MathWorks®
MATLAB Examples

Using a Data Dictionary to Manage the Data for a Fuel Control System

By MathWorks

Explore:
Simulink

Try it in MATLAB

View in: Documentation

This example shows how to use data dictionaries to manage the data for a fuel rate control system designed using Simulink® and Stateflow®. To familiarize yourself with the fuel rate control model see sldemo_fuelsys.

Contents

- Familiarize Yourself with the Model
- Open and Compile the Test Harness Model
- View the Engine Gas Dynamics System (Plant)
- View the Fuel Rate Control System (Controller)
- Investigate the Data Used by the Controller
- Investigate the Units Used by the Components
- Simulate the Test Harness Model
- Close the Example
- Related Examples

Familiarize Yourself with the Model

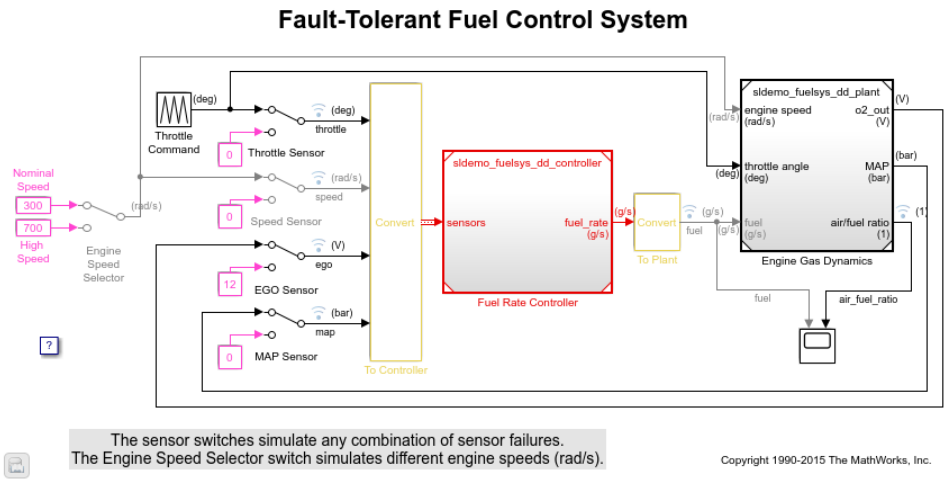
The sldemo_fuelsys_dd model is a closed-loop system containing a "plant" and "controller". The plant is used to validate the design of the controller. In this example, the plant and controller are represented by separate models that are referenced from the test harness model. Let's take a look at these models.

Open and Compile the Test Harness Model

Related Examples

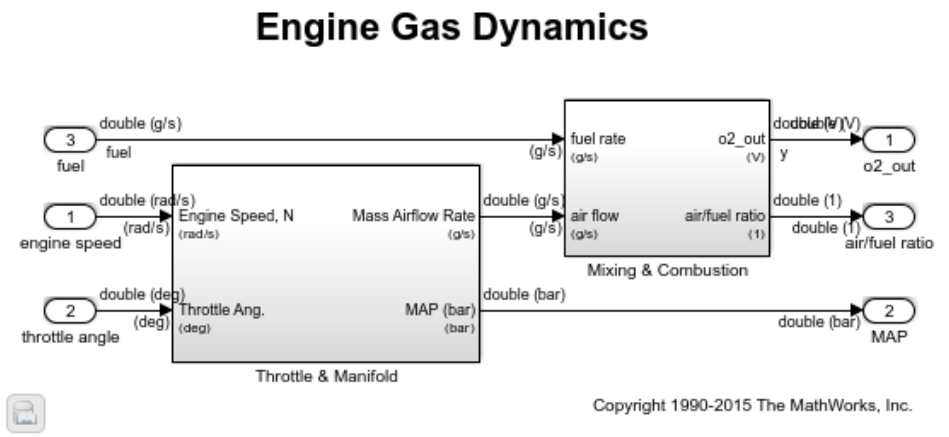
Fixed-Point Fuel Rate C System

This example shows how t a floating-point and a fixed- simulation of a fuel rate cor system designed using Sim



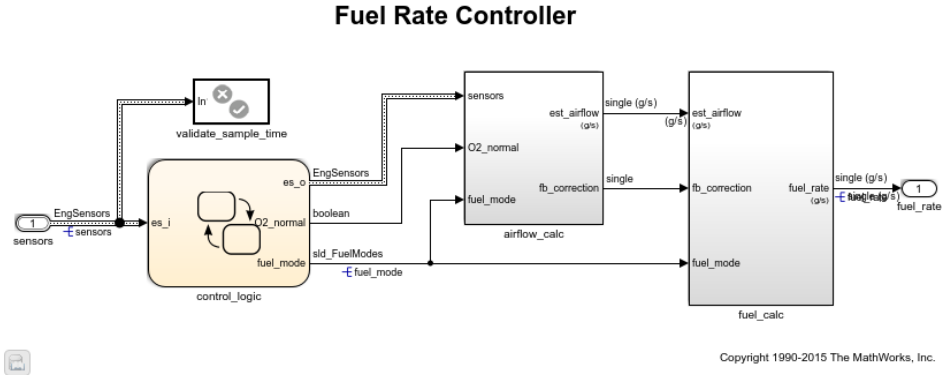
View the Engine Gas Dynamics System (Plant)

Double-click on the Engine Gas Dynamics block to open the plant model.



View the Fuel Rate Control System (Controller)

Double-click on the Fuel Rate Controller block to open the controller model.



Investigate the Data Used by the Controller

Configure Generated C According to Interface Document

Import specifications from a interface control document configure code generation for a model according to th

The global design data for the controller model is defined in a data dictionary. Using data dictionaries has many advantages over defining data in the base workspace. To learn about data dictionaries see the documentation for Simulink Data Dictionary.

The controller model is explicitly linked to a data dictionary. This link is set up on the Data tab of the Model Properties dialog.

The icon in the lower-left corner of the model window shows that this model is linked to a data dictionary. You can open the data dictionary in the Model Explorer by clicking on this icon.

This data dictionary contains parameter and signal objects that are used to configure the controller algorithm for simulation and code generation. It also contains a reference to another data dictionary that defines the data type objects used by this model.

Initially, this data dictionary is configured for a float-point controller, as is seen by the data type display on the signal lines in the controller model. You can easily switch between floating-point and fixed-point controllers by changing the types dictionary (sldemo_fuelsys_dd_types.sldd) to reference the appropriate dictionary:

- Floating-point types: sldemo_fuelsys_dd_float.sldd
- Fixed-point types: sldemo_fuelsys_dd_fixpt.sldd

Investigate the Units Used by the Components

Notice that units are visible on the model and subsystem icons and signal lines. Units are specified on the ports and on the bus, signal and parameter objects in the data dictionary. To learn more about units in Simulink see Simulink Units.

Simulate the Test Harness Model

The test harness model is also linked to a data dictionary (sldemo_fuelsys_dd.sldd). This data dictionary contains references to the data dictionaries for the plant and controller models but it does not contain any additional data.

Simulate the test harness model to validate the behavior of the controller in either the floating-point or fixed-point configuration.

Close the Example

Close the models and data dictionaries from this example

Related Examples

Refer to other examples related to sldemo_fuelsys:

- Floating-point design: sldemo_fuelsys

- Fixed-point design: fxpdemo_fuelsys
- Production C/C++ code generation: rtwdemo_fuelsys
- Fixed-point production C/C++ code generation: rtwdemo_fuelsys_fxp

mathworks.com