FMI WORKSHOP

INCOSE International Workshop, Los Angeles, CA, 2015

Contents

Introduction

Functional Mockup Interface (FMI) is a tool independent standard to support both model exchange and co-simulation of dynamic models. This workshop provides hands-on experience with FMI-based workflows in a sample of FMI-compliant tools including the following:

- FMU export from Dymola
- FMU import and simulation in Dymola
- FMU import and simulation in PyFMI
- FMU import and simulation with FMI Add-in for Excel
- FMU import and simulation with FMI Toolbox for MATLAB/Simulink
- FMU export from Simulink with FMI Toolbox for MATLAB/Simulink

The goal of this workshop is to illustrate the creation and use of FMUs in several different tools. While only a sampling of available FMI-compliant tools, this workshop aims to demonstrate the power and flexibility of FMI-based workflows for CAE toolchain integration. The tools above are provided as part of the material distributed with this workshop along with evaluation licenses as required.

Modelon

Model Overview

The dynamic model used throughout this tutoria is a Vehicle Thermal Management (VTM) system model for studying vehicle-level interactions between thermal management systems. This particular example is a conventional four wheel passenger sedan chassis and drivetrain architecture with a spark ignition engine and standard transmission. These mechanical systems are created using components from the Vehicle Dynamics Library (VDL) from Modelon. The model also includes a representation of the coolant and oil fluid loops for the engine and transmission in conjunction with a four heat exchanger stack for the thermal domain. These portions of the model are constructed from components of the Liquid Cooling Library (LCL) from Modelon. A snapshot of the entire system is shown in the following figure.

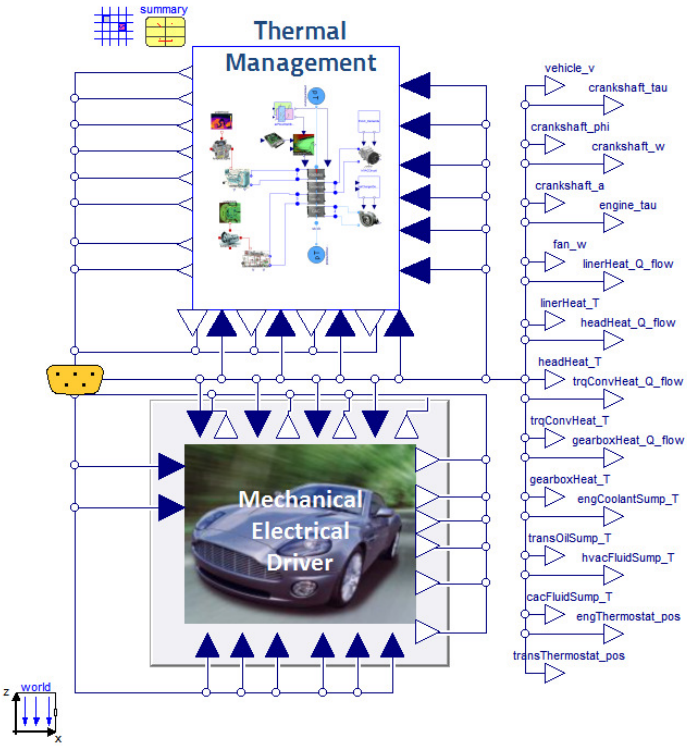


Figure 1 Complete system model

Model Systems

The key systems within the model are:

- Thermal Management
  - Lumped engine thermal mass
  - Lumped transmission thermal mass
  - Engine coolant fluid circuit
  - Transmission oil cooling circuit
  - Grill shutters and controller
  - Heat exchanger stack
- Mechanical/Electrical/Driver
  - Driver

    - Vehicle
        - Engine
        - Transmission
        - Driveline
        - Chassis
        - Aerodynamics
        - Other external loads
        - Brakes
    - Low voltage battery
    - Alternator
    - Cooling fan and controller

Model Features

The key features of these systems are:

- Driver
    - closed loop speed control for drive cycle following with automatic gear shifting
    - Standard list of drive cycles
- Vehicle – Engine
    - Torque map from throttle and speed with heat generation
- Vehicle – Transmission
    - Standard transmission with efficiency losses for heat generation
- Vehicle – Driveline
    - Rear wheel drive with parameterized final drive ratio
- Vehicle – Chassis
    - VDL compatible interfaces
    - 1D longitudinal dynamics
    - Ideal suspension and wheels
- Fluid coolant circuits
    - Heat transfer pathways with thermal masses
    - Crankshaft pump loads
    - Thermostat fluid control
    - Fluid flow resistances
- Heat exchanger stack
    - Parameterized geometry, efficiency, resistances
    - Stack ordering effects
    - Air flow effects due to vehicle speed, grill position, and fan speed
- Alternator
    - Crankshaft load
- Cooling fan and controller
    - Power supplied by alternator
- Grill shutters and controller
    - Variable position
    - Heat exchanger stack air flow modification

- Aerodynamic drag effects

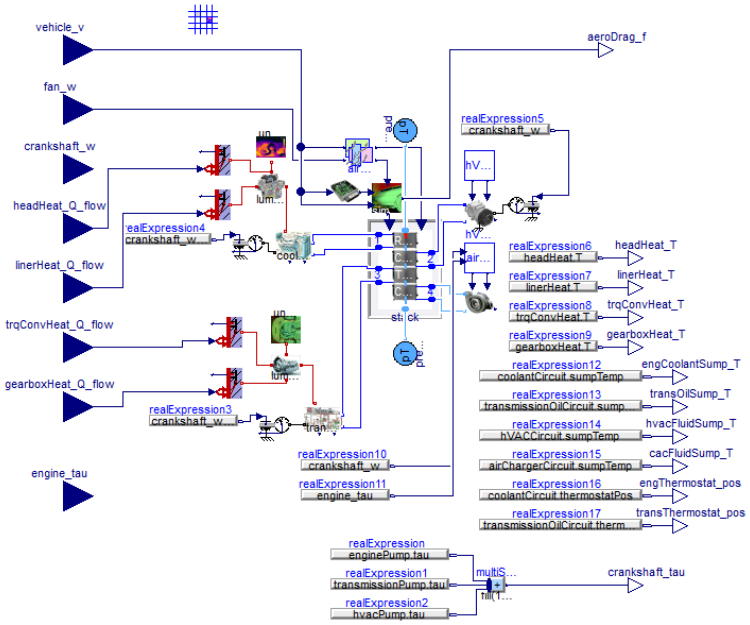


Figure 2 Thermal management system

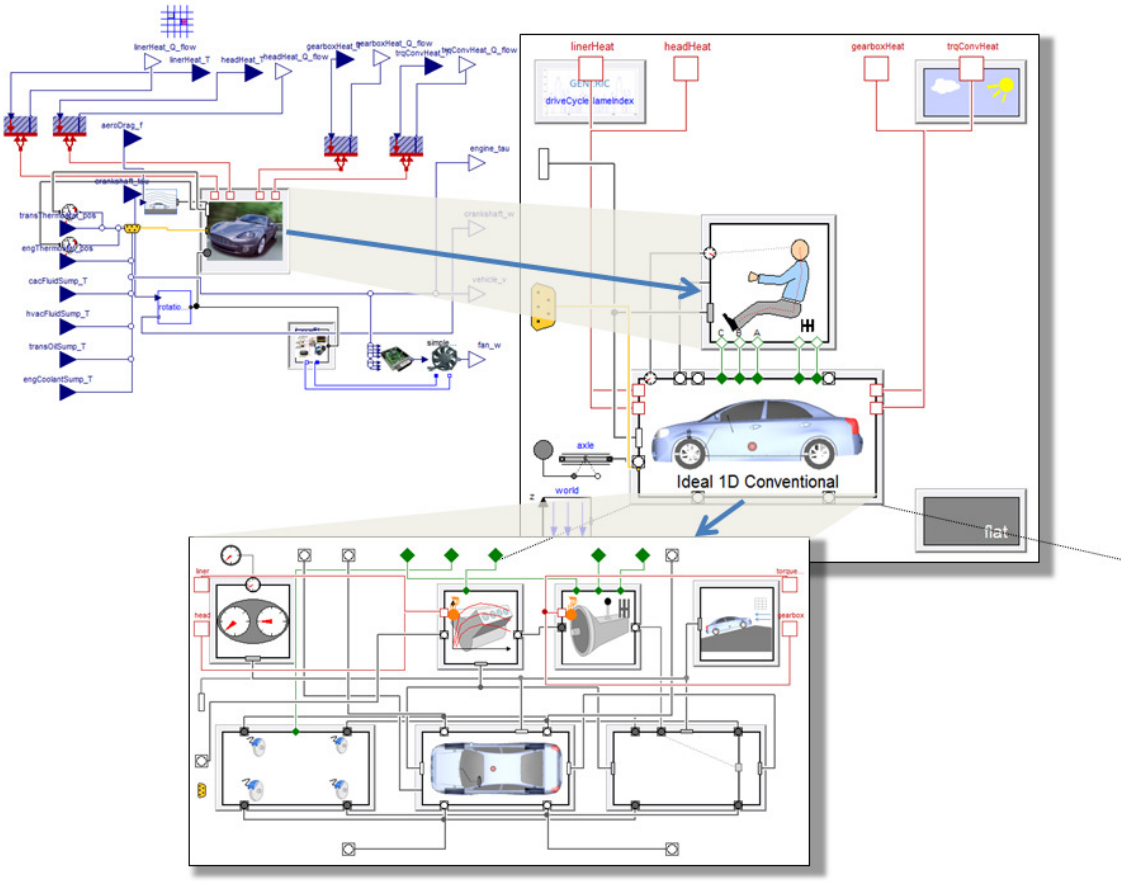


Figure 3 Mechanical, electrical, and driver system

Since the purpose of this model is to study vehicle thermal dynamics, a simplified 1D longitudinal dynamics chassis model is used rather than a full 3D body model. This approach allows for faster simulations of the typically long duration drive cycles.

During the simulation, heat generated by the engine is stored within the engine thermal mass and then rejected to the coolant-to-air heat exchanger (radiator) through a coolant fluid loop. A similar loop and heat exchanger also exists for the transmission oil.

The model is well suited to thermal management controller design, studying tradeoffs between thermal management energy demands and fuel economy, heat exchanger efficiency and sizing, and coolant fluid flow dynamics.

For this work, the model was partitioned into separate FMU executables by dividing the model along domain boundaries. In this case, the vehicle mechanics, electrical, and driver were grouped into one model while the fluid and thermal portions of the model were grouped into another. In order to achieve this structure, the physical connections that are bisected by the boundaries must be converted to causal signals. As an example for the engine, the heat is generated within the mechanical portion of the model.

The heat is directed to the lumped thermal model, within the thermal portion of the model, which determines the thermal mass temperature. Images of these two systems are shown below.

Key Parameters

Parameter Path	Default Value	Description
driverVehicle.vehicle.body_mass	1600 kg	Overall vehicle mass
driverVehicle.vehicle.R0_front	0.3 m	Front wheel radius
driverVehicle.vehicle.R0_rear	0.3 m	Rear wheel radius
driverVehicle.vehicle.wheel_i_xx_front	1 kg.m2	Individual front wheel inertia
driverVehicle.vehicle.wheel_i_xx_rear	1 kg.m2	Individual rear wheel inertia
driverVehicle.vehicle.driveline.rear_ratio	4	Final drive ratio
driverVehicle.vehicle.engine.J	0.1	Engine inertia
driverVehicle.vehicle.transmission.JA	0.04	Transmission engine side inertia
driverVehicle.vehicle.transmission.JB	0.07	Transmission driveline side inertia
driverVehicle.vehicle.c_w_front	0.38	Aerodynamic drag coefficient
driverVehicle.vehicle.a_front	2.7 m2	Vehicle frontal area
driverVehicle.vehicle.externalLoads.scale	1	Additional drag scale factor
driverVehicle.vehicle.engine.frictionScale	1.0	Engine friction scale factor
driverVehicle.vehicle.transmission.frictionScale	1.0	Transmission friction scale factor
driverVehicle.vehicle.engine.k_tau	1.5	Engine torque scale factor (note does not affect fuel consumption)
driverVehicle.vehicle.brakes.brake_*N*.clamp_factor	2	Individual brake clamping factor (for *N* = wheel number 1 – 4)
driverVehicle.vehicle.brakes.brake_*N*.mu	0.3	Individual brake coefficient of friction (for *N* = wheel number 1 – 4)
driverVehicle.vehicle.brakes.brake_*N*.J_disc	0.05 kg.m2	Individual brake inertia (for *N* = wheel number 1 – 4)

File Structure

The tutorial files are organized as shown in Figure 4. "Licenses" contains evaluation licenses for FMI Add-in for Excel and FMI Toolbox for MATLAB/Simulink. These licenses are good through October 20, 2014. The installers for the various software components are in the "Software" folder. The "Demos" folder contains all the relevant files for the tutorials using the different tools with the required FMUs in the "FMUs" folder. Note that the FMUs expire on Nov. 5, 2014.

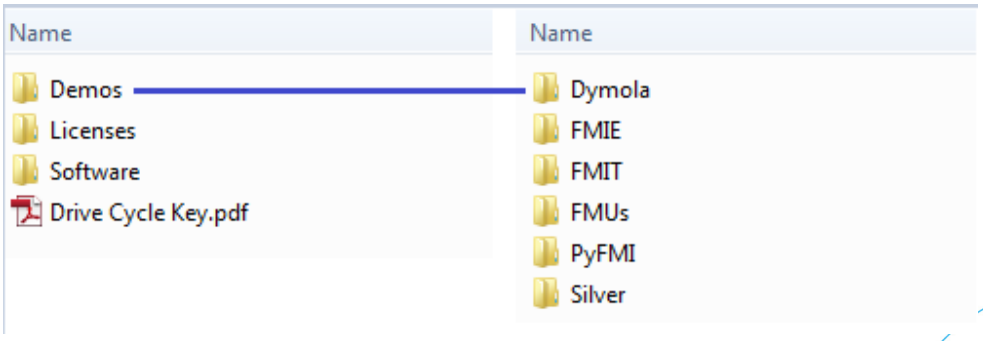
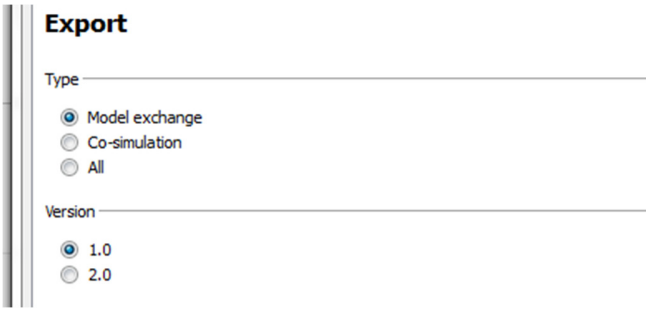


Figure 4 Tutorial files

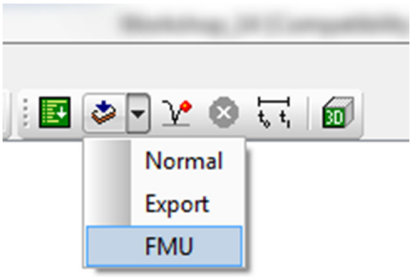
Demonstration: FMU export from Dymola (optional , requires Dymola)

This tutorial walks through the steps to create both a Model Exchange (ME) and Co-Simulation (CS) FMU with Dymola with a sample model from the Modelica Standard Library. The resulting FMU file structure is also explained. Note that this tutorial is optional as FMUs have already been provided as part of the workshop material for import into other tools.

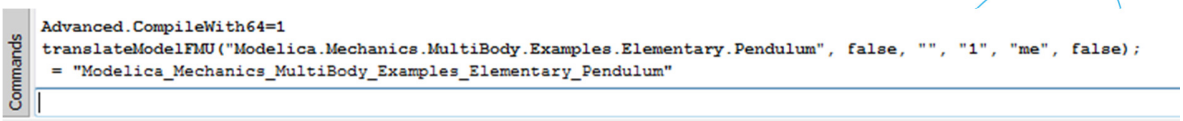
1) Go to the *Simulation* > *Setup* > *FMI* dialog in Dymola and change the export properties so that Dymola generates Model Exchange FMUs of FMI version 1.0.



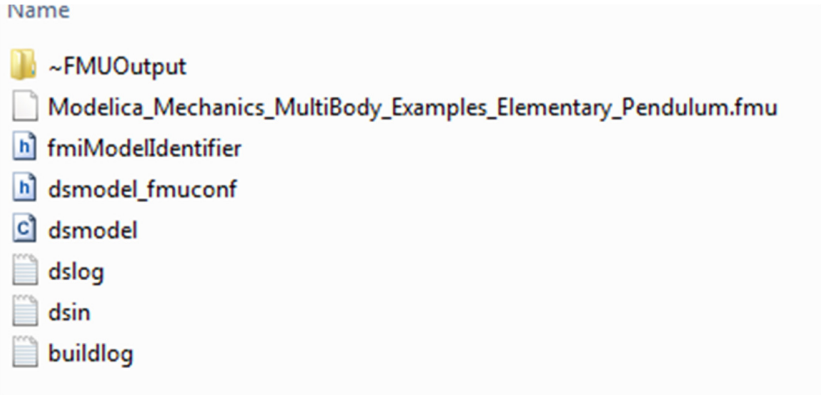
2) Press OK to close the dialog
3) Open Modelica.Mechaniscs.MultiBody.Examples.Elementary.Pendulum from the package browser. Then switch to Simulation mode of Dymola.
4) Click on the little arrow next to the Translate button and select FMU to translate the model into a Model Exchange FMU of version 1.0. You may need to force Dymola into generating a 32bit FMU in case you are using a 32bit compiler together with 64bit Dymola by selecting the appropriate Binaries option from the FMI tab in Simulation Setup. The recommendation is to use 32bit Dymola if you only have a 32bit compiler installed (free Visual Studio compilers). 64bit FMU compilation is possible with Visual Studio 2012 express and newer, older versions of the free (express) editions of Visual Studio only support 32 bit.



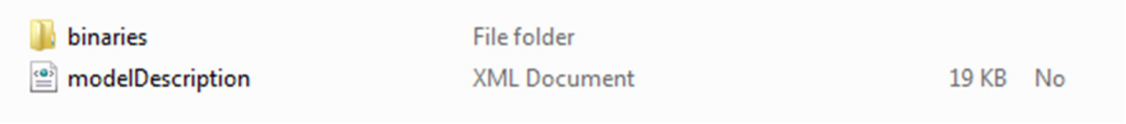
Note that Dymola translates the translate action into a command translateModelFMU() and runs it from the command prompt. This command can be useful if you want to generate FMUs directly from the command line or from Modelica scripts.



5) Navigate to the folder where the FMU has been generated (your current directory). A file Modelica_Mechanics_MultiBody_Examples_Elementary_Pendulum.fmu should be present in the folder.



6) Unzip the FMU with a standard zip tool. Change the file extension of the file to .zip to use the built in zip tool in Windows.
7) The FMU consists of a modelDescription.xml file and a folder with binaries (compiled model). Open the modelDescription.xml and investigate its contents. Rename the file "modelexchangeXML.xml" and save it to a separate folder (not to the FMU).



8) Go back to Dymola and open the *Simulation* > *Setup* > *FMI* dialog. Change the properties to export Co-Simulation FMUs of version 1.0.
9) Repeat step 2-7 and compare the content of the Co-Simulation XML file with the contents of modelexchangeXML.xml. In addition to the information of the Model Exchange XML file, the CS XML file also contains a section <*Implementation*> with some general information about the solver at the very end of the file.

```
  <Implementation>
    <CoSimulation_StandAlone>
      <Capabilities
        canHandleVariableCommunicationStepSize="true"
        canHandleEvents="true"
        canBeInstantiatedOnlyOncePerProcess="true"/>
    </CoSimulation_StandAlone>
  </Implementation>
</fmiModelDescription>
```

With other programs that can generate and export FMUs, the process is fairly similar even though the user interface and details can vary. In every tool, the options are similar as they are governed by the standard, and most tools will do the same basic process: they compile the model into an executable dll that can be deployed to other CAE platforms.

Demonstration: FMU import in Dymola (optional , requires Dymola)

This tutorial walks through the steps to import an FMU into Dymola and simulate.

1) In Dymola, change the FMU import type to Co-simulation, and in order to save time for this example, disable the import of all variables. These settings are in the FMI tab of the Simulation Setup dialog. The format of this tab has changed slightly for the later versions of Dymola with the inclusion of a couple different options, but in general it should look like the following figure.

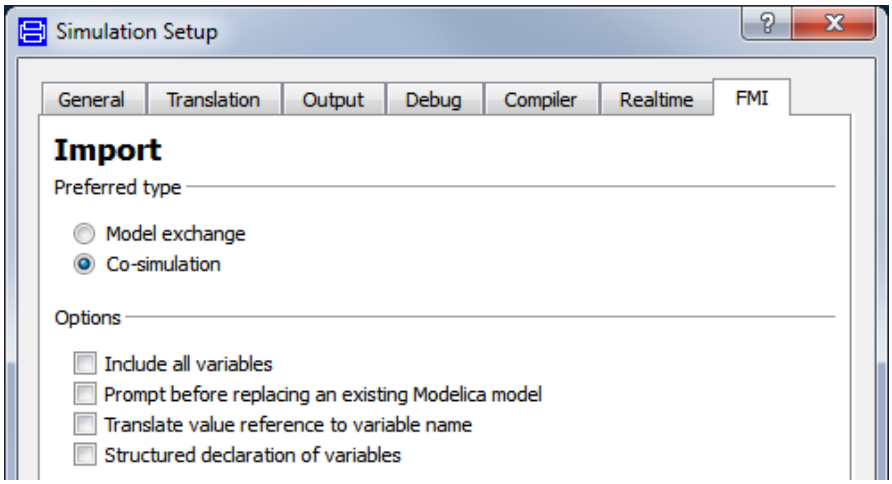


Figure 5 Dymola FMU import settings dialog

2) Import the vehicle model FMU *VTMModels_Ideal1DSedanCycleFollower.fmu*. From Dymola, click *File* > *Import* > *FMU* and select the FMU from the FMUs\CS directory as shown below.

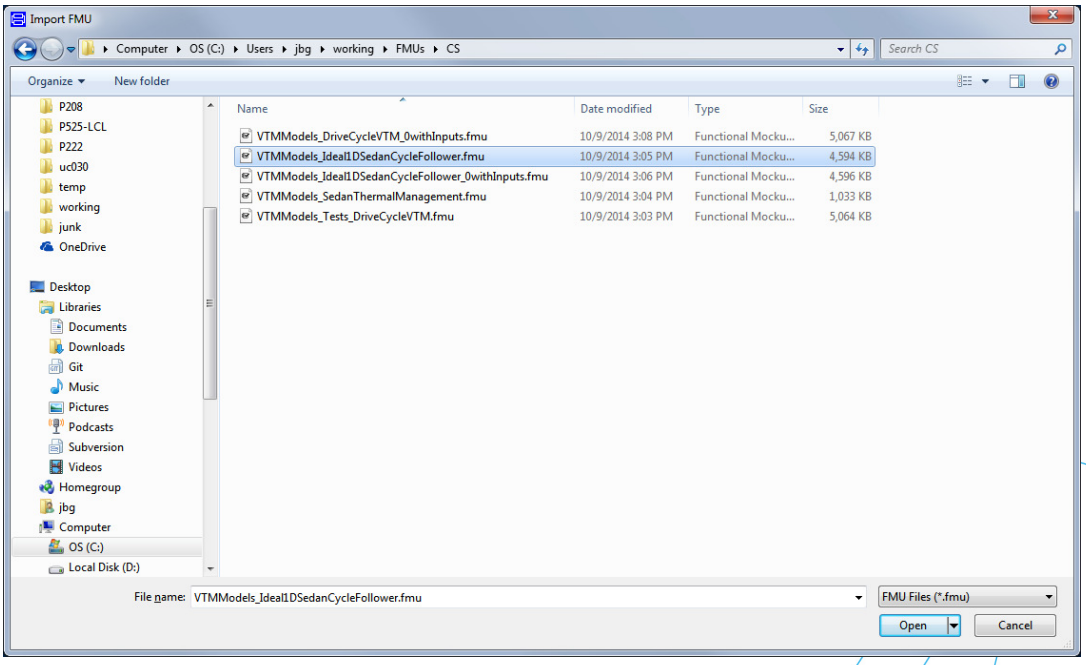


Figure 6 Dymola FMU file import dialog

Note that the action is translated into the command `importFMU()` that can also be run from the command prompt in Dymola.

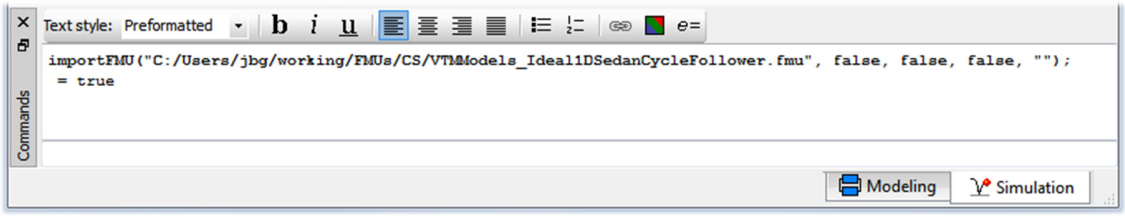


Figure 7 Dymola importFMU command

3) Repeat the above two steps to import the thermal management FMU *VTMModels_SedanThermalManagement.fmu*.
4) The imported FMU models will appear in the *Package Browser*.

Note that there are some options that govern the appearance of FMUs after import. In Dymola 2015, if the option flag Advanced.FMI.OverlappingIO is set to true, inputs of the same type will be drawn on top of one another to improve the graphics for FMUs with many inputs. In Dymola 2015 FD01 this has been refined to an integer flag Advanced.FMI.OverlappingIOThreshold that will group signals of the same type if there are more of them than the threshold number.

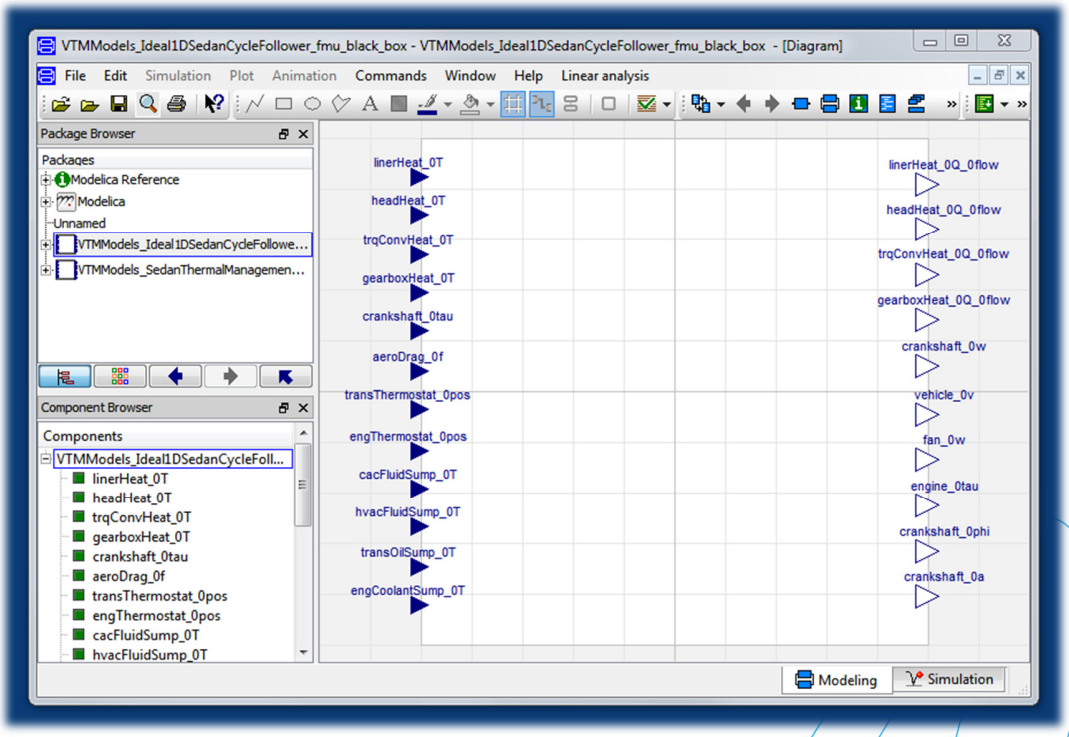


Figure 8 Dymola imported FMU instance

5) Load the *Demos\Dymola\FMUTests* Modelica package in Dymola.

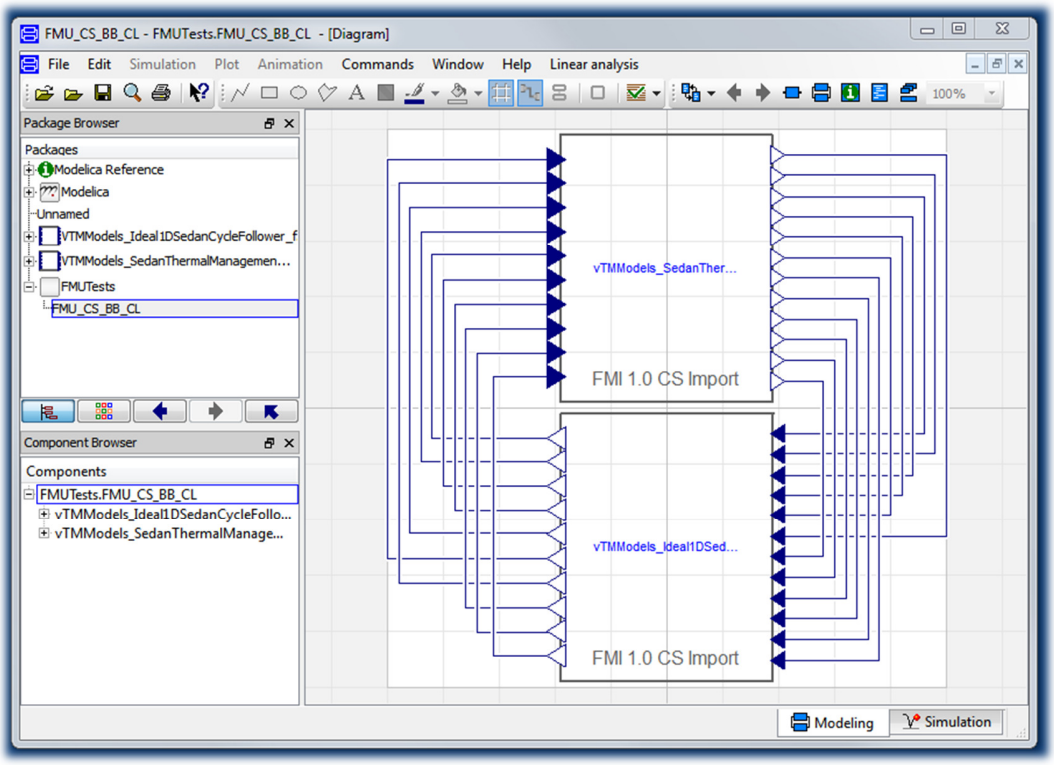


Figure 9 Dymola closed loop connected FMU model

The package contains the FMU_CS_BB_CL model with fully connected and parameterized instances of the vehicle and thermal management FMUs.

6) Change to the simulation tab in Dymola and simulate the model.
7) Generate a figure of the vehicle speed (the variable called vehicle_0v from either the vehicle or thermal management model).

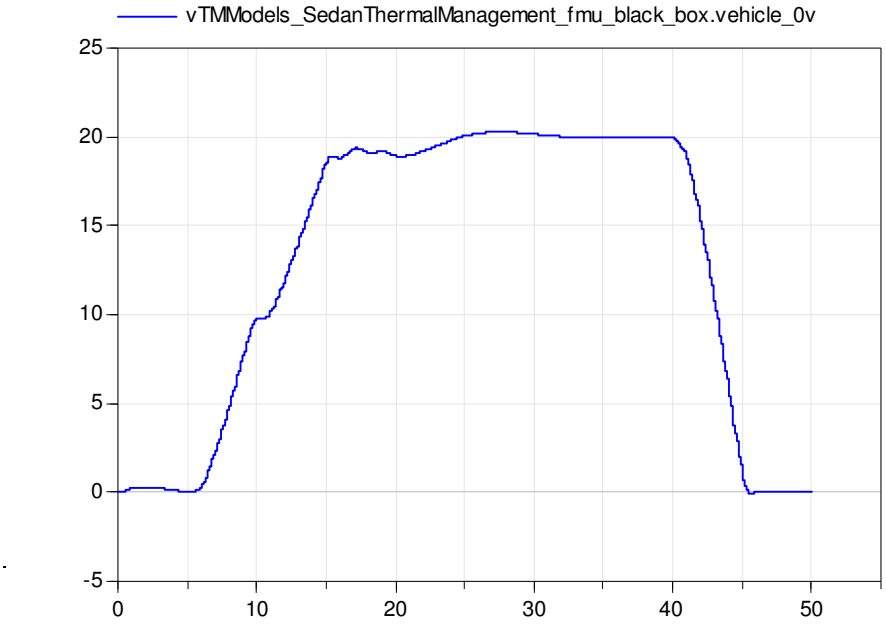


Figure 10 Dymola vehicle speed trajectory

Zoom in on a section of the trajectory and observe the signal. Note that the trajectory appears as a series of steps at the same rate as the communication interval set by the top level parameter *fmi_CommunicationStepSize*.

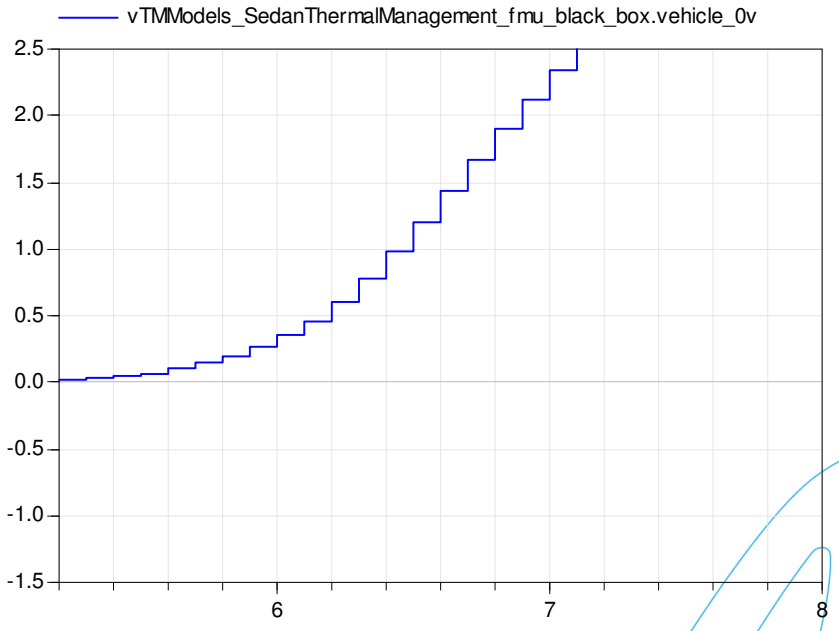


Figure 11 Dymola vehicle speed trajectory zoom

8) Observe the statistics section of the translation log. Note that it shows there are no continuous time states.

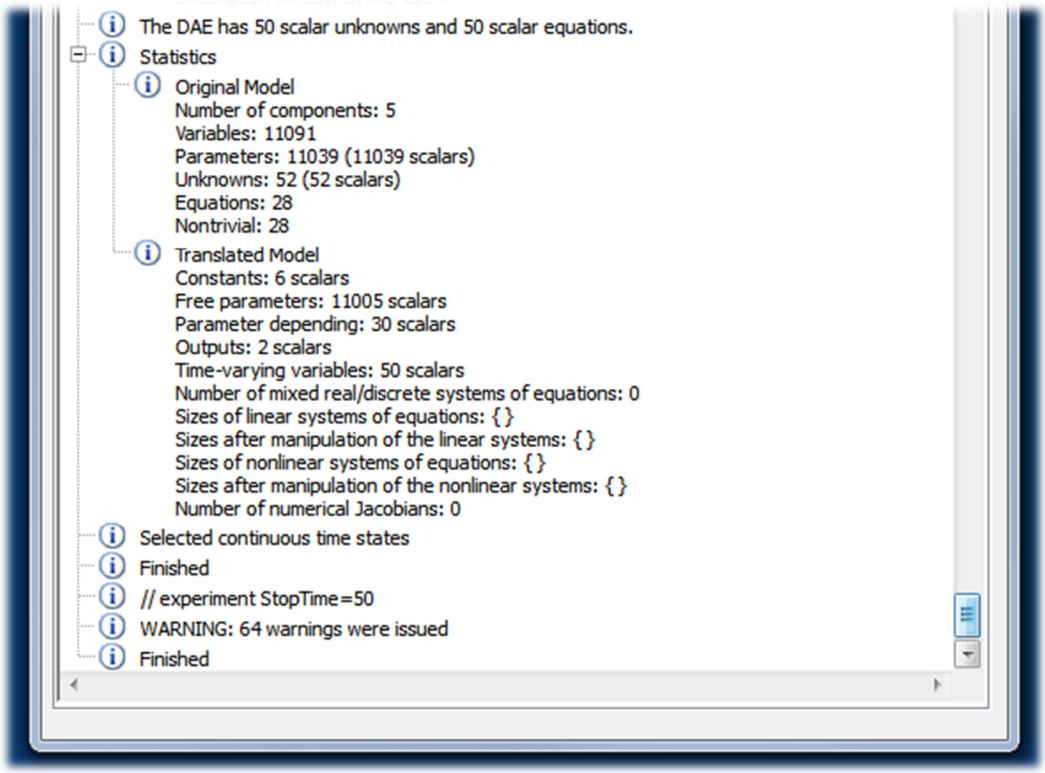


Figure 12 Dymola translation log

Demonstration: FMU import and simulation using PyFMI (optional, hands-on)

This tutorial shows the import of FMUs into PyFMI, an open source tool that provides a Python environment for FMI. Note that this tutorial requires the installation of JModelica which was provided as part of the tutorial material.

In this tutorial, we will run a parameter sweep with the vehicle model with inputs for the throttle command and the brake force. In the script, we will construct the inputs for a wide open throttle (WOT) run and look at the maximum vehicle speed while varying the final drive ratio.

1) *Preparation:* Install JModelica via installer *JModelica.org-1.15.exe* accepting all defaults.
2) Navigate to Demos\PyFMI folder for the workshop material in Windows Explorer. Type *cmd* in the file location field of the Explorer window to start the Windows command prompt in the PyFMI directory.
3) Type the command shown below to set some environment variables that Python needs assuming the default installation location for JModelica:

4) Execute the script *ParameterSweep.py* by typing the command shown below assuming the default installation location for Python (note that the full path to Python is used here as there is no assumption that Python is added to the Windows Path environment variable):

```
C:\Temp\Demos\PyFMI>C:\Python27\python.exe ParameterSweep.py
```

5) The simulations will execute and provide feedback in the command window. Once the simulations are complete, examine the plots for vehicle speed, gear command, engine speed, and engine torque as a function of time for the simulations. A plot is also included that shows the maximum vehicle speed as a function of the final drive ratio. Does it make sense to run different drive ratios with the same transmission shift control logic? Why is there a sudden drop in engine torque for one of the simulations?
6) Open the Python script in a text editor and look at the various sections. The script is commented to show the key syntax for defining the sweep parameters, providing the inputs, loading the FMU, setting the parameters, simulating the FMU, and extracting the results.

Optional exercises:

- Increase the number of points in the sweep to better understand the shape of the max vehicle speed vs. final drive curve.
- Refine the parameter sweep around the maximum vehicle speed to better identify the "optimum" final drive ratio (assuming same control strategy, etc.).
- Investigate 0-60 mph time (~26.8 m/s) by shortening the simulation time. Does the same drive ratio that maximized vehicle speed also give the best 0-60 time?
- Construct different throttlePostion inputs to simulate different types of drives.
- Copy the script and modify it to investigate the impact of vehicle mass on max vehicle speed and 0-60 time. The vehicle mass parameter is *driverVehicle.vehicle.body_mass*.

PyFMI includes a number of different examples of FMI-based workflows including compilation of FMUs, different simulations with FMUs, etc. This tutorial is meant to only show the basics for loading and simulating an FMU. More information about PyFMI can be found at http://www.jmodelica.org/page/4924.

Hands-on: FMU import and simulation using FMI Add-in for Excel®

This tutorial demonstrates an FMI based analysis workflow using Microsoft Excel® and the FMI Add-in for Excel® (FMIE) from Modelon. A demo license of the add-in will be provided as part of this tutorial. Note that FMI Add-in for Excel requires 32-bit Microsoft Excel which is the default for most Office installations.

1) *Preparation:* Install the FMI Add-in for Excel® via installer *FMI Add-In for Excel 1.3.3.exe* accepting all defaults. This installation will add the **FMI** entry to the ribbon bar. The demo license for FMIE is in the Licenses folder of the demo distribution and should be placed in the AppData folder as described in the FMIE User's Guide (i.e.

C:\Users\username\AppData\Roaming\Modelon\Licenses\Nodelocked). To activate the license, perform the following steps:

- In the FMI Ribbon tab shown below, click on License.
- Using the drop-down list next to the label **Active feature**, select “Standard”.
- Click the button **Change...** and select "OK" when prompted with a verification question.
- Restart Excel for the changes to take effect.

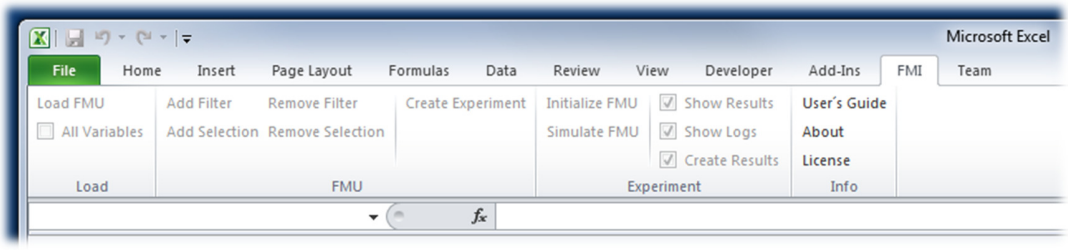


Figure 13 FMI Ribbon Tab

Note that the FMI tab contains different groupings of buttons. Some of the buttons are disabled by default. They are enabled when the active sheet contains data in a matching format.

The following steps document the process used to generate the Demos/FMIE/DriveCycleVTM.xlsx Excel® workbook. The workbook is already available with all the FMI related worksheets and simulation results completed. Interested users can feel free to go through the steps separately to investigate the process further.

2) Start a new workbook in Excel® and click the **Load FMU** button on the FMI tab to open a File Open dialog box. Browse to the *VTMModels_Tests_DriveCycleVTM.fmu* file within the *FMUs/CS* folder and click Open. This FMU is a co-simulation FMU based on the model shown in Figure 1.

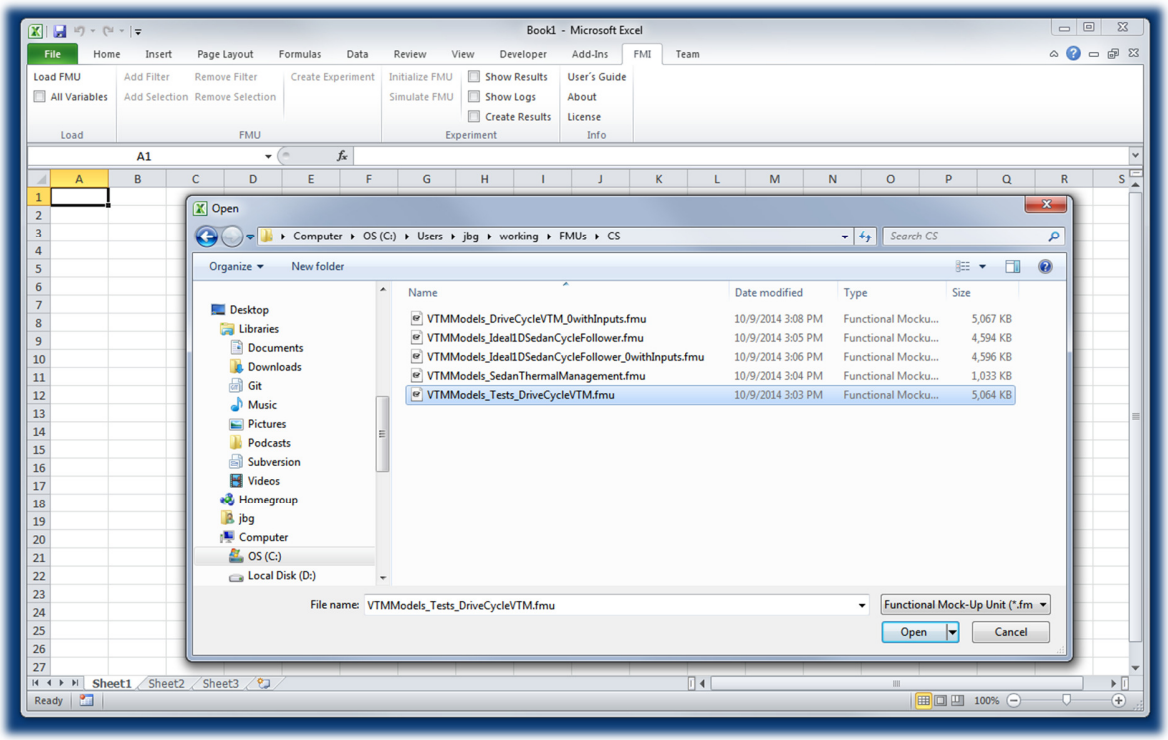
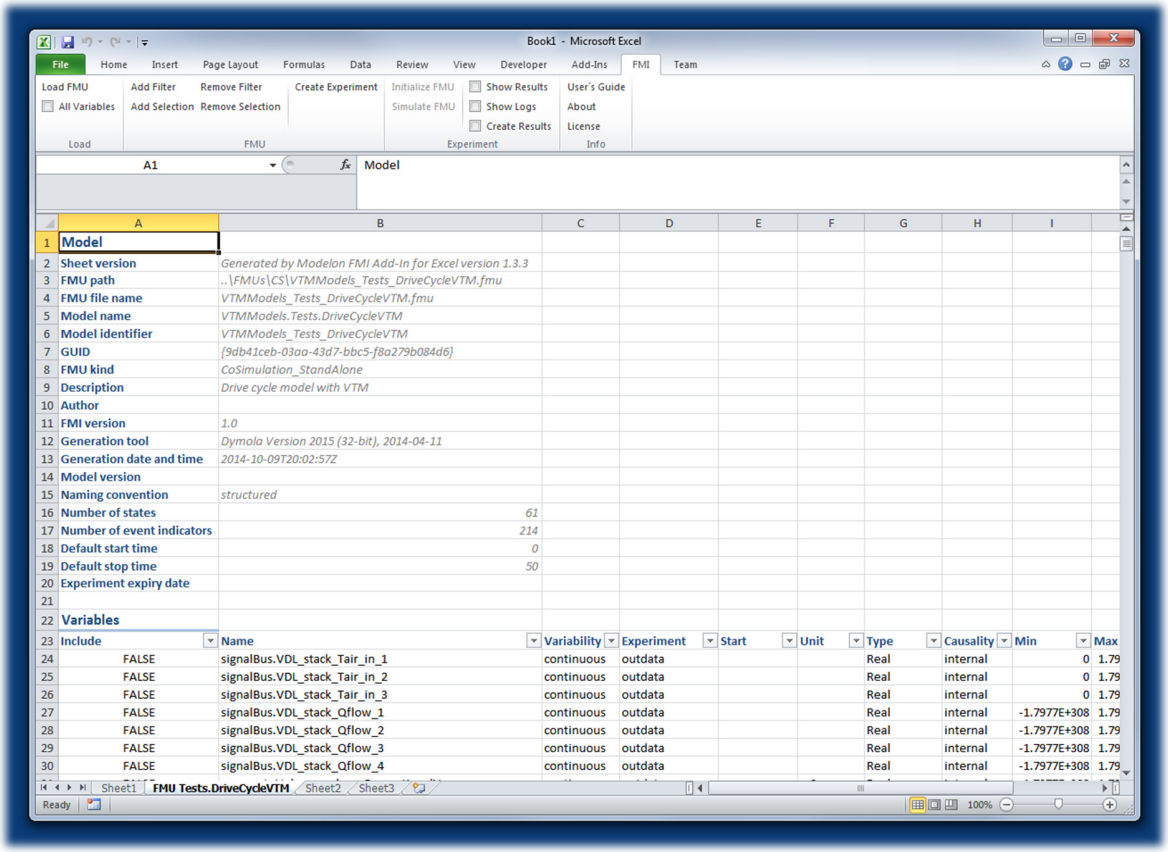


Figure 14 FMU Open Dialog

3) A new sheet is created with the prefix “FMU” (e.g. FMU Tests.DriveCycleVTM). There are two areas in the new sheet, one for model metadata ("Model") and one for the variables in the FMU ("Variables"), see Figure 15. The metadata section contains general information about the FMU, e.g. model name, FMU kind, number of state variables, etc. The Variables area lists the variables with name and their corresponding attributes, including start value, unit, data type, etc. There are two columns that do not correspond to any FMI attribute: the "Include" column and the "Experiment" column. The Include column indicates if the variable will be included in the experiment or not. The Experiment column indicates if a variable will be used as an input or an output.



	A	B	C	D	E	F	G	H	I
1	**Model**								
2	**Sheet version**	*Generated by Modelon FMI Add-In for Excel version 1.3.3*							
3	**FMU path**	*..\FMUs\CS\VTMModels_Tests_DriveCycleVTM.fmu*							
4	**FMU file name**	*VTMModels_Tests_DriveCycleVTM.fmu*							
5	**Model name**	*VTMModels.Tests.DriveCycleVTM*							
6	**Model identifier**	*VTMModels_Tests_DriveCycleVTM*							
7	**GUID**	*{9db41ceb-03aa-43d7-bbc5-f8a279b084d6}*							
8	**FMU kind**	*CoSimulation_StandAlone*							
9	**Description**	*Drive cycle model with VTM*							
10	**Author**								
11	**FMI version**	*1.0*							
12	**Generation tool**	*Dymola Version 2015 (32-bit), 2014-04-11*							
13	**Generation date and time**	*2014-10-09T20:02:57Z*							
14	**Model version**								
15	**Naming convention**	*structured*							
16	**Number of states**	*61*							
17	**Number of event indicators**	*214*							
18	**Default start time**	*0*							
19	**Default stop time**	*50*							
20	**Experiment expiry date**								
21									
22	**Variables**								
23	**Include**	**Name**	**Variability**	**Experiment**	**Start**	**Unit**	**Type**	**Causality**	**Min**	**Max**
24	FALSE	signalBus.VDL_stack_Tair_in_1	continuous	outdata			Real	internal	0	1.79
25	FALSE	signalBus.VDL_stack_Tair_in_2	continuous	outdata			Real	internal	0	1.79
26	FALSE	signalBus.VDL_stack_Tair_in_3	continuous	outdata			Real	internal	0	1.79
27	FALSE	signalBus.VDL_stack_Qflow_1	continuous	outdata			Real	internal	-1.7977E+308	1.79
28	FALSE	signalBus.VDL_stack_Qflow_2	continuous	outdata			Real	internal	-1.7977E+308	1.79
29	FALSE	signalBus.VDL_stack_Qflow_3	continuous	outdata			Real	internal	-1.7977E+308	1.79
30	FALSE	signalBus.VDL_stack_Qflow_4	continuous	outdata			Real	internal	-1.7977E+308	1.79

Figure 15 FMU Worksheet

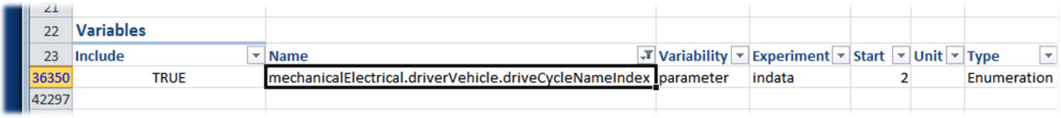
Note that the **Create Experiment** button in the ribbon was enabled when the FMU sheet was created. It is only enabled when the active sheet is an FMU sheet. Also, by default the FMU path field includes the full system path to the FMU. In this tutorial image the path is shown relative to the Excel® working folder. The working folder can be determined by clicking File → Open. The initial path shown in the Open dialog is the current working folder. The working folder is changed to the folder of files opened by this dialog. It can often be easiest to keep the FMU paths as absolute when not sharing workbooks. For example, if this FMU was placed in a sibling folder called FMU relative to the workbook, the relative path would be *FMU\VTMModels_Tests_DriveCycleVTM.fmu.*

4) The value in the Experiment column indicates whether a variable is included or not in an experiment. By default, no variable is included. Variables will be included by filtering the names and setting the Include field to TRUE.
   a. Using the list filter in the Name column search for “driveCycleNameIndex”.

21							
22	**Variables**						
23	**Include**	**Name**	**Variability**	**Experiment**	**Start**	**Unit**	**Type**
36350	FALSE	mechanicalElectrical.driverVehicle.driveCycleNameIndex	parameter	indata	2		Enumeration
42297							

Figure 16 FMIE filtered Name list

This step should result in at least one entry found with “parameter” variability. Select any cell in this row and click **Add Filter** from the FMI tab. This action will mark the Include field of all the variables found with the filter as TRUE.

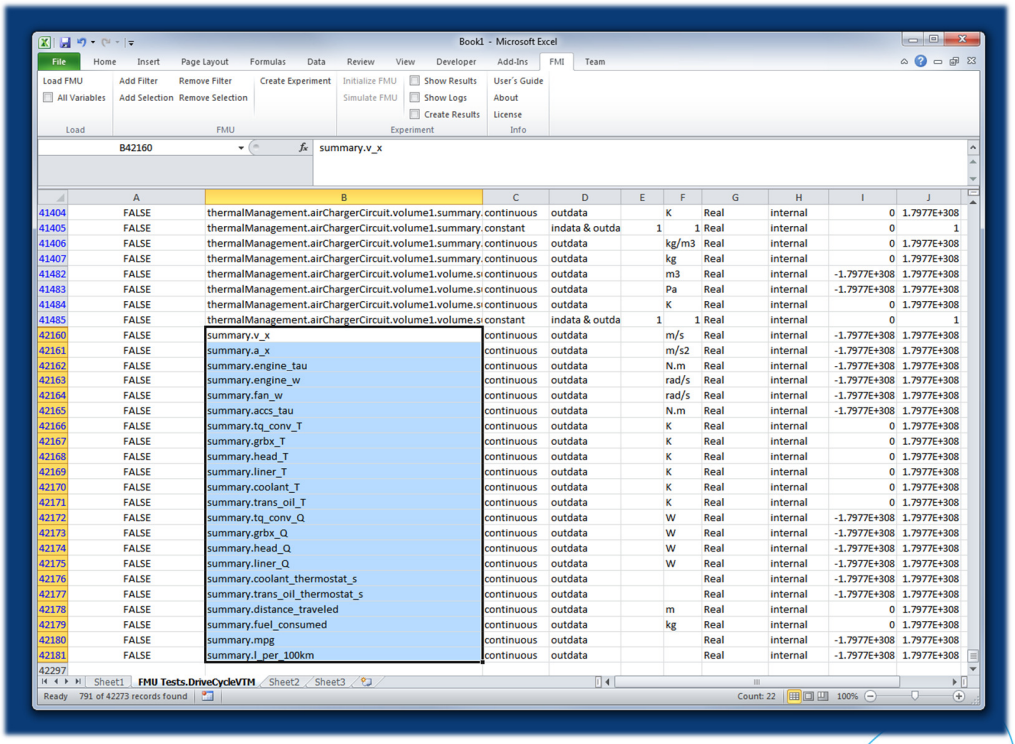


22	Variables						
23	Include	Name	Variability	Experiment	Start	Unit	Type
36350	TRUE	mechanicalElectrical.driverVehicle.driveCycleNameIndex	parameter	indata	2		Enumeration
42297							

Figure 17 FMIE toggled variable include flag

b. Now search for “driveCycleScale” and “rear_ratio” from the Name column and repeat the above procedure to include the parameters in the experiment.

c. Finally some output (“outdata”) variables must also be included in the experiment. Search for “summary” in the Name column. This filter should yield many valid entries. Toward the bottom of the list will be the top level summary record entries as shown in the following figure.



41404	FALSE	thermalManagement.airChargerCircuit.volume1.summary.	continuous	outdata		K	Real	internal	0	1.7977E+308
41405	FALSE	thermalManagement.airChargerCircuit.volume1.summary.	constant	indata & outda	1	1	Real	internal	0	1
41406	FALSE	thermalManagement.airChargerCircuit.volume1.summary.	continuous	outdata		kg/m3	Real	internal	0	1.7977E+308
41407	FALSE	thermalManagement.airChargerCircuit.volume1.summary.	continuous	outdata		kg	Real	internal	0	1.7977E+308
41482	FALSE	thermalManagement.airChargerCircuit.volume1.volume.s	continuous	outdata		m3	Real	internal	-1.7977E+308	1.7977E+308
41483	FALSE	thermalManagement.airChargerCircuit.volume1.volume.s	continuous	outdata		Pa	Real	internal	-1.7977E+308	1.7977E+308
41484	FALSE	thermalManagement.airChargerCircuit.volume1.volume.s	continuous	outdata		K	Real	internal	0	1.7977E+308
41485	FALSE	thermalManagement.airChargerCircuit.volume1.volume.s	constant	indata & outda	1	1	Real	internal	0	1
42160	FALSE	summary.v_x	continuous	outdata		m/s	Real	internal	-1.7977E+308	1.7977E+308
42161	FALSE	summary.a_x	continuous	outdata		m/s2	Real	internal	-1.7977E+308	1.7977E+308
42162	FALSE	summary.engine_tau	continuous	outdata		N.m	Real	internal	-1.7977E+308	1.7977E+308
42163	FALSE	summary.engine_w	continuous	outdata		rad/s	Real	internal	-1.7977E+308	1.7977E+308
42164	FALSE	summary.fan_w	continuous	outdata		rad/s	Real	internal	-1.7977E+308	1.7977E+308
42165	FALSE	summary.accs_tau	continuous	outdata		N.m	Real	internal	-1.7977E+308	1.7977E+308
42166	FALSE	summary.tq_conv_T	continuous	outdata		K	Real	internal	0	1.7977E+308
42167	FALSE	summary.grbx_T	continuous	outdata		K	Real	internal	0	1.7977E+308
42168	FALSE	summary.head_T	continuous	outdata		K	Real	internal	0	1.7977E+308
42169	FALSE	summary.liner_T	continuous	outdata		K	Real	internal	0	1.7977E+308
42170	FALSE	summary.coolant_T	continuous	outdata		K	Real	internal	0	1.7977E+308
42171	FALSE	summary.trans_oil_T	continuous	outdata		K	Real	internal	0	1.7977E+308
42172	FALSE	summary.tq_conv_Q	continuous	outdata		W	Real	internal	-1.7977E+308	1.7977E+308
42173	FALSE	summary.grbx_Q	continuous	outdata		W	Real	internal	-1.7977E+308	1.7977E+308
42174	FALSE	summary.head_Q	continuous	outdata		W	Real	internal	-1.7977E+308	1.7977E+308
42175	FALSE	summary.liner_Q	continuous	outdata		W	Real	internal	-1.7977E+308	1.7977E+308
42176	FALSE	summary.coolant_thermostat_s	continuous	outdata			Real	internal	-1.7977E+308	1.7977E+308
42177	FALSE	summary.trans_oil_thermostat_s	continuous	outdata			Real	internal	-1.7977E+308	1.7977E+308
42178	FALSE	summary.distance_traveled	continuous	outdata		m	Real	internal	0	1.7977E+308
42179	FALSE	summary.fuel_consumed	continuous	outdata		kg	Real	internal	0	1.7977E+308
42180	FALSE	summary.mpg	continuous	outdata			Real	internal	-1.7977E+308	1.7977E+308
42181	FALSE	summary.l_per_100km	continuous	outdata			Real	internal	-1.7977E+308	1.7977E+308

Figure 18 Filtered summary record variables

Select all the desired output variables then click the **Add Selection** button from the FMI tab. This step will mark all the selected variables as included. Any valid multi-selection combination of cells (adjacent or not) will work. Likewise the **Remove Filter** and **Remove Selection** buttons can be used to toggle the Include field to FALSE. Manually toggling the Include field is also valid although a bit tedious when selecting many variables.

Finally verify the included variable list by filtering on the Include column for variables marked TRUE. The list should be similar to the following. Note that it includes variables that are inputs (indata) as well as outputs (outdata).

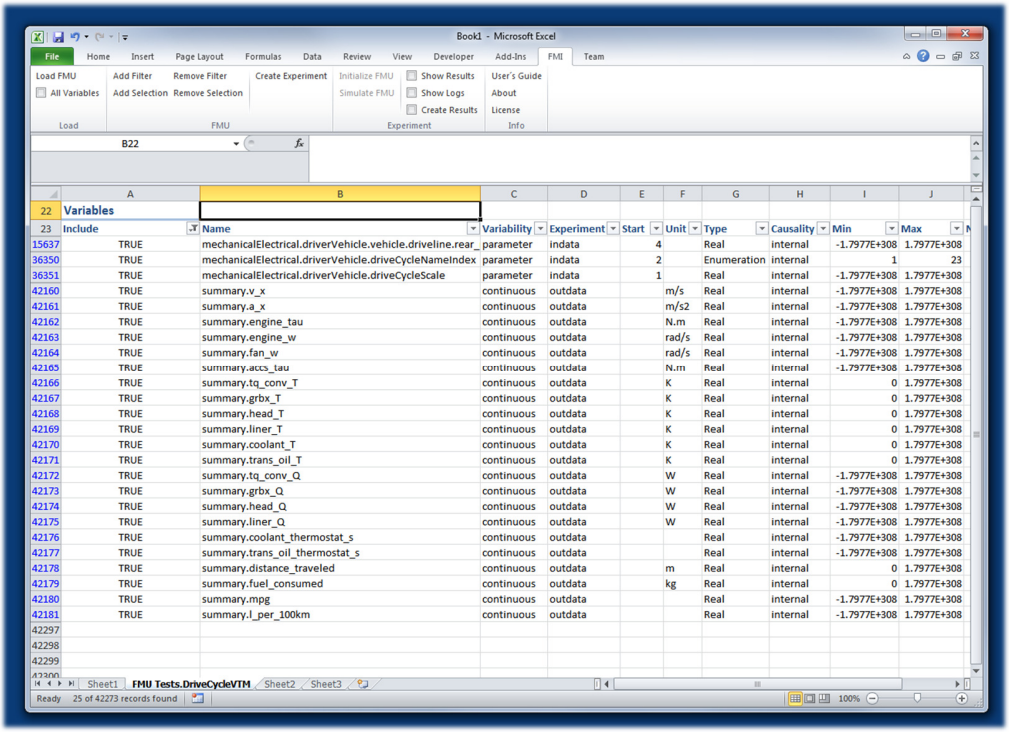


Figure 19 FMIE included variable list

5) Now that the desired variables have been marked as included, click the **Create Experiment** button from the FMI ribbon tab. This generates a new sheet with the prefix “Exp”. In this case the sheet name is “Exp Tests.DriveCycleVTM”. The sheet has four areas, "Model", "Settings", "Indata" and "Outdata". The Indata and Outdata areas were generated based on the variables in the Load FMU sheet with their Include value set to TRUE and their corresponding Experiment selections. The number of simulation cases is defined by columns called "Case 1", "Case 2", etc. Three cases are generated by default, and more can be added by inserting columns. To ensure that the ranges in the Excel sheet are maintained properly, select any cell in a column corresponding to a simulation case except “Case 1”.

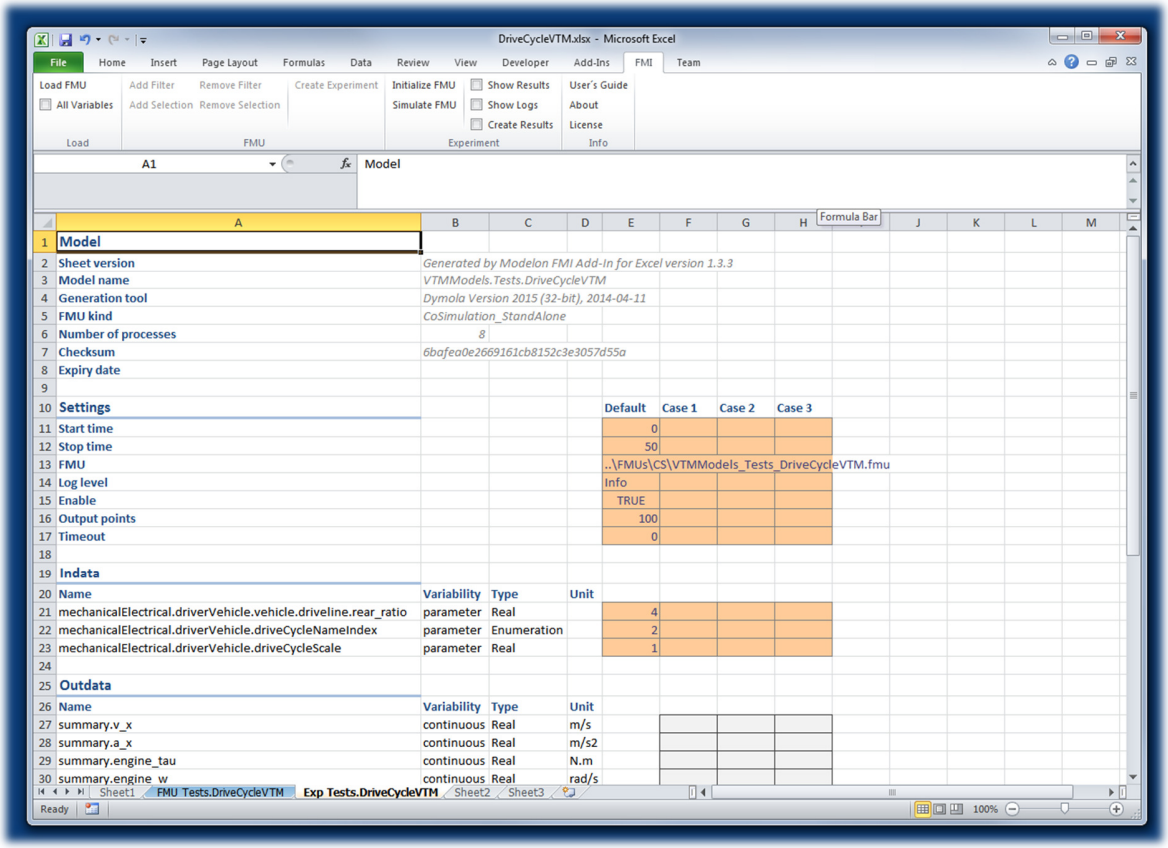


Figure 20 FMIE experiment sheet

6) This experiment will test the effects of the vehicle final drive ratio on the vehicle fuel economy during a UDDS drive cycle test.
   a. The UDDS cycle is number 19 (refer to the Drive Cycle Key.pdf document for a list of predefined cycles and their profiles). Change the *driveCycleNameIndex* default value from 2 to 19.
   b. Since the duration of the UDDS cycle is approximately 1370 seconds, increase the *Stop time* in the Default column to 1400. Also, to capture a reasonable resolution of the results increase the *Output points* in the Default column to 2800. This value will capture data at ½ second intervals.
   c. Next, to test the effects of varying the final drive ratio, enter the values 3, 4, and 5 for the *rear_ratio* parameter for cases 1, 2, and 3 respectively. Note that the default value of a parameter is overwritten by entering values in a case column.
   d. Finally enable result trajectory creation by checking the **Show Results** and **Create Results** checkboxes on the FMI ribbon tab. By default these fields are unchecked and only the final value of output data is returned to the workbook. When these fields are checked, the full time trajectory (at the selected output interval rate) is returned in a new sheet with the prefix "Res".
   e. The final experiment sheet should look similar to the following:

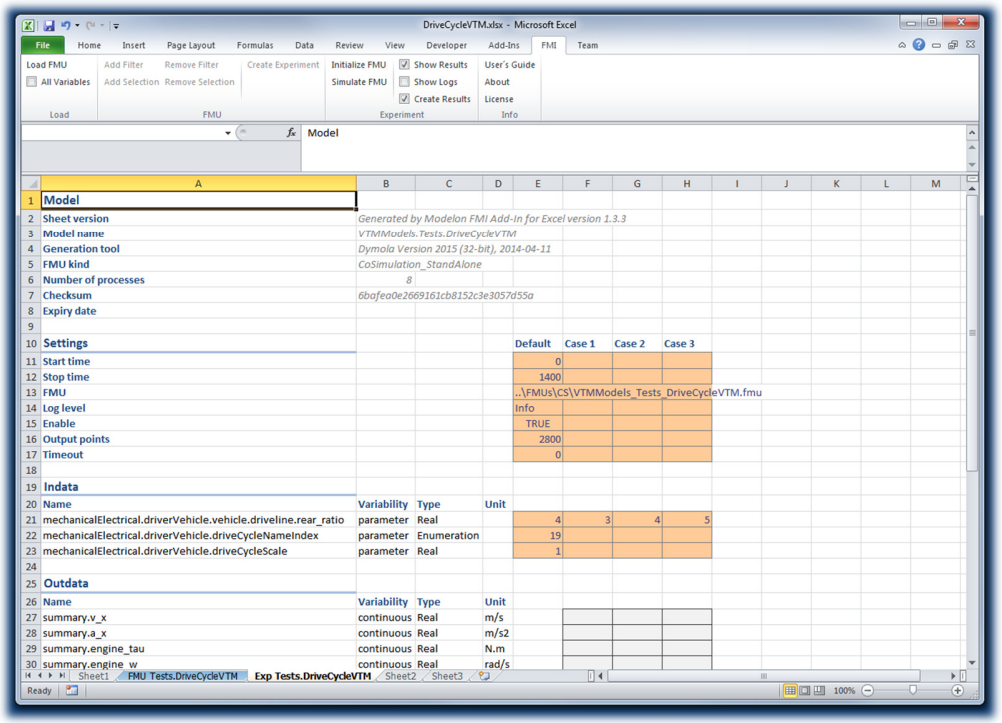


Figure 21 FMIE fully specified experiment sheet

This experiment sheet will run 3 separate simulations of the UDDS cycle with final drive ratios of 3:1, 4:1, and 5:1 respectively.

7) Now that the experiment sheet is created and ready, click the **Initialize FMU** button on the FMI ribbon tab. If successful, the worksheet should now show the initial values for each of the output data variables and the **Message** row should show *OK*.

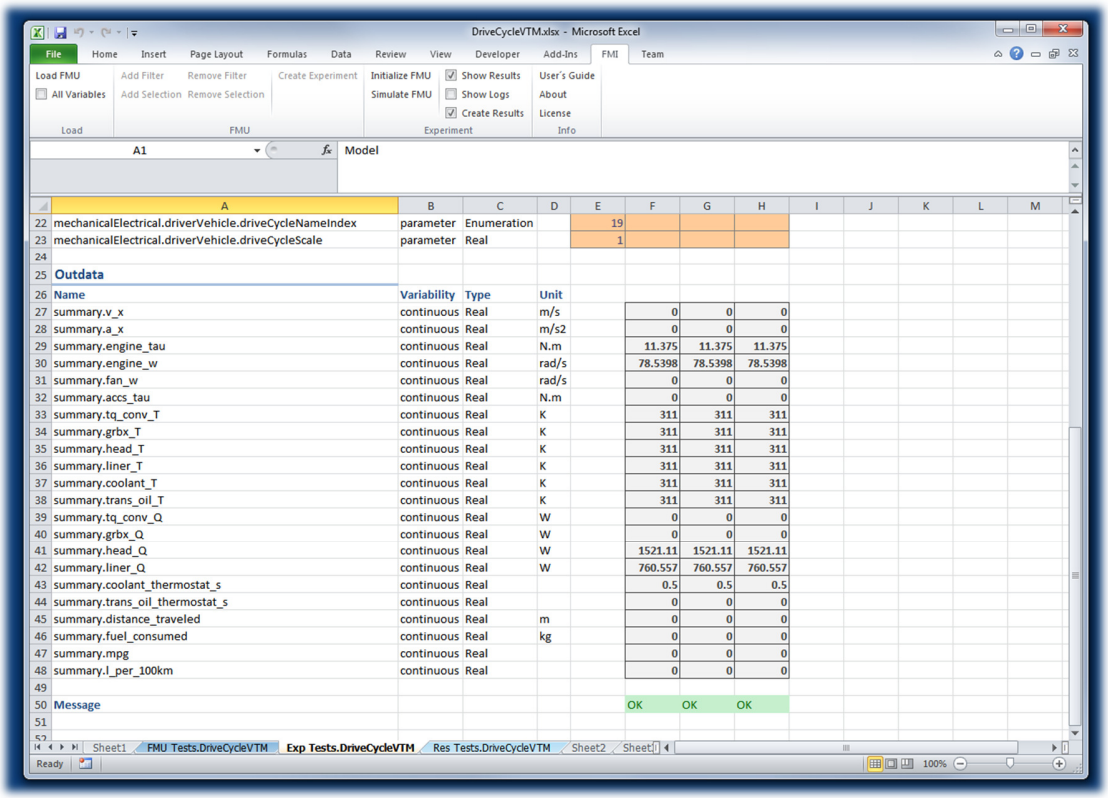
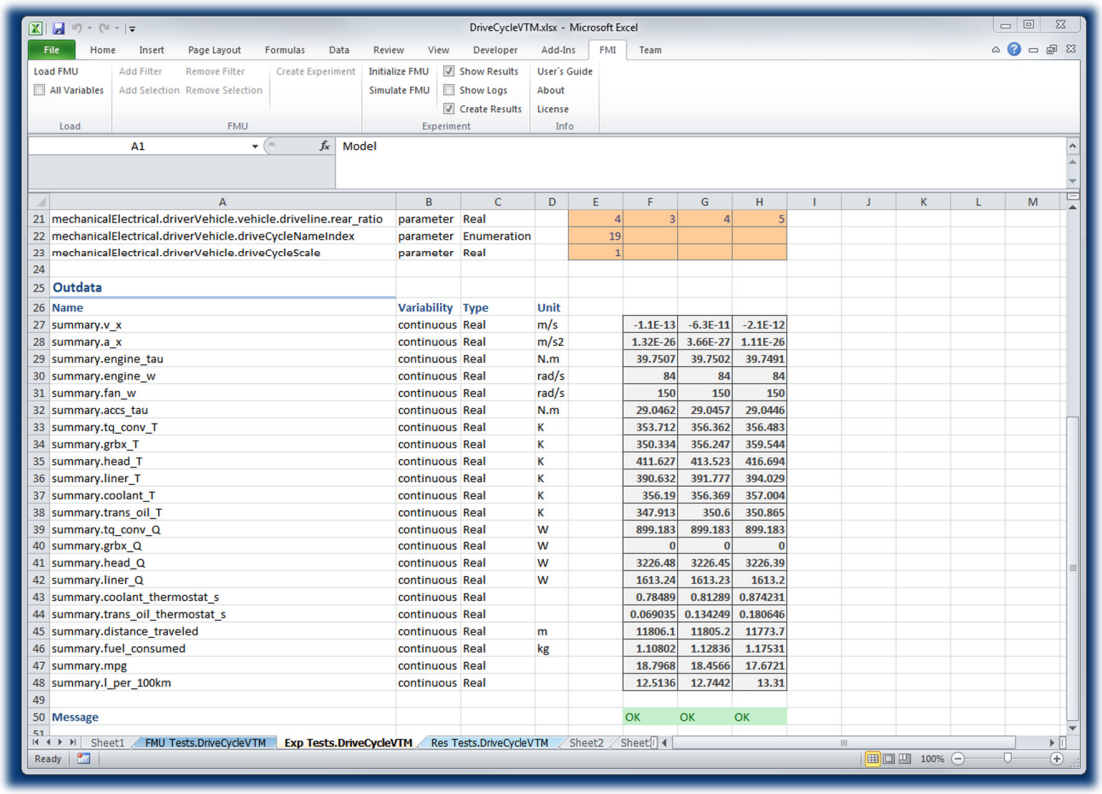


Figure 22 FMIE initialized experiment sheet

8) If the initialization of the FMU was successful, click the **Simulate FMU** button from the FMI ribbon tab. This step will launch each case in parallel on a separate processor, up to the number of processors on the machine and the *Number of processes* specified on the experiment sheet. Note that this simulation is fairly long and dynamic. It can take a few minutes (~10 minutes) to complete.
9) When the simulation completes, the final values of each output variable are returned to the experiment sheet.

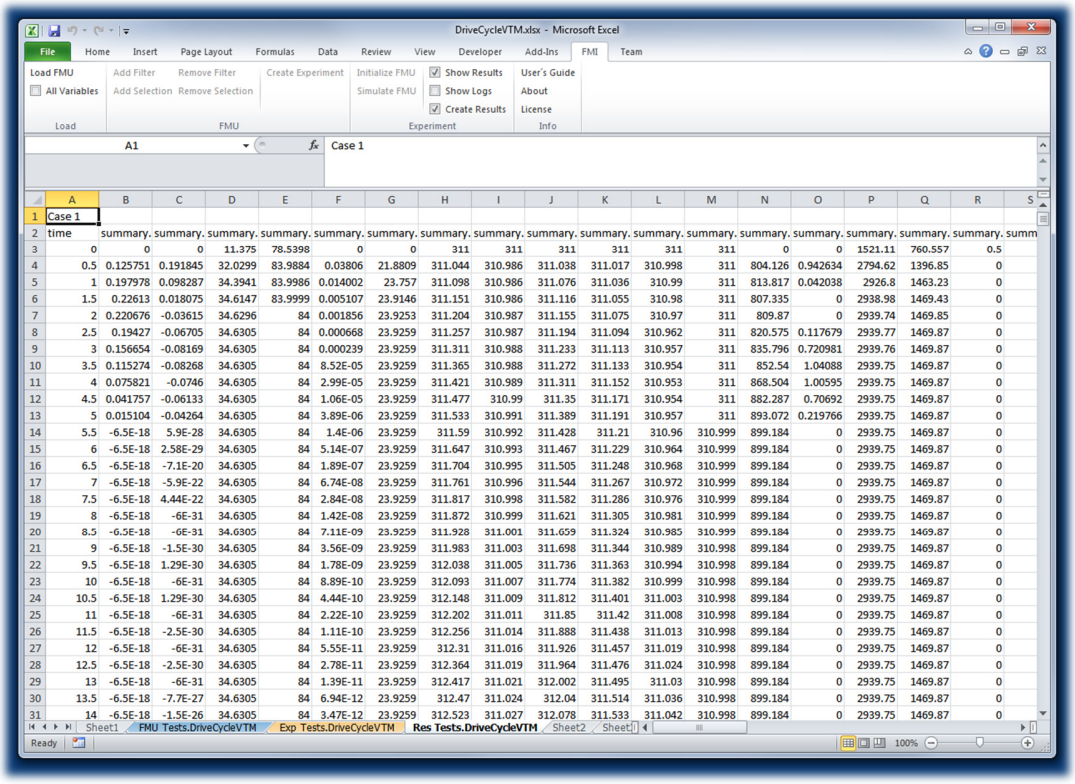


	A	B	C	D	E	F	G	H
21	mechanicalElectrical.driverVehicle.vehicle.driveline.rear_ratio	parameter	Real		4	3	4	5
22	mechanicalElectrical.driverVehicle.driveCycleNameIndex	parameter	Enumeration		19			
23	mechanicalElectrical.driverVehicle.driveCycleScale	parameter	Real		1			
24								
25	**Outdata**							
26	**Name**	**Variability**	**Type**	**Unit**				
27	summary.v_x	continuous	Real	m/s		-1.1E-13	-6.3E-11	-2.1E-12
28	summary.a_x	continuous	Real	m/s2		1.32E-26	3.66E-27	1.11E-26
29	summary.engine_tau	continuous	Real	N.m		39.7507	39.7502	39.7491
30	summary.engine_w	continuous	Real	rad/s		84	84	84
31	summary.fan_w	continuous	Real	rad/s		150	150	150
32	summary.accs_tau	continuous	Real	N.m		29.0462	29.0457	29.0446
33	summary.tq_conv_T	continuous	Real	K		353.712	356.362	356.483
34	summary.grbx_T	continuous	Real	K		350.334	356.247	359.544
35	summary.head_T	continuous	Real	K		411.627	413.523	416.694
36	summary.liner_T	continuous	Real	K		390.632	391.777	394.029
37	summary.coolant_T	continuous	Real	K		356.19	356.369	357.004
38	summary.trans_oil_T	continuous	Real	K		347.913	350.6	350.865
39	summary.tq_conv_Q	continuous	Real	W		899.183	899.183	899.183
40	summary.grbx_Q	continuous	Real	W		0	0	0
41	summary.head_Q	continuous	Real	W		3226.48	3226.45	3226.39
42	summary.liner_Q	continuous	Real	W		1613.24	1613.23	1613.2
43	summary.coolant_thermostat_s	continuous	Real			0.78489	0.81289	0.874231
44	summary.trans_oil_thermostat_s	continuous	Real			0.069035	0.134249	0.180646
45	summary.distance_traveled	continuous	Real	m		11806.1	11805.2	11773.7
46	summary.fuel_consumed	continuous	Real	kg		1.10802	1.12836	1.17531
47	summary.mpg	continuous	Real			18.7968	18.4566	17.6721
48	summary.l_per_100km	continuous	Real			12.5136	12.7442	13.31
49								
50	**Message**					OK	OK	OK

Figure 23 FMIE completed simulation experiment sheet

In this case, the final fuel economy metrics show that the case with the 3:1 final drive ratio was slightly better than the case with the 4:1 ratio.

10) Also, when the simulation completes, a new worksheet will be created with the prefix “Res”. In this case, it is something similar to *Res Tests.DriveCycleVTM*. The structure of this sheet is columns of output data with time in the first column and variable trajectories in subsequent columns for each simulated case. Any cases that were disabled on the experiment sheet or failed simulations will contain empty columns. For this experiment sheet, the result sheet should look similar to the following if the simulations completed successfully:



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
1	Case 1																		
2	time	summary.	summary.	summary.	summary.	summary.	summary.	summary.	summary.	summary.	summary.	summary.	summary.	summary.	summary.	summary.	summary.	summary.	summ
3	0	0	0	11.375	78.5398	0	0	311	311	311	311	311	311	0	0	1521.11	760.557	0.5	
4	0.5	0.125751	0.191845	32.0299	83.9884	0.03806	21.8809	311.044	310.986	311.038	311.017	310.998	311	804.126	0.942634	2794.62	1396.85	0	
5	1	0.197978	0.098287	34.3941	83.9986	0.014002	23.757	311.098	310.986	311.076	311.036	310.99	311	813.817	0.042038	2926.8	1463.23	0	
6	1.5	0.22613	0.018075	34.6147	83.9999	0.005107	23.9146	311.151	310.986	311.116	311.055	310.98	311	807.335	0	2938.98	1469.43	0	
7	2	0.220676	-0.03615	34.6296	84	0.001856	23.9253	311.204	310.987	311.155	311.075	310.97	311	809.87	0	2939.74	1469.85	0	
8	2.5	0.19427	-0.06705	34.6305	84	0.000668	23.9259	311.257	310.987	311.194	311.094	310.962	311	820.575	0.117679	2939.77	1469.87	0	
9	3	0.156654	-0.08169	34.6305	84	0.000239	23.9259	311.311	310.988	311.233	311.113	310.957	311	835.796	0.720981	2939.76	1469.87	0	
10	3.5	0.115274	-0.08268	34.6305	84	8.52E-05	23.9259	311.365	310.988	311.272	311.133	310.954	311	852.54	1.04088	2939.75	1469.87	0	
11	4	0.075821	-0.0746	34.6305	84	2.99E-05	23.9259	311.421	310.989	311.311	311.152	310.953	311	868.504	1.00595	2939.75	1469.87	0	
12	4.5	0.041757	-0.06133	34.6305	84	1.06E-05	23.9259	311.477	310.99	311.35	311.171	310.954	311	882.287	0.70692	2939.75	1469.87	0	
13	5	0.015104	-0.04264	34.6305	84	3.89E-06	23.9259	311.533	310.991	311.389	311.191	310.957	311	893.072	0.219766	2939.75	1469.87	0	
14	5.5	-6.5E-18	5.9E-28	34.6305	84	1.4E-06	23.9259	311.59	310.992	311.428	311.21	310.96	310.999	899.184	0	2939.75	1469.87	0	
15	6	-6.5E-18	2.58E-29	34.6305	84	5.14E-07	23.9259	311.647	310.993	311.467	311.229	310.964	310.999	899.184	0	2939.75	1469.87	0	
16	6.5	-6.5E-18	-7.1E-20	34.6305	84	1.89E-07	23.9259	311.704	310.995	311.505	311.248	310.968	310.999	899.184	0	2939.75	1469.87	0	
17	7	-6.5E-18	-5.9E-22	34.6305	84	6.74E-08	23.9259	311.761	310.996	311.544	311.267	310.972	310.999	899.184	0	2939.75	1469.87	0	
18	7.5	-6.5E-18	4.44E-22	34.6305	84	2.84E-08	23.9259	311.817	310.998	311.582	311.286	310.976	310.999	899.184	0	2939.75	1469.87	0	
19	8	-6.5E-18	-6E-31	34.6305	84	1.42E-08	23.9259	311.872	310.999	311.621	311.305	310.981	310.999	899.184	0	2939.75	1469.87	0	
20	8.5	-6.5E-18	-6E-31	34.6305	84	7.11E-09	23.9259	311.928	311.001	311.659	311.324	310.985	310.999	899.184	0	2939.75	1469.87	0	
21	9	-6.5E-18	-1.5E-30	34.6305	84	3.56E-09	23.9259	311.983	311.003	311.698	311.344	310.989	310.998	899.184	0	2939.75	1469.87	0	
22	9.5	-6.5E-18	1.29E-30	34.6305	84	1.78E-09	23.9259	312.038	311.005	311.736	311.363	310.994	310.998	899.184	0	2939.75	1469.87	0	
23	10	-6.5E-18	-6E-31	34.6305	84	8.89E-10	23.9259	312.093	311.007	311.774	311.382	310.999	310.998	899.184	0	2939.75	1469.87	0	
24	10.5	-6.5E-18	1.29E-30	34.6305	84	4.44E-10	23.9259	312.148	311.009	311.812	311.401	311.003	310.998	899.184	0	2939.75	1469.87	0	
25	11	-6.5E-18	-6E-31	34.6305	84	2.22E-10	23.9259	312.202	311.011	311.85	311.42	311.008	310.998	899.184	0	2939.75	1469.87	0	
26	11.5	-6.5E-18	-2.5E-30	34.6305	84	1.11E-10	23.9259	312.256	311.014	311.888	311.438	311.013	310.998	899.184	0	2939.75	1469.87	0	
27	12	-6.5E-18	-6E-31	34.6305	84	5.55E-11	23.9259	312.31	311.016	311.926	311.457	311.019	310.998	899.184	0	2939.75	1469.87	0	
28	12.5	-6.5E-18	-2.5E-30	34.6305	84	2.78E-11	23.9259	312.364	311.019	311.964	311.476	311.024	310.998	899.184	0	2939.75	1469.87	0	
29	13	-6.5E-18	-6E-31	34.6305	84	1.39E-11	23.9259	312.417	311.021	312.002	311.495	311.03	310.998	899.184	0	2939.75	1469.87	0	
30	13.5	-6.5E-18	-7.7E-27	34.6305	84	6.94E-12	23.9259	312.47	311.024	312.04	311.514	311.036	310.998	899.184	0	2939.75	1469.87	0	
31	14	-6.5E-18	-1.5E-26	34.6305	84	3.47E-12	23.9259	312.523	311.027	312.078	311.533	311.042	310.998	899.184	0	2939.75	1469.87	0	

Figure 24 FMIE result sheet

11) With the variable time trajectories returned to the result sheet, any standard Excel® routine can be used to review the results.

a. Plot the vehicle velocity for each case by selecting the corresponding time and velocity (column *summary.v_x*) for each case and adding to a scatter plot chart.

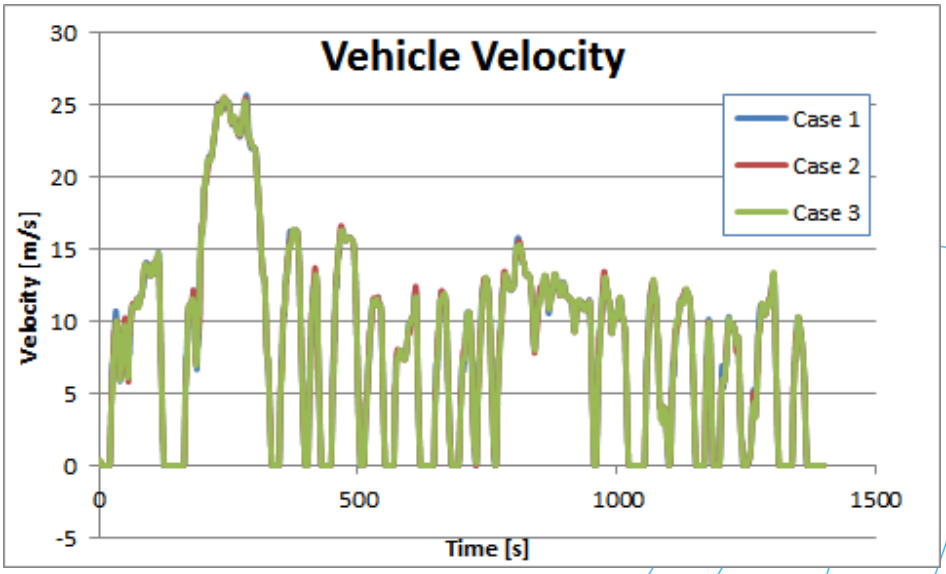


Figure 25 FMIE vehicle velocity comparison figure

b. Next plot the engine speed from the column labeled *summary.mpg*:

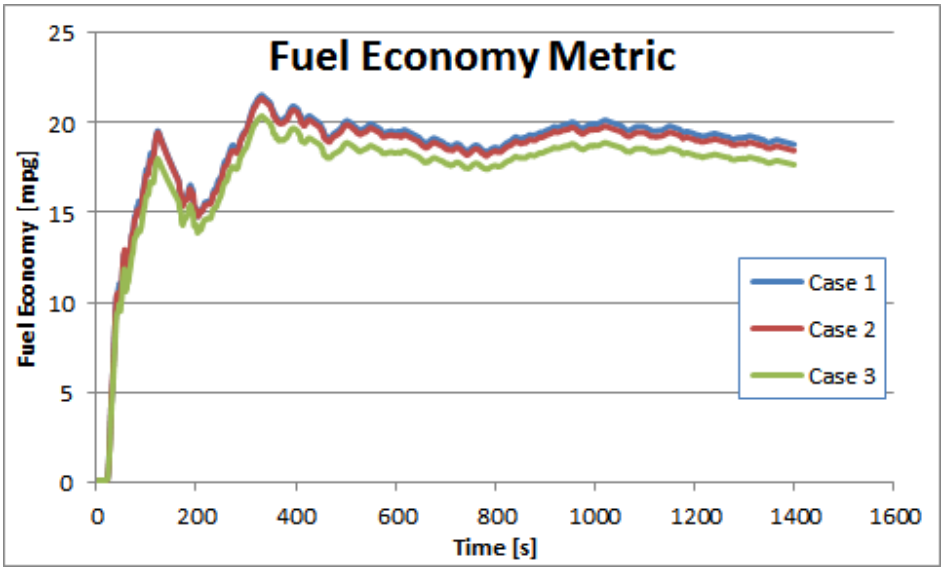


Figure 26 FMIE fuel economy metric comparison figure

c. Other interesting trajectories are the engine speed (*summary.engine_w*), coolant temperature (*summary.coolant_T*) and thermostat positions (*summary.coolant_thermostat_s*) shown below.

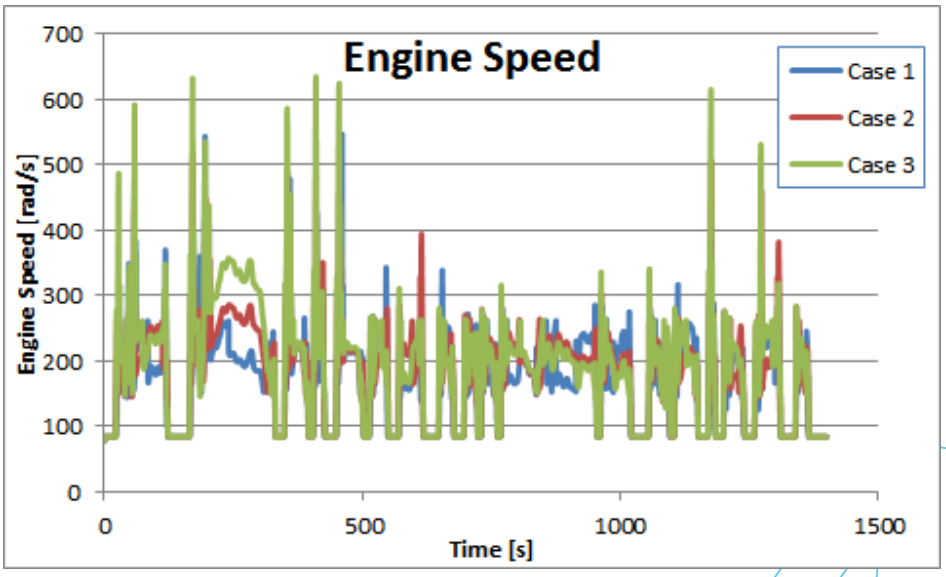


Figure 27 FMIE engine speed comparison figure

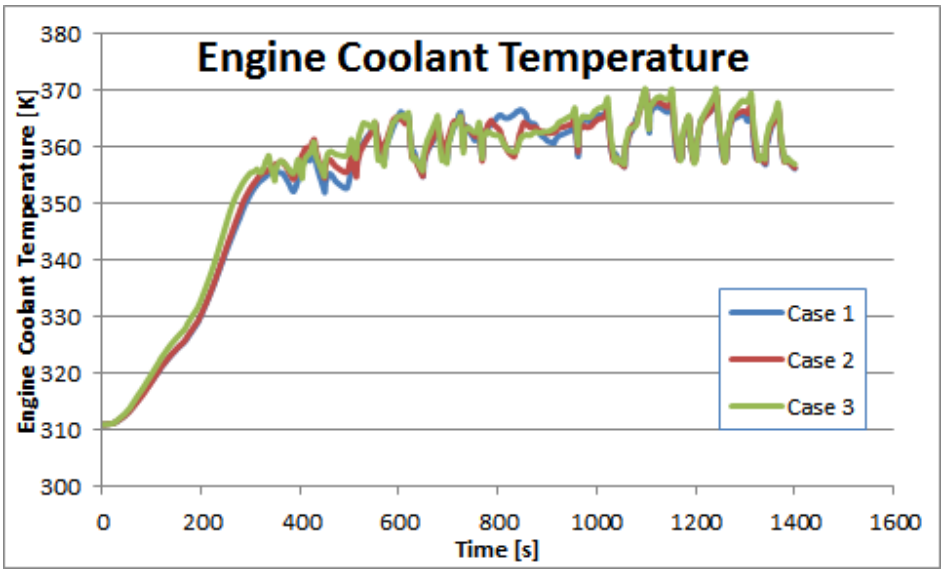


Figure 28 FMIE engine coolant temperature comparison figure

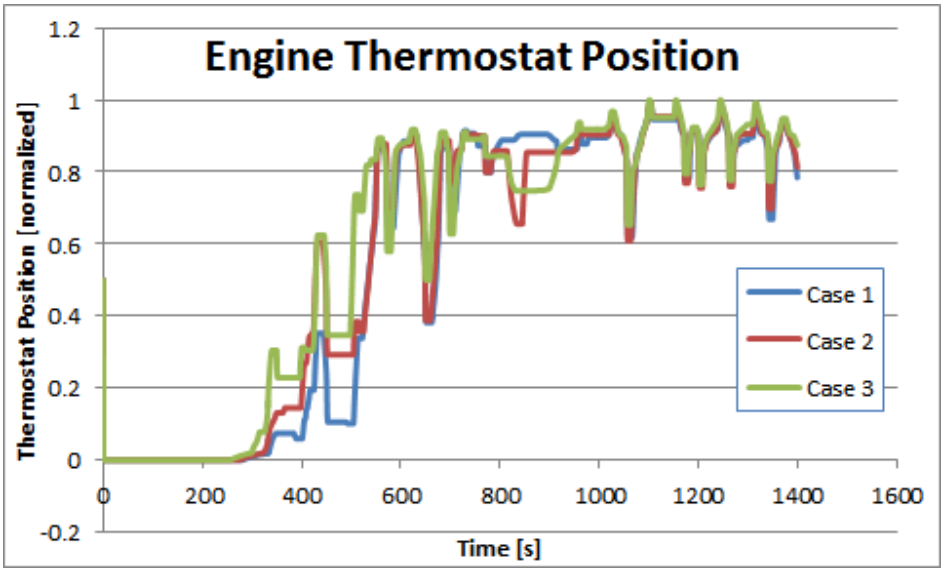


Figure 29 FMIE engine thermostat comparison figure

12) The last example in this section shows how to apply a time varying input to an FMU. To start use the process in step 2 to load the *VTMModels_DriveCycleVTM_0withInputs.fmu* from the FMUs\CS folder. The model used to create this FMU is identical to the model shown in Figure 1except that the throttle and brake signals are no longer generated by the internal driver model but will be set as time trajectory inputs from Excel®.
13) When the FMU is loaded, use the list filter from the Name column to search for and include the desired variables.
    a. From the top level *throttlePosition* input variable. This variable will be named *throttlePosition* without any additional levels of hierarchy. Toggle the Include field to TRUE.

42286	TRUE	throttlePosition	continuous	indata & outdata	0	Real	input

Figure 30 FMIE top level throttle pedal position input

Do the same for the *brakeForce* input variable.

42287	TRUE	brakeForce	continuous	indata & outdata	0	Real	input

Figure 31 FMIE top level brake pedal force input

b. Use the Name list filter to search for "driver.summary". Then choose the *summary_s_acc* and *summary_f_brk* output variables as shown.

22	Variables							
23	Include	Name	Variability	Experiment	Start	Unit	Type	
34966	TRUE	mechanicalElectrical.driverVehicle.driver.summary_s_acc	continuous	outdata		1	Real	
34967	FALSE	mechanicalElectrical.driverVehicle.driver.summary_f_acc	continuous	outdata		N	Real	
34968	FALSE	mechanicalElectrical.driverVehicle.driver.summary_s_brk	continuous	outdata		1	Real	
34969	TRUE	mechanicalElectrical.driverVehicle.driver.summary_f_brk	continuous	outdata		N	Real	
34970	FALSE	mechanicalElectrical.driverVehicle.driver.summary_s_clt	continuous	outdata		1	Real	
34971	FALSE	mechanicalElectrical.driverVehicle.driver.summary_f_clt	continuous	outdata		N	Real	
42360								

Figure 32 FMIE throttle pedal position and brake force output variables

c. Next use the steps from section 4)c above to select the summary record output variables.

The final list of variables should appear similar to the following:

21								
22	Variables							
23	Include	Name	Variability	Experiment	Start	Unit	Type	
34966	TRUE	mechanicalElectrical.driverVehicle.driver.summary_s_acc	continuous	outdata		1	Real	
34969	TRUE	mechanicalElectrical.driverVehicle.driver.summary_f_brk	continuous	outdata		N	Real	
42221	TRUE	summary.v_x	continuous	outdata		m/s	Real	
42222	TRUE	summary.a_x	continuous	outdata		m/s2	Real	
42223	TRUE	summary.engine_tau	continuous	outdata		N.m	Real	
42224	TRUE	summary.engine_w	continuous	outdata		rad/s	Real	
42225	TRUE	summary.fan_w	continuous	outdata		rad/s	Real	
42226	TRUE	summary.accs_tau	continuous	outdata		N.m	Real	
42227	TRUE	summary.tq_conv_T	continuous	outdata		K	Real	
42228	TRUE	summary.grbx_T	continuous	outdata		K	Real	
42229	TRUE	summary.head_T	continuous	outdata		K	Real	
42230	TRUE	summary.liner_T	continuous	outdata		K	Real	
42231	TRUE	summary.coolant_T	continuous	outdata		K	Real	
42232	TRUE	summary.trans_oil_T	continuous	outdata		K	Real	
42233	TRUE	summary.tq_conv_Q	continuous	outdata		W	Real	
42234	TRUE	summary.grbx_Q	continuous	outdata		W	Real	
42235	TRUE	summary.head_Q	continuous	outdata		W	Real	
42236	TRUE	summary.liner_Q	continuous	outdata		W	Real	
42237	TRUE	summary.coolant_thermostat_s	continuous	outdata			Real	
42238	TRUE	summary.trans_oil_thermostat_s	continuous	outdata			Real	
42239	TRUE	summary.distance_traveled	continuous	outdata		m	Real	
42240	TRUE	summary.fuel_consumed	continuous	outdata		kg	Real	
42241	TRUE	summary.mpg	continuous	outdata			Real	
42242	TRUE	summary.l_per_100km	continuous	outdata			Real	
42286	TRUE	throttlePosition	continuous	indata & outdat	0		Real	
42287	TRUE	brakeForce	continuous	indata & outdat	0		Real	
42360								

Figure 33 FMIE included variable list

14) Next click the **Create Experiment** button to create the experiment sheet. A new worksheet will be created with a name similar to *Exp DriveCycleVTM_withInputs*.
15) In this test, cases 2 and 3 will be disabled by toggling the *Enable* flag to FALSE as shown.

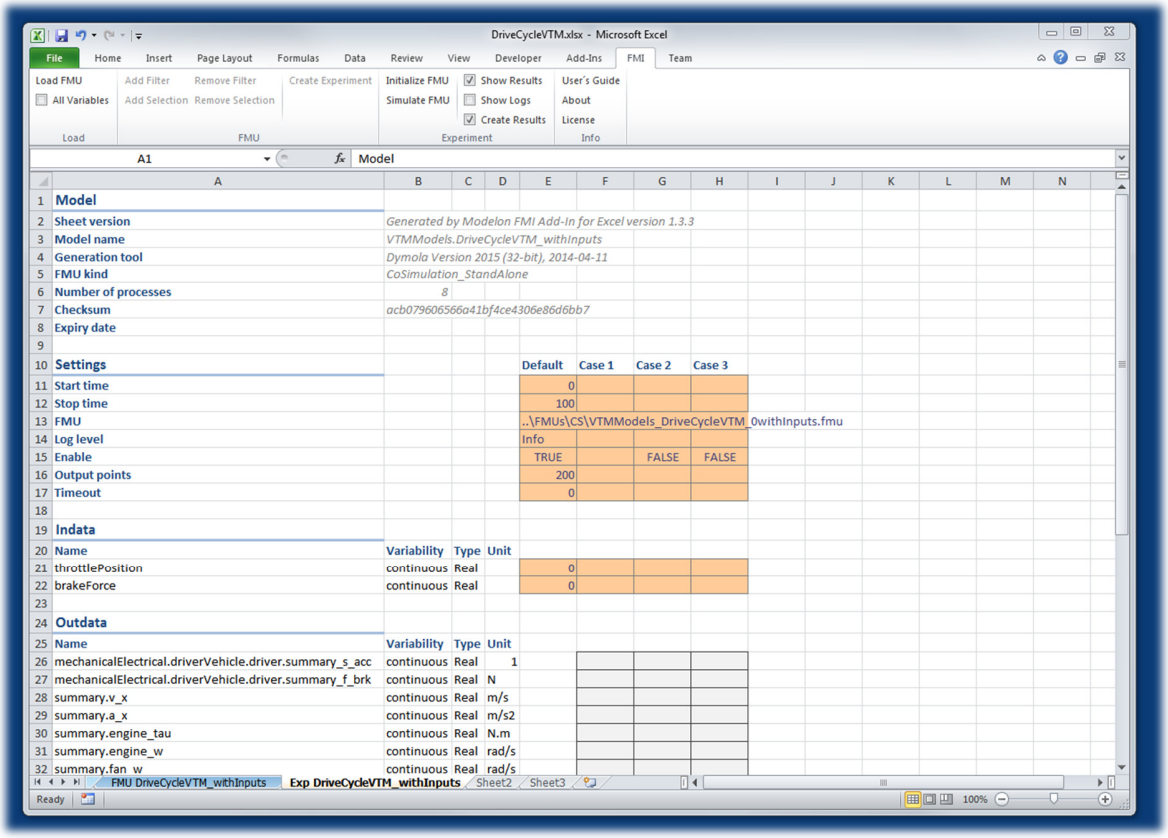


Figure 34 Experiment sheet for the input trajectories

Also set the stop time and number of output points as desired. In this example, simulating for 100 seconds with 200 output points is reasonable.

16) Now the input throttle and brake trajectories can be defined.
    a. On a separate worksheet, define the throttle position and brake force time trajectory arrays with time in the first row (or column) and the data in the second row (or column). The input trajectory range must have dimensions either 2 x M or M x 2, where M is the number of data points. The first row or column must contain the time points and the second row or column must contain the data values. For 2 x 2 sets, the time points are in the first row. In this example the following two trajectories were defined as shown in the following figure.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	time	0	5	10	15	20	25	30	35	40	75	80	100	
2	throttle position	0	0	0.3	0.3	0.5	0.5	1	1	0.4	0.4	0	0	
3														
4	time	0	5	5.1	80	85	100							
5	brake force	100	100	0	0	200	200							

Figure 35 Throttle pedal position and brake pedal force array specification

b. Use the mouse to select the cell for the *throttlePosition* input variable for Case 1 on the experiment sheet. In Figure 34 this cell is F21.

c. Start typing an equal sign (=) in the selected cell as shown.

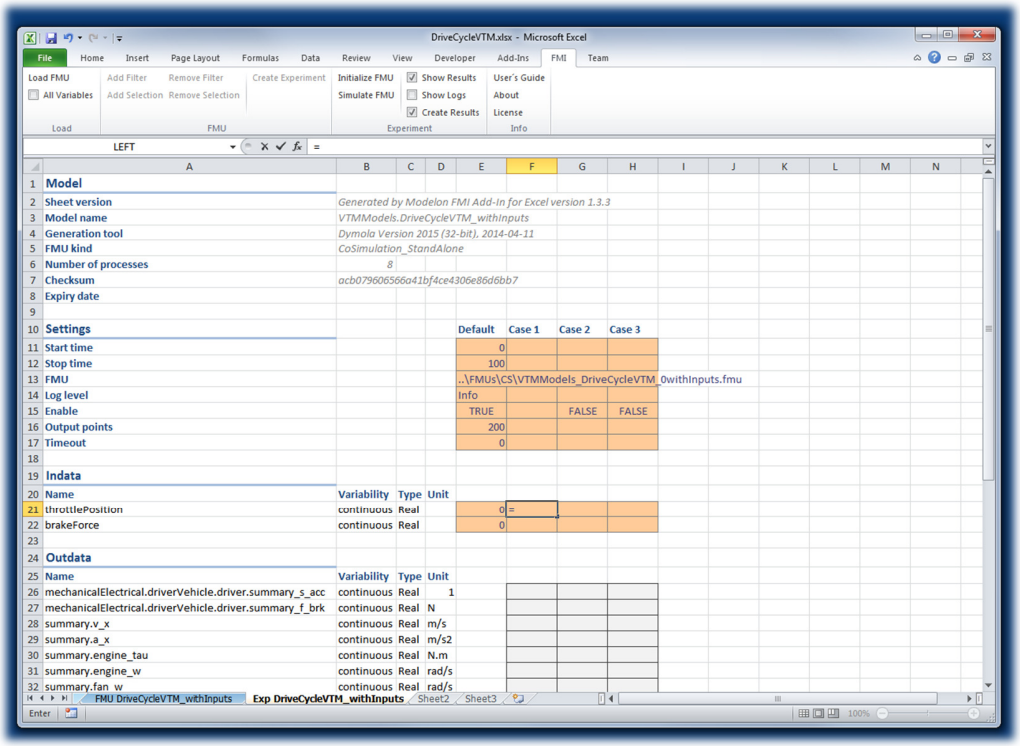


Figure 36 FMIE array formula entry

d. Next select the throttle position trajectory array as shown.

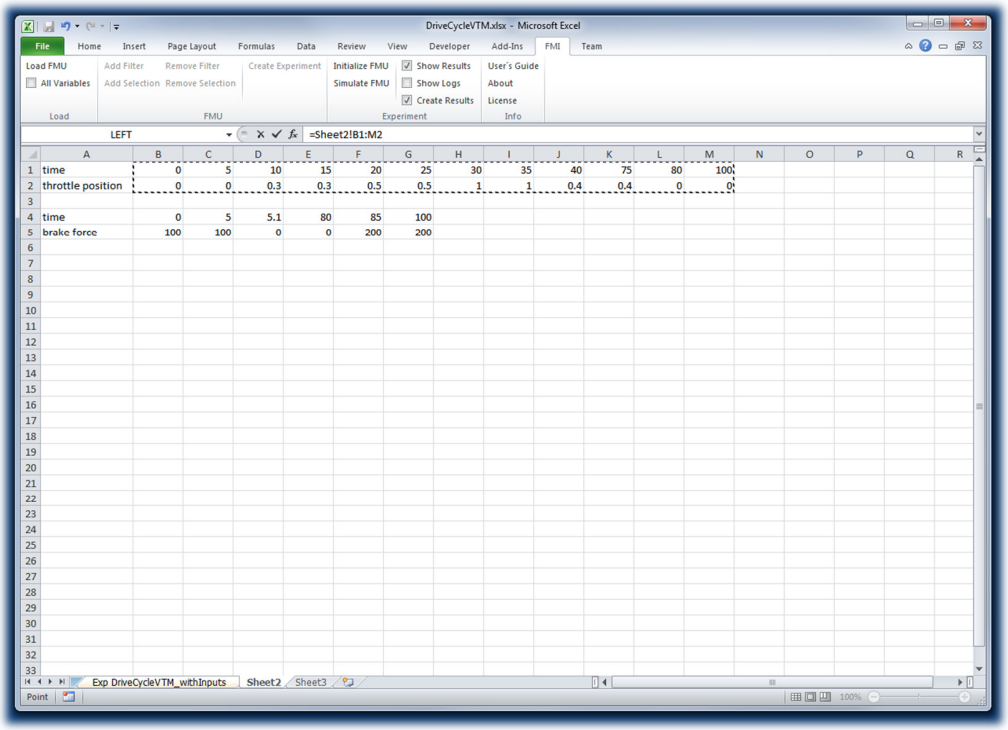


Figure 37 FMIE array formula selection

e. Finally click `Ctrl` + `Shift` + `Enter` to create an array formula. If successful, the first value in the input trajectory range will be visible in the cell, in this case 0. Otherwise "`#VALUE!`" will be displayed in the cell. The formula bar should show an array formula similar to *`{=Sheet2!B1:M2}`*.

f. Repeat steps b – e above for the *brakeForce* input variable.

17) Finally click Simulate FMU from the FMI ribbon tab to run the simulation. This action will run the single test Case 1. The final values of the output variables are returned to the experiment sheet and also create the new result sheet with a name similar to *Res DriveCycleVTM_witInputs*.

Some trajectories from the result sheet to observe are the throttle (*mechanicalElectrical.driverVehicle.driver.summary_s_acc*) and brake (*mechanicalElectrical.driverVehicle.driver.summary_f_brk*) commands, the vehicle speed (*summary.v_x*) and the engine speed (*summary.engine_w*).

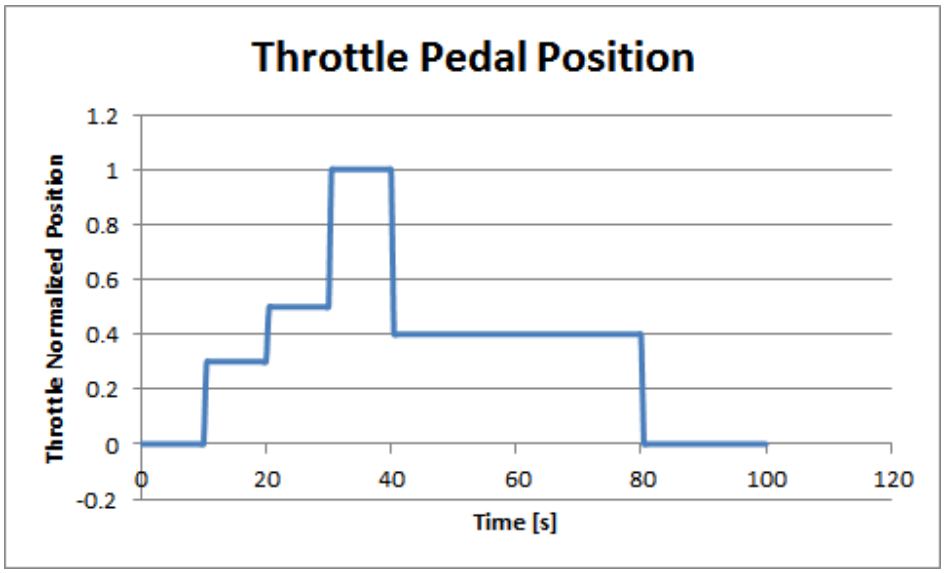


Figure 38 FMIE throttle pedal position figure

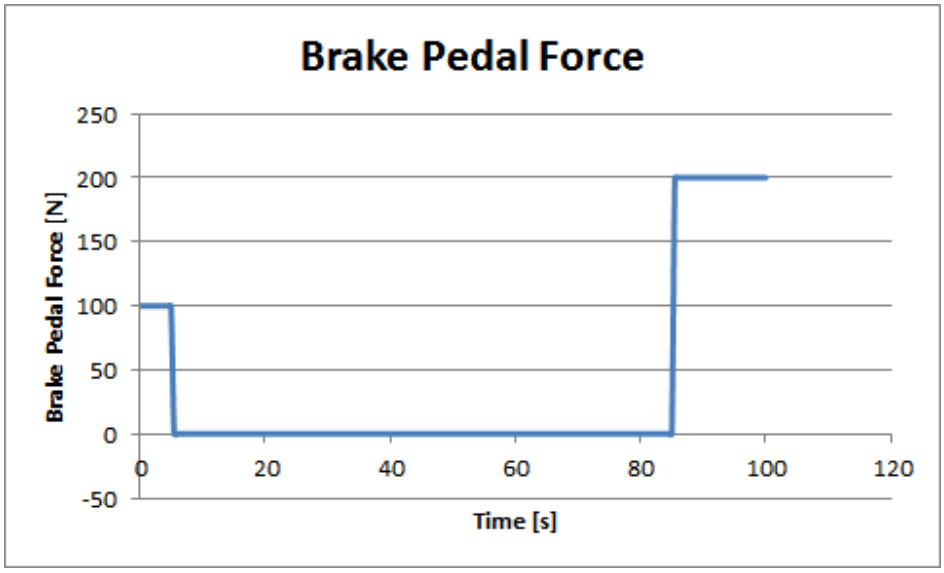


Figure 39 FMIE brake pedal force figure

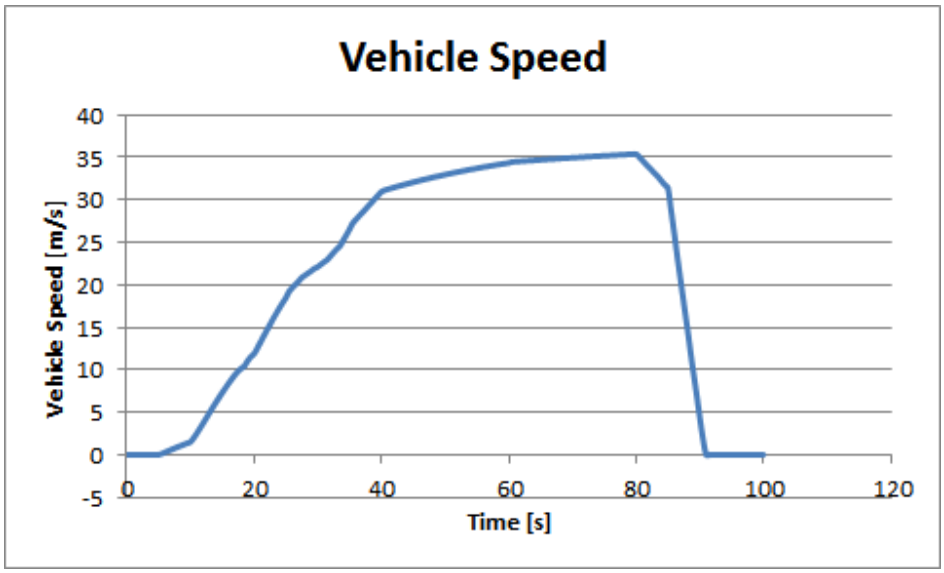


Figure 40 FMIE vehicle speed figure

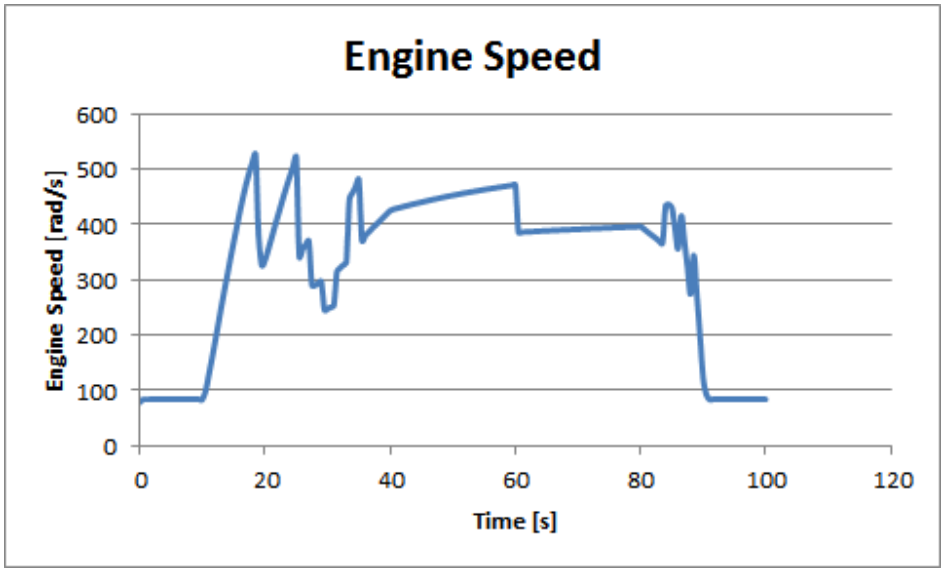


Figure 41 FMIE engine speed figure

The FMI Add-in User's Guide contains a number of different examples of FMI-based workflows with Excel, including a scripting API for automated simulation and analysis. All FMIE demos can be run even after the evaluation license has expired.

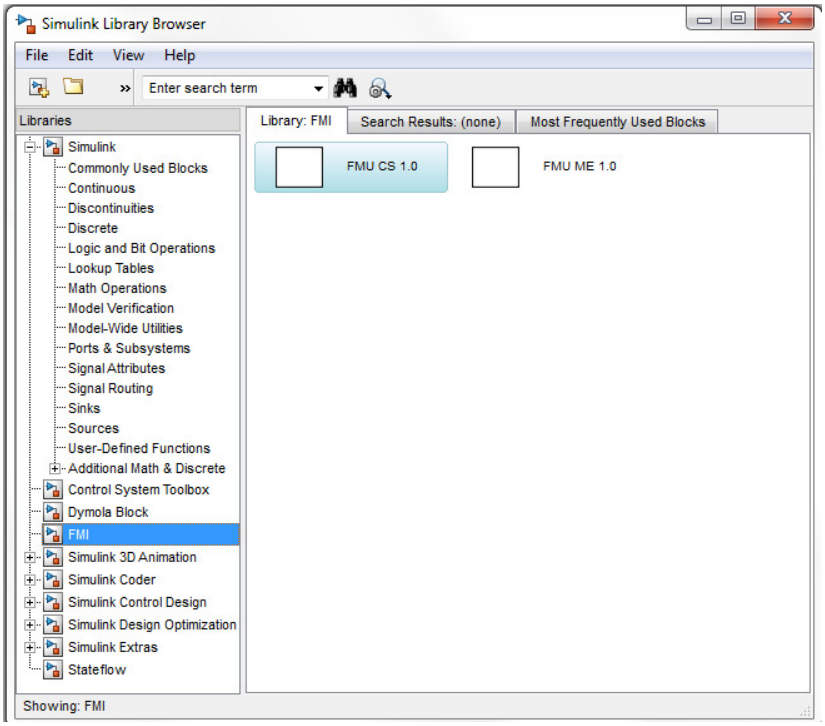
FMU workflows in MATLAB/Simulink using FMI Toolbox

This tutorial illustrates sample FMI-based workflows in MATLAB/Simulink with FMI Toolbox for MATLAB from Modelon.

Demonstration: FMU import in Simulink (optional hands-on)

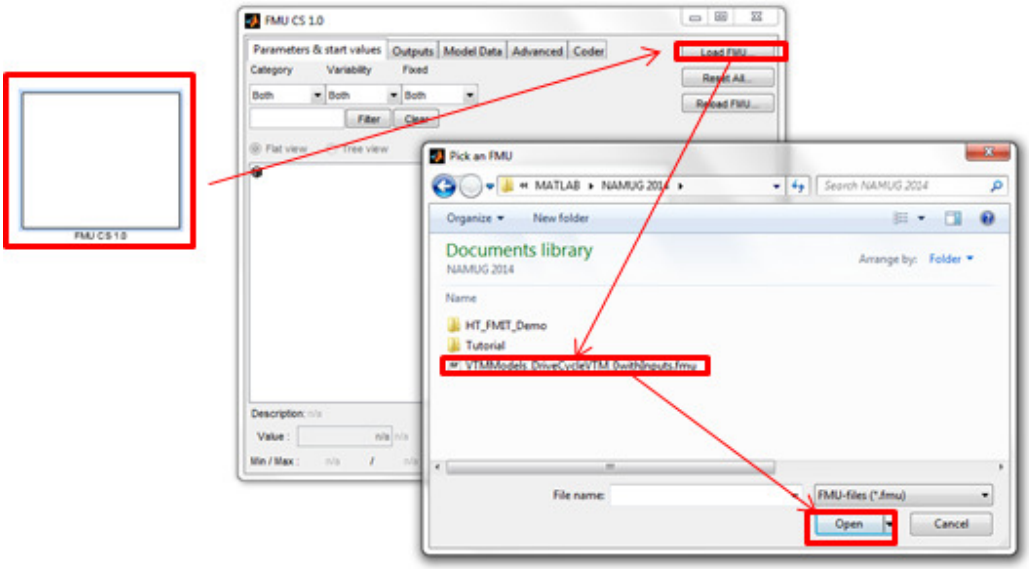
This tutorial shows the import of FMUs into MATLAB/Simulink using FMI Toolbox for MATLAB/Simulink (FMIT). Integration of FMUs with Simulink models supports FMI-based workflows for controls development or when Simulink is used as an integration platform. FMI Toolbox can optionally also export FMUs from Simulink models for integration into other FMI-compliant tools. Bidirectional integration with Simulink will be illustrated in this tutorial.

1) *Preparation:* Install FMI Toolbox via installer *FMI Toolbox-1.8.5-win.exe* accepting all defaults. The demo license for FMIT is in the Licenses folder and should be placed in the AppData folder as described in the FMIT User's Guide (i.e. C:\Users\username\AppData\Roaming\Modelon\Licenses\Nodelocked). In MATLAB, change current directory to the FMI Toolbox installation directory (default: C:\Program Files (x86)\Modelon\FMI Toolbox 1.8.5) and run the "setup" function. See Section 3.3 of the FMIT User's Guide for more details.
2) Launch MATLAB.
3) Open the Simulink Library Browser and look for the FMI library (see figure below). The FMI library contains blocks for importing both co-simulation & model exchange FMUs.



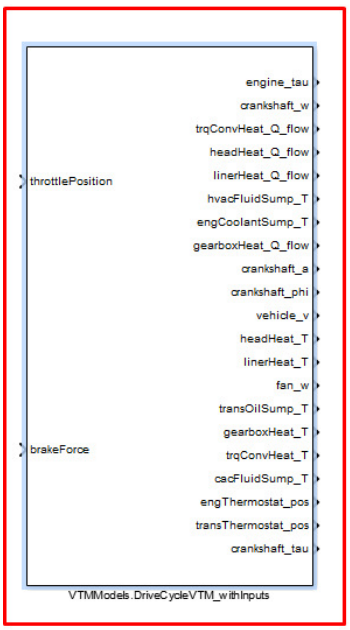
4) Drag the co-simulation (CS) block into a new Simulink model.

5) Double-click on the CS block to load the *"VTMModels_DriveCycleVTM_0withInputs.fmu"* as shown below.

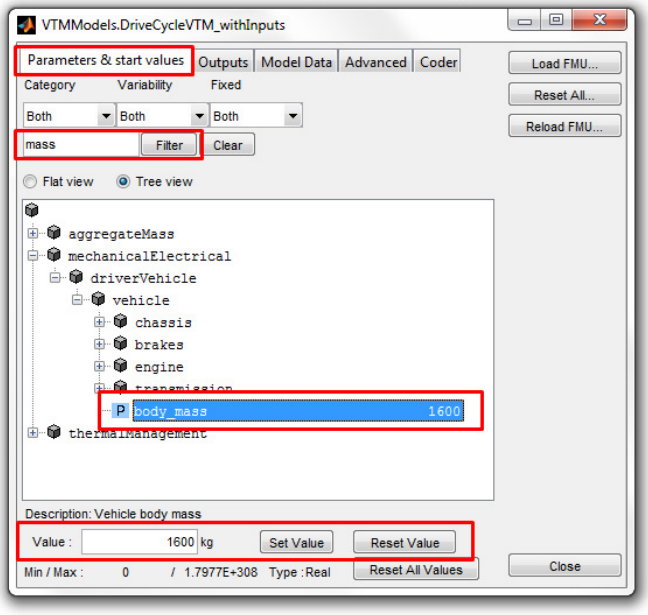


6) Once loaded, the toolbox populates the parameters of the model and the corresponding start values. The other tabs provide additional information/options about the outputs, model data etc.
7) Click on *Close* in the FMI dialog to load the model. This step loads the FMU in Simulink with the appropriate inputs and the (default) outputs set in the original tool that exported the FMU.

FMU loaded in Simulink

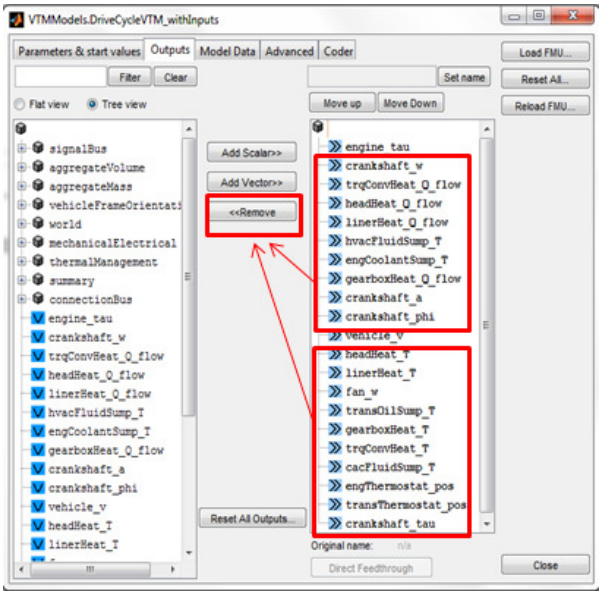


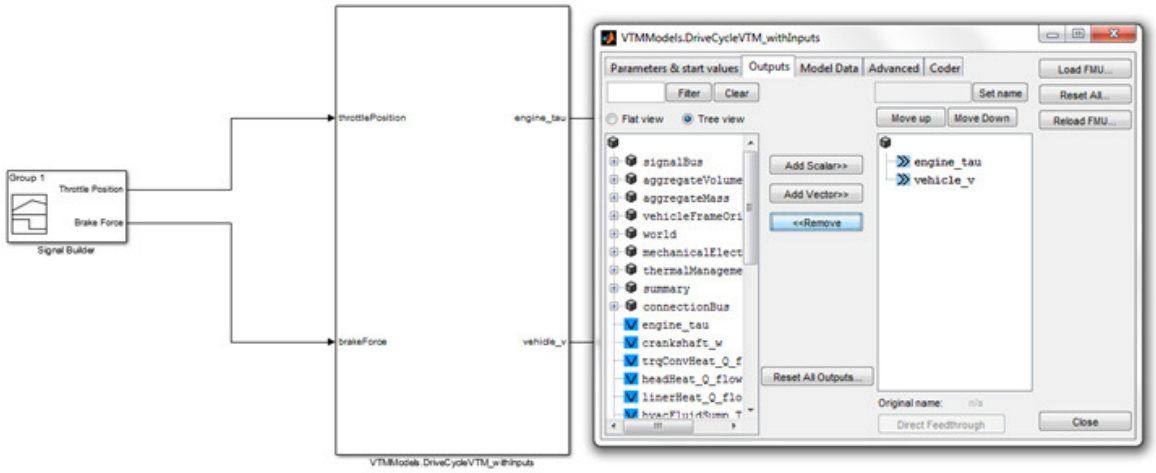
Parameter set/reset dialog



8) In this experiment, we will vary the throttle position and brake force and observe the engine torque (*engine_tau*) and vehicle velocity (*vehicle_v).*
9) In the FMU dialog, remove all other outputs except for *engine_tau* and *vehicle_v*.

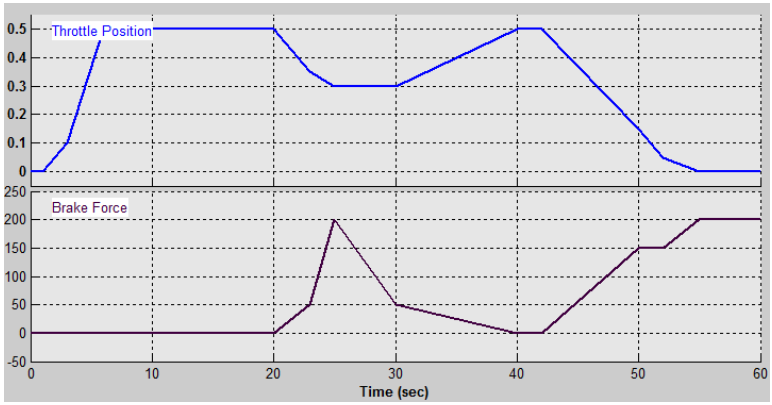
Note: The user is also allowed to add additional output signals to the FMU through the *Add Scalar* and *Add Vector* buttons. Adding outputs is done by navigating the variable tree on the left, choosing the variable to be added, and then clicking on the appropriate *Add* button.



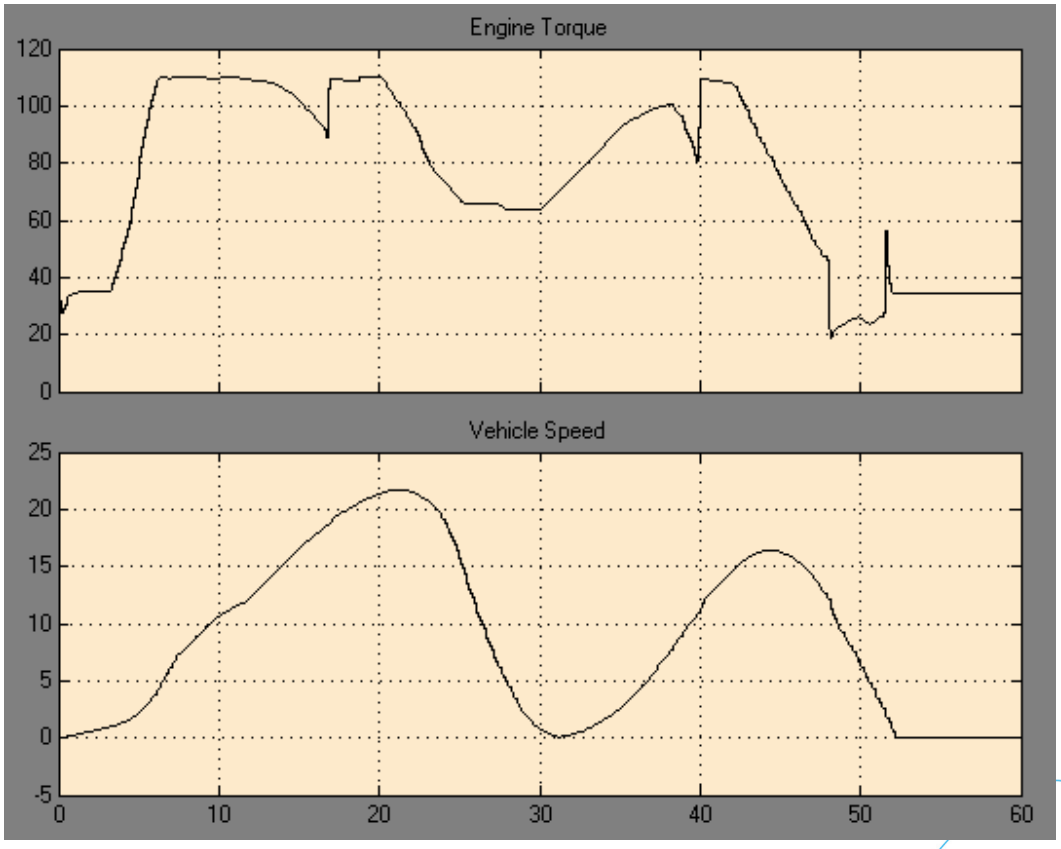


10) Add a signal builder from the Simulink sources library. Create 2 custom signals with the following time traces and name these as *Throttle Positon* and *Brake Force* respectively.

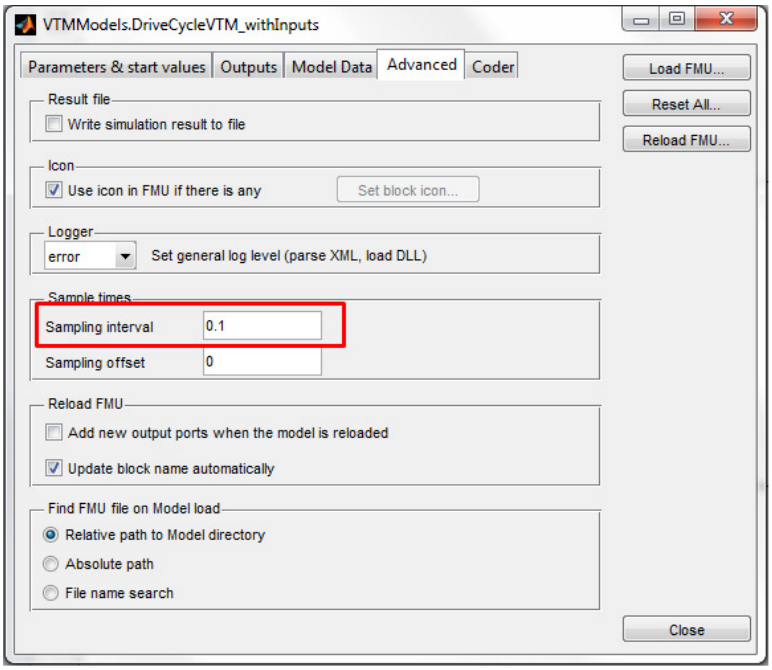
Time	0	1	3	6	20	23	25	30	40	42	50	52	55	60
Throttle Position	0	0	0.1	0.5	0.5	0.35	0.3	0.3	0.5	0.5	0.15	0.05	0	0
Brake Force	0	0	0	0	0	50	200	50	0	0	150	150	200	200



11) Connect these signals to the corresponding inputs to the FMU. Add a scope and connect the *engine_tau* and *vehicle_v* outputs to the scope as shown.
12) Simulate the model for 60 seconds and observe the scope.



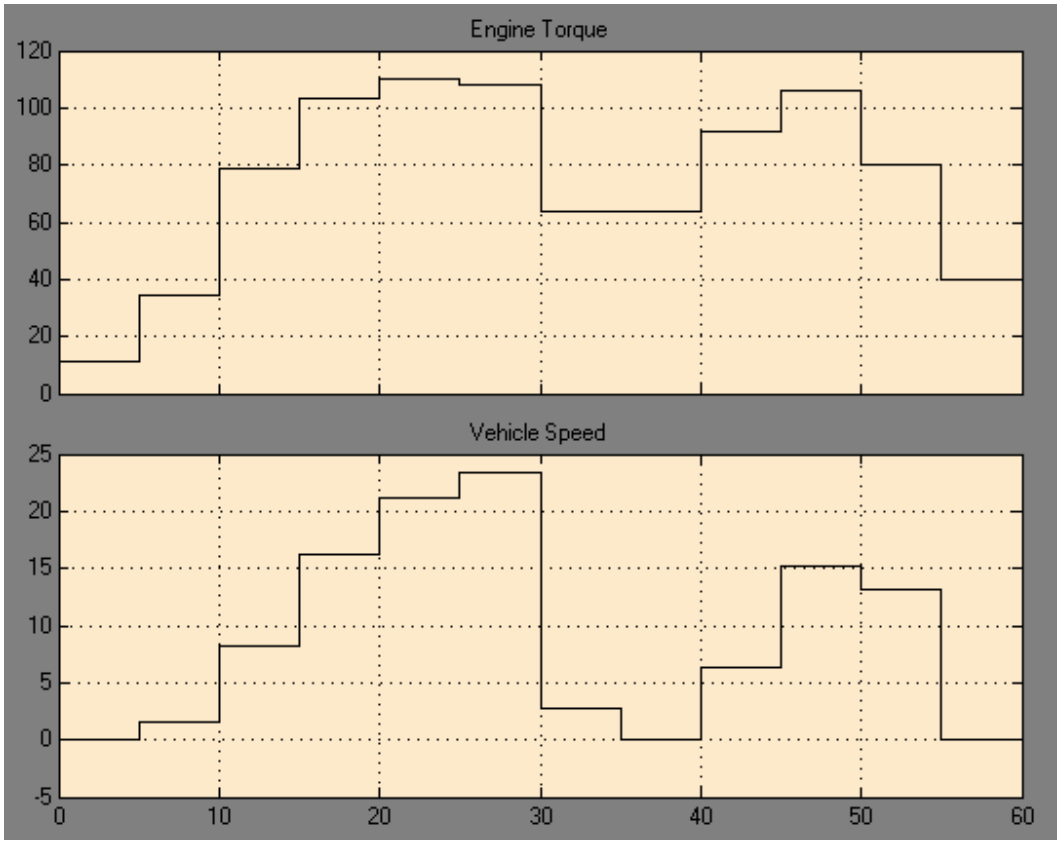
13) Now, the Simulink solver (*Master*) and the solver embedded within the FMU (*slave*) communicate at the default 0.1 sec interval. This setting can be found in the *Advanced* tab in the FMU dialog.



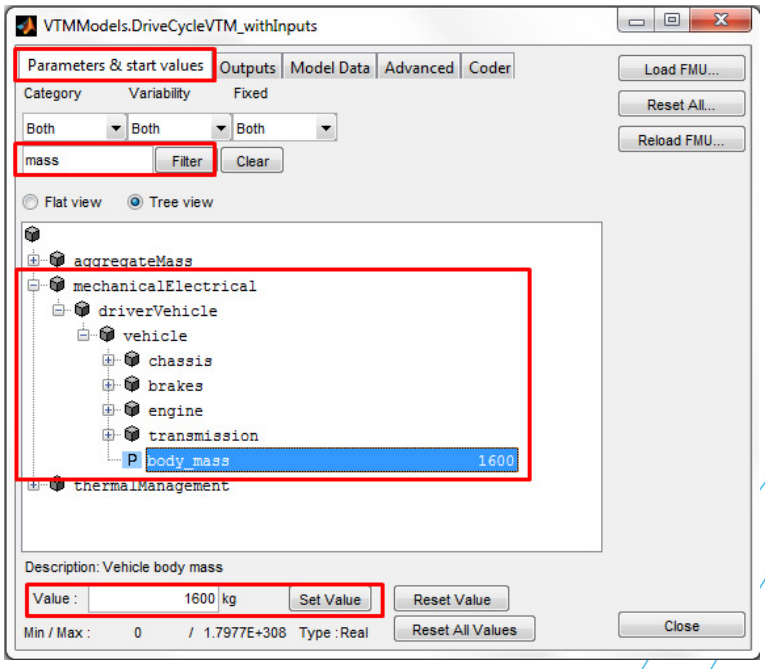
14) Change this sampling interval to 5 sec and simulate the model to 60 sec and observe the difference in the trajectory of the outputs compared to the previous experiment.

15) Do you notice a difference in the trajectories of the outputs between the two experiments? Why?

The communication between the master and slave algorithm introduces a zero-order hold of the inputs sampled at the communication time interval. The sampling is noticeable in the output behavior in Step 13.



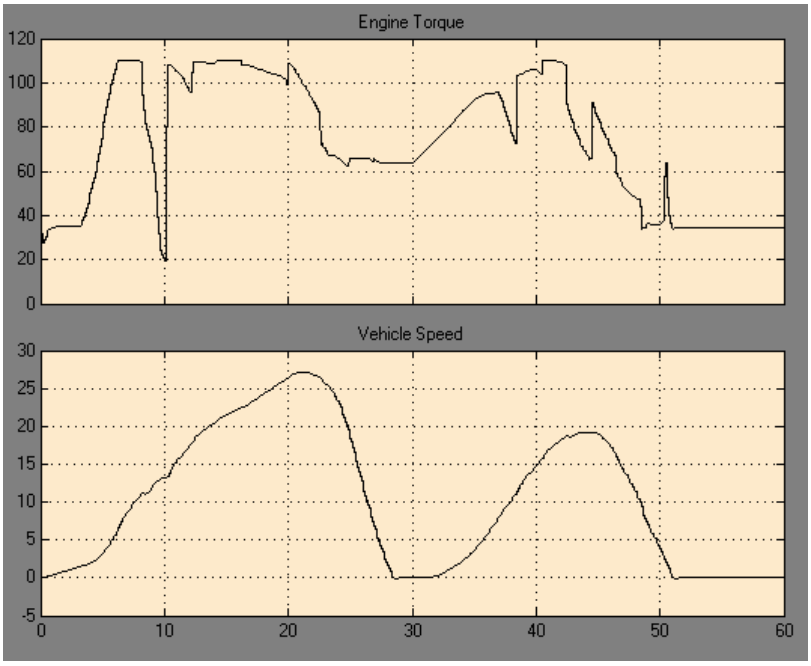
16) In the next experiment, let us change the mass of the vehicle. Double click on the fmu and filter out all variables with the string *mass*.



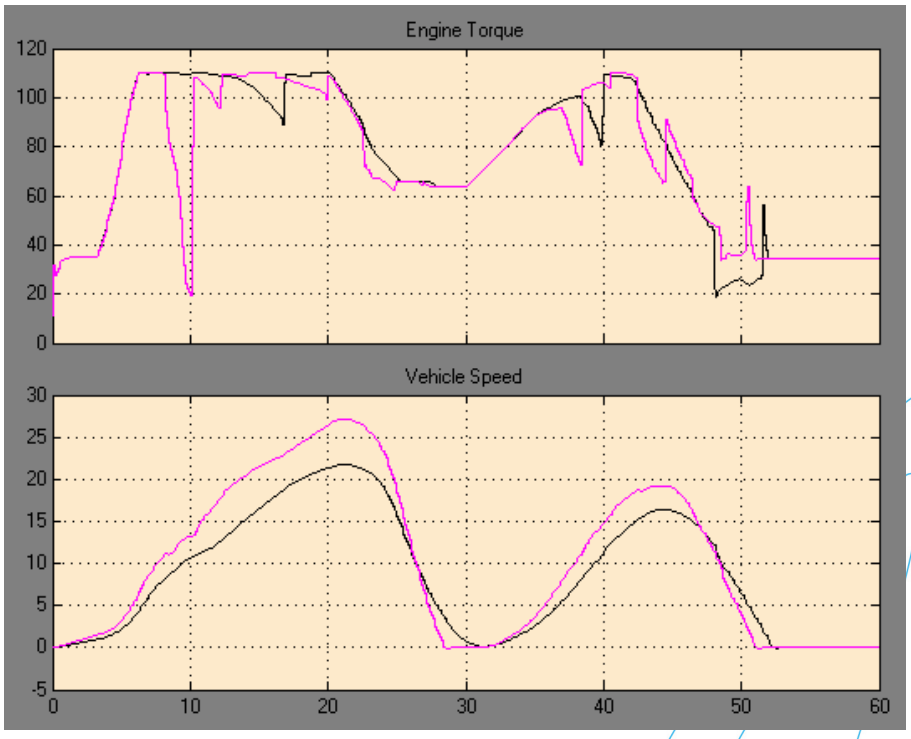
17) Set the mass to 1000 kg. Once the parameter is changed from the default, notice a small asterisk appearing next to the parameter name in the dialog.

Note: The parameter values can be restored to the default values individually or *en masse* with the *Reset Value* and *Reset All Values* buttons.

18) Reset the communication interval to 0.1 sec in the *Advanced* tab and simulate the model. Observe the trajectories of the outputs and compare these with the previous experiment.



The following figure plots the engine torque & vehicle speed for the simulation cases with vehicle mass 1600 kg (black) and 1000 kg (magenta)



Demonstration: FMU export from Simulink using the FMI Toolbox

FMI Toolbox includes an optional Coder feature to export Simulink models as FMUs for import into other FMI-compliant tools. This feature requires MATLAB/Simulink and Simulink Coder. Attendees with Simulink Coder can follow the instructions in Chapter 5 of the FMIT User's Guide to create a simple model and export it as an FMU. The FMU can then be imported into other FMI compliant tools as documented in this tutorial.

The FMI Toolbox User's Guide contains a number of different examples of FMI-based workflows including Design of Experiments, execution of FMUs in MATLAB scripts, and integration of FMUs with other MATLAB toolboxes. All FMIT demos can be run even after the evaluation license has expired.