Perspective
Volume 11:1, 2022

Journal of Industrial Engineering and Management

ISSN: 2169-0316 Open Access

Model-based Mechatronic System Design

Veena Priyadarshini*

Department of Life Sciences, BSAR University, Chennai, Tamil Nadu, India

Perspective

Electromechanical systems without some form of embedded computer system are unusual today. In fact, an embedded system's intelligence is frequently the essential feature that sets a piece of equipment apart from its competitors. However, the advantages of an embedded system come at a cost. As more powerful microprocessors are used in mechatronic systems, the interplay between hardware and software gets more complicated. Hardware and software engineering teams, who define requirements, describe difficulties, and test and implement their ideas in various methods, may find it difficult to manage this complexity. The fact that most of these systems use closed-loop control techniques to correct for electromechanical interactions and external disturbances adds to the complexity. These difficulties necessitate the use of open-loop supervisory control for tasks such as startup and shutdown, people and equipment safety, and problem detection and correction. Mechanical modelling is emphasized in today's mechatronic-design procedures before any hardware is developed. When it comes to embedded systems, however, we rarely see this approach. In most situations, software validation is addressed late in the development process, with developers simply testing their programmed-on hardware prototypes. At this level, errors in hardware or software cause significant delays. It can take a long time to track down the source of an issue. Model-Based Design is a solution to such problems. It facilitates the development of mechatronic systems by offering a common design and communication environment for engineers from all disciplines. Model-Based Design has its origins in the aerospace and automobile industries, which were rapidly incorporating more microprocessors into their products in the early 1990s. Engineers in the aerospace and automotive industries recognized the benefits of simulating multidomain systems while developing embedded controllers.

Simulation of control algorithms led to the automatic creation of code from the math model in the mid-1990s. These industries' successful implementation of Model-Based Design shown that simulation and automatic code generation for embedded systems are cost-effective and time-saving methods for designing mechatronic systems. Other industries involved in mechatronic development have discovered Model-Based Design in the last five years. Model-Based Design has spread in the same manner that CAD/FEA software did due to the combined benefits of more sophisticated software and falling cost per-MIP of desktop computers. To adequately represent the total system, Model-Based Design combines the dynamics and performance requirements. Because this method is based on software, developers may quickly research competing designs and test new ideas without having to invest in expensive hardware. A system-level model is used in Model-Based Design. This paradigm creates an executable specification by mathematically characterizing the equipment's natural and regulated behavior. Engineers can put the model into action by modelling the system's actual dynamics and performance. The model offers an explicit mathematical definition of the mechatronic system's intended performance. The model comprises inputs (signals from outside agencies) and outputs (metrics of how well the system is working). Voltage, temperature, and pH are examples of real-world inputs and outputs. The diagram's blocks indicate mathematical processes between the model's input and output signals. Some blocks, referred to as the plant or process, represent the mechatronic system's natural behavior. A motor, for example, could be represented by a block in the model. The motor's mathematical model may be straightforward, translating a voltage input to an output torque [1-3].

Enhancing the System Design

Adding inputs to the model, such as voltage noise, or characteristics, such as temperature and magnetic saturation effects, can make the motor model more complex. The compensation or control in the system can be represented by a single block or a collection of blocks filtering and processing signals based on output faults or events in the model. A lumped parameter mathematical model that describes the physics of the system is the foundation of a system level model. The input-to-output relationship of mechatronic systems is expressed using ordinary differential equations (ODEs) and differential algebraic equations (DAEs). An ODE describes the relationship between the voltage input and the shaft output torque in the motor case. Instead of utilizing a tool like FEA, which is based on partial differential equations, differential equations offer a more computationally efficient technique to represent lumped dynamics. Detail work, such as solving for the torque-induced stress distribution at a crucial slot in the motor shaft, is more likely to be done with FEA software. Engineers can use Model-Based Design to start with a less detailed system-level model and gradually raise its accuracy as the project advances. To assist engineers swiftly remove concepts with limited potential, a proof-of-concept model written as a lower-order ODE may be all that's required at first. Within the equipment, the open-loop control conducts supervisory and mode control, as well as handling interactions with the operator. Developers may now design increasingly sophisticated self-diagnostics, fault-detection, and safety-shutdown systems because to the availability of more powerful microprocessors. Engineers can use Model-Based Design to create and evaluate more complicated open-loop control systems for such functions against a system-level model. Simulation enables testing to begin early in the design phase, allowing engineers to optimize equipment ergonomics and identify any scenarios that could result in damage or harmful conditions. Several closed-loop control systems can be used in a mechatronic system to operate under a variety of scenarios. The use of a system-level model assists in the design and tuning of controllers in coupled loops. Engineers can use Model-Based Design to undertake cost trade-off evaluations within the control system. The system-level model is a technique for determining whether a less expensive sensor with a larger tolerance will provide the needed accuracy and performance. Engineers can compare the cost of virtually any component used in mechatronic systems to the influence on system performance in this way. Throughout the development process, Model-Based Design entails continual testing and verification. When designing a control system, developers usually write standard tests. Engineers can analyze the changing system-level model in a consistent manner and with the same set of metrics by using standard tests or a test harness. Electronic linkages can often be used to connect requirement documents with test criteria like pass/fail and tolerance bands. Continuous testing with a common test harness reveals the impact of each design change on system outputs right away, allowing you to rapidly pinpoint the source of the change [4-5].

***Address for Correspondence:** Veena Priyadarshini, Department of Life Sciences, BSAR University, Chennai, Tamil Nadu, India; E-mail: veenapriya31@gmail.com*

Received: 05-Jan-2022, Manuscript No: iem-22-52939; **Editor assigned:** 07 Jan 2022, Pre QC-No. P- 52939; **Reviewed:** 12 Jan -2022, QC No. Q- 52939; **Revised:** 17 Jan 2022, Manuscript No. R- 52939; **Published:** 22 Jan 2022, DOI: 10.37421/2169-0316.22.11.333

References

1. Barbieri, Giacomo, Cesare Fantuzzi, and Roberto Borsari. "A model-based design methodology for the development of mechatronic systems." Mechatronics 24 (2014): 833-843.
2. Sadlauer, Alfred, and Peter Hehenberger. "Using design languages in model-based mechatronic system design processes." Int J Agile Syst Manag 10 (2017): 73-91.
3. Qamar, Ahsan. "An integrated approach towards model-based mechatronic design." PhD diss., KTH Royal Institute of Technology, 2011.
4. Zheng, Chen, Xiansheng Qin, Benoît Eynard, Jing Li, Jing Bai, Yicha Zhang, and Samuel Gomes. "Interface model-based configuration design of mechatronic systems for industrial manufacturing applications." Robot Comput Integr Manuf 59 (2019): 373-384.
5. Dieterle, Werner. "Mechatronic systems: Automotive applications and modern design methodologies." Annual Reviews in Control 29(2005): 273-277.

How to cite this article: Priyadarshini, Veena. "Model-based mechatronic system design" *J. Ind. Eng. Manag*. 11(2022):333