Simulink Design Documenter Tool

May 2017

McMaster Centre for Software Certification (McSCert)

1 Introduction

The purpose of the Simulink Design Documenter is to facilitate the production of useful Software Design Description documents for software systems developed with Simulink. While producing documentation is widely seen as a tedious task, it is often not feasible to ignore it since many projects are too large and complex to easily understand. The existence of documentation will reduce difficulty of and time required for maintenance.

Further details than those provided in this document as well as additional options and capabilities can be found in the full guide.

What the Simulink Design Documenter provides

The Simulink Design Documenter generates a Word document with a title page, clickable table of contents, Changelog, Document Purpose section, Scope section, design details, a glossary, appendix, and a summary of warnings (if there were any during generation). The table of contents as well as overall formatting of the document are handled completely automatically, while the title page, Document Purpose, and Scope are also included automatically though may be configured or edited.

To provide design details for the system, the Simulink Design Documenter approaches documentation with the mentality that there will be subsystems within the 'main system' which are complex enough to warrant documentation for themselves and as such the Simulink Design Documenter allows the user to designate subsystems to document and will nest a subsystem's design details within an upper-level system's design details. For each system/subsystem being documented the Simulink Design Documenter will automatically produce a picture of the system, a list of subsystems within it, and an Interface section with information about the blocks which are involved in the interface. Beyond this each system will be documented with Purpose, Internal Design, Rationale, and Anticipated Changes sections as well as a Requirements Specification section if desired.

A note on other software documentation

A Software Design Description document is not the only document that *should* be produced in a project that uses Simulink. Among other documents, Software Requirements Specification documents are also very important to a successful project and they should not be overlooked. However, the Simulink Design Documenter focuses only on Software Design Description documents.

2 How to Use the Tool

2.1 Prerequisites

Please ensure the following, before using the tool:

- Use MATLAB/Simulink 2016b or newer to generate (the model can be saved in earlier versions).

- Simulink Report Generator is installed (check for it with ver command in MATLAB).

- The Simulink Design Documenter is present in your MATLAB path. If it is not, go to `File > Set Path...`, press `Add with Subfolders`, and select the `McMaster_Tools` folder. Restart MATLAB after doing so.

- The Signature Tool is present in your MATLAB path. See `https://www.mathworks.com/matlabcentral/fileexchange/49897-signature-tool` to download.

- The model is open (or loaded, for command line use).

2.2 Getting Started

The Simulink Design Documenter can be used via the Simulink Context Menu, which can be viewed by right-clicking in a model. The following option will be available in the Context Menu (as shown in Figure 1):

- *Generate Simulink Design Document*

2.3 Functionality

This section describes the tool functionality when being used from the Simulink Context Menu.

Generate Simulink Design Document

Right-clicking anywhere in the model and then selecting `Generate Simulink Design Document` from the Context Menu will begin generation of a Software Design Description document which will be named $SDD\_ < MySystem > .docx$ (where "$< MySystem >$" should be replaced with the name of the system being documented) and saved in the current folder in MATLAB (the cd command will return your current folder).

Prior to generation a file should be placed on the MATLAB path named "$< MySystem > \_SDD\_Config.m$" (where "$< MySystem >$" should be replaced with the name of the system being documented). Variables set in that file will be used by the Simulink Design Documenter to configure the way the document will generate. A template config file which explains the different options can

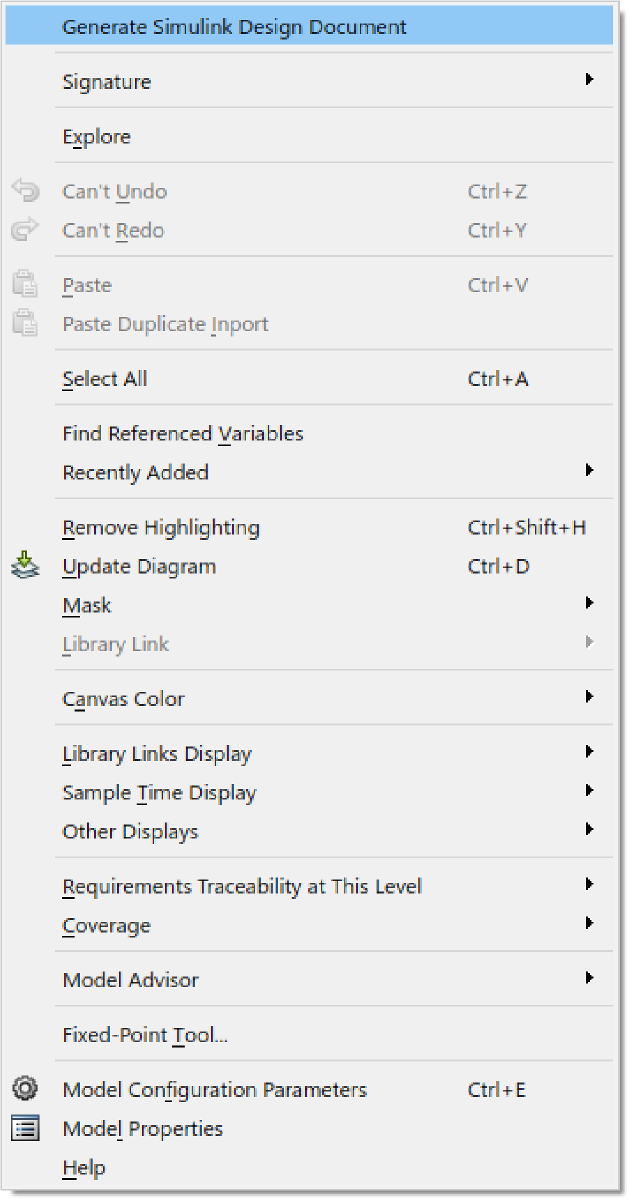


Figure 1: Simulink Context Menu with tool option visible.

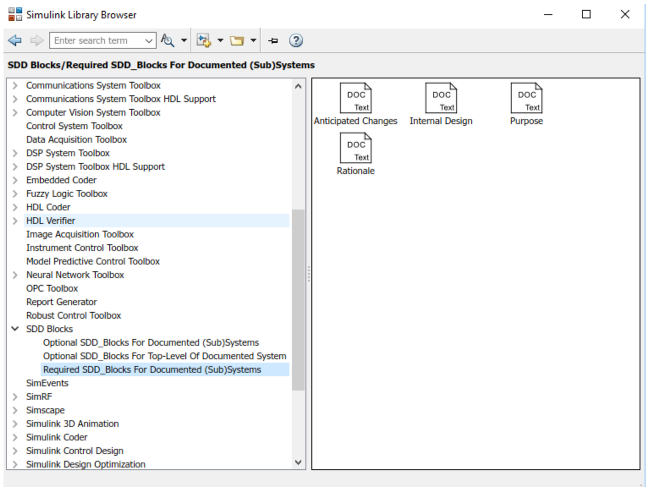

Simulink Library Browser
Enter search term
SDD Blocks/Required SDD_Blocks For Documented (Sub)Systems
Communications System Toolbox
Communications System Toolbox HDL Support
Computer Vision System Toolbox
Control System Toolbox
Data Acquisition Toolbox
DSP System Toolbox
DSP System Toolbox HDL Support
Embedded Coder
Fuzzy Logic Toolbox
HDL Coder
HDL Verifier
Image Acquisition Toolbox
Instrument Control Toolbox
Model Predictive Control Toolbox
Neural Network Toolbox
OPC Toolbox
Report Generator
Robust Control Toolbox
SDD Blocks
Optional SDD_Blocks For Documented (Sub)Systems
Optional SDD_Blocks For Top-Level Of Documented System
Required SDD_Blocks For Documented (Sub)Systems
SimEvents
SimRF
Simscape
Simulink 3D Animation
Simulink Coder
Simulink Control Design
Simulink Design Optimization
DOC Text
Anticipated Changes
DOC Text
Internal Design
DOC Text
Purpose
DOC Text
Rationale

Figure 2: SDD Blocks library in the library browser showing which DocBlocks are required for each system/subsystem being documented.

be found at Report_Specific_Files/TopsysName_SDD_Config.m. In particular, users are urged to look into the subsystemList variable described in that file and to choose a list of subsystems to document within the system.

Next, the user should add DocBlocks to the system being documented as well as those listed by the subsystemList variable. The DocBlocks may be ordinary ones, however they must have specific names, to simplify adding these blocks, we have created a block library called "SDD Blocks" which contains optional DocBlocks as well as mandatory ones (the mandatory ones are shown in Figure 2).

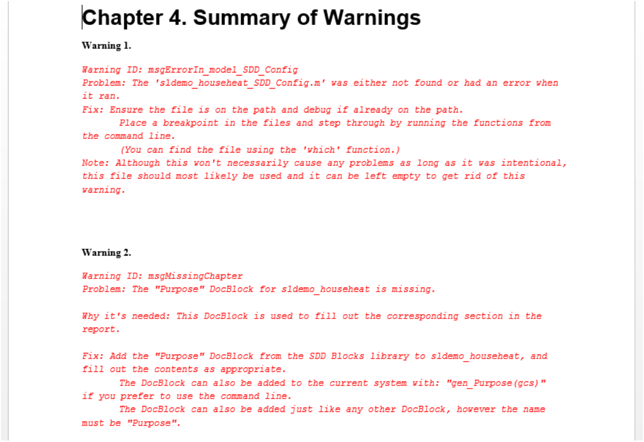
If a user attempts to generate without creating the config file or adding the DocBlocks, the generated report will give warnings at the bottom which explain what to do (as shown in Figure 3):

2.4 Errors and Warnings

Any errors or warnings during tool use will be visible in the MATLAB Command Window or at the bottom of the generated document. Typically, warnings will occur when any of the Purpose, Internal Design, Rationale, or Anticipated Changes DocBlocks are missing or if an appropriately named config file is not found.

3 Example

This section will only provide limited detail on an example usage of the Simulink Design Documenter, for a somewhat more in depth example please see the full



Chapter 4. Summary of Warnings

Warning 1.

```
Warning ID: msgErrorIn_model_SDD_Config
Problem: The 'sldemo_househeat_SDD_Config.m' was either not found or had an error when
it ran.
Fix: Ensure the file is on the path and debug if already on the path.
       Place a breakpoint in the files and step through by running the functions from
the command line.
       (You can find the file using the 'which' function.)
Note: Although this won't necessarily cause any problems as long as it was intentional,
this file should most likely be used and it can be left empty to get rid of this
warning.
```

Warning 2.

```
Warning ID: msgMissingChapter
Problem: The "Purpose" DocBlock for sldemo_househeat is missing.

Why it's needed: This DocBlock is used to fill out the corresponding section in the
report.

Fix: Add the "Purpose" DocBlock from the SDD Blocks library to sldemo_househeat, and
fill out the contents as appropriate.
       The DocBlock can also be added to the current system with: "gen_Purpose(gcs)"
if you prefer to use the command line.
       The DocBlock can also be added just like any other DocBlock, however the name
must be "Purpose".
```

Figure 3: Summary of Warnings section in a report generated for the sldemo_househeat system which comes with MATLAB.

guide.

Let's assume that we wish to create documentation for the top-level of the sldemo_househeat model which comes with MATLAB. The following steps will create a simple Software Design Description document for sldemo_househeat.

1. Open `sldemo_househeat`
2. Create a file on the MATLAB path called "`sldemo_househeat_SDD_Config.m`"
   - Set "`author = '<your name>';`" and set "`subsystemList = {};`"
3. Add 4 DocBlocks to the top-level system with the following names: Purpose, Internal Design, Rationale, and Anticipated Changes. Normally you should then open each DocBlock and write something meaningful (described below), but instead just write something arbitrary to get a feel for the program.
   - The Purpose DocBlock should normally describe the purpose of the system.
   - The Internal Design DocBlock should normally describe how the system works to accomplish the purpose.
   - The Rationale DocBlock should normally describe why certain design decisions were made.
   - The Anticipated Changes DocBlock should normally describe expected changes to the system (such as from changes in project scope or requirements).
4. Go to the MATLAB command line and enter "`GenSDD('sldemo_househeat')`"

- The document should open automatically when generation is complete.

4 Matlab Commands

The tool can also be used via the Matlab command line, with the following classes and functions.

Function	`GenSDD`
Syntax	`GenSDD`(*topsys*)
Description	Generate the Software Design Description document for *topsys*.
Inputs	*topsys*: Name of the system to generate the documentation for. It should be a specific subsystem name.
Outputs	N/A