Simulink® para el diseño de sistemas automotrices.

Curso presencial

Objetivo del curso:

Creating and modifying Simulink models and simulating system dynamics, modeling continuous-time, discrete-time, and hybrid systems, modifying solver settings for simulation accuracy and speed, building hierarchy into a Simulink® model, creating reusable model components using subsystems, libraries, and model references.

Requisito(s):

MATLAB® Fundamentals for Automotive Applications.

Descripción del curso:

Demonstrate how to apply basic modeling techniques and tools to develop Simulink® block diagrams.

Imparte: Emmanuel Olivar

Ingeniero en Sistemas Electrónicos Industriales

Emmanuel Olivar Domínguez, egresado de la Universidad Autónoma de la Ciudad de México (UACM), cuenta con amplia experiencia en el mantenimiento preventivo y correctivo de equipo electrónico industrial y comercial, además de utilizar tecnología Mathworks (MATLAB y Simulink) por más de cinco años. Se especializa en las áreas de control e instrumentación con la finalidad de apoyar en la solución de las problemáticas que se puedan presentar en éstas, además de utilizar diversas herramientas con el fin de desarrollar controladores aplicados a la industria.

Actualmente ocupa el cargo de Ingeniero de Aplicación MATLAB en MultiON Consulting S.A. de C.V. en donde se dedica a la resolución de problemas de índole computacional que enfrentan organizaciones tanto públicas como privadas.

Temario

**El manual del curso se encuentra en inglés.*

1. Introduction.
   - 1.1 MathWorks® at a Glance.
   - 1.2 MathWorks® Product Overview.
   - 1.3 Diverse Users.
   - 1.4 Computer Setup.
   - 1.5 What Can You Do with Simulink®?
   - 1.6 System Design Process.
   - 1.7 Model-Based Design with Simulink®
   - 1.8 Overview of Dynamic Systems.
   - 1.9 Course Example.

2. Creating and Simulating a Model.
   - 2.1 Modeling the System with Equations.
   - 2.2 Creating a New Simulink® Model.
   - 2.3 Using the Simulink® Library Browser.
   - 2.4 Adding and Connecting Blocks.
   - 2.5 Labeling Blocks and Signals.
   - 2.6 Defining Block Parameters.
   - 2.7 Defining Lookup Table Parameters.
   - 2.8 Defining System Inputs.
   - 2.9 Defining System Outputs.
   - 2.10 Simulating the Model.
   - 2.11 Overview of Simulink® Solver.
   - 2.12 Simple Models – Default Step Size.
   - 2.13 Adding Signal Viewers.
   - 2.14 Overview of Signal Viewers.
   - 2.15 Customizing Signal Viewers.
   - 2.16 Additional I/O Blocks.
   - 2.17 Managing Model Parameters.

3. Modeling Programming Constructs.
   - 3.1 Approaches to Modeling Programming Constructs.
   - 3.2 Modeling Comparisons.
   - 3.3 Modeling Decision Statements.
   - 3.4 Extracting and Combining Data Using Vectors.
   - 3.5 Course Example: PWM Conversion.
   - 3.6 Modeling the System with Equations.
   - 3.7 Building a Block Diagram.

3.8 Zero-Crossing Detection.
3.9 Zero-Crossing Solver Support.
3.10 Modeling with MATLAB® Function Blocks.
3.11 Understanding the MATLAB® Function Block.

4. Modeling Discrete Systems.
4.1 Sample Times.
4.2 Simulating a Model That Contains Discrete Signals.
4.3 Defining Discrete-State Systems.
4.4 Modeling Difference Equations.
4.5 Course Example: PI Controller.
4.6 Modeling the System with Equations.
4.7 Building a Block Diagram.
4.8 Creating Discrete State-Space and Transfer Function Models.
4.9 Modeling Multirate Systems.
4.10 Simulating a Model That Contains Multirate Discrete Signals.

5. Modeling Continuous Systems.
5.1 Defining Continuous-State Systems.
5.2 Modeling Continuous-State Systems with Simulink®.
5.3 Modeling Differential Equations.
5.4 Course Example: Throttle Body.
5.5 Modeling the System with Equations.
5.6 Building a Block Diagram.
5.7 Defining Block Parameters.
5.8 Simulating a Model That Contains Continuous States.
5.9 Creating Continuous State-Space and Transfer Function Models.
5.10 Modeling Physical Boundaries.

6. Solver Selection.
6.1 Accuracy and Speed.
6.2 Understanding the Simulink® Solver.
6.3 Solving Simple Models.
6.4 Solving Models with Discrete States.
6.5 Solving Multirate Models.
6.6 Solving Models with Continuous States.
6.7 Common Performance Concerns.
6.8 Default Solvers.
6.9 Continuous-State Dynamics – Solver Type.
6.10 Continuous-State Dynamics – Solver Selection.
6.11 Continuous-State Dynamics – Tolerances.
6.12 Zero Crossings.

- 6.13 Handling Consecutive Zero Crossings.
- 6.14 Algebraic Loops.
- 6.15 Effects of Algebraic Loops.
- 6.16 Handling Algebraic Loops.
- 6.17 Profiling Solver Behavior.

7. Developing Model Hierarchy.

- 7.1 Introducing Subsystems.
- 7.2 Creating Subsystems.
- 7.3 Handling Cross-System Gateways.
- 7.4 Creating Visual Hierarchy.
- 7.5 Creating Functional Hierarchy.
- 7.6 Reducing Signal Congestion.
- 7.7 Masking Blocks.
- 7.8 Creating a Mask.
- 7.9 Dialog Parameters.
- 7.10 Mask Workspaces.
- 7.11 Initialization Commands.
- 7.12 The Icon Editor.
- 7.13 Possible Icons.
- 7.14 Block Documentation.

8. Modeling Conditionally Executed Algorithms.

- 8.1 Block Sample Times.
- 8.2 Types of Subsystems.
- 8.3 Conditionally Executed Subsystems.
- 8.4 Enabled Subsystems.
- 8.5 Enabled Subsystems – State/Output Behavior.
- 8.6 Triggered Subsystems.
- 8.7 Types of Triggers.
- 8.8 Enabled and Triggered Subsystems.
- 8.9 Course Example: Handling Invalid Input.

9. Combining Models into Diagrams.

- 9.1 Subsystems and Model Referencing.
- 9.2 Model Referencing Workflow.
- 9.3 Defining Model Reference Inputs and Outputs.
- 9.4 Testing the Controller Model.
- 9.5 Referencing Models.
- 9.6 Simulating the Model (Normal Mode).
- 9.7 Simulating the Model (Accelerator Mode).
- 9.8 Data Scope Considerations.

Joel Cervantes
Asesor Comercial LATAM
MultiON Consulting, S.A. de C.V
Cómputo Científico y Técnico: software y hardware especializado.
Tel: +52 (55) 5559-4050 Ext. 119 | cursos@multion.com
www.multion.com