Latest Features in Simulink Coder

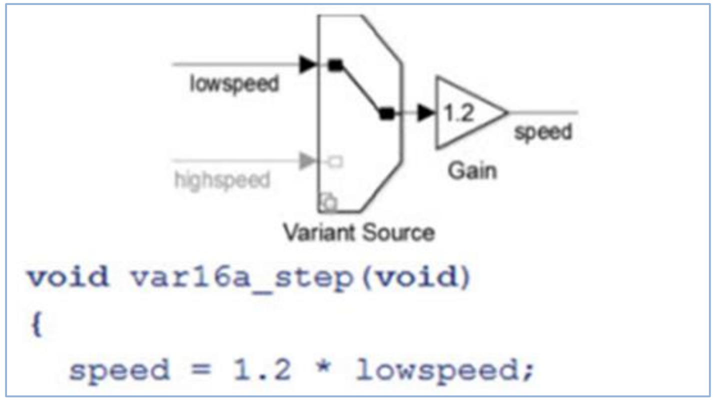
March 2016

R2016a

Design Variants

Generate code for active design variant using Variant Source/ Sink blocks

- Use Variant Source/Sink blocks to design variants instead of placing blocks inside variant subsystems or model variants
- Connections to inactive ports are ignored during simulation

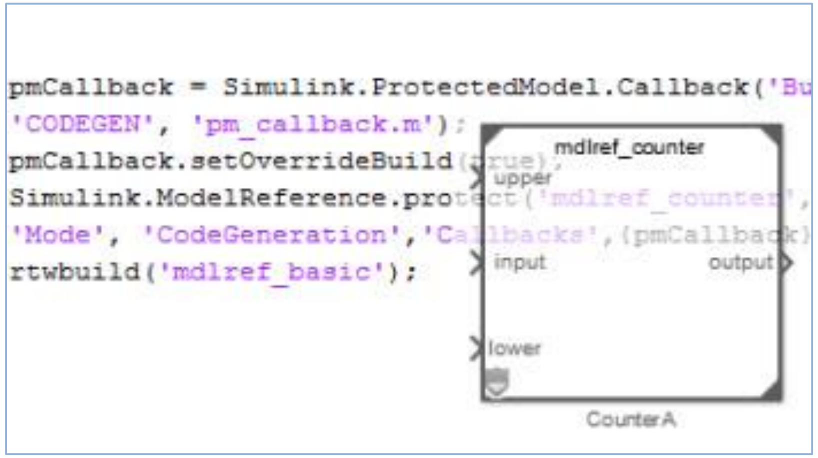


```
void var16a_step(void)
{
  speed = 1.2 * lowspeed;
```

Protected Model Callbacks

Define callbacks for customized protected models

- Specify code to execute when viewing, simulating, or generating code for protected model

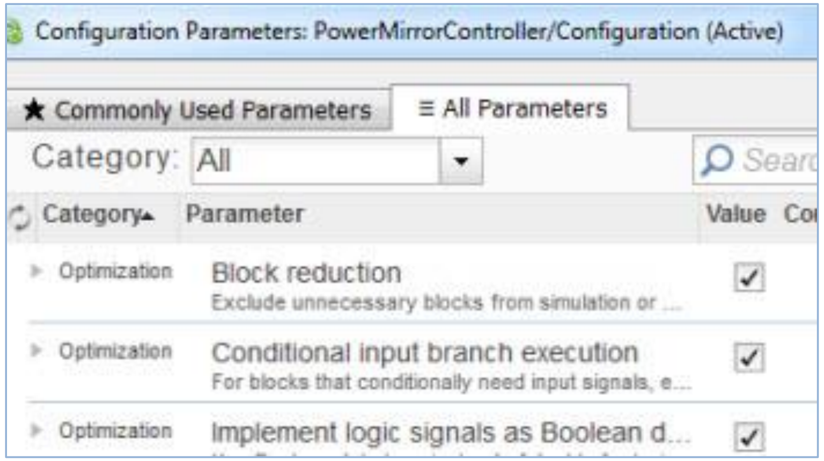


```
pmCallback = Simulink.ProtectedModel.Callback('Bu
'CODEGEN', 'pm_callback.m');
pmCallback.setOverrideBuild(true);
Simulink.ModelReference.protect('mdlref_counter',
'Mode', 'CodeGeneration','Callbacks',{pmCallback}
rtwbuild('mdlref_basic');
```

Simplified Configuration Parameters

Configure models more easily via simplified code generation panes

- Category panes display only parameters most useful for configuring models for code generation



Configuration Parameters: PowerMirrorController/Configuration (Active)

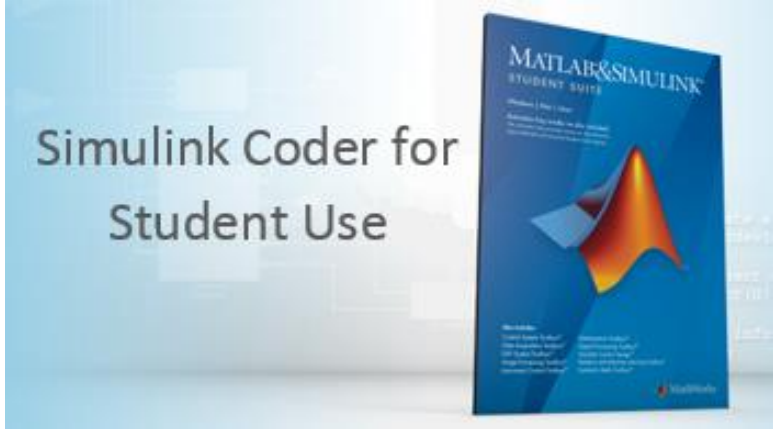
Category: All

Category▲	Parameter	Value	Co
Optimization	Block reduction Exclude unnecessary blocks from simulation or ...	☑	
Optimization	Conditional input branch execution For blocks that conditionally need input signals, e...	☑	
Optimization	Implement logic signals as Boolean d...	☑	

Simulink Coder Student Access

Use Simulink Coder as student-use add-on product

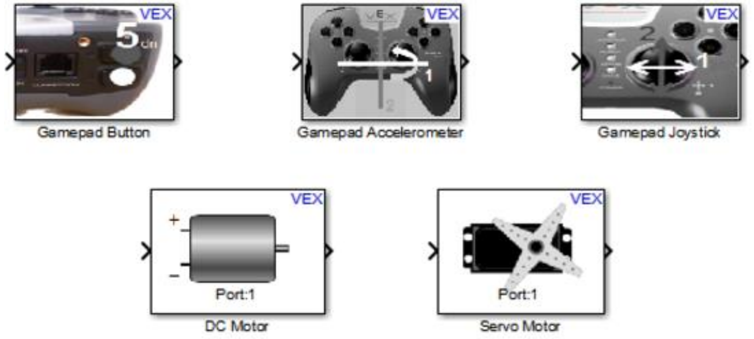
- Get as add-on product with MATLAB Primary and Secondary School Suite
- Student-use software provides the same tools that professional engineers and scientists use



Support for Vex from Simulink Coder

Code generation support for VEX Cortex Microcontroller used in VEX Robotics and BEST Robotics

- Provides blocks for servomotors, DC motors, switches, potentiometers, and VEX gamepad inputs
- Previously required Embedded Coder



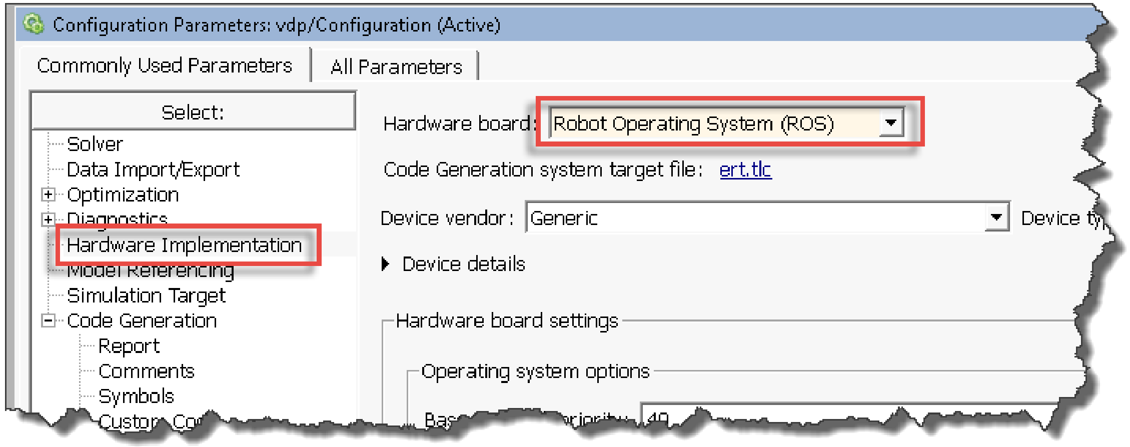
Learn more at www.mathworks.com/hardware-support/vex-arm-cortex.html

R2016a

Generate Standalone ROS Nodes

Code generation for Robot Operating System

- Generate standalone Robot Operating System (ROS) nodes from Simulink models with just MATLAB Coder and Simulink Coder
- Embedded Coder can optionally be used to customize the generated code

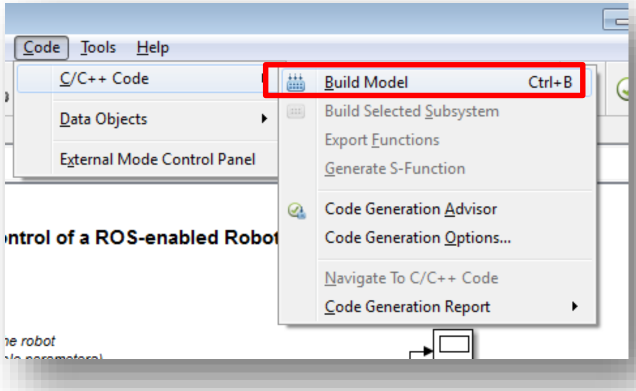


Configuration Parameters: vdp/Configuration (Active)

Hardware board: Robot Operating System (ROS)

Code Generation system target file: ert.tlc

Device vendor: Generic



ntrol of a ROS-enabled Robot

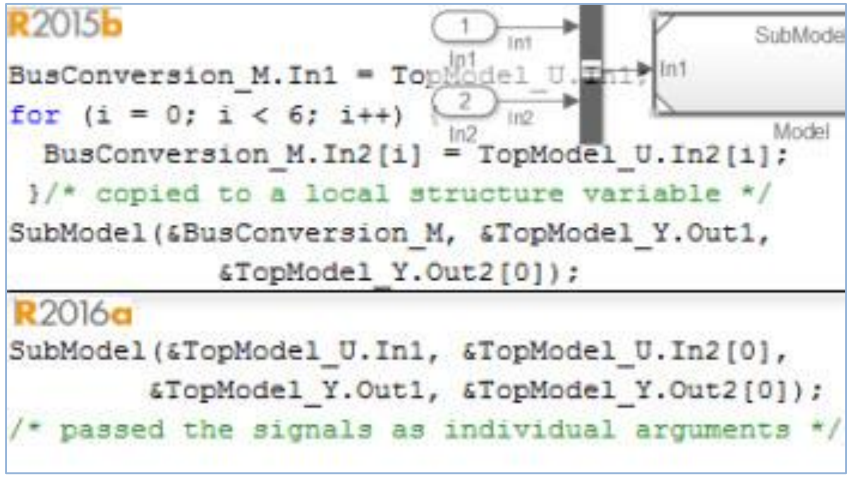
C++ code for standalone ROS node

» robotROSCodeGenerationExample

Model Block Virtual Buses

Use virtual bus to interface with Model blocks

- Exchange signal data through variables or pointers corresponding to bus element, instead of structure
- Improves efficiency by eliminating memory consumption of structure



R2015b

```
BusConversion_M.In1 = TopModel_U.In1;
for (i = 0; i < 6; i++)
  BusConversion_M.In2[i] = TopModel_U.In2[i];
 }/* copied to a local structure variable */
SubModel(&BusConversion_M, &TopModel_Y.Out1,
              &TopModel_Y.Out2[0]);
```

R2016a

```
SubModel(&TopModel_U.In1, &TopModel_U.In2[0],
         &TopModel_Y.Out1, &TopModel_Y.Out2[0]);
/* passed the signals as individual arguments */
```