Automating Model Based Design and Code Generation using MATLAB and Simulink Tools

Authors: Umamaheswara Reddy Karumuri, Jamuna Konda, Venkat Ramana Reddy Anchuri
Infotech Enterprises Ltd, Hyderabad, India

August 8, 2013
Agenda

- Challenges in multi-domain modeling
- Problem definition
- Solution
- Methodology
- Automation Module Overview
- Conclusion
- References
Challenges in multi-domain modeling

- Engineers working on development of complex systems such as Aircraft systems, face tough real world challenges on a daily basis.

Some of them are,

- Increase in the complexity of systems,
- Dealing with huge number of requirements,
- Coordinating with multiple disciplines and domains,
- Changes in requirements at later stages of software development life cycle, etc…

- One of the effective ways to deal with the above challenges is, to automate the development process steps wherever feasible. It improves the code efficiency, reduces effort and cost.
Problem Definition

- Engineers working in respective keep designing Simulink models (more than ~100 per sub system). With change in current requirements or additional requirements, those models are required to be updated by the respective engineers. Thereafter, the models are integrated into a single model which is used to generate embedded code.

- Different models are handled by different engineers. The configuration settings for each model has to be set manually by respective engineer, which is time consuming and tedious. If there is an update in any single model, the integrated model needs an update before generating the code.

- With huge number of Simulink models, the manual process results in longer development time & consequently more cost. Hence if this process step is automated, the challenge, due to numerous requirements and changes, can be met efficiently.
In order to reduce the manual process of integrating the models, setting configurations for each model and generation of code, an automation module is developed using MATLAB script.

Using automation script, one can compile the generated code and build a dynamic link library. MATLAB, Simulink & Simulink coder are used to develop the automation module.

The automation module reduces the risk of re-coding when the requirements change. The Simulink models can be updated incrementally with respect to the new requirements & the automation module is run on the models for production code & DLL generation.

This process is simple & effective, irrespective of the number of models to be handled, to apply predefined configuration settings for all the models.

Customized configuration settings and TLC file help to generate optimized code in terms of memory and computation time which will also be compatible with the interface based on its requirements.
MATLAB scripting is used to accomplish the following methodology.

1. **Group all the selected Simulink diagrams into a single Simulink model**
2. **Link the diagrams**
3. **Compile the model**
4. **Apply model configuration settings**
5. **Apply customized Target Language Compiler**
6. **Generate embedded code**
Automation Module Overview

- Automation script groups all the Simulink diagrams into a single Simulink model using Simulink model building commands, links the diagrams, compiles the model, invokes Real Time Workshop Embedded Coder for RTW configuration settings and code generation using customized target language compiler, compiles the RTW code and generates the DLL.
- The automation module checks & updates the configuration settings & block parameters of each Simulink model before integration.
- The automation module also generates embedded code for each Simulink model & integrated model.
Conclusion

- Model Based Design (MBD) using MATLAB and Simulink is quick to set up, easy to handle and offers superior data repeatability that can save both time and costs.
- Automation script groups all the selected Simulink diagrams into a single Simulink model using Simulink model construction commands, links the diagrams, compiles the model, apply model configuration settings, applies customized Target Language Compiler (TLC) and generate embedded code.
- The automation module reduces the risk of re-coding when the requirements change and minimizes the effort to integrate the models.
- In our experience, approximately ~95% of the time is saved by using our MATLAB automation module as compared to the manual process of model integration.
References

- MATLAB, The Mathworks Inc. MATLAB is a software tool for doing mathematical computations
- Mathworks website:
  http://www.mathworks.in/products/matlab/?s_cid=wiki_matlab_15
- SIMULINK product page on the Mathworks website.
  http://www.mathworks.in/products/simulink/
- SIMULINK Embedded Coder tool
  http://www.mathworks.in/products/embedded-coder/
Thank You

www.infotech-enterprises.com

We deliver Global Engineering Solutions. Efficiently.

© 2011 Infotech Enterprises. All Rights Reserved

This page contains no technical data – Subject to the EAR or the ITAR