Code generation for STM32 MCUs using MATLAB® and simulink®

March 2020
• Mandatory Software

  • From Mathworks
    • MATLAB®, Simulink and Embedded Coder (2018B)
  • From STMicroelectronics
    • STM32CubeMX (5.6.0)
    • STM32-MAT/TARGET toolkit to develop STM32 applications (STM32MatTarget_5.6.0_setup.zip)

• One of following Toolchain
  • EWARM from IAR (8.32.3)
  • MDK-ARM from Keil (5.29.0)
  • STM32CubeIDE from STMicroelectronics (1.3.0)

• Hardware

  • Any STM32 based electronic application board with SWD/JTAG connection.
  • STLinkV2/V3 or 3rd party debug probe
  • Virtual Com Port implemented between STM32 and Computer
• Arm®/ST/MathWorks initiative
  • Optimized Cortex-M code for Embedded Coder based on Arm’s CMSIS DSP SW library
  • Simulink Peripheral blockset provided by ST based on STM32Cube abstraction layers supporting most of STM32 portfolio
  • Compiler / IDE integration for MATLAB, based on Arm Keil, IAR, and STM32CubeIDE

• Enables Model-Based Design Workflow for STM32 MCUs
  • Create executable models
  • Generate code automatically
  • Automate software integration

• Reduces cost and time, improves quality
  • ~33% cost & time reduction in ECU development
  • ~50% fewer specification errors
Why matlab/simulink on STM32?

• Simulation maybe of interest when creating a mathematical algorithm
• Matlab/Simulink enable STM32 simulation and on chip algorithm execution
• The algorithm can use the actual STM32 peripherals, ending up with a complete processing 100% done in the MCU, with no written lines of code and a complete model-driven approach, less error prone
# Quick description of tools

<table>
<thead>
<tr>
<th>Tool</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Matlab</td>
<td>High level language for complex calculations</td>
</tr>
<tr>
<td>Simulink</td>
<td>Graphical development environment, complete environment of simulation and implementation of embedded systems.</td>
</tr>
<tr>
<td>Embedded Coder</td>
<td>C code generation for embedded systems, embedded system interface</td>
</tr>
<tr>
<td>STM32Cube Embedded Software</td>
<td>Collection of embedded software components, highly portable from one STM32 to another</td>
</tr>
<tr>
<td>STM32CubeMX</td>
<td>Configuration software tool on the PC, able to generate initialization C code based on user choices</td>
</tr>
</tbody>
</table>

## Toolchain

A toolchain from one of our partners is required to compile and link C code generated by Embedded Coder, STM32CubeMX and STM32Cube embedded software.
Using scenarios and tools

• **Step 1: Pure simulation**
  • Everything done on the PC

• **Step 2: Processor-in-the-loop (PIL)**
  • Algorithm fully executed on STM32 MCU
  • Data (Input or output) exchanged between MATLAB/Simulink and STM32 MCU via UART

• **Step 3: Everything on STM32**
  • Data (input or output) gathered from STM32 peripherals (ADC, Timers, …) and algorithm fully executed on STM32

• **Step 4: External mode**
  • Same behavior than step 3. Data monitoring from Simulink via UART. Not possible to modify STM32 configuration during External Mode.

---

<table>
<thead>
<tr>
<th>MATLAB/Simulink</th>
<th>STM32Cube Embedded Software</th>
<th>Toolchain</th>
</tr>
</thead>
<tbody>
<tr>
<td>Embedded Coder</td>
<td>STM32CubeMX</td>
<td></td>
</tr>
</tbody>
</table>

*: used only for UART
Step 1: pure simulation
MATLAB® & simulink

- Not a subject for this presentation: You do your model ready, simulate and test it on the PC.
- General purpose for MATLAB®/Simulink
- Doesn’t need STM32-MAT/TARGET toolkit.
- Contact MathWorks for MATLAB®, Simulink trainings
Step 2: processor-in-the-loop
• PIL is used to run MATLAB® algorithm on STM32 device
  • To see if STM32’s computational results are numerically equivalent to PC-based simulation results.
  • To measure directly the time to execute the Simulink model steps (profiling)

• PIL simulation uses serial port to send data from Simulink to STM32 and receive Simulink processed data back from STM32

• PIL simulation doesn’t run in «real-time»
  • Simulink is «master» sending asynchronously data to STM32 through serial port.
  • STM32 device waits for data from Simulink, process data (in real-time) and sends result back to Simulink through serial-port.

• PIL simulation doesn’t process real data from STM32 peripherals.
  • STM32 peripherals (ADC etc…) are not used.
  • Only STM32 USART peripheral is used to communicate with Simulink.
Set simulation duration time and click run simulation green button

- Example: Simulate 1 second (given 1000 values from testdata, one for each 1 ms)
• PIL referenced model runs into STM32 target while the simulation model runs on PC
  • Automatic conversion of PIL model to .c/.h files
  • Automatic call to STM32CubeMX to create project
  • Manually build and download project to STM32 target from selected IDE
Step 3: everything on STM32
Handle data through real peripherals

- Configure your peripherals through a bridge with the STM32CubeMX tool
- Flash your device and run!
- More info at www.st.com/stm32-mat-target
• Generate code for the Simulink application
  • Press «Build Model» to automatically generate C code and IDE project.
• STM32 peripherals driver code is generated in .c/.h files which name is created using name of Simulink model and peripheral name.
STM32CubeMX code generation

- **STM32CubeMX process**
  - STM32CubeMX is automatically called from MATLAB® when «Download Application» has been selected from the Model Configuration window.
  
  - STM32CubeMX generates configuration code.
  
  - STM32CubeMX adds necessary library files .c/.h library files from HAL STM32 libraries.
  
  - STM32CubeMX generates toolchain project including files generated from MATLAB®
  
  - The project generated by STM32CubeMX can be opened

Click « Open Project » to automatically open project using selected toolchain.
Step 4: external mode
• Monitoring
  • As long as STM32 application is running, Simulink model parameters can be edited

• Monitoring means that only parameters can be modified. C generated code can't be modified. For example, it is not possible to replace x with + as x C code function has been generated.
• New Support Package for MATLAB, Simulink, and Embedded Coder
  • Generate optimized code
  • Automate build, integration and verification steps

• Now with ST and MathWorks you can:
  • Accelerate time to market
  • Reduce development costs
  • Improve product quality

• …Join the Design Community
Thank you