Design of Mechatronics Systems

Steps of mechatronics system design, possible design solutions.

Mechatronics System design

- Mechatronic systems are complex and multidisciplinary, integrating mechanical, electrical, and software components to achieve specific functions and goals.
- Designing a mechatronic system for optimal performance and reliability requires a systematic and iterative approach that considers the interactions and trade-offs among the subsystems and the environment.
- Mechatronics system design basically an synergetic Integration of Different Disciplines
- Here, we will learn the basic steps and principles of mechatronic system design and some examples of mechatronic applications.

Steps of Mechatronics system design

1. Define the problem

- The first step in any engineering design process is to define the problem and the requirements of the system.
- You need to identify the purpose, function, and specifications of the mechatronic system, as well as the constraints, assumptions, and criteria for evaluation.
- You also need to conduct a market and customer analysis, a feasibility study, and a risk assessment to ensure that the system is viable, desirable, and safe.

2. Model the system

- The next step is to model the system using mathematical, physical, and logical tools.
- We need to describe the system's behavior, structure, and dynamics using equations, diagrams, and algorithms.
- We also need to identify the inputs, outputs, and feedback loops of the system, as well as the sources of uncertainty, noise, and disturbance.
- Modeling the system helps us to understand, analyze, and simulate the system's performance and response under different conditions and scenarios.

3. Select the components

- The third step is to select the components that will form the mechatronic system.
- We need to choose the appropriate sensors, actuators, controllers, and communication devices that will meet the system's requirements and specifications.
- We also need to consider the compatibility, integration, and interfacing of the components, as well as their cost, availability, and reliability.
- Selecting the components involves comparing and evaluating different options and alternatives based on their advantages and disadvantages.

4. Design the software

- The fourth step is to design the software that will control and coordinate the mechatronic system.
- We need to develop the algorithms, programs, and codes that will implement the system's logic, functionality, and intelligence.
- We also need to follow the software engineering principles and practices, such as modularization, documentation, testing, and debugging, to ensure that the software is robust, efficient, and maintainable.
- Designing the software requires using the appropriate programming languages, tools, and platforms that suit the system's needs and capabilities.

5. Test and optimize the system

- The fifth step is to test and optimize the system using experimental and computational methods.
- We need to verify and validate that the system meets the design objectives and requirements, as well as the standards and regulations.
- We also need to measure and evaluate the system's performance and reliability using appropriate metrics, indicators, and benchmarks.
- Testing and optimizing the system involves conducting experiments, simulations, and analyses to identify and correct any errors, flaws, or weaknesses in the system.

6. Implement and maintain the system

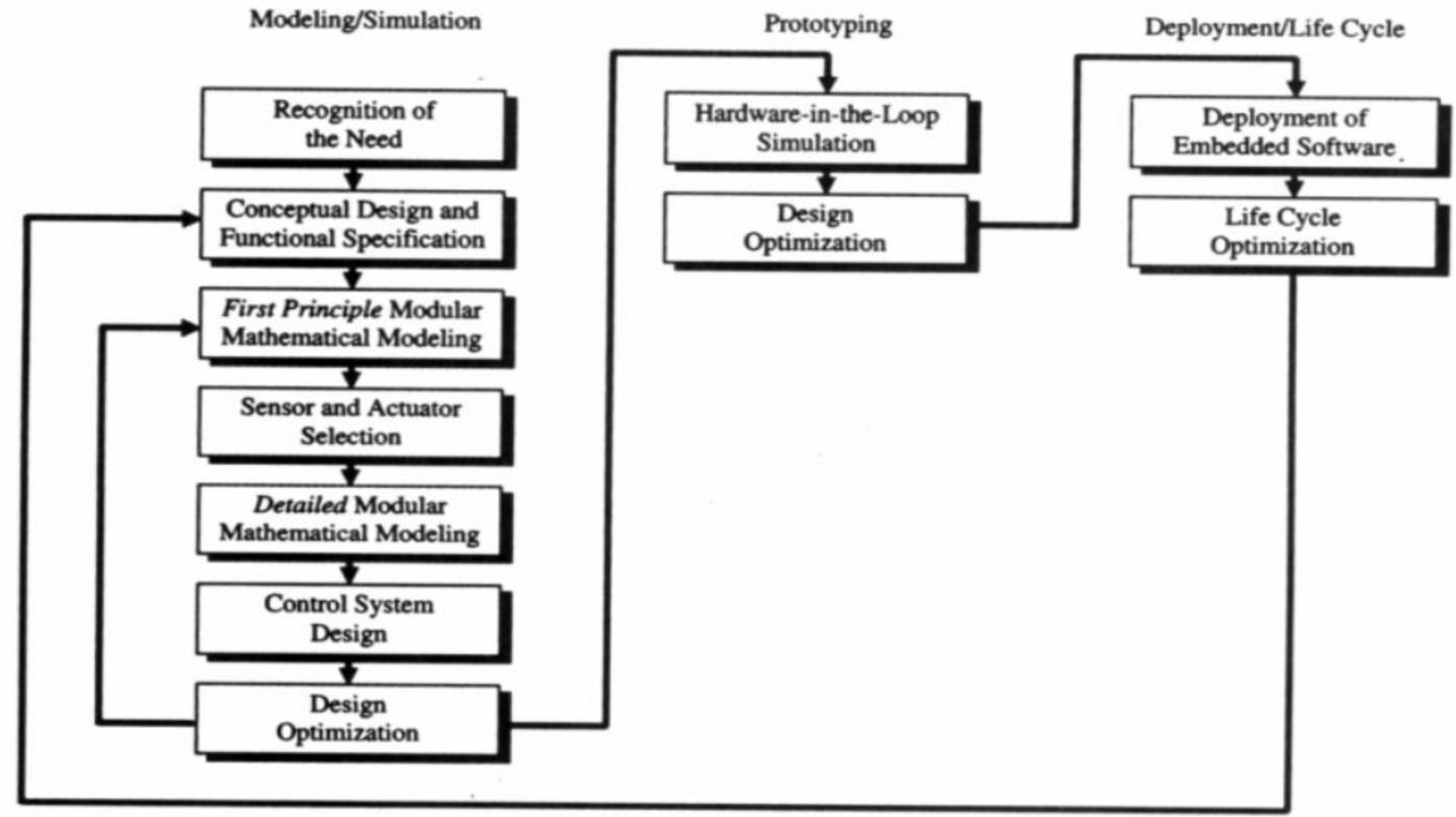
- The final step is to implement and maintain the system in the real-world environment.
- We need to install, operate, and monitor the system using the necessary hardware, software, and network resources.
- We also need to provide training, support, and feedback to the users, customers, and stakeholders of the system.
- Implementing and maintaining the system requires ensuring that the system is adaptable, scalable, and upgradable to cope with changing needs and conditions.

Mechatronic design process

- The mechatronic design process consists of three phases:

  1. Modeling and Simulation 2. Prototyping 3. deployment.
- All modeling should be modular in structure.
- Complex models may be created by connecting the modules, or blocks, together.
- Each block represents a subsystem that corresponds to some physically of functionally realizable operations that can be encapsulated into a block with input/output limited to input signals, parameters, and output signals.
- Because of their modularity, mechatronic systems are well suited for applications that require reconfigurations.
- Such products can be reconfigured either during the design stage by substituting various subsystem modules or during the life span of the products.

Steps of Mechatronics system design



Modeling:

- Modeling is the process of representing the behavior of a real system by a collection of mathematical equations and logic.
- Models are cause-and-effect structures—they accept external information and process it with their logic and equations to produce one or more outputs.
- Models can be text-based programming or block diagram.

Simulation:

- Simulation is the process of solving the model and is performed on a computer.
- Simulation process can be divided into three sections:
    - Initialization
    - Iteration,
    - Termination

Steps of Modeling & Simulation

Recognition of the need:

- Need recognition refers to the phenomenon that occurs when a person becomes aware of a disparity between their actual circumstances and those they consider ideal or desirable.
- Before people can begin looking for a solution, they must first acknowledge that they have a problem, to begin with.

Conceptual design and functional specification:

- In the conceptual design phase the functional requirements are translated to the design parameters, working principles, and physical structures.
- Conceptual design involves concept generation and concept evaluation.
- A good design concept must be logically feasible, functionally simple, and physically certain.

First principle modular mathematical modeling:

- Mathematical modeling refers to the process of creating a mathematical representation of a real-world scenario to make a prediction or provide insight
- A first principle ('knowledge-based‘) model is a simple model that captures some of the fundamental behavior of a subsystem.

Sensors and actuators selection:

- Actuators and sensors often depend on each other to perform certain tasks.
- Where both are present, the actuator relies on the sensor to power its function.
- If one or the other fails to work correctly, the system will malfunction.
- Selection of these depend on the requirements of the project.

Detailed modular mathematical modeling:

- A detailed model is an extension of the first principle model providing more function and accuracy than the first-level model.
- Detailed mathematical modeling process explain with six stages:

1. Understanding the problem
2. Choosing variables
3. Making assumptions
4. Solving the equations
5. Interpreting the solution
6. Validating the model
7. Criticizing and improving the model.

(Validation is the action of making or declaring something legally or officially acceptable)

Control system design:

- The objective of control system design is to construct a system that has a desirable response to standard inputs.
- A desirable steady-state response is one that follows the desired output with sufficient accuracy.

Design optimization:

- Design optimization is the process of finding the best design parameters that satisfy project requirements.
- Engineers typically use design of experiments (DOE), statistics, and optimization techniques to evaluate tradeoffs and determine the best design.
- Optimization solves the problem of distributing limited resources throughout a system so that pre-specified aspects of its behavior are satisfied

Prototyping

- Prototyping is an experimental process where design teams implement ideas into tangible forms from paper to digital.
- Teams build prototypes of varying degrees of fidelity to capture design concepts and test on users.
- Using basic sketches and rough materials, the prototype may be a simple drawing or rough model that helps innovators determine what they need to improve and fix in their design.

Hardware-in-the-loop simulation:

- Hardware-in-the-loop (**HIL**) **simulation**, HWIL, or HITL, is a technique that is used in the development and testing of complex real-time embedded systems.
- HIL simulation can be used to test controller design.
- HIL simulation shows how controller responds in real time to realistic virtual stimuli.
- HIL can also be used to determine if a physical system (plant) model is valid.

Design optimization:

Deployment

- Deployment is the action of bringing resources into effective action.
- Following deployment, the life cycle functions begin, and information from these functions should be made available for upgrades to the product.

Deployment of imbedded software:

- Embedded software is computer software, written to control machines or devices that are not typically thought of as computers, commonly known as embedded systems.
- It is typically specialized for the particular hardware that it runs on and has time and memory constraints.
- This term is sometimes used interchangeably with firmware

Life cycle optimization:

- Life cycle optimization (LCO) is a method that integrates life cycle assessment (LCA) with optimization analysis for enhancing product sustainability.

Life cycle factors

- **Delivery**: Time, cost, and medium
- **Reliability:** Failure rate, materials, and tolerances.
- **Maintainability:** Modular design.
- **Serviceability**: On board diagnostics, prognostics, and modular design.
- **Upgradeability:** Future compatibility with current designs.
- **Disposability:** Recycling and disposal of hazardous materials

- In the mechatronic design approach, life cycle factors are included during the product design stages, resulting in products that are designed from conception to retirement.

- Because no single model can ever flawlessly reproduce reality, there will always be error between the behavior of a product model and the actual products.
- These errors, referred to as unmodeled errors, are the reason why so many model-based designs fail when deployed top the product.
- Mechatronic design approach also uses a model-based approach, relying heavily on modeling and simulation; unmodeled errors are accounted for in the prototyping step.
- Their effects are absorbed into the design, which significantly raiser the probability of successful product deployment.
- In the prototyping step many of the noncomputer subsystems of the model are replaced with actual hardware.

- The resulting model is part mathematical and par real.
- The real part of the model inherently evolves in real time and the mathematical part evolves in simulated time, it is essential that the two parts be synchronized.
- This process of fusing and synchronizing model, sensor, and actuator information is called real-time interfacing or hardware-in-the-loop simulation and is and essential ingredient in the modeling an simulating environment.

Advanced approaches in mechatronics :

- In addition to influencing the way the products are designed, recent developments in mechatronics are also creating opportunities in intelligent manufacturing.
- In order to produce high-quality products at low life cycle costs, many manufacturing plants have been reducing their workforce.

Comparison

Conventional Design	Mechatronic Design
Added components	**Integration of components (hardware)**
1 Bulky	Compact
2 Complex mechanisms	Simple mechanisms
3 Cable problems	Bus or wireless communication
4 Connected components	Autonomous units
Simple control	**Integration by information processing (software)**
5 Stiff construction	Elastic construction with damping by electronic feedback
6 Feedforward control, linear (analog) control	Programmable feedback (nonlinear) digital control
7 Precision through narrow tolerances	Precision through measurement and feedback control
8 Nonmeasurable quantities change arbitrarily	Control of nonmeasurable estimated quantities
9 Simple monitoring	Supervision with fault diagnosis
10 Fixed abilities	Learning abilities