

Creating C DLL's

sT-Embed Training

Ric Kolk
Altair Engineering
rkolk@altair.com

C DLL's & Software Installation

DLL's can be used for several purposes including:

- Improving the execution speed of a sTE model
- Creating a “Black Box” version of a sTE model
- Software in the Loop (SIL): Evaluating the C Code equivalent of a sTE model for correct execution order, initialization, performance, and other characteristics (before code is generated for a target).

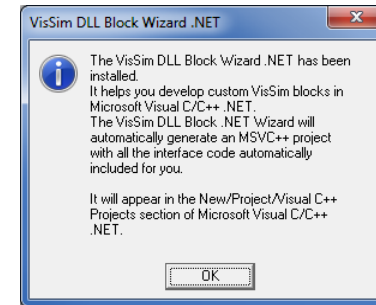
Software Installation:

1. Microsoft Visual Studio Express 2010 (or newer) must be installed on your computer. Use the following link to download and install MSVD 2010 Express

<http://getintopc.com/software/development/visual-studio-express-2010-edition-free-download/>

2. Install or Reinstall solid Thinking EMBED. sTE will detect the MSVC and automatically install the sTE support blocks necessary to create a DLL. You will see the DLL Block Wizard dialog block (right) during the install or reinstall.

3. Install the “sTE/C-Code” Add-On available from the sT-Embed website www.sT-Embed.com under “/Downloads/sT-Embed Software”.



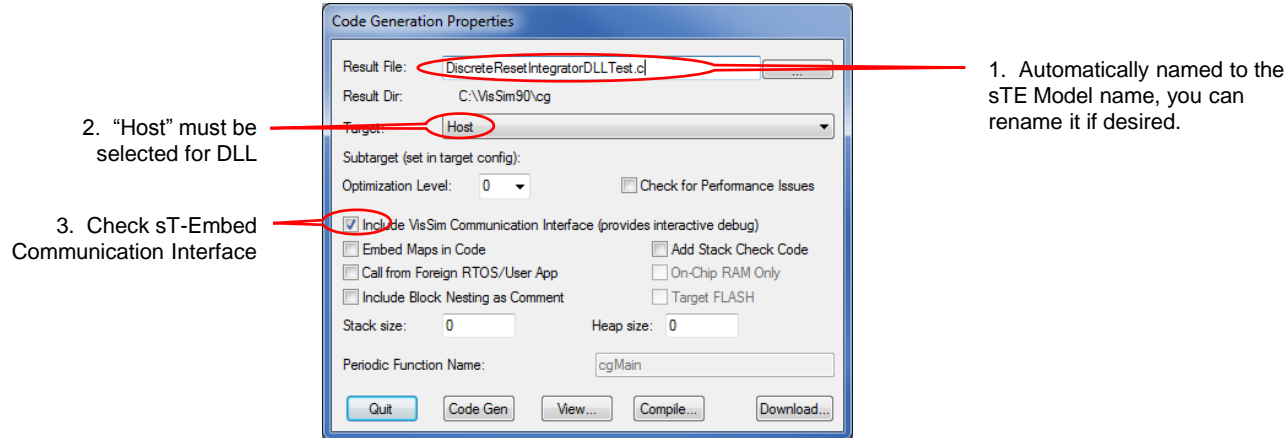
Creating a C DLL from a sT-Embed Compound Block

Step 1: Preparing the compound block for DLL generation:

- A. Using the compound block “Connector Properties”, label all input and output signals.
- B. Add additional inputs to define all internally defined parameters

Step 2: Create the DLL:

- A. Select (Lasso) the compound block you wish to produce a DLL for.
- B. Under the “Tools” menu, select “Code Gen...” and configure the screen as follows;



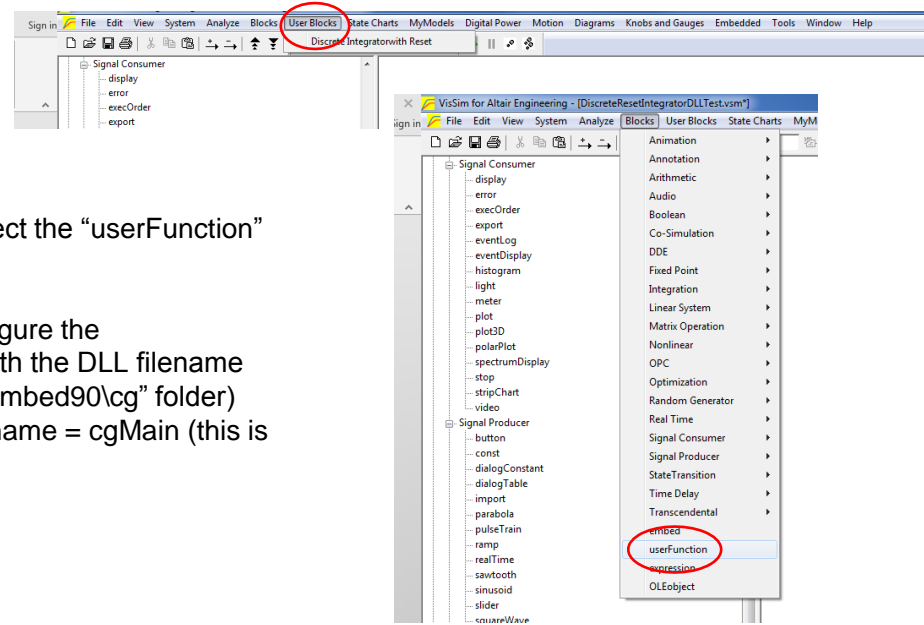
C. Click “Compile...” and acknowledge the “DOS” prompt.

[Creating a C DLL with scalar and vector IO](#)

Creating a C DLL from a sTE Compound Block

Step 3: Accessing the DLL – 2 Methods:

A. In the sTE model, a new “User Blocks” dropdown menu will be added, the DLL you just created will be present in the dropdown menu.

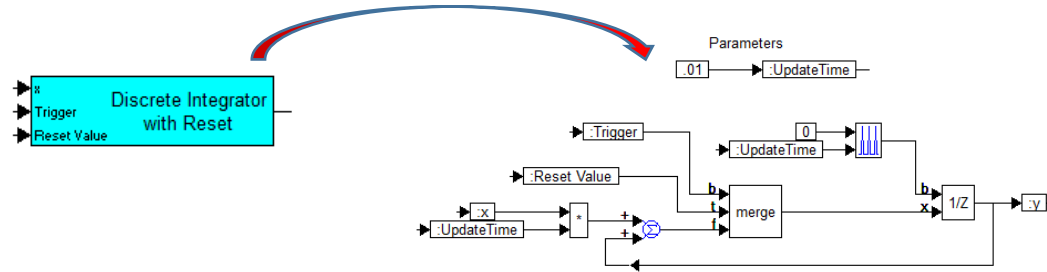


B. Under “Blocks”, select the “userFunction” block.

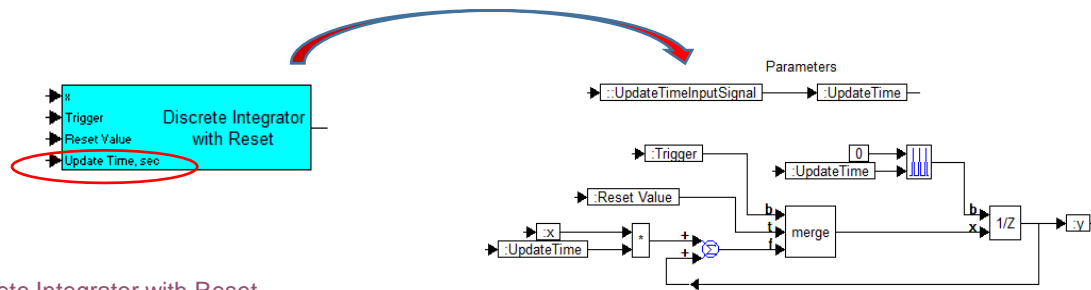
NOTE: You must configure the “userFunction” block with the DLL filename (located in the “C:\sT-Embed90\cg” folder) and the base function name = cgMain (this is a fixed name).

Example

Let's create a DLL for a compound block named “Discrete Integrator with Reset”, shown below:



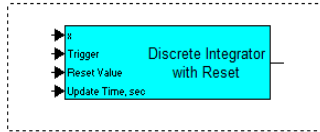
To have access to the “`::UpdateTime`” parameter, it must be defined as an additional input to the compound block, we'll name this input “`::UpdateTimeInputSignal`”;



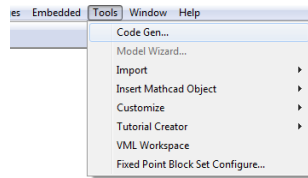
Creating a C DLL for Discrete Integrator with Reset

Example (1/2)

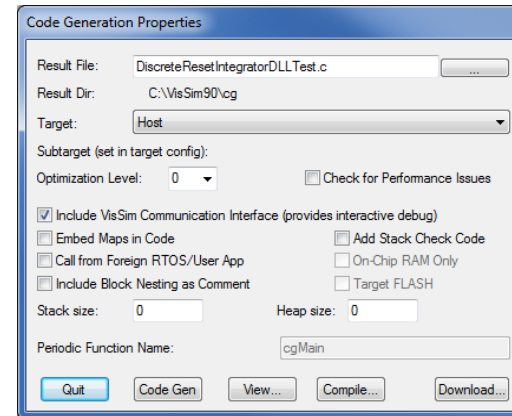
Now we'll create the DLL for the compound block “Discrete Integrator with Reset”;



Step 2A. Lasso the compound block to select it



Step 2B. Click on “Tools”, “Code Gen...”



Step 2B. Configure as shown, “Compile...”

```

VisSim
Setting environment for using Microsoft Visual Studio 2010 x86 tools.
C:\VisSim90\cvg>set USERLIBS=
C:\VisSim90\cvg>set LMOD=vs9
C:\VisSim90\cvg>make -f d1132.nak T-DiscreteResetIntegrator LMOD=vs9
Microsoft (R) Program Maintenance Utility Version 10.00.30319.01
Copyright (C) Microsoft Corporation. All rights reserved.

cl -c -DWIN32 -nologo -Zi -O2 -Gx -IC:\VisSim90\cvg\include -IC:\VisSim90\
cvsdk\include -c DiscreteResetIntegrator.c
link -nologo -incremental -dll -debug DiscreteResetIntegrator.obj C:\
VisSim90\cvg\lib\cgdll32-99.lib C:\VisSim90\cvg\lib\cglib32.lib user32.lib no
n132.lib C:\VisSim90\cvg\lib\cgdll.res C:\VisSim90\cvg\lib\cgstub.lib
Generating library DiscreteResetIntegrator.lib and object DiscreteResetInteg
or.exp
C:\VisSim90\cvg>echo done creation of DiscreteResetIntegrator.dll
done creation of DiscreteResetIntegrator.dll
C:\VisSim90\cvg>pause
Press any key to continue . . .
  
```

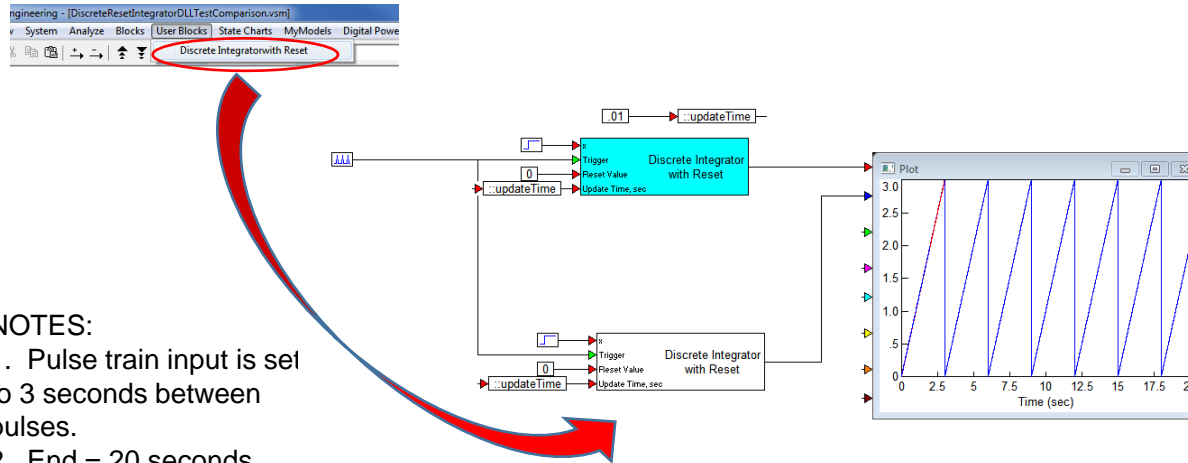
Step 2C.
Acknowledge DOS
Prompt

The DLL is now available either

- Under the menu “UserBlocks” OR
- Under the menu “Blocks”, “userFunction”

Example (2/2)

Now we'll subject the sT-Embed model block and the DLL version to the same input and observe the results.



NOTES:

1. Pulse train input is set to 3 seconds between pulses.
2. End = 20 seconds, Time step = .001 seconds.

End of Section