

Practical Course Matlab/Simulink

Code Generation

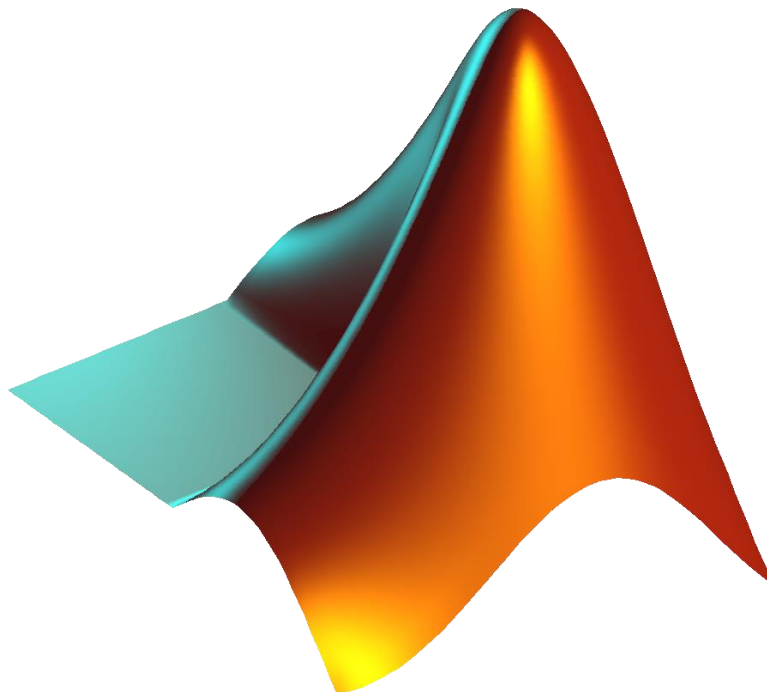


Table of Contents

Table of Contents.....	2
0 General Information and Advice	3
0.1 If you use your own computer.....	3
1 Generate Code from MATLAB Function	3
2 Generate Code from Simulink Model	4

0 General Information and Advice

This tutorial covers the toolboxes MATLAB Coder and Simulink Coder. Please keep in mind that the course has been designed for **MATLAB R2019a** and that other versions may require different setups or different approaches, especially in this tutorial.

0.1 If you use your own computer

For any type of code generation, MATLAB requires an additional (external) C/C++ compiler, such as for example *Microsoft Visual* or *MinGW*. Once the compiler is installed, open MATLAB and type `mex -setup` in the command window and hit enter. You will be then guided through a short configuration process, where you can select one of the compatible compilers installed on your computer.

1 Generate Code from MATLAB Function

Suppose you have created a fancy new **averaging filter**, implemented as a MATLAB function. You would like to run it with large amounts of data on a low-performance computer. However, you find that execution time is excessive, and you therefore have to wait for the results for too long. To speed up the execution of the filter algorithm, you decide to generate a MATLAB executable, also called MEX-file.

Exercise

- (1) The M-file of the filter is provided as `averaging_filter.m` in the folder Exercise 1. Make sure this file is on the MATLAB path or in the MATLAB Current Folder.
- (2) In the MATLAB main window, select the tab “Apps”
- (3) In the Apps tool strip, click the MATLAB Coder icon. The MATLAB Coder GUI will open.
- (4) In the field “Generate code for function:”, type “`averaging_filter`”. Hit Enter. Click Next.
- (5) Select “Let me enter input or global types directly”
- (6) Specify the input variable `x` as **`1xn double`**, with `n` unbounded (`double 1 x :inf`). Click Next.
- (7) For the purpose of this tutorial, you can skip the run-time issue check by clicking Next.
- (8) Select MEX as build type and C as language. You can choose a custom file name. Then click “Generate” to initiate code generation.
- (9) After successful code generation, you will see all the files generated on the left side of the GUI. Click “Continue” to see the generated C-Code.
- (10) Compare the original M-file with the C-Code. Note that the comments can be found in both files.
- (11) You can close the MATLAB Coder App.
- (12) Open the script `filter_test.m`. It generates a noisy signal and then calls both the M-file and the MEX-file to compare the execution times.
- (13) Make sure the MEX file name is the same you specified for code generation. Then execute the script. What are the execution times? Depending on the performance of your computer, you may need to adjust the length of the vector `v`.

2 Generate Code from Simulink Model

Usually, the second example in this tutorial covers the generation of executable code for an Arduino robot, and how to flash the software on the target hardware. However, since this is only useful with having the hardware ready and all necessary drivers installed, and therefore difficult to achieve for this online / remote practical course, we encourage you to go through the following example from the Mathworks website instead:

<https://de.mathworks.com/help/dsp/ug/generate-c-code-from-simulink-model.html>

To start the exercise, call the command `open_system('ex_codegen_dsp')`. This file is already installed with the MATLAB toolboxes. The focus with this example is on how to generate code with the Simulink Coder and to experiment with the configuration parameters (model settings).

It is not required to upload any files for this tutorial. The reason is that it is possible to run into any compiler configuration problems for the MATLAB or Simulink Coder, and we cannot ensure that every student is able to set up the toolchain on their computer. Nevertheless, feel free to use the Moodle forum and ask about any problems during the Q&A session.