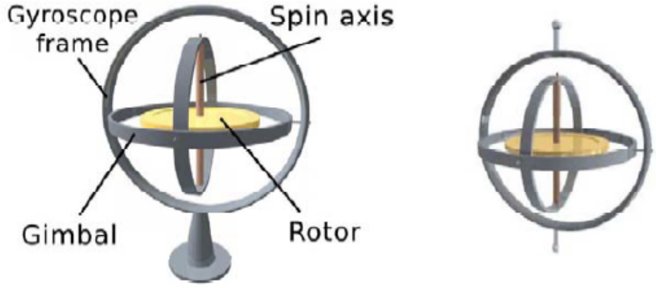
MEMS Accelerometer testing



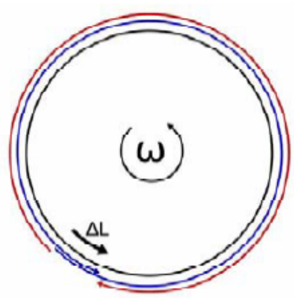
Gyroscope

1. Convert angular velocity into directly measurable quantity
2. 2.Convert this quantity into electrical quantity or parameter
3. 3.Process the output - measure the electrical quantity or parameter

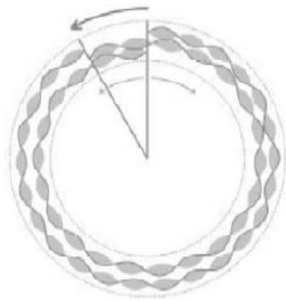
Mechanical gyroscope



Optical gyroscopes — based on Sagnac effect

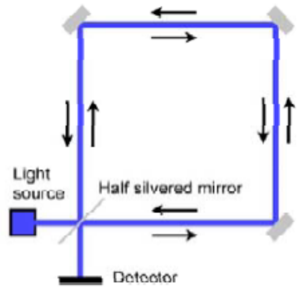


Light traveling opposite directions go different distances before reaching the moving source again



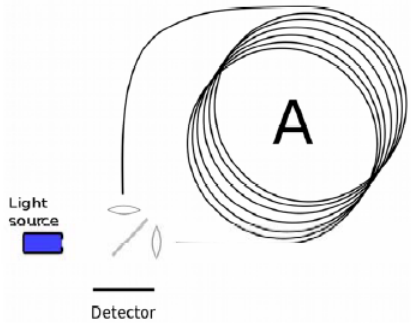
Schematic representation of the frequency shift when a ring laser interferometer is rotating

Sagnac interferometer



- A beam of light is split and the two beams follow a trajectory in opposite directions
- In the detector an interference pattern is obtained
- The position of the interference fringes is dependent on the angular velocity of the setup

Fiber optic gyroscope (FOG)



- The interference on a Sagnac interferometer is proportional to the enclosed area.
- A looped fiber-optic coil multiplies the effective area by the number of loops.
- A coil of optical fiber can be as long as 5 km

Differential system

Two transducers differentially connected for elimination of shock and linear acceleration

MEMS and signal conditioning on a single piece of silicone

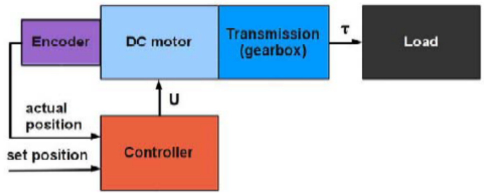
+ Synergy?
+ Added value?

4. Examples of mechatronic systems -— direct drives

Classical servomechanism

- Cannot operate at low speeds

- Low torque
- Low efficiency -> temperature (rotor)
- Nominal speed -> 3000rpm



- Restricted power and torque
- Gearbox needed
- Brushes
  - Maintenance costs
  - Reliability

Gearbox

- Efficiency
- Vibrations
- Maintenance costs
- Reliability
- Elasticity
  - Sensor localization problem
- Large additional inertia
- Hard nonlinearities
  - Friction
  - Backlash
- Cost

Controller

- Comparatively simple
- Comparatively cheap

List of wishes

- High constant torque at low speed (at rest)
- High stiffness
- Low moment of inertia
- Low weight
- Compact
- High reliability
- Low maintenance costs

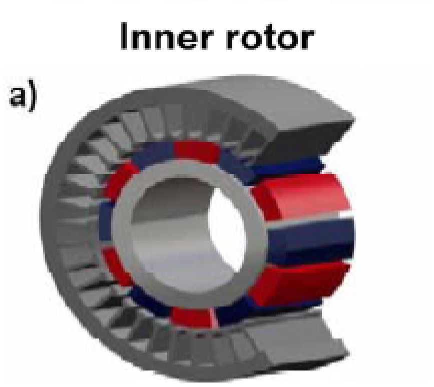
Direct drive

- Mechanical structure — inverted construction of a DC motor with brushes
- Rotor — multipole permanent magnet

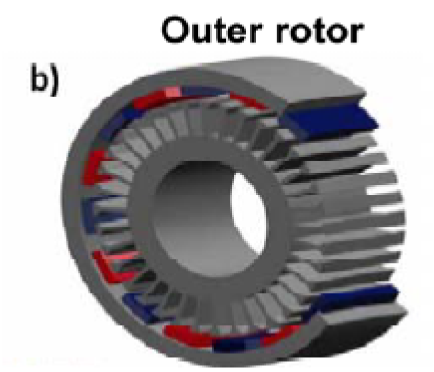
- Stator -— multiple windings connected in a delta (series) or wye “Y” (“star” - parallel) configuration
- Electronic commutation — the controller regulates relative position of two magnetic field vectors (rotor and stator)

Cross-section of a direct drive

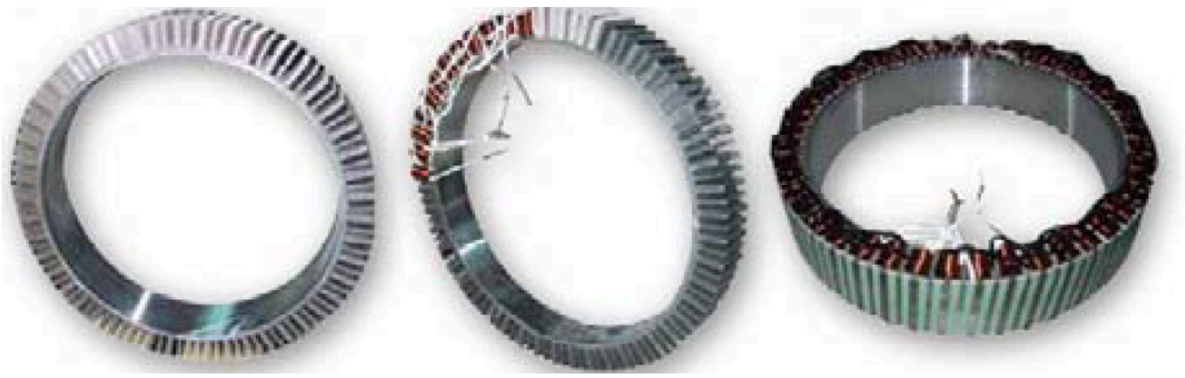
a) Inner rotor



b) Outer rotor

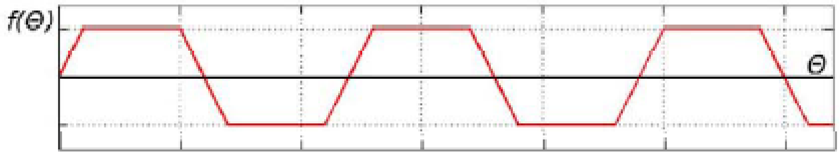


c) Stators

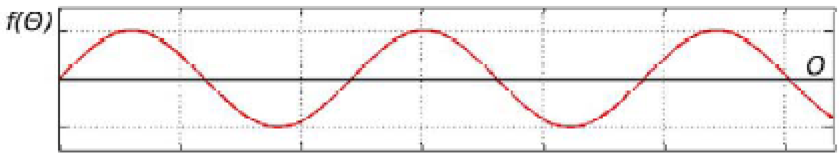


Magnetic field distribution in the slot between stator and rotor

a) Trapezoidal - BLDC



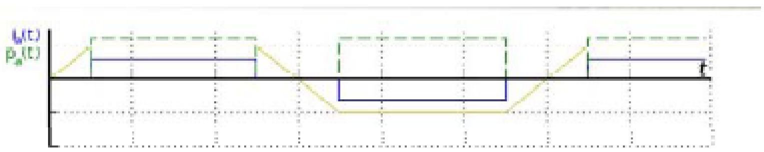
b) Sinusoidal - PMSM



BLDC

- Brushless Direct Current motor
- At constant speed square phase currents
- At constant speed square back EMF
- Requires square supply voltage

BLDC at constant speed

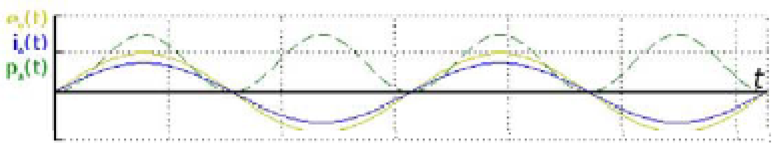


PMSM

Permanent Magnet Synchronous Motor

- At constant speed sinusoidal phase currents
- At constant speed sinusoidal back EMF
- Requires sinusoidal supply voltage at constant speed

PMSM at constant speed

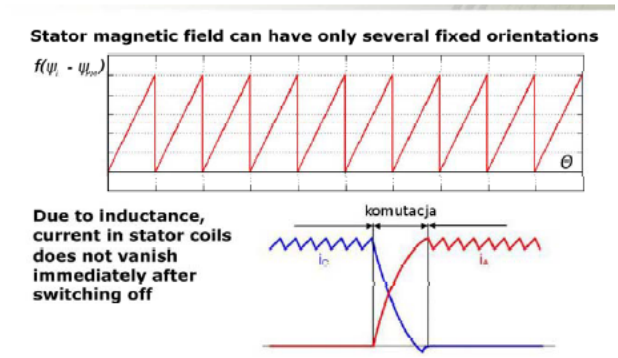


Control of direct drives

- Requires “electronic commutator”, usually microprocessor controller that switches stator coils
  - Current controller — high frequency PWM

- Commutator coupled with a rotor (magnetic field) angular position sensor
- Often both functions in a single circuit
  - three-phase inverter with MOSFETs or IGBTs

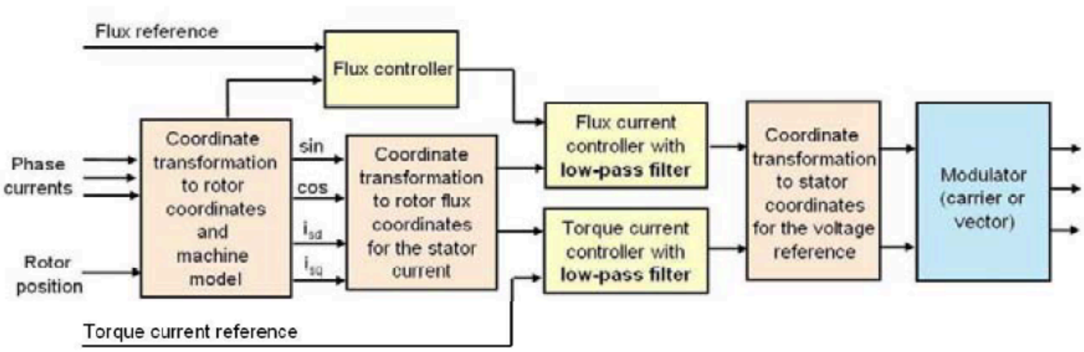
BLDC torque ripple



Control of PMSM

- To maintain constant angle between stator and rotor magnetic flux vectors
- High frequency PWM - average currents in coils changes sinusoidally (at constant speed)
- Field Oriented Control (FOC)
  - Amplitude, frequency and phase of magnetic flux vectors
  - In transient and steady state

Field Oriented Control



List of wishes

- High constant torque at low speed (at rest)
- High stiffness
- Low moment of inertia
- Low weight

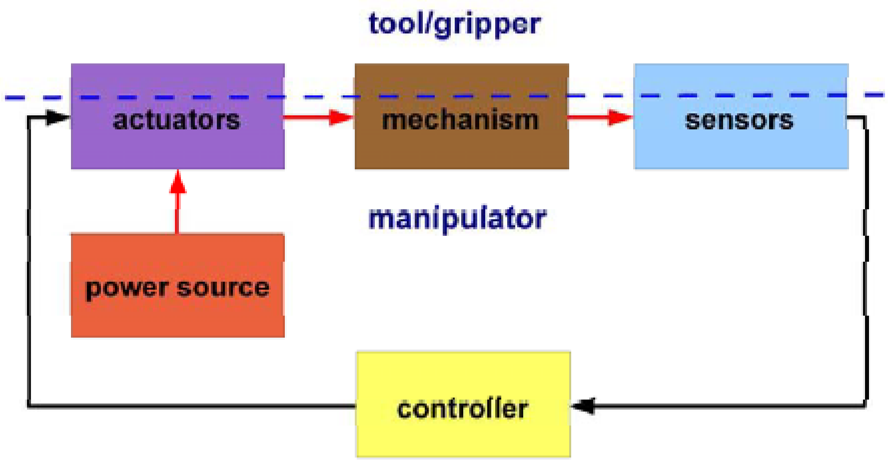
- Compact
- High reliability
- Low maintenance costs

Mechatronic aspects

- Design?
- Synergy?
- Added value?

5. Examples of mechatronic systems — industrial robots

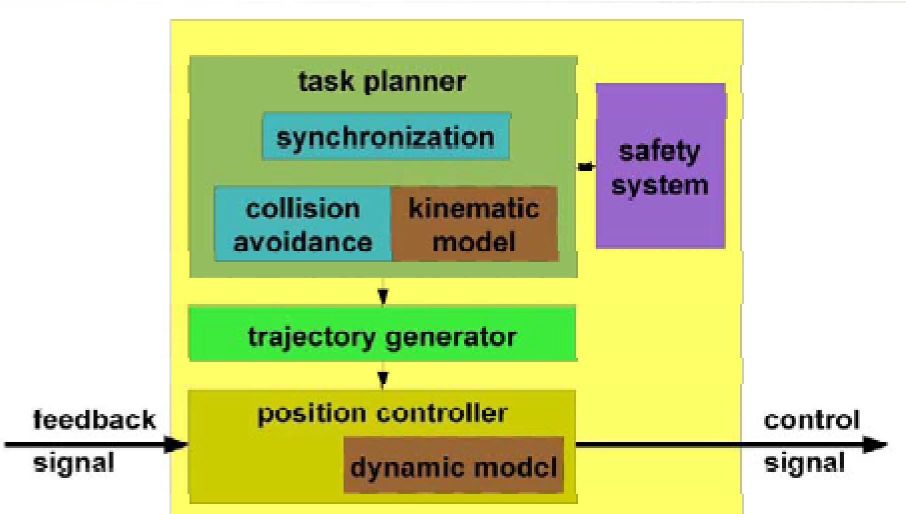
Structure of an industrial robot



Mechanism

- Provides end-effector mobility and orientation in required workspace with required redundancy (DOF) in required time
- Consists of links and joints
- Kinematics
  - DOF, workspace, singular positions
  - transformation between joint (local) and global (usually Cartesian) coordinates
- Dynamics
  - to analyze mechanism movement in time

Controller



6. Techniques of mechatronic design

Techniques of mechatronic design

Design procedure depends on actual device, but there are commonly used techniques:

- Virtual prototyping
- Real-time simulation
  - Fast prototyping
  - Hardware-in-the-Loop-Simulation (HILS)
- (fast prototyping on a target platform)
  - because of problems with implementation of signal processing algorithms
- Implementation of control and signal processing algorithms

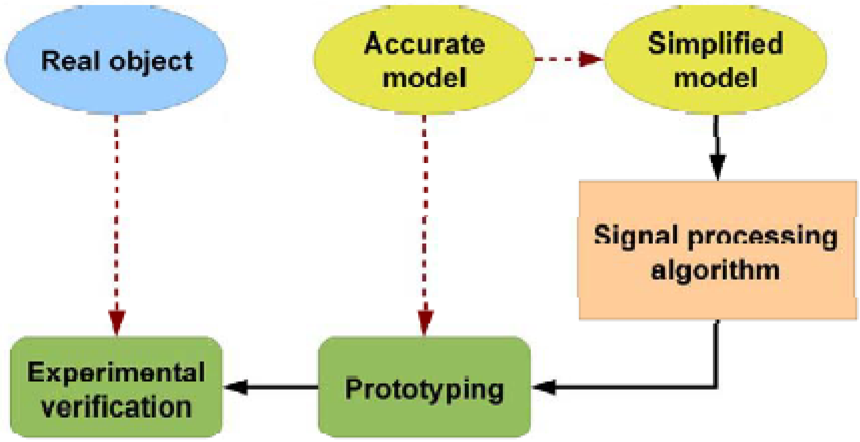
When are those techniques necessary?

- A controller (monitoring system) is design for a physically not yet existing system or experiments are dangerous
  - Virtual prototyping
- System is unstable or poorly damped -> cannot be experimentally tested without a stabilizing closed-loop controller that cannot be selected through an experiment
  - Virtual prototyping + fast prototyping (for fine tuning)

When are those techniques necessary? (contd.)

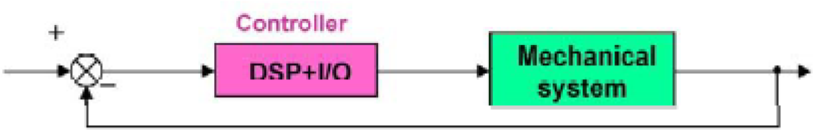
- System is complex, nonlinear, hard to theoretical modeling ~ experimental identification, sometimes iterative
    - Time consuming without fast prototyping
- Control, monitoring or other signal processing algorithm is implemented when the device is not yet existing physically
    - Experimental verification possible only through HILS

Modeling of an object



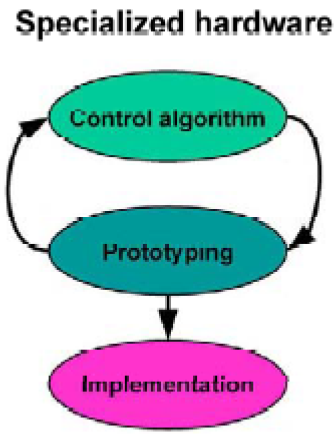
Fast prototyping

- Automatic code generation and compilation from a diagram
- Easy modification of parameters
- Display and recording of signals

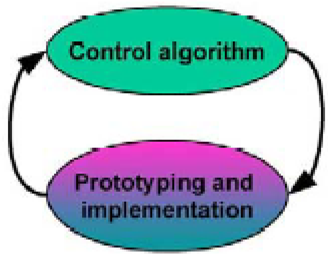


Fast prototyping on a target hardware platform

a) Specialized hardware

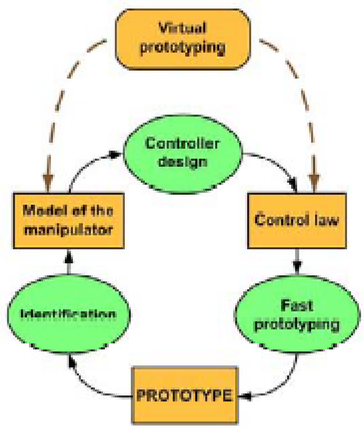


b) Target hardware

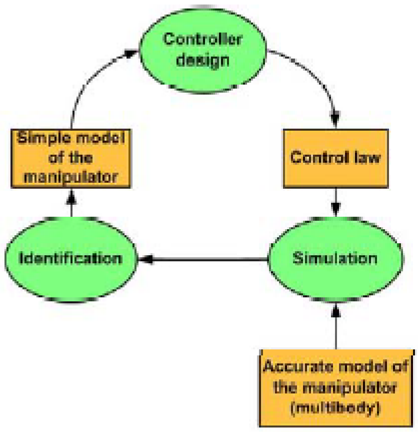


Identification in closed-loop

a) experimental

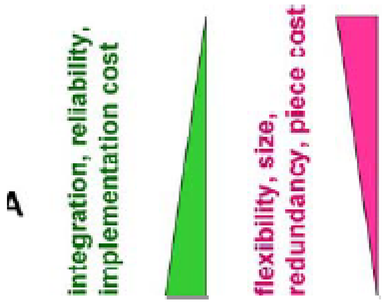


b) virtual



Hardware platform form implementation of algorithms

- Industrial computer
  - Single piece production
- Embedded system
  - Middle lot production
- Circuit based on ASIC/FPGA
  - Large lot production



Specialized systems for fast prototyping

- Flexible, large computing power
- Expensive
- Different from target hardware

Industrial computers

- Easy and flexible configuration
  - Variety of peripherals available
- High reliability in industrial conditions
- Sufficient computing power
- Use of recognized software standards
  - (RT)OS
  - Reliability
  - Efficient and well-tested compilers
  - Availability of tools
- Expensive
- Suboptimal

Embedded systems

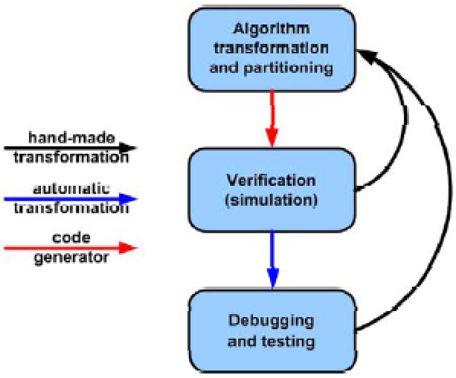
- Specialized, complete, non-reconfigurable microprocessor system

- Based on:
- general-purpose CPU -> Irregular, especially floating-point algorithms
- DSP
- Regular, simple algorithms fast processing large amounts of data
- FPGA

Transformations of algorithms being implemented

- Time discretization
    - Always and partitioning
- Amplitude quantization
    - Hardware implementation, recommended always
- Coding in HDL
    - Hardware implementation

Mixed, hardware-software algorithm implementation

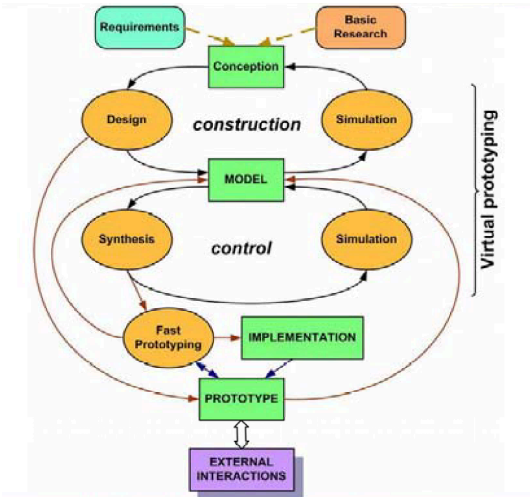


7. An example of mechatronic design — a parallel robot

Why mechatronic approach?

- Each kinematic chain acts directly on a platform (end-effector) — analysis and design of individual limbs separately is impossible
- Closed kinematic chains improve properties (increase stiffness) of a robot, but complicate kinematic equations and control synthesis
- Modification of elements shape alternates character of kinematics equations
- Actuators usually play also a role of structural members of a manipulator and their mechanical properties influence dynamics (actuators depend on the construction and construction depends on actuators)

The general procedure


Requirements
Basic Research
Conception
Design
construction
Simulation
MODEL
Synthesis
control
Simulation
Virtual prototyping
Fast Prototyping
IMPLEMENTATION
PROTOTYPE
EXTERNAL INTERACTIONS