Tutorial 5 — FMU Export (20-sim)

Overview

This INTO-CPS tutorial will show you how to:

1. Generate a new physical model simulator FMU in 20-sim
   (a) Import a model description into 20-sim
   (b) Complete the skeleton model to produce a working simulator
   (c) Export the simulator FMU
2. Associate the new simulator FMU with a multi-model configuration
3. Execute a co-simulation using the new simulator

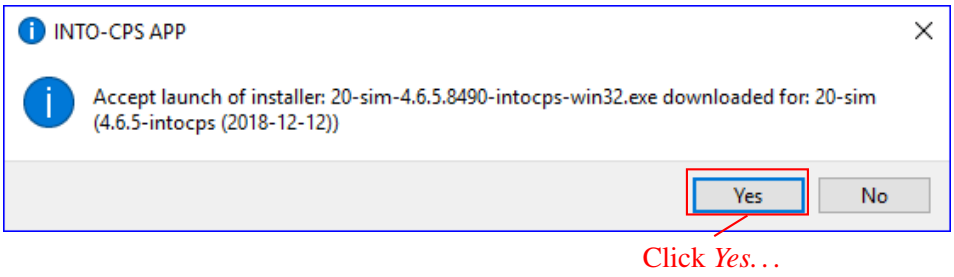
Requirements

This tutorial requires the following tools from the INTO-CPS tool chain to be installed:

- INTO-CPS Application
- COE (Co-simulation Orchestration Engine) accessible to the Application
- 20-sim - Modeling and simulation tool

1 Download and install the 20-sim tool (Windows only)

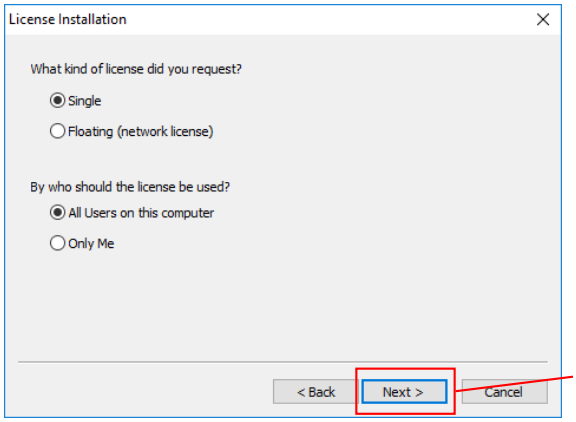
Step 1. Launch the *INTO-CPS Application*. Download the 20-sim modeling tool from the *Window* > *Show Download Manager*. If after the download click *Yes*. If the window does not appear please find the installer inside the *into-cps-projects/install_downloads* folder.



Step 2. Follow the installation wizard until the following license activation screen appears and click *Activation*.

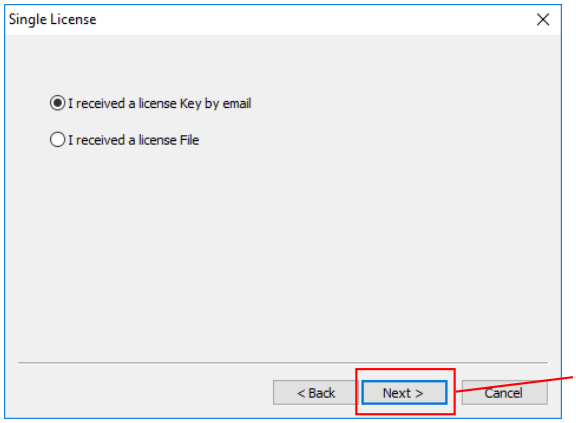
Click *Activation*...

Step 3. The following options will appear, click *Next* to continue.



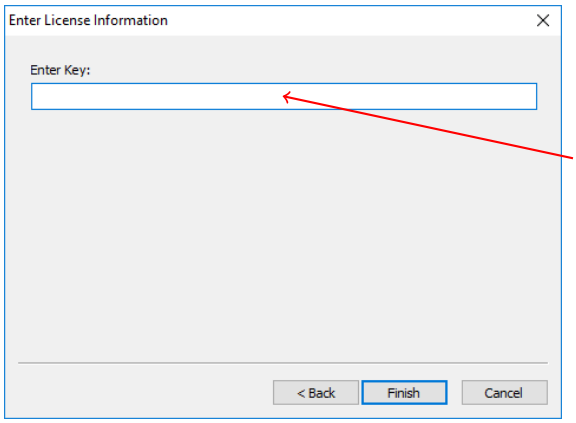
Click *Next*...

Step 4. Select *I received a license Key by email* and click *Next* to continue.

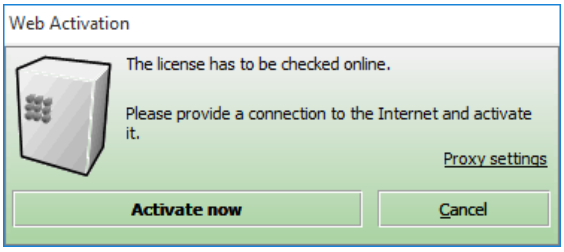


Click *Next*...

Step 5. Please fill in the license key you were given (Available in the blackboard system) and click *Finish*.



Step 6. You will be asked go online. Click on *Active Now* to finish activation.



2 Download and install the 20-sim tool (Linux and macOS only)

It is possible to run 20-sim on Linux and macOS systems after installing Wine.

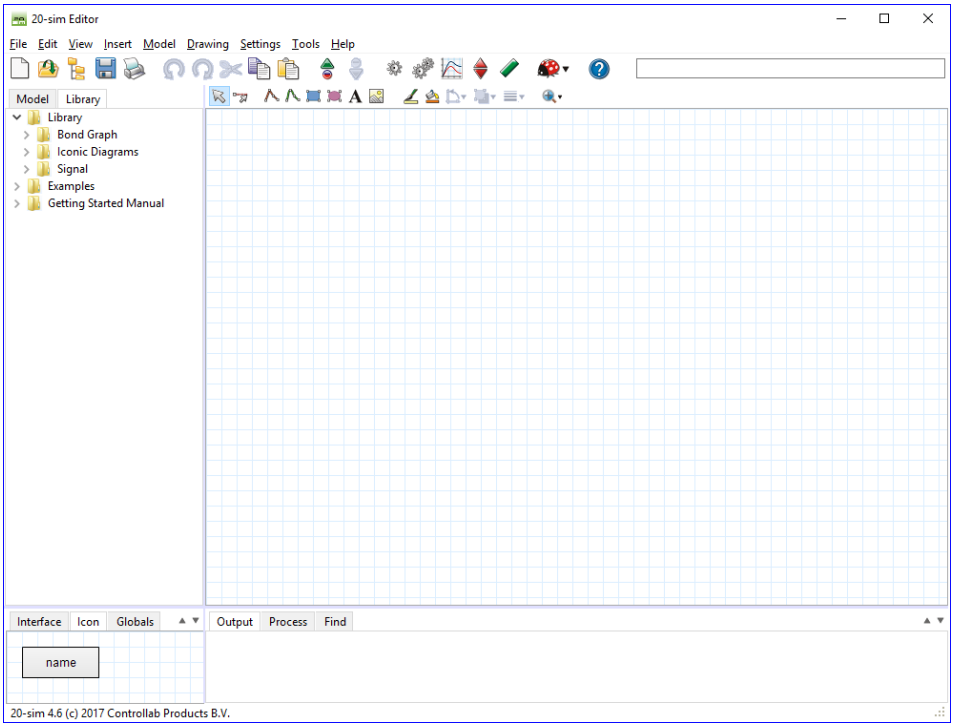
- For the macOS there is a disk image file (.dmg) ready to install. Please ask us.
- For Linux

  1. Install the Wine package for your Linux distribution or follow the instructions for your Linux distribution from `https://wiki.winehq.org/Download`
  2. Use the winetricks tool to install a few Microsoft DLLs to fix the license activation and some drawing issues:
     - winetricks msxml4
     - winetricks wininet
     - winetricks gdiplus
  3. Now download 20-sim from `http://www.20sim.com/downloads/files/20-sim-4.6.5.8490-intocps-win32.exe` and install it using: `wine 20sim.exe`

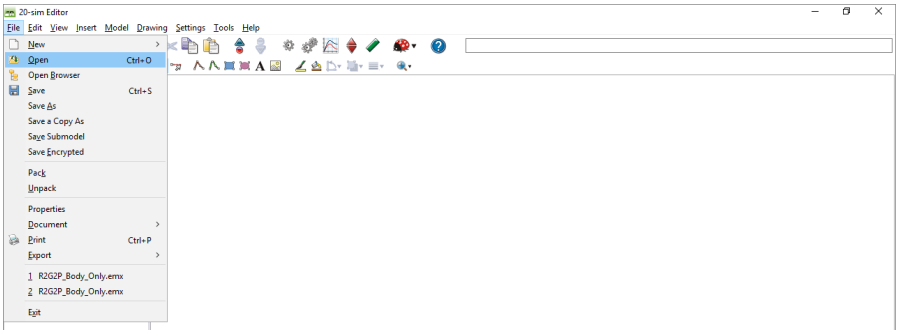
  This should result in a working 20-sim version on Linux. Tested here with various Debian and Ubuntu versions and Wine version 1.5 up to Wine version 3.0.

3 Loading a Model in 20-sim

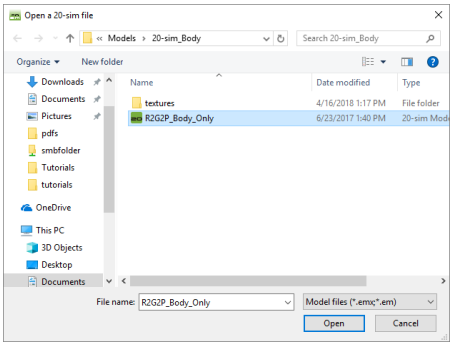
Step 7. Launch the *20-sim* tool. You should see the following 20-sim window.



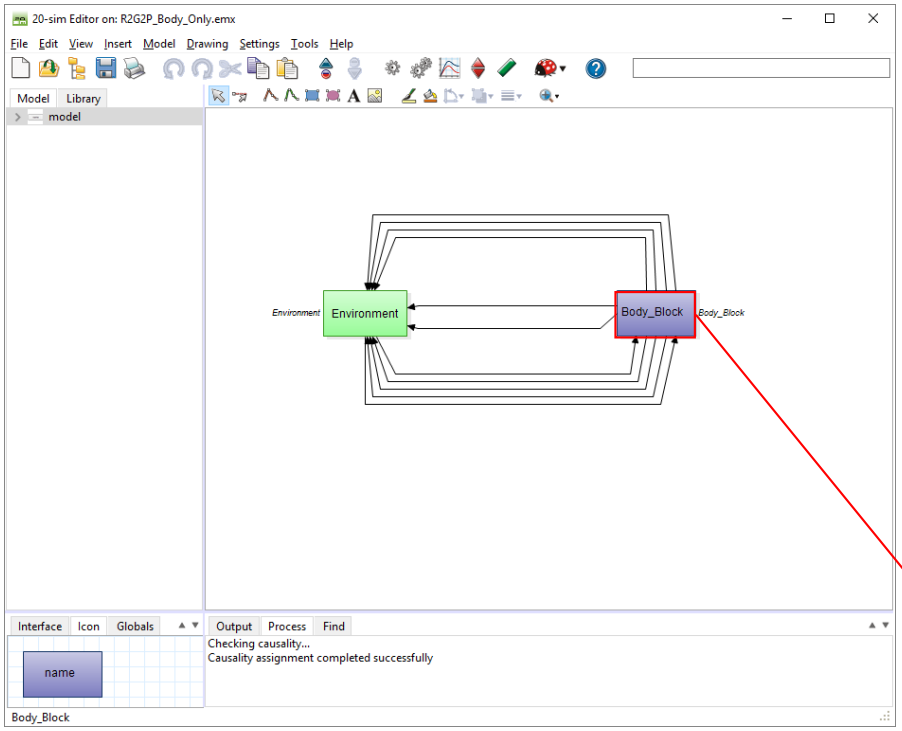
Step 8. Open the 20-sim physical model of the line follower robot by selecting *File* > *Open*.



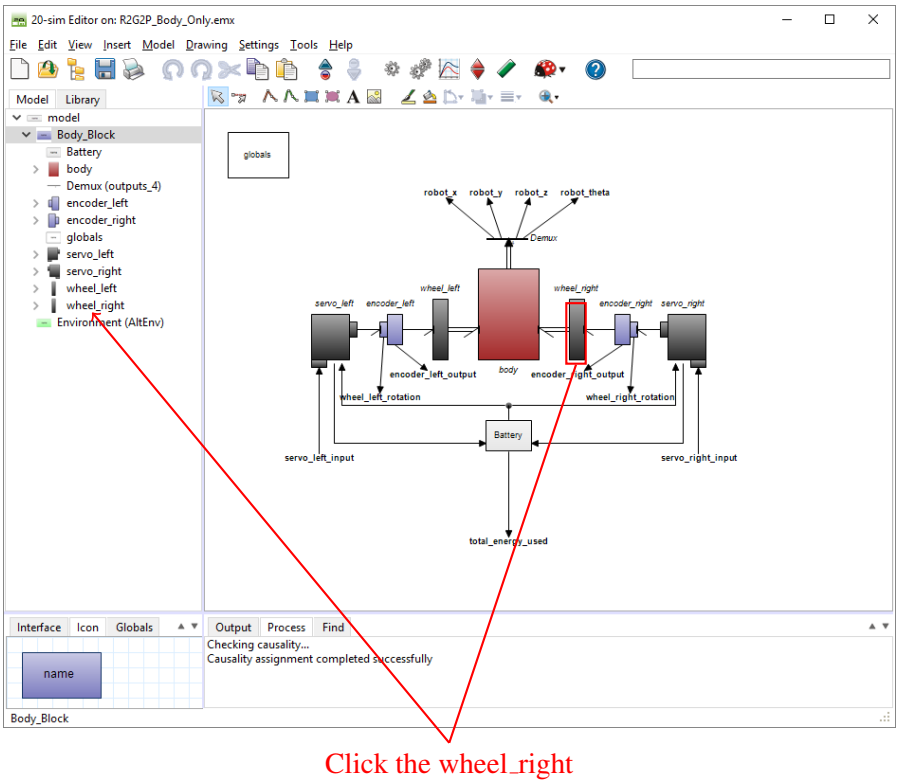
Step 9. Navigate to the location of *tutorial_5/Models/20-sim_Body* and *Open* the *R2G2P_Body_Only* model.



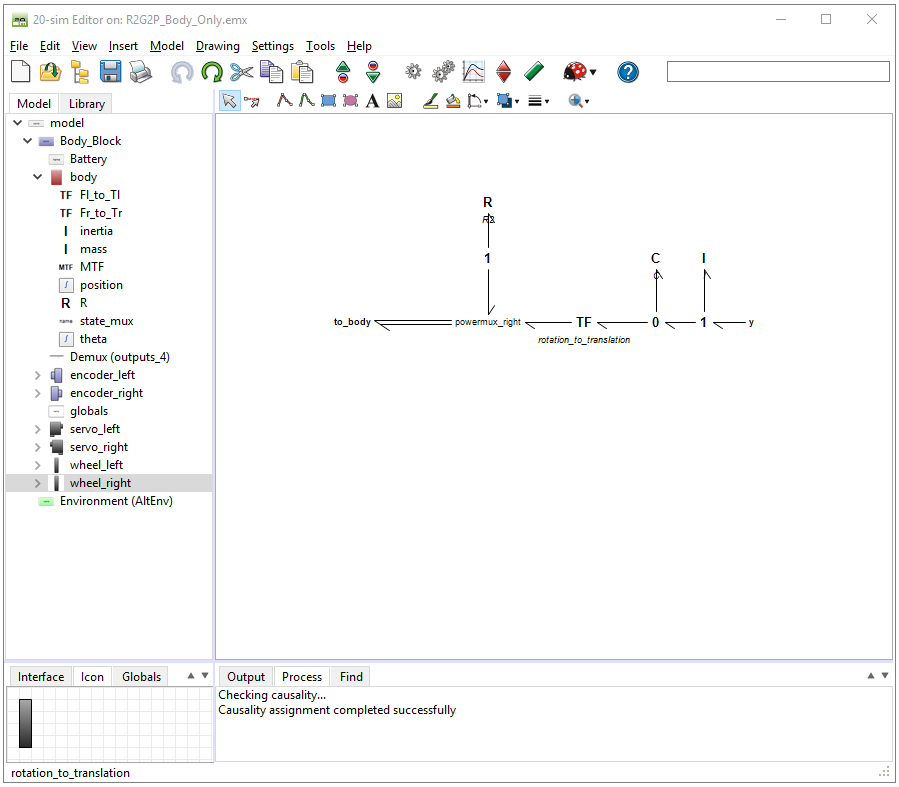
Step 10. You should see the block diagram for the model. The diagram is composed of two blocks: The *Environment* and the *Body_Block*. Click on the Body_Block to visualize its contents.



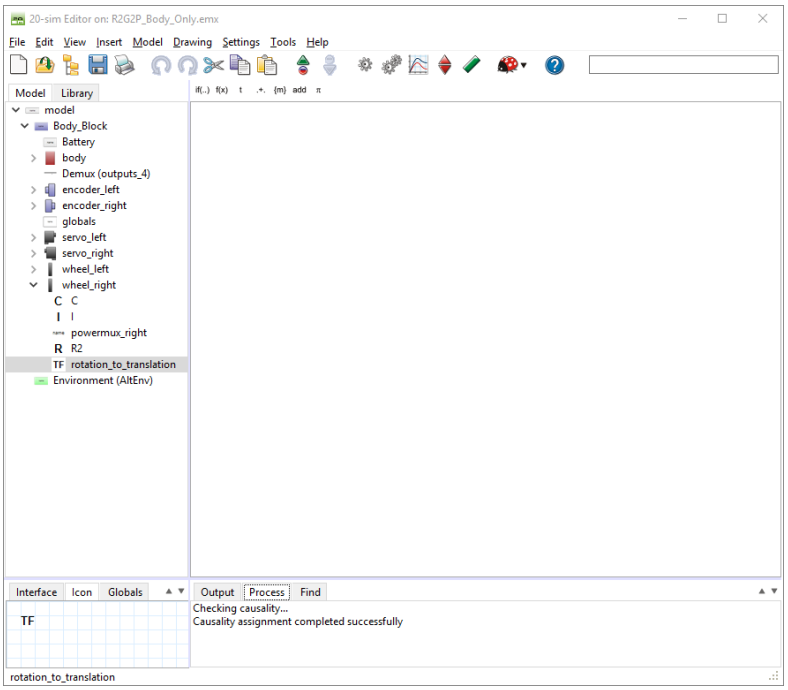
Step 11. At this step you can explore the model further by clicking on blocks or by clicking on the Model view. To continue our tutorial open the *wheel_right* submodel.



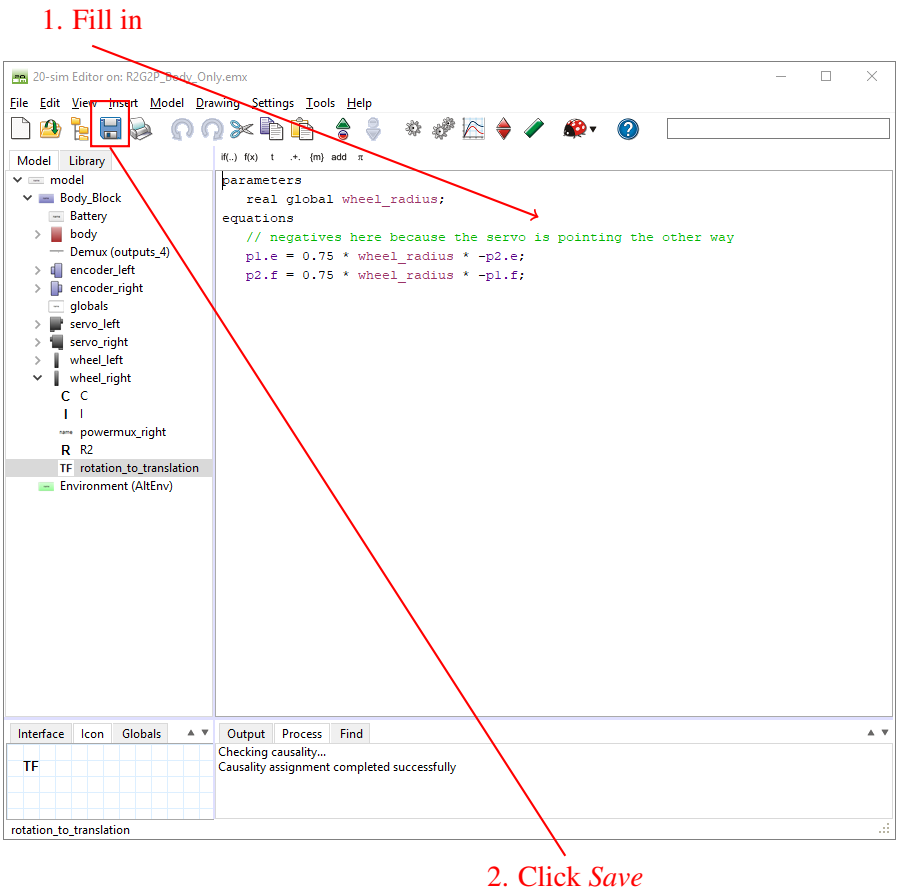
Step 12. The *wheel_right* models the physical behaviour of the line following robot wheel. You can explore the model but we will focus on the *rotation_to_translation* element. This element is a transformer which as it implied by its name transforms the rotation of the wheel into the translation of the line following robot body structure.



Step 13. After double clicking the transformer you should find its contents to be empty.



Step 14. Fill in the *rotation_to_translation* with the content shown on the figure, and save afterwards.

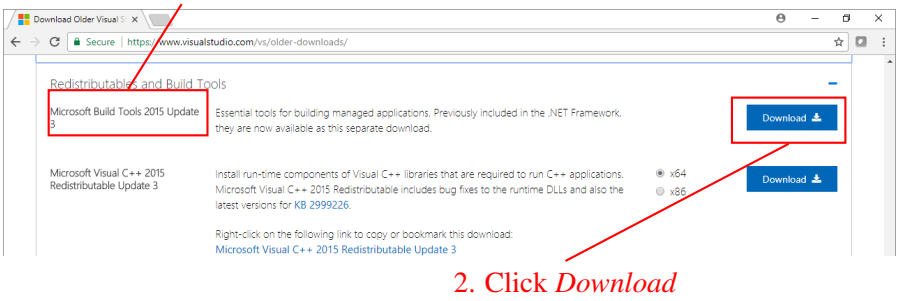


Congratulations! You edited your first 20-sim physical simulator.
In the next sections one need either a Windows system or an emulated Windows platform to be able to export an FMU for the physical model of the Line Following Robot. The simulator FMU can be provided to you and you can continue in Section 6. Nevertheless, it is interesting if you can follow the next steps by joining one of your colleagues.

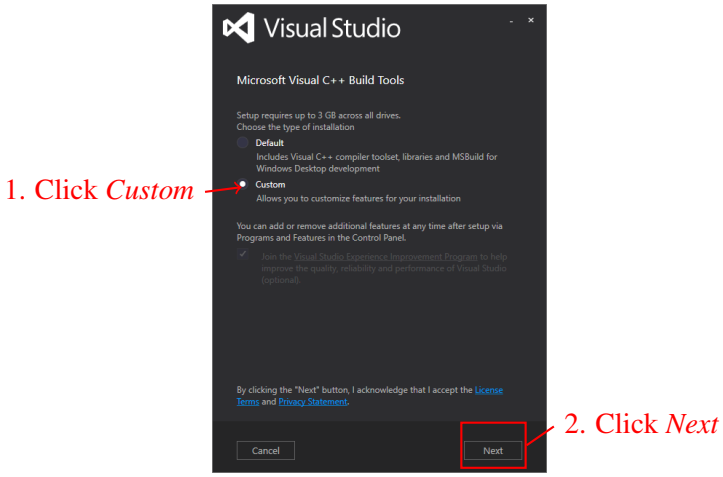
4 Installing Microsoft Build Tools (Windows Only)

The 20-sim is able to export new FMUs however *this feature depends on Microsoft Build Tools available for Windows platform only*.

Step 15. *The Microsoft Build Tools 2015* can be downloaded from:
`https://www.visualstudio.com/vs/older-downloads`
Scroll down to find the Microsoft Build Tools, and click download.

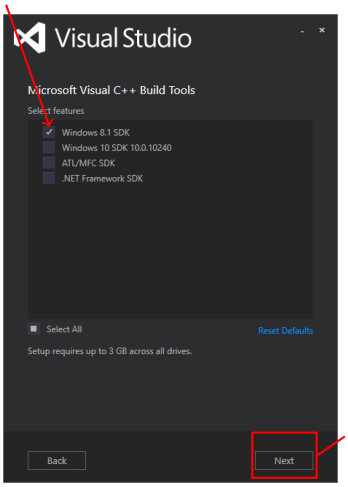


Step 16. You may need to login or create an account with Microsoft first. After executing the installer the following screen appears. Select the custom install and click next to continue.



Step 17. Confirm the Windows 8.1 SDK is selected and proceed with the installation by clicking Next.

1. Check selection

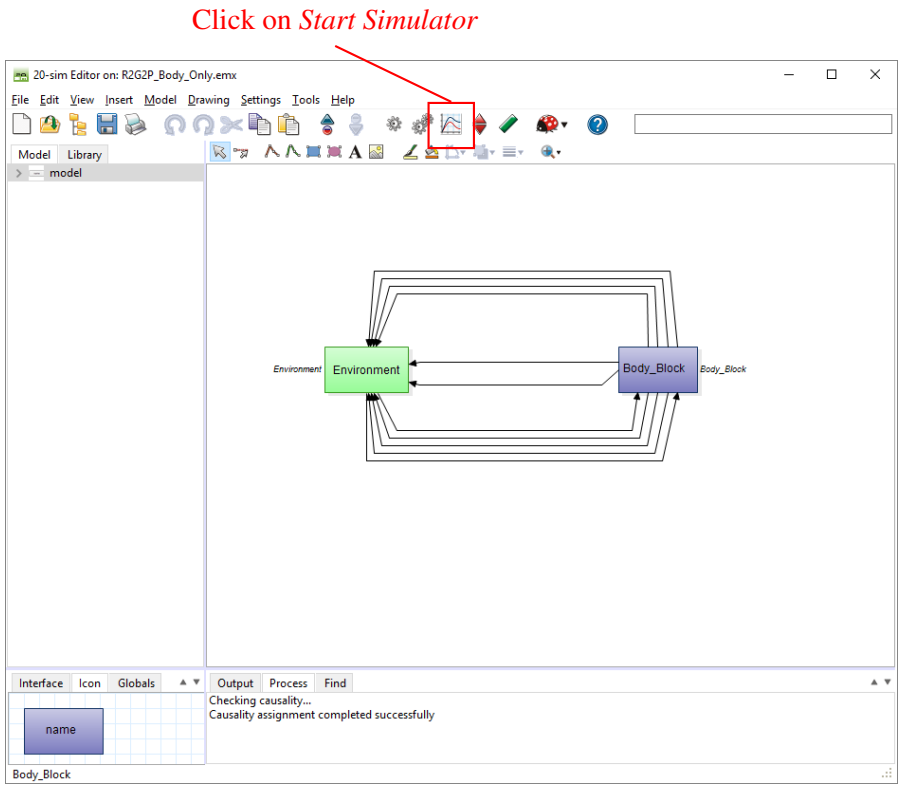


2. Click *Next*

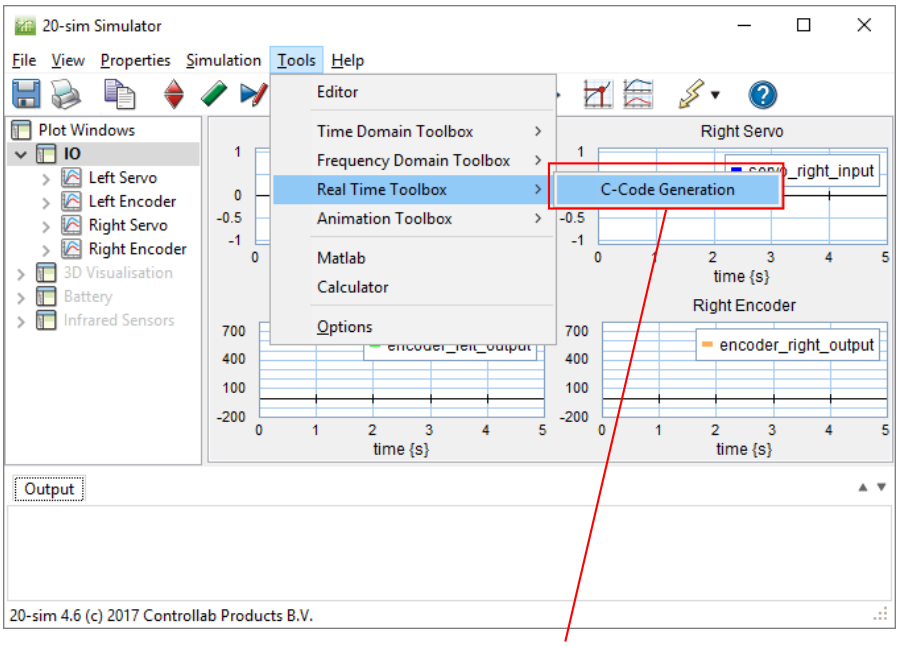
After finishing the install your Windows system is ready to generate an FMU from within the 20-sim tool.

5 Exporting an FMU and Adding it to a Multi-model

Step 18. Reload the 20-sim model again by repeating Step 7. to Step 9. and click *Start Simulator* to open the Simulator Window.



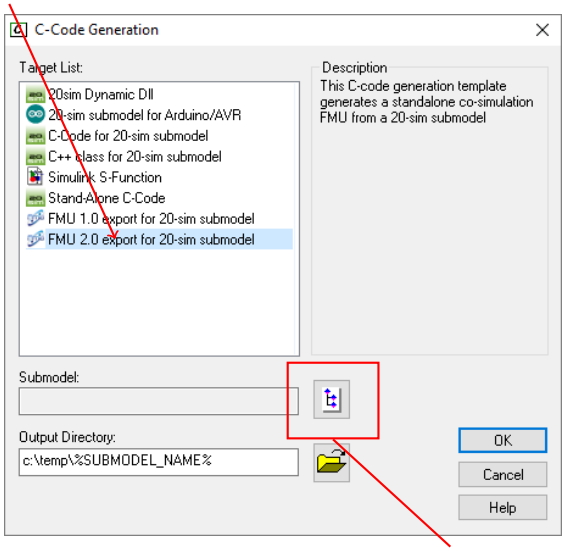
Step 19. After the *Simulator Window* appears, navigate to *Tools> Real Time Toolbox> C-Code Generation* and click *C-Code Generation*.



Click *C-Code Generation*

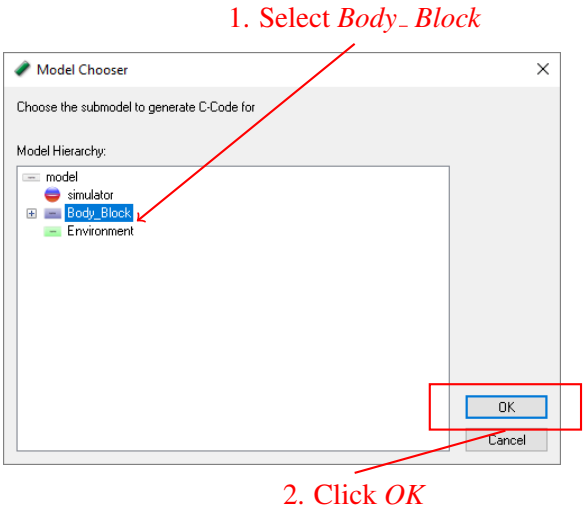
Step 20. Select *FMU 2.0 export for 20-sim submodel* from the *Target List* and click on the *Submodel* icon.

1. Click on *FMU 2.0 export...*

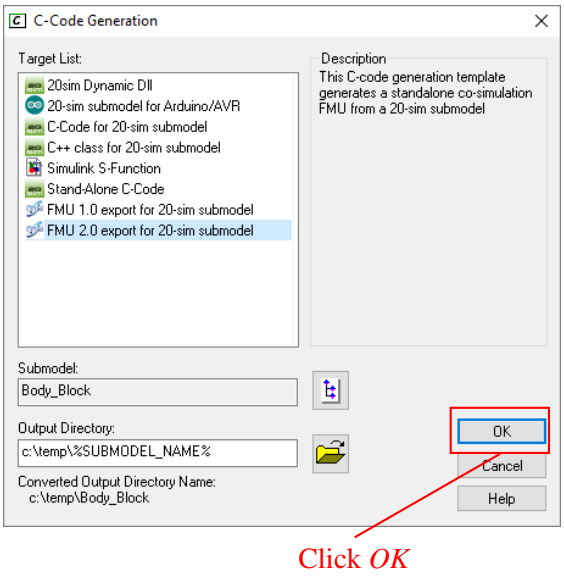


2. Click on the *Submodel* icon

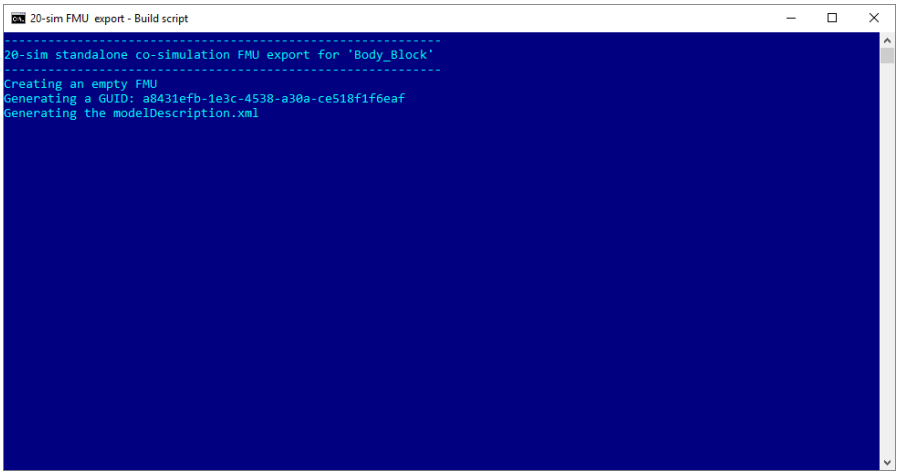
Step 21. A model chooser appears. Select the *Body_block* in the Model Hierarchy.



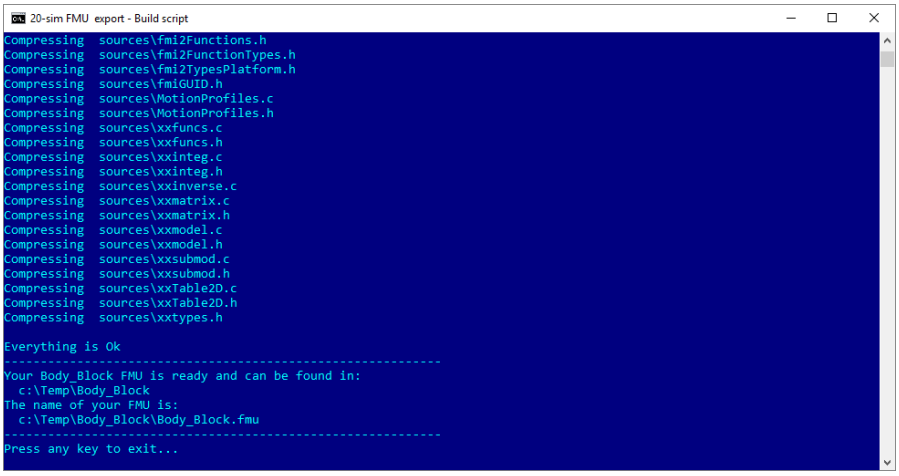
Step 22. You may change the location for the FMU output by changing the *Output Directory* or leave it to its default location and click *OK* to generate the FMU.



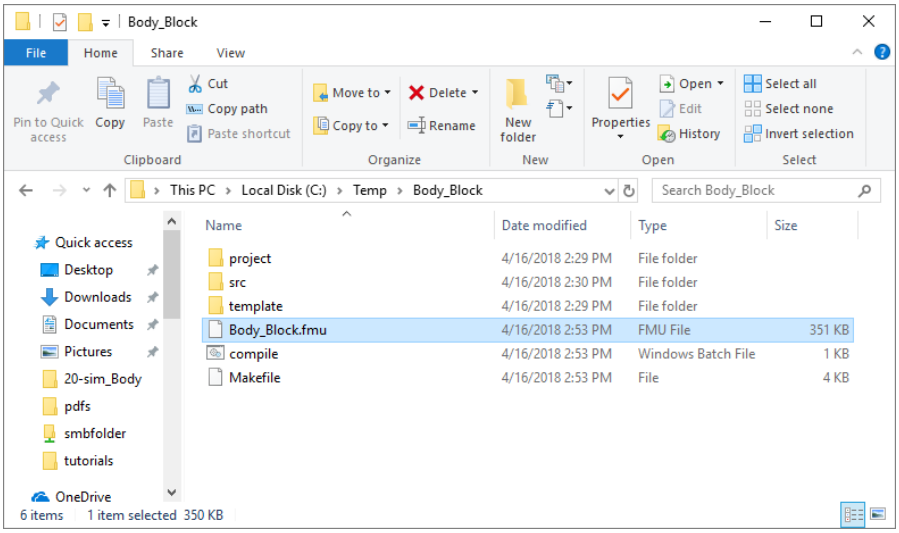
Step 23. The following shell should pop up and the script to be run may take a while to complete...



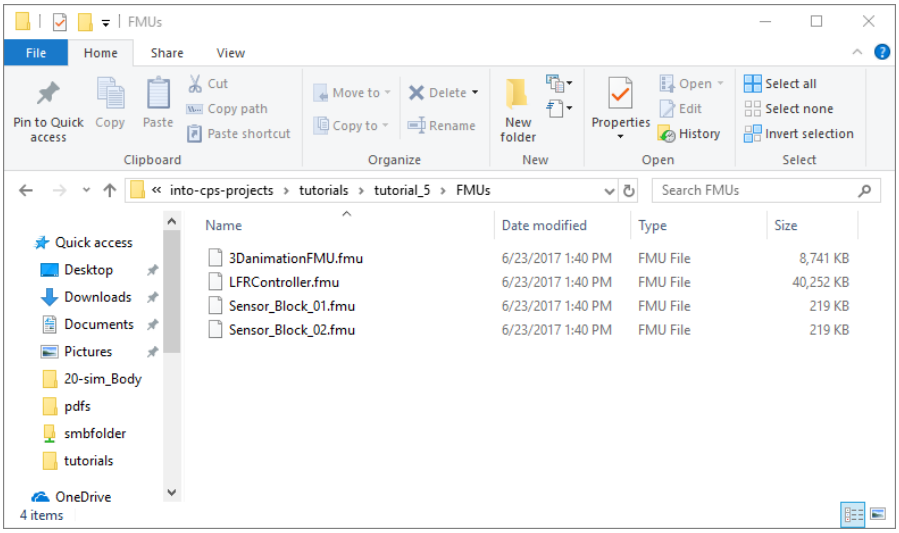
Step 24. When the script is finished and the FMU generation succeeded you should see the following message.



Step 25. Navigate to the chosen folder and copy the *Body_Block.fmu*.



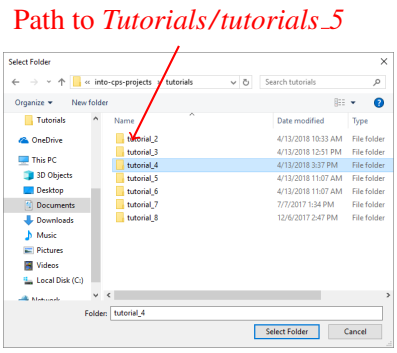
Step 26. Paste the *Body_Block.fmu* into the FMUs folder inside tutorial 5..



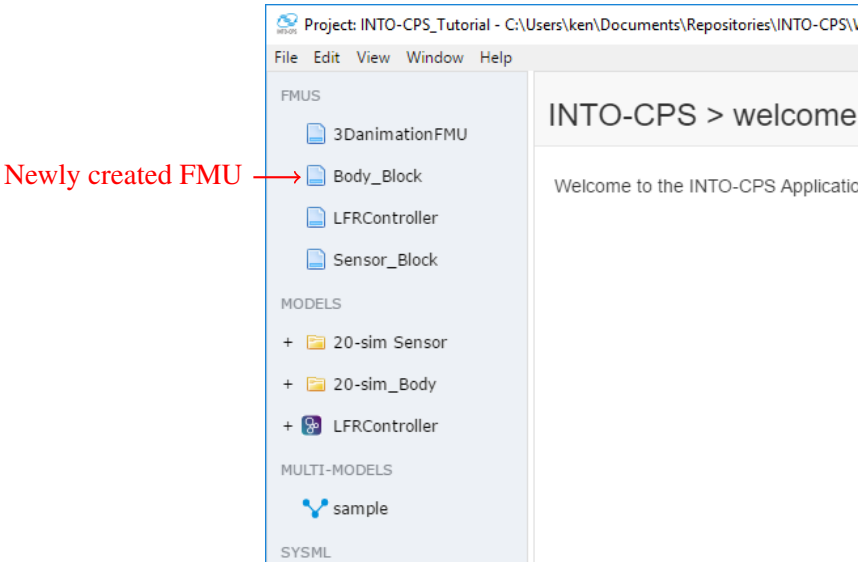
Congratulations! You are now ready to use the generated FMU in a co-simulation.

6 Co-simulating with the New Simulator

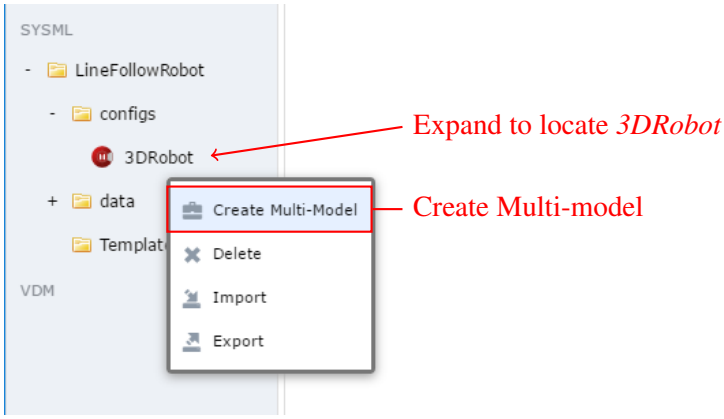
Step 27. Launch the *INTO-CPS Application* and select *File* > *Open Project*. Set the *Project root path* to the location of *Tutorials/tutorials_5* and click *Open*. You can browse using the *Folder* button.



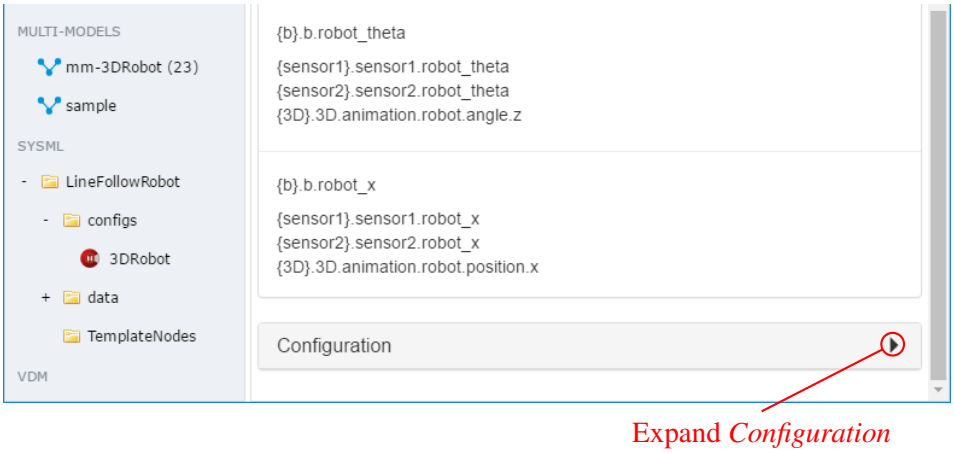
You should see the newly export *Body_Block* FMU in the list.



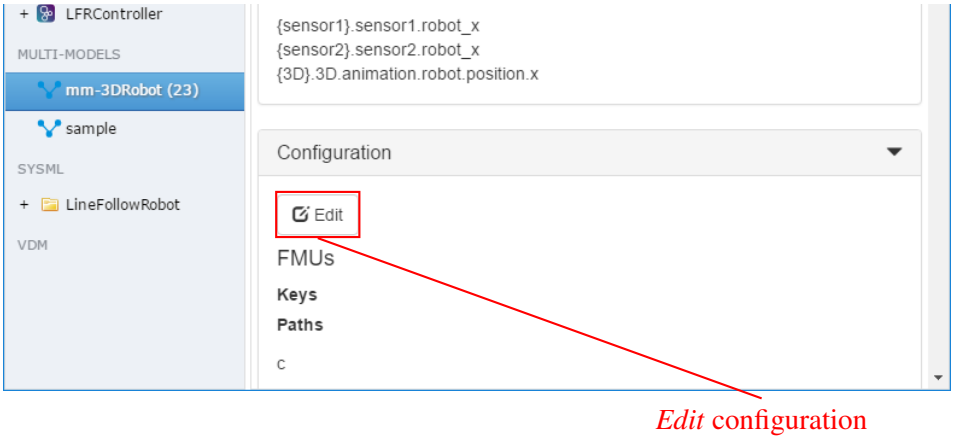
Step 28. In the SysML entry of the project browser, expand the *LineFollowRobot* folder, then *config* folders. Right-click on *3DRobot* and select *Create Multi-Model*.



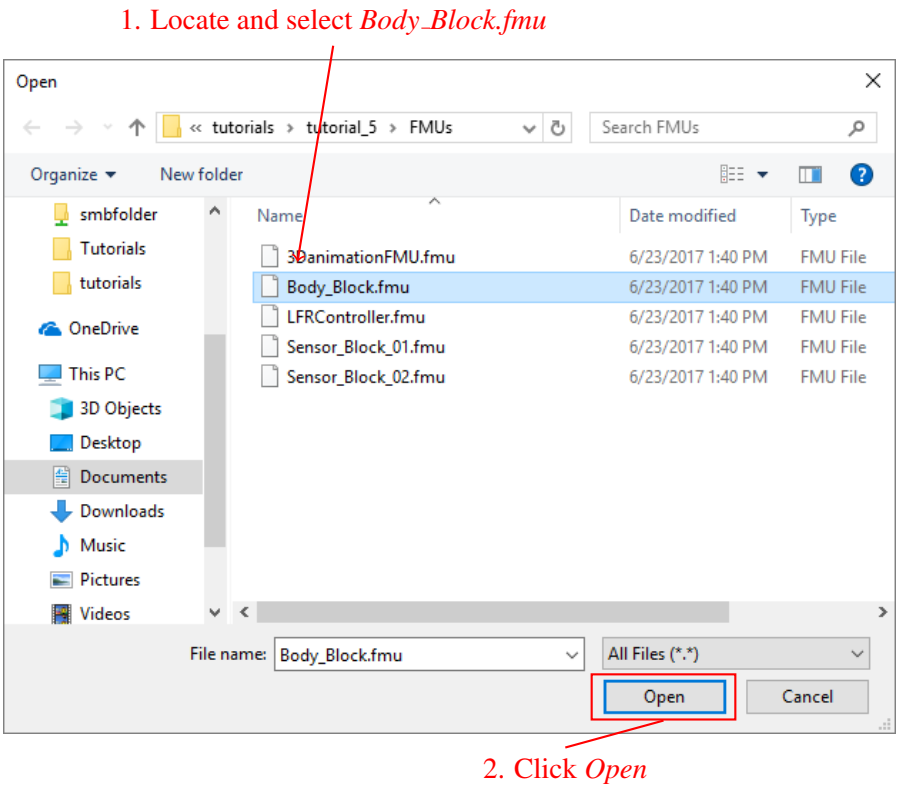
Step 29. We now need to associate FMUs to the multi-model as we did in *Tutorial 2*. Scroll down to find the *Configuration* panel and expand it by clicking the arrow.



Step 30. Scroll down and click *Edit*.



Step 31. As in *Tutorial 2*, in the FMUs section press *File* next to the Body_ Block element, *b*. A file browser window will open and show five FMUs (if the file browser does not show the FMUs, navigate to *tutorials_5/FMUs*). Select *Body_Block.fmu* and click *Open*.

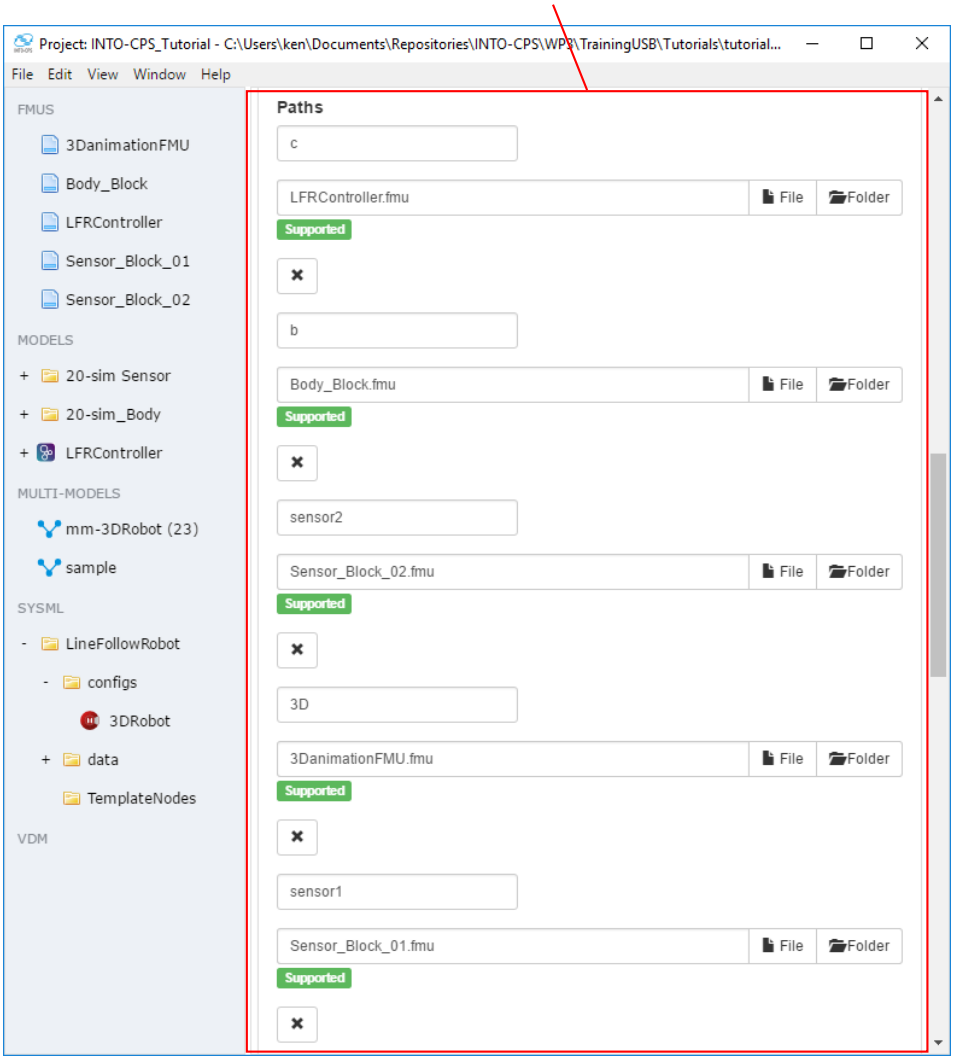


Step 32. Repeat this for the remaining elements:

- *c* : *LFRController.fmu*
- *3D* : *3DanimationFMU.fmu*
- *sensor1* : *Sensor_Block_01.fmu*
- *sensor2* : *Sensor_Block_02.fmu*

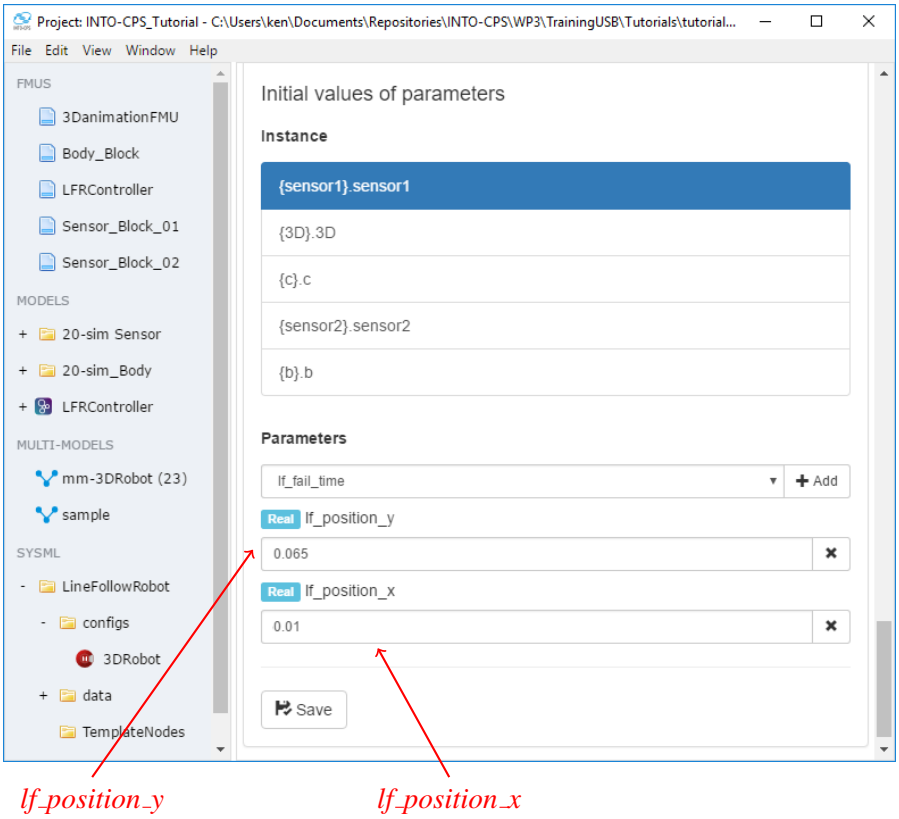
The complete set of FMUs will look like this:

FMUs added


Project: INTO-CPS_Tutorial - C:\Users\ken\Documents\Repositories\INTO-CPS\WP3\TrainingUSB\Tutorials\tutorial...
File Edit View Window Help
FMUS
3DanimationFMU
Body_Block
LFRController
Sensor_Block_01
Sensor_Block_02
MODELS
20-sim Sensor
20-sim_Body
LFRController
MULTI-MODELS
mm-3DRobot (23)
sample
SYSML
LineFollowRobot
configs
3DRobot
data
TemplateNodes
VDM
Paths
c
LFRController.fmu
File
Folder
Supported
b
Body_Block.fmu
File
Folder
Supported
sensor2
Sensor_Block_02.fmu
File
Folder
Supported
3D
3DanimationFMU.fmu
File
Folder
Supported
sensor1
Sensor_Block_01.fmu
File
Folder
Supported

Step 33. Scroll down to the *Initial values of parameters section*, and click {*sensor1*}.*sensor1*. In the *Parameters* section, enter the following values:

- *lf_position_y* = 0.065
- *lf_position_x* = 0.01

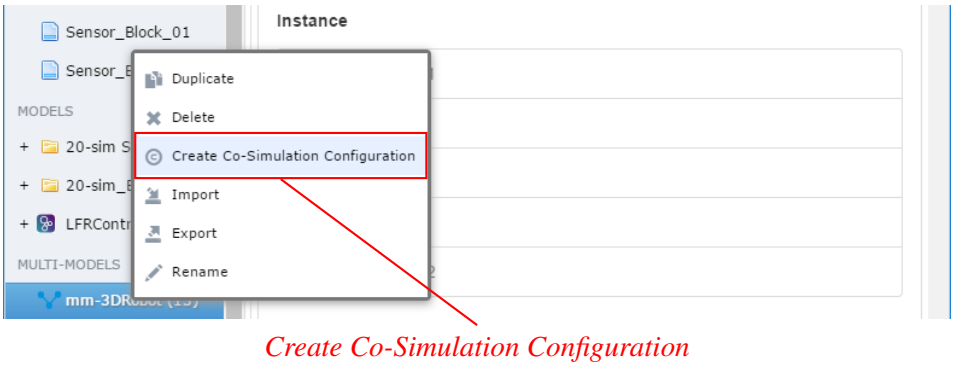


Step 34. Repeat the previous step for the second sensor, {*sensor2*}.*sensor2*, with the following values:

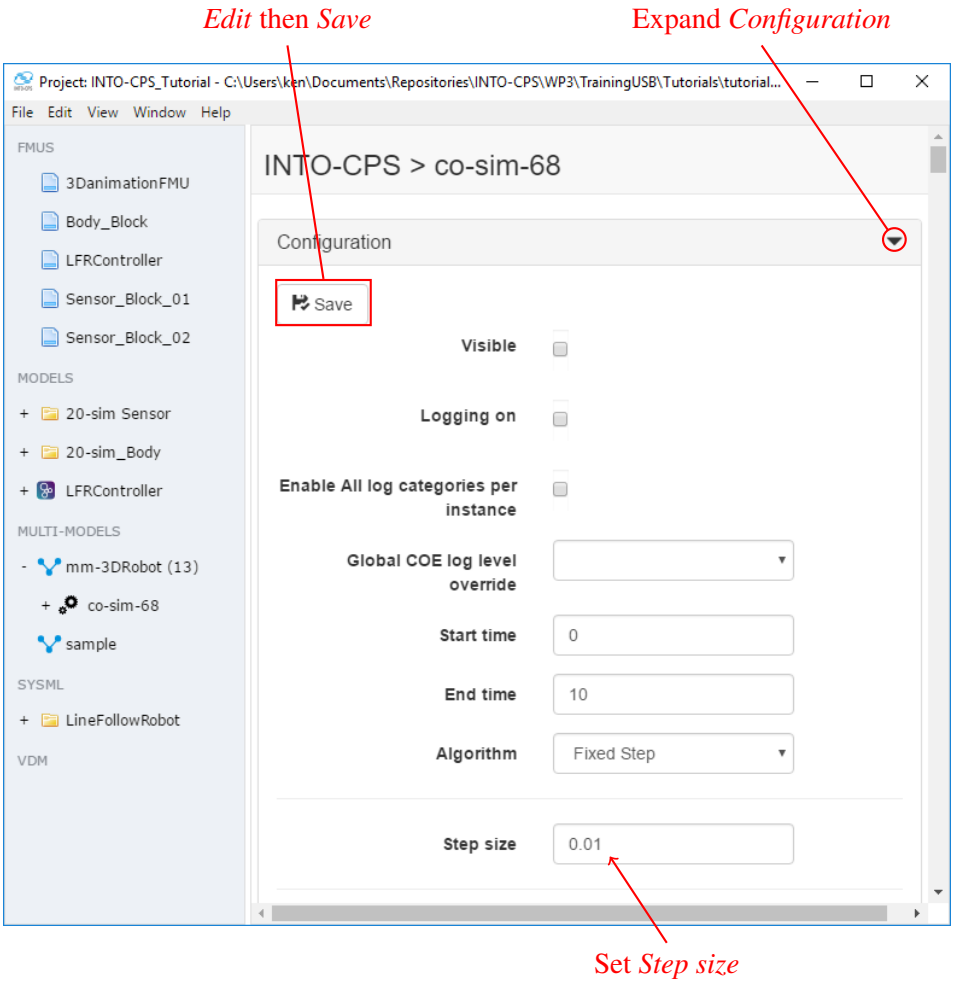
- *lf_position_x* = -0.01
- *lf_position_y* = 0.065

Step 35. *Save* the *Configuration*.

Step 36. Right-click on the new multi-model configuration and select *Create Co-simulation Configuration*.

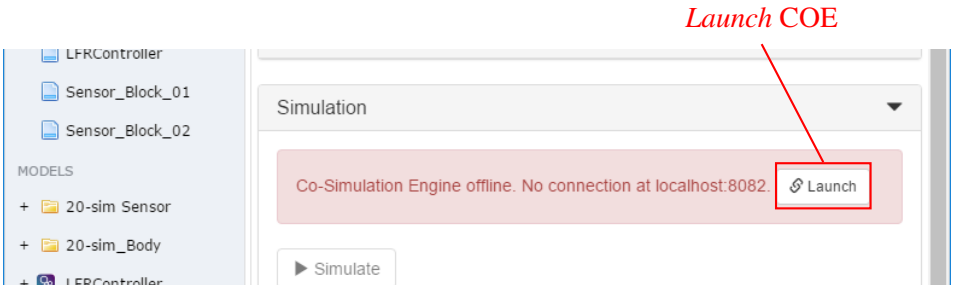


Step 37. Set the *Step size* to 0.01. Don't forget to press *Edit* then *Save*.

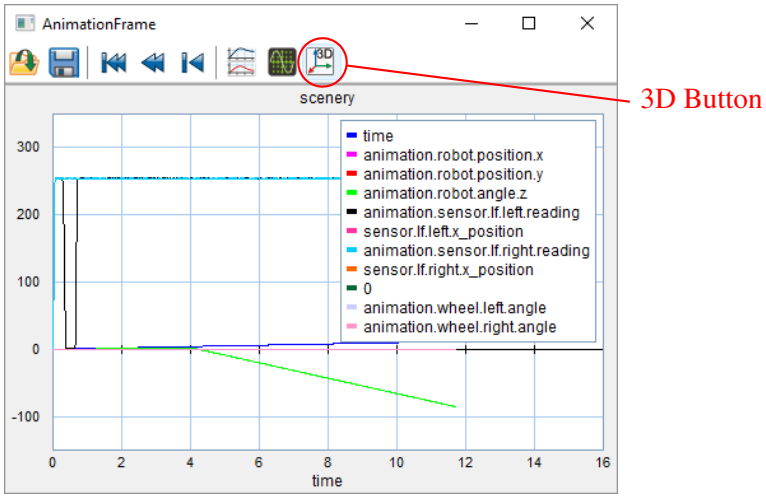


Step 38. Check *lf_1_sensor_reading* from {*sensor1*}.*sensor1* and {*sensor2*}.*sensor2* to see the sensor values appear in the *Live Plotting*.

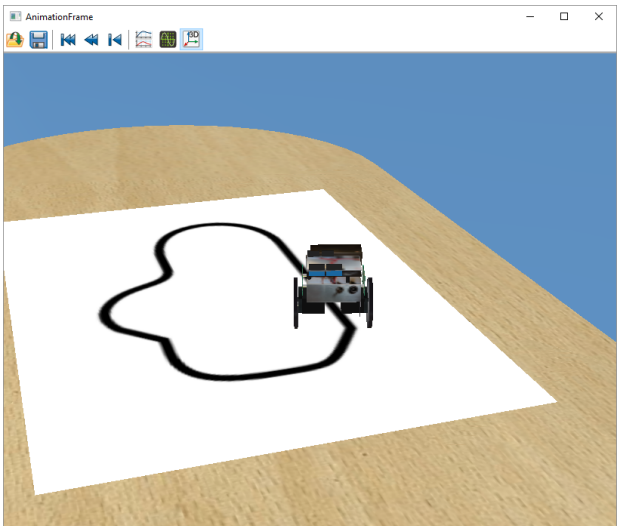
Step 39. Launch the COE if necessary (see *Tutorial 1 — First Co-simulation* for a reminder if needed).



Step 40. When the COE is running, click the *Simulate* button. The *Animation Frame* should appear. You can click the *3D* button to see the 3D visualisation of the robot.



Step 41. If everything went well, the robot should follow the line as in *Tutorial 2 — Adding FMUs*. Although with a slight *difference*...



Can you spot the *difference*? You can go back to *20-sim* and explore[1] the physical model. Try to make some changes to obtain the correct behaviour. Just repeat Step 18. to Step 26. to regenerate and copy the FMU, then press *Simulate*.

[1] Ask for a hint in case you feel lost...