### Chapter 1. Standard Blockset

**Revision History**

1. Generating FIR Compiler MegaCore Function Creates Errors
2. SignalTap II and HIL Design Flows Fail with Development Boards
3. State Machine Editor Block Fails to Integrate
4. HDL Import Block Fails to Import a RAM Initialization File
5. Reset Signal Inverted if _n Appended to Signal Name
6. DSP Block Naming Issue in Software Version 8.1
7. Incorrect VHDL if Single Simulink Inport Block is Connected to Multiple ChannelIn Ports
8. Bad Port Indexing for Complex Data on ChannelOut and ChannelIn
9. Driving Simulink Scopes With Complex Signals Gives Errors
10. No Support for 1-Input Mode
11. Imaginary Output from Complex to Real Imaginary Gives Internal Errors
12. Hardware Tcl Files Do Not Have Correct Port Names for Vector Ports
13. Multiplier Optimizations Can Cause Signal Width Mismatch in Synthesis
14. HIL Simulation of Combined Blockset Design Outputs All Zeros
15. Latency Not Balanced for ModelBus Blocks Outside Primitive Subsystem
16. Bus and System Clocks Must Match When Using Bus Stimulus Block
17. Primitive Automatic Testbench Flow Incorrect with Many Integer Bits
18. Limit of 50 Subsystem Ports Anywhere in a Design
19. Saturation Limitation When using the Scale Block
20. Very High Hard Multiplier Threshold Does Not Force the Use of Logic
21. No Forward Flow Control in Primitive Subsystems
22. Limitations of the ModelSim Testbench Flow
23. Cannot Import a Combined Blockset Design into SOPC Builder
24. Incorrect Simulation Results if Word Length Greater Than 64 bits
25. Compilation Error if Unused Channel Out Port is Terminated

**Errata**

1. MATLAB Runs Out of Java Virtual Machine Heap Memory
2. VCD Sink Block Does Not Generate HDL Files on Linux
3. Errors Issued With Imported Quartus II Project
4. Cannot Compare Simulink Against ModelSim When Using VIP Suite IP
5. MATLAB Hangs If You Press Enter in Signal Compiler Device Box
6. Additional Project Assignments Required for User Libraries
7. Dual-Clock FIFO Simulation Does Not Match ModelSim

### Chapter 2. Advanced Blockset

**Revision History**

1. MATLAB Supported Versions
2. Simulation Fails
3. DSP Builder Advanced User Guide has No Revision History for v9.1
4. Simulation Error With Cyclone Series Devices
5. DSP Builder Advanced Blockset Reports an Incorrect Version Number
6. Primitive Latency Constraints Do Not Work
7. ATB Generation Does Not Follow Links out of Subsystems
8. Incorrect VHDL if Single Simulink Inport Block is Connected to Multiple ChannelIn Ports
9. Bad Port Indexing for Complex Data on ChannelOut and ChannelIn
10. Driving Simulink Scopes With Complex Signals Gives Errors
11. No Support for 1-Input Mode
12. Imaginary Output from Complex to Real Imaginary Gives Internal Errors
13. Hardware Tcl Files Do Not Have Correct Port Names for Vector Ports
14. Multiplier Optimizations Can Cause Signal Width Mismatch in Synthesis
15. HIL Simulation of Combined Blockset Design Outputs All Zeros
16. Latency Not Balanced for ModelBus Blocks Outside Primitive Subsystem
17. Bus and System Clocks Must Match When Using Bus Stimulus Block
18. Primitive Automatic Testbench Flow Incorrect with Many Integer Bits
19. Limit of 50 Subsystem Ports Anywhere in a Design
20. Saturation Limitation When using the Scale Block
21. Very High Hard Multiplier Threshold Does Not Force the Use of Logic
22. No Forward Flow Control in Primitive Subsystems
23. Limitations of the ModelSim Testbench Flow
24. Cannot Import a Combined Blockset Design into SOPC Builder
25. Incorrect Simulation Results if Word Length Greater Than 64 bits
26. Compilation Error if Unused Channel Out Port is Terminated
1. Standard Blockset

This chapter describes new features, revision history, known errata and documentation changes for the DSP Builder standard blockset.

Revision History

Table 1–1 lists the revision history for the DSP Builder standard blockset v10.0, v9.1, and v9.0.

Table 1–1. DSP Builder Standard Blockset Revision History

<table>
<thead>
<tr>
<th>Version</th>
<th>Date</th>
<th>Revision</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>July 2010</td>
<td>■ Maintenance release.</td>
</tr>
<tr>
<td>9.1</td>
<td>November 2009</td>
<td>■ Maintenance release.</td>
</tr>
<tr>
<td>9.0 SP1</td>
<td>May 2009</td>
<td>■ Maintenance release.</td>
</tr>
<tr>
<td>9.0</td>
<td>March 2009</td>
<td>■ Support for Linux operating systems.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ Support for Arria® II GX.</td>
</tr>
<tr>
<td>8.1</td>
<td>November 2008</td>
<td>■ Removed obsolete Simulation Accelerator block and deprecated support for the Video and Image Processing Suite.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ Updated Hardware in the loop (HIL) block user interface.</td>
</tr>
</tbody>
</table>

Errata

Errata are functional defects or errors, which may cause DSP Builder to deviate from published specifications. Table 1–2 shows the errata that affect the DSP Builder standard blockset v10.0, v9.1, and v9.0.

For the latest errata information, refer to the DSP Builder Release Notes and Errata on the Altera® literature website.

Table 1–2. DSP Builder Standard Blockset Errata (Part 1 of 2)

<table>
<thead>
<tr>
<th>Added or Updated</th>
<th>Issue</th>
<th>Affected Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>15 Jul 10</td>
<td>Generating FIR Compiler MegaCore Function Creates Errors</td>
<td>10.0 9.1 9.0 SP1 9.0</td>
</tr>
<tr>
<td></td>
<td>SignalTap II and HIL Design Flows Fail with Development Boards</td>
<td></td>
</tr>
<tr>
<td></td>
<td>State Machine Editor Block Fails to Integrate</td>
<td></td>
</tr>
<tr>
<td></td>
<td>HDL Import Block Fails to Import a RAM Initialization File</td>
<td></td>
</tr>
<tr>
<td>15 Nov 09</td>
<td>Reset Signal Inverted if _n Appended to Signal Name</td>
<td>Fixed</td>
</tr>
<tr>
<td>15 Jun 09</td>
<td>DSP Block Naming Issue in Software Version 8.1</td>
<td></td>
</tr>
<tr>
<td>15 May 09</td>
<td>Incorrect Pin Assignments Test2S60Board.mdl Design Example</td>
<td>Fixed</td>
</tr>
<tr>
<td>01 Apr 09</td>
<td>MATLAB Runs Out of Java Virtual Machine Heap Memory</td>
<td></td>
</tr>
</tbody>
</table>
Table 1–2. DSP Builder Standard Blockset Errata (Part 2 of 2)

<table>
<thead>
<tr>
<th>Added or Updated</th>
<th>Issue</th>
<th>Affected Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>15 Mar 09</td>
<td>VCD Sink Block Does Not Generate HDL Files on Linux</td>
<td>✓</td>
</tr>
<tr>
<td></td>
<td>Errors Issued With Imported Quartus II Project</td>
<td>✓</td>
</tr>
<tr>
<td></td>
<td>Cannot Compare Simulink Against ModelSim When Using VIP Suite IP</td>
<td>✓</td>
</tr>
<tr>
<td>01 Nov 08</td>
<td>MATLAB Hangs If You Press Enter in Signal Compiler Device Box</td>
<td>✓</td>
</tr>
<tr>
<td>15 May 08</td>
<td>Additional Project Assignments Required for User Libraries</td>
<td>✓</td>
</tr>
<tr>
<td>01 May 07</td>
<td>Dual-Clock FIFO Simulation Does Not Match ModelSim</td>
<td>✓</td>
</tr>
</tbody>
</table>

Generating FIR Compiler MegaCore Function Creates Errors

You may receive errors when you generate instances of the FIR Compiler MegaCore function.

Affected Configurations

DSP Builder designs that contain FIR Compiler MegaCore functions on the Windows OS.

Design Impact

You cannot generates instances of FIR Compiler MegaCore functions.

Workaround

Ensure that you use DSP Builder within an environment where the PATH variable includes the location of the Quartus II executable.

To prepare such an environment, in a DOS prompt, type the following command:

```
set PATH=%PATH%;c:\altera\10.0\quartus\bin;
```

Solution Status

This issue will be fixed in a future release of DSP Builder.

SignalTap II and HIL Design Flows Fail with Development Boards

The SignalTap® II and HIL design flows may fail when your design attempts to communicate with development boards.

Affected Configurations

Designs that use the SignalTap II or HIL design flows on the Windows OS.

Design Impact

The SignalTap II and HIL design flows cannot program devices successfully.

Workaround

Ensure that you use DSP Builder within an environment where the PATH variable includes the location of the Quartus II executable.
To prepare such an environment, in a DOS prompt, type the following command:

```
set PATH=%PATH%;c:\altera\10.0\quartus\bin;
```

**Solution Status**

This issue will be fixed in a future release of DSP Builder.

**State Machine Editor Block Fails to Integrate**

If you create a state machine design with the State Machine Editor you do not get do correctly parameterized blocks in DSP Builder.

**Affected Configurations**

This issue affects designs that contain State Machine Editor blocks, which do not have a state machine defined.

**Design Impact**

State Machine Configurations cannot be created from the unparameterised (default) State Machine Editor block.

**Workaround**

To workaround this issue, follow these steps:

1. In the directory that contains the Simulink model, `<modelname>.mdl`, which incorporates a State Machine Editor block, `<statemachine_name>`, execute the following Matlab commands in the Matlab Command Window:

   ```matlab
   mkdir('DSPBuilder_<modelname>_import');
   csvwrite('DSPBuilder_<modelname>_import<statemachine_name>.smf', ');
   ```

   Ensure that the name of the State Machine Editor block is a valid name in VHDL.

2. Double-click on the State Machine Editor block, use the GUI to configure the State Machine, and save the state machine.

3. Edit the `<statemachine_name>.smf` file, which is in the `DSPBuilder_<modelname>_import` folder, to replace the "VLOG" in the GENERAL section of the file with "VHDL".

4) Close the State Machine Editor GUI.

5) Use the State Machine Editor block as normal.

**Solution Status**

This issue will be fixed in a future release of DSP Builder.

**HDL Import Block Fails to Import a RAM Initialization File**

The `HDLImport` block fails to import RAM initialization (.hex) files that lack a line feed at the end of the file.
Affected Configurations
This issue affects designs that contain an HDLImport block that sources a .hex file that does not end with a line feed.

Design Impact
Memory elements in the HDL import system may produce uninitialized values during Simulink simulation.

Workaround
Add a line feed to the end of the .hex file.

Solution Status
This issue will be fixed in a future release of DSP Builder.

Reset Signal Inverted if _n Appended to Signal Name
The reset signal inverts if you append _n to the signal name specified in the Clock or Clock_Derived block.

Affected Configurations
This issue affects designs that include a Clock or Clock_Derived block.

Design Impact
The inverted reset signal is propagated through the design.

Workaround
Do not append _n to the reset signal specified in a Clock or Clock_Derived block unless you explicitly want the signal to invert. You can append _N (uppercase) without inverting the reset signal.

Solution Status
This issue is fixed in DSP Builder v9.1.

DSP Block Naming Issue in Software Version 8.1
The Stratix III DSP block was renamed to DSP block in the v8.0 software because it also supports Stratix® IV devices. A forwarding table in the library model ensures that any blocks named Stratix III DSP block upgraded from the previous release are recognized as a DSP block in the v8.0 software.

The name of the block incorrectly reverts to Stratix III DSP in the v8.1 software.

Affected Configurations
This issue affects designs upgraded from v8.0 that use the DSP block.

Design Impact
A DSP block in a design upgraded from v8.0 is not recognized in the v8.1 software.
Workaround

If you upgrade to v9.0, any blocks named DSP or Stratix III DSP are recognized as DSP blocks.

Alternatively, you can edit the `<install directory>\quartus\dsp_builder\bin\mdllibrary\allblocks_alteradspbuilder2.mdl` file in the v8.1 software and add the following reverse forwarding table string:

```
ForwardingTableString "||__slOldName__||allblocks_alteradspbuilder2/DSP Block||__slNewName__||allblocks_alteradspbuilder2/Stratix III DSP Block"
```

Solution Status

This issue is fixed in DSP Builder v9.0.

Incorrect Pin Assignments Test2S60Board.mdl Design Example

The pin assignments for the PLL output clock resets are incorrect in the design example Test2S60Board.mdl.

Affected Configurations

This issue affects design example Test2S60Board.mdl.

Design Impact

DSP Builder issues errors reporting invalid pin assignments when you run the Quartus® II Fitter from the Signal Compiler block.

Workaround

To correct the pin assignments, follow these steps:

1. In MATLAB, navigate to the directory `<install directory>\quartus\dsp_builder\DesignExamples\Demos\Board\StratixII_DSPBoard_2S60` and open the model Test2S60Board.mdl.
2. Double-click on the Quartus II assignments block to open the subsystem window.
3. Double-click on the Quartus II Pinout Assignments1 block to open the Block Parameters dialog box for PLL clock reset out. Change the pin location assignments to Pin_AD32, Pin_AB30, Pin_M30, Pin_N29. Click OK to apply and save the changes.
4. Save the model.
5. Use the normal procedures to compile and download the design to the board, and run SignalTap® II Analysis on the design.

Solution Status

This issue is fixed in DSP Builder v9.0 SP1.

MATLAB Runs Out of Java Virtual Machine Heap Memory

For a very large design, MATLAB may have insufficient heap memory available for the Java virtual machine.
Affected Configurations
This issue affects very large designs containing many thousand blocks.

Design Impact
Compilation fails and MATLAB issues an error message of the form:

"OutofMemoryError: Java heap space"

Workaround
Increase the heap space available to the Java virtual machine. For more information, refer to:

http://www.mathworks.com/support/solutions/data/1-18I2C.html

Solution Status
This issue will be documented in a future release of DSP Builder.

VCD Sink Block Does Not Generate HDL Files on Linux
The VCD Sink block does not generate HDL files on Linux systems.

Affected Configurations
This issue affects Linux configurations.

Design Impact
No HDL is generated for the VCD Sink block.

Workaround
None.

Solution Status
This issue will be fixed in a future release of DSP Builder.

Errors Issued With Imported Quartus II Project
HDL Import blocks that import Quartus II projects may cause errors while executing the signal compiler or testbench flows.

Affected Configurations
Models with HDL Import blocks that reference Quartus II projects with SEARCH_PATH assignments, rather than USER_LIBRARIES assignments.

Design Impact
Designs with models containing HDL Import blocks that import Quartus II projects generated with the Quartus II software v8.0 may fail to generate hardware because the library settings are unrecognized.
Workaround
Add a **USER_LIBRARIES** setting to the `.qsf` file for the Quartus II project to be imported. The assignment value should be the concatenation of all the project's **SEARCH_PATH** assignment values, separated by semicolons and enclosed within quotation marks. These required search path values may be in the top-level `.qsf` file, or in `.qip` files included in the project. Then recompile the HDL Import block.

Solution Status
This issue is fixed in v9.0 of DSP Builder.

Cannot Compare Simulink Against ModelSim When Using VIP Suite IP
Execution of the **Compare against HDL** or **Generate HDL** steps in the Testbench Generator dialog box issues an error for Video and Image Processing Suite MegaCore® functions.

Affected Configurations
Designs using Video and Image Processing Suite MegaCore function blocks.

Design Impact
Testbench generation does not complete successfully, and automatic comparison of the Simulink simulation results against the results from the ModelSim® simulator is not possible.

Workaround
Altera provides a MATLAB m-script that give access to the testbench flow, and that send the the flow output to the MATLAB command window. You require the following command to use this script is:

```matlab
alt_dspbuilder_compare_VIP_design_against_HDL(<model>, <performcompare>)
```

where:
- `<model>` is the name of the `.mdl` file for which the flow is to be run.
- `<performcompare>` is optional, setting it to 0 causes the final compare results step to be skipped. The default value of 1 causes the compare results step to be run.

After the script is run, you can use the features available in the **Advanced** tab of the Testbench Generator dialog box (excluding the **Generate HDL** step).

Solution Status
DSP Builder v9.0 or later does not support the Video and Image Processing Suite MegaCore functions.
MATLAB Hangs If You Press Enter in Signal Compiler Device Box

MATLAB hangs if you press the Enter or Return key in the Device box on the Signal Compiler dialog box.

Affected Configurations
This issue affects all configurations.

Design Impact
None.

Workaround
Avoid pressing the Enter or Return key after typing the device name.

Solution Status
This issue will be fixed in a future release of DSP Builder.

Additional Project Assignments Required for User Libraries

Additional project assignments may be required when using the Export HDL flow with user libraries.

Affected Configurations
Configurations that include MegaCore function, HDL Import, or State Machine Editor blocks.

Design Impact
Compilation fails in the Quartus II software with an error of the form:

```
Error: Node instance <block_name> instantiates undefined entity <entity_name>
```

Workaround
Add the following to the Quartus II project:

- The .qip file corresponding to the entity named in the error message. (This file is located in the import subdirectory corresponding to the library model.)
- In some cases you may also need to add any libraries which are referenced by HDL Import block(s) in the library model.

Solution Status
This issue will be fixed in a future release of DSP Builder.

Dual-Clock FIFO Simulation Does Not Match ModelSim

The Dual-Clock FIFO simulation in Simulink is functionally equivalent to hardware, but not cycle-accurate.

Affected Configurations
This issue affects most configurations.
Design Impact
The delay between the write-side adding data and the read-side seeing it, and between the read-side clearing space in the FIFO and the write-side seeing it, does not match hardware.

Workaround
Do not rely on these timing characteristics for a correct design.

Solution Status
This issue will be fixed in a future release of DSP Builder.
Chapter 1: Standard Blockset

Errata
This chapter describes new features, revision history, known errata and documentation changes for the DSP Builder advanced blockset.

Revision History

Table 2–1 lists the revision history for the DSP Builder advanced blockset v10.0, v9.1, and v9.0.

Table 2–1. DSP Builder Advanced Blockset Revision History

<table>
<thead>
<tr>
<th>Version</th>
<th>Date</th>
<th>Revision</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>July 2010</td>
<td>■ Added new blocks</td>
</tr>
<tr>
<td>9.1</td>
<td>November 2009</td>
<td>■ Primitive block support for:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>▪ Folding and time-division multiplexing</td>
</tr>
<tr>
<td></td>
<td></td>
<td>▪ Simulink Complex type support in arithmetic and memory primitive blocks</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ Simulink 1-D vector (array) support in arithmetic and memory primitive blocks</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ Improved resource utilization</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ Run ModelSim system-level testbench now verifies ModelSim simulation against Simulink, and there are no longer restrictions on the stimulus blocks supported</td>
</tr>
<tr>
<td>9.0 SP1</td>
<td>May 2009</td>
<td>Maintenance release.</td>
</tr>
<tr>
<td>9.0</td>
<td>March 2009</td>
<td>Support for Linux operating systems. Support for Arria® II GX.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Updated SOPC Builder support and procedures for instantiating a design.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Support for running all the automatic testbenches in a design.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Improved documentation for the NCO block including multichannel design examples with support for multiple banks.</td>
</tr>
</tbody>
</table>

Errata

Errata are functional defects or errors, which may cause DSP Builder to deviate from published specifications. Table 2–2 shows the errata that affect the DSP Builder advanced blockset v10.0, v9.1, and v9.0.

For the latest errata information, refer to the DSP Builder Release Notes and Errata on the Altera literature website.
## Table 2–2. DSP Builder Advanced Blockset Errata

<table>
<thead>
<tr>
<th>Added or Updated</th>
<th>Issue</th>
<th>Affected Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>15 Jul 10</td>
<td>MATLAB Supported Versions</td>
<td>10.0</td>
</tr>
<tr>
<td></td>
<td>Simulation Fails</td>
<td>✔</td>
</tr>
<tr>
<td></td>
<td>DSP Builder Advanced User Guide has No Revision History for v9.1</td>
<td>✔</td>
</tr>
<tr>
<td>15 Nov 09</td>
<td>Simulation Error With Cyclone Series Devices</td>
<td>✔</td>
</tr>
<tr>
<td></td>
<td>DSP Builder Advanced Blockset Reports an Incorrect Version Number</td>
<td>✔</td>
</tr>
<tr>
<td></td>
<td>Primitive Latency Constraints Do Not Work</td>
<td>✔</td>
</tr>
<tr>
<td></td>
<td>ATB Generation Does Not Follow Links out of Subsystems</td>
<td>✔</td>
</tr>
<tr>
<td></td>
<td>Incorrect VHDL if Single Simulink Import Block is Connected to Multiple ChannelIn Ports</td>
<td>✔</td>
</tr>
<tr>
<td></td>
<td>Bad Port Indexing for Complex Data on ChannelOut and ChannelIn</td>
<td>✔</td>
</tr>
<tr>
<td></td>
<td>Driving Simulink Scopes With Complex Signals Gives Errors</td>
<td>✔</td>
</tr>
<tr>
<td></td>
<td>No Support for 1-Input Mode</td>
<td>✔</td>
</tr>
<tr>
<td></td>
<td>Imaginary Output from Complex to Real Imaginary Gives Internal Errors</td>
<td>✔</td>
</tr>
<tr>
<td></td>
<td>Hardware Tcl Files Do Not Have Correct Port Names for Vector Ports</td>
<td>—</td>
</tr>
<tr>
<td>15 May 09</td>
<td>Multiplier Optimizations Can Cause Signal Width Mismatch in Synthesis</td>
<td>✔</td>
</tr>
<tr>
<td></td>
<td>HIL Simulation of Combined Blockset Design Outputs All Zeros</td>
<td>—</td>
</tr>
<tr>
<td>01 Apr 09</td>
<td>Latency Not Balanced for ModelBus Blocks Outside Primitive Subsystem</td>
<td>✔</td>
</tr>
<tr>
<td>15 Mar 09</td>
<td>Bus and System Clocks Must Match When Using Bus Stimulus Block</td>
<td>✔</td>
</tr>
<tr>
<td></td>
<td>Primitive Automatic Testbench Flow Incorrect with Many Integer Bits</td>
<td>—</td>
</tr>
<tr>
<td></td>
<td>Limit of 50 Subsystem Ports Anywhere in a Design</td>
<td>—</td>
</tr>
<tr>
<td>01 Nov 08</td>
<td>Saturation Limitation When using the Scale Block</td>
<td>✔</td>
</tr>
<tr>
<td></td>
<td>Very High Hard Multiplier Threshold Does Not Force the Use of Logic</td>
<td>✔</td>
</tr>
<tr>
<td></td>
<td>No Forward Flow Control in Primitive Subsystems</td>
<td>✔</td>
</tr>
<tr>
<td></td>
<td>Limitations of the ModelSim Testbench Flow</td>
<td>✔</td>
</tr>
<tr>
<td>15 May 08</td>
<td>Cannot Import a Combined Blockset Design into SOPC Builder</td>
<td>✔</td>
</tr>
<tr>
<td></td>
<td>Incorrect Simulation Results if Word Length Greater Than 64 bits</td>
<td>✔</td>
</tr>
<tr>
<td></td>
<td>Compilation Error if Unused Channel Out Port is Terminated</td>
<td>✔</td>
</tr>
</tbody>
</table>

### MATLAB Supported Versions

DSP Builder v10.0 does not support MATLAB version 2010.

### Affected Configurations

This issue affects all configurations.

### Design Impact

You cannot implement your design.
Workaround
To workaround this issue use a supported version of MATLAB (2009 and earlier).

Solution Status
This issue will be fixed in a future release of the DSP Builder advanced blockset.

Simulation Fails
Simulation may fail to simulate because a real signal, which DSP Builder identifies as complex, drives a block input that does not support complex signals.

Affected Configurations
This issue affects some designs that use complex signals.

Design Impact
Designs that use complex signals may unexpectedly fail to simulate, and you may not be able to generate hardware.

Workaround
Where propagation incorrectly identifies a signal as complex, insert a configuration consisting of the following blocks into the design:
- primitive Const
- Simulink Real-Imag to Complex
- Simulink Complex to Real-Imag
- Simulink terminator

Figure 2–1 shows how to insert the blocks into your design.

Figure 2–1. Simulation Failure Workaround

Solution Status
This issue will be fixed in a future release of the DSP Builder advanced blockset.

DSP Builder Advanced User Guide has No Revision History for v9.1
The DSP Builder Advanced User Guide has no revision history for v9.1. The following revision history is correct:
- Added appendix on folding
- Added descriptions on complex datatype and vectorization
Solution Status
This issue is fixed in the DSP Builder advanced blockset v10.0.

Simulation Error With Cyclone Series Devices

Primitive Const blocks that generate a signal driving a Mult-Add block sequence may cause a simulation error when targeting Cyclone® series devices.

When you target any of the Cyclone devices, simulation may fail in designs that contain a signal from a primitive Const block that drives a Mult block that drives an Add block.

Affected Configurations
This issue affects configurations that target any Cyclone series devices and contain a primitive Const driving a Mult-Add block sequence.

Design Impact
The following assertion failure occurs, which causes MATLAB to crash.

`outputClock != m_model->m_infiniteClock.`

Workaround
Connect the constant signal to the Mult block using a Convert block.

Solution Status
This issue will be fixed in a future release of the DSP Builder advanced blockset.

DSP Builder Advanced Blockset Reports an Incorrect Version Number

DSP Builder advanced blockset reports an incorrect version number.

Affected Configurations
This issue affects all configurations.

Design Impact
The version number reported by DSP Builder advanced blockset within MATLAB (for example, through the `ver` MATLAB command) is 0.0.0. This incorrect version number also appears in the header comment of generated HDL files.

Workaround
Ignore the incorrect version numbers.

Solution Status
This issue will be fixed in a future release of the DSP Builder advanced blockset.

Primitive Latency Constraints Do Not Work

In primitive subsystems, an option on the SynthesisInfo block should allow you to set a latency constraint for the subsystem. However, this feature does not work.
Chapter 2: Advanced Blockset

Errata

© July 2010 Altera Corporation
Software Version 10.0 DSP Builder Release Notes and Errata

**Affected Configurations**
This issue affects all configurations.

**Design Impact**
There is no design impact.

**Workaround**
This issue has no workaround.

**Solution Status**
This issue will be fixed in a future release of the DSP Builder advanced blockset.

**ATB Generation Does Not Follow Links out of Subsystems**
Synthesizable subsystems where Simulink Inports directly connect to Simulink Outports may lead to ATB VHDL errors.

**Affected Configurations**
This issue affects synthesizable subsystems where a Simulink Inport block feeds directly into a Simulink Outport block.

**Design Impact**
When you use the generated ATB (for example by double-clicking Run ModelSim), you receive the following error, because the ATB is trying to read simulation data from a file that does not exist:

```
# ** Error: (vsim-7) Failed to open VHDL file 
"../MATLAB/../rtl/device_atb_problem/DUT/Wire/Out1_auto.stm" in rb
```

**Workaround**
To workaround the issue, replace direct connections from a Simulink Inport block to a Simulink Outport block with direct connections outside the corresponding subsystem.

**Solution Status**
This issue will be fixed in a future release of the DSP Builder advanced blockset.

**Incorrect VHDL if Single Simulink Inport Block is Connected to Multiple ChannelIn Ports**
Connecting single Simulink Inport block to multiple ChannelIn ports may lead to VHDL error in synthesis.

**Affected Configurations**
This issue affects primitive subsystems where a Simulink Inport block feeds into multiple ChannelIn ports
Design Impact
When you compile for synthesis in the Quartus II software, you receive the following error, because the same port name appears multiple times in the port list:

Error (10465): VHDL error at 
one_inport_to_many_channelin_ports_Chip.vhd(40): name "d0" cannot be 
used because it is already used for a previously declared item

Workaround
Add a Simulink Inport block for each ChannelIn input in your subsystem, and feed the desired signals to those ports outside the subsystem.

Solution Status
This issue will be fixed in a future release of the DSP Builder advanced blockset.

Bad Port Indexing for Complex Data on ChannelOut and ChannelIn

You should always connect the channel (DC) input port of the ChannelOut ModelPrim block, if your design uses complex data types. If your design does not have the DC input port connected, as it is not required, you should drive it with a constant zero uint(8).

Similarly, you should always connect the channel (DC) input port of the ChannelIn ModelPrim block, if your design uses complex data types. If your design does not have the DC input port connected, as it is not required, you should connect it up through the subsystem levels and drive it with a constant zero uint(8) in the testbench.

Affected Configurations
This issue affects all configurations.

Design Impact
Without these imaginary components any complex data signals through the channel out may not be correctly connected, and the imaginary components are lost. In addition for the ChannelIn ModelPrim block, if completely unconnected a port ChannelIn_2 is created on the entity, but not on the components.

Workaround
This issue has no workaround.

Solution Status
This issue will be fixed in a future release of the DSP Builder advanced blockset.

Driving Simulink Scopes With Complex Signals Gives Errors
You cannot drive Simulink scopes with complex signals. They only accept real signals (including real vectors). If you attempt to drive a scope with a complex signal in your design, Simulink assumes this signal should be real and back-propagates the real type through the design, until such time as there is a clash, where a block is explicitly driven by a complex type.
Affected Configurations
This issue affects all configurations.

Design Impact
This clash results in the following error message:

"Attempted to set output port (n) complexity after it was already set to opposite sense."

If you see this error for a complex port, check that the signal downstream is not driving a scope directly.

Workaround
Add a Simulink Complex to Real-Imag block (Simulink > Math Operations library) and connect the real and imaginary components to the scope individually.

Blocks other than Simulink scopes may also cause similar errors, if they only accept real data types. The same workaround applies.

Solution Status
This issue will be fixed in a future release of the DSP Builder advanced blockset.

No Support for 1-Input Mode
There is no support for 1-input mode on AND, OR, or other logical ModelPrim blocks.

Affected Configurations
This issue affects all configurations.

Design Impact
You see the following error:


Workaround
This issue has no workaround.

Solution Status
This issue will be fixed in a future release of the DSP Builder advanced blockset.

Imaginary Output from Complex to Real Imaginary Gives Internal Errors
You cannot use the imaginary output from Complex to Real-Imag block, if the input signal is real, (despite this imaginary output signal appearing in Simulink). Attempts to use the imaginary output leads to internal errors. In synthesizable subsystems, you receive an error message; in subsystems not for synthesis, you receive a warning message.
Affected Configurations
This issue affects all configurations.

Design Impact
If you use the imaginary output, DSP Builder issues internal errors.

Workaround
Ensure the source signal is complex or remove the Complex to Real-Imag block and generate a constant zero to drive the elements connected to the imaginary signal.

Solution Status
This issue will be fixed in a future release of the DSP Builder advanced blockset.

Hardware Tcl Files Do Not Have Correct Port Names for Vector Ports
If any input ports to the synthesizable system have vector inputs (for example if they feed into a FIR filter with more than one channel and a sample rate matching the clock rate), the `<hardware>_hw.tcl` file generated for SOPC Builder has misnamed ports.

Affected Configurations
Any synthesizable system that has vector input ports.

Design Impact
The port names in the `<hardware>_hw.tcl` file are not vectorized. For example, it might only list an input `a` and an output `q`, although the VHDL actually contains input port names `a0, a1,...` and output port names `q0, q1,...`

Workaround
Manually correct the Tcl file using the generated VHDL as a reference. For example:

```bash
add_interface_port exp a export Input 17
```
becomes:

```bash
add_interface_port exp a0 export Input 17
add_interface_port exp a1 export Input 17
add_interface_port exp a2 export Input 17
...  
...  
```

Solution Status
This issue is fixed in the DSP Builder advanced blockset v9.1.

Multiplier Optimizations Can Cause Signal Width Mismatch in Synthesis
Multiplier optimizations can lead to signal bit width mismatches and VHDL errors in synthesis.
Affected Configurations
Primitive subsystems where a multiplier feeds directly into an output port.

Design Impact
When you compile for synthesis in the Quartus II software, an error is issued of the form:

```
Error: Actual width (34) of port "q1" on instance "<model>:<submod>"
      is not compatible with the formal port width (30) declared by the
      instantiated entity
```

This is because the output signal is smaller than expected because of a multiplier optimization not being propagated on to a fixed width entity (usually an subsystem boundary or other fixed-width block).

Workaround
Use a Convert block from the ModelPrim library to explicitly set the required signal format on the output of the multiplier.

Solution Status
This issue will be fixed in a future release of the DSP Builder advanced blockset.

HIL Simulation of Combined Blockset Design Outputs All Zeros
A hardware in the loop (HIL) design containing blocks from both the standard and advanced blocksets sometimes outputs all zeros when simulated. This problem occurs when the reset type is set to Low in the Signals block.

The DSP Builder Handbook states an embedded advanced blockset design should have the same reset type as the standard blockset design in which it is embedded, which is incorrect. The advanced blockset design must always have an active-high reset.

Affected Configurations
HIL designs derived from a model created with the advanced blockset.

Design Impact
The HIL simulation output remains zero.

Workaround
Change the reset type specified in the Signals block to High, by following these steps:
1. In the original combined blockset design, set the Reset name to aclr and Reset active to High in the Signals block.
2. Delete all the generated files from the rtl directory specified as the Hardware destination directory in the Control block.
3. Delete all the files from the DSPBuilder_<top level>_import directory.
4. Delete all the files from <top level>_dspbuilder directory.
5. Run Simulink simulation (to generate HDL for the advanced blockset).
6. Compile the design in the Signal Compiler block (generate HDL and create a Quartus II project for the whole design).
7. Open the HIL model and run the normal HIL flow.

   The reset level specified in the Clock block is not relevant.

Solution Status
This issue is fixed in the latest version of the DSP Builder Handbook

Latency Not Balanced for ModelBus Blocks Outside Primitive Subsystem
If you are using ModelBus blocks outside of a primitive subsystem with Scheduled synthesis style, it is likely that the latencies on the address, write enable and data paths created for these blocks may not be balanced in v9.0.

Affected Configurations
This issue affects configurations using ModelBus blocks outside of a primitive subsystem with Scheduled synthesis style. This issue does not affect ModelBus blocks inside a scheduled subsystem.

Design Impact
The latencies on the address, write enable and data paths created for ModelBus blocks outside of a primitive subsystem may not be balanced.

Workaround
To ensure the data path is registered in line with the address and write enable for such unscheduled blocks, define the following workspace variable before performing simulation and HDL generation (for example in the model InitFnc callbacks):

   DSPBA_Features.useScheduledDataBus = false;

The default v9.0 behavior can be restored by setting this variable to true.

Solution Status
This issue will be fixed in a future release of the DSP Builder advanced blockset.

Bus and System Clocks Must Match When Using Bus Stimulus Block
Testbench simulation results are incorrect when using the Bus Stimulus block if the bus clock frequency is different from the frequency of the system clock.

Affected Configurations
Any design that uses the Bus Stimulus block.

Design Impact
The testbench simulation results are incorrect.
Workaround
The bus clock frequency must be the same as the frequency of the system clock. Turn on Separate Bus Clock and Bus Clock Synchronous with System Clock in the Signal block for your model.

Solution Status
This issue will be fixed in a future release of the DSP Builder advanced blockset.

Primitive Automatic Testbench Flow Incorrect with Many Integer Bits
The primitive automatic testbench flow may be incorrect when using fractional representations with many integer bits.

Affected Configurations
This issue affects designs that require a large number of integer bits.

Design Impact
DSP Builder reports unexpected simulation mismatches.

Workaround
Reduce the number of integer bits required by your model.

Solution Status
This issue is fixed in the DSP Builder advanced blockset v9.0.

Limit of 50 Subsystem Ports Anywhere in a Design
All blocks that become functional units are restricted to a maximum of 50 ports.

Affected Configurations
This issue affects any block or subsystem that has more than 50 ports.

Design Impact
An assert m_inCount <= m_maxInCount which crashes MATLAB.

Workaround
Avoid creating a subsystem or block with more than 50 ports.

Solution Status
This issue is fixed in the DSP Builder advanced blockset v9.0.

Saturation Limitation When using the Scale Block
When you increase the number of fractional bits used for a signal, the MSB may be truncated.

Affected Configurations
Configurations using the Scale block.
Design Impact
The MSB may be truncated.

Workaround
Use a Convert block inside the primitive subsystem.

Solution Status
This issue will be fixed in a future release of the DSP Builder advanced blockset.

Very High Hard Multiplier Threshold Does Not Force the Use of Logic
Setting a very high hard multiplier threshold should avoid using any DSP blocks.

Affected Configurations
This issue affects all configurations.

Design Impact
Unwanted use of DSP blocks.

Workaround
None.

Solution Status
This issue will be fixed in a future release of the DSP Builder advanced blockset.

No Forward Flow Control in Primitive Subsystems
‘Bursty’ input data is not supported for primitive subsystems, for designs where the valid signal toggles high and low repeatedly.

Affected Configurations
This issue affects any primitive subsystem.

Design Impact
The simulation results are incorrect.

Workaround
Avoid using bursts of data.

Solution Status
This issue will be fixed in a future release of the DSP Builder advanced blockset.

Limitations of the ModelSim Testbench Flow
You can use the Run ModelSim block to simulate a model in ModelSim. This flow attempts to make a ModelSim testbench by synthesizing all the block (including any Simulink blocks in the top-level). However only a limited number of Simulink blocks are synthesizable and an empty subsystem is created for the unsupported blocks.
**Affected Configurations**

Any model with unsynthesizable Simulink blocks in the top-level design.

**Design Impact**

You cannot simulate the testbench.

**Workaround**

To workaround this issue, use only the supported blocks in the testbench for your design or write your own HDL for the empty subsystems. (A full list of synthesizable blocks is given in the description of the Run ModelSim block.) Alternatively, you can use the automatic testbench flow to simulate each subsystem in your design separately.

**Solution Status**

This issue will be fixed in a future release of the DSP Builder advanced blockset.

---

**Cannot Import a Combined Blockset Design into SOPC Builder**

You cannot import a design that combines blocks from the standard and advanced blocksets into SOPC Builder.

**Affected Configurations**

This issue affects configurations that combine blocks from the standard and advanced blocksets.

**Design Impact**

If you attempt to import a combined blockset design into SOPC Builder, DSP Builder issues warnings stating that there are multiple clocks with the same name and the component is not added to the system.

**Workaround**

There is no workaround.

**Solution Status**

This issue will be fixed in a future release of the DSP Builder advanced blockset.

---

**Incorrect Simulation Results if Word Length Greater Than 64 bits**

The simulation results are incorrect for ModelIP blocks if the word length is greater than 64 bits.

**Affected Configurations**

This issue affects configurations that include ModelIP blocks. However, ModelPrim blocks do support word lengths greater than 64 bits.

**Design Impact**

The simulation results are incorrect.
Workaround
Restrict the word length used by ModelIP blocks to 64 bits.

Solution Status
This issue will be fixed in a future release of the DSP Builder advanced blockset.

Compilation Error if Unused Channel Out Port is Terminated
Quartus II compilation fails if an unused channel or valid signal from a ChannelOut block is connected to a Simulink Terminator block inside a primitive subsystem.

Affected Configurations
Primitive subsystems with unused channel or valid output signals.

Design Impact
Incorrect HDL is generated and errors are issued when you attempt to compile the design in the Quartus II software.

Workaround
Leave the unused signals unconnected or terminate them outside the subsystem.

Solution Status
This issue will be fixed in a future release of the DSP Builder advanced blockset.
Revision History

The following table displays the revision history for these release notes.

<table>
<thead>
<tr>
<th>Date</th>
<th>Version</th>
<th>Changes Made</th>
</tr>
</thead>
<tbody>
<tr>
<td>15 July 2010</td>
<td>10.0</td>
<td>Updated for software version 10.0.</td>
</tr>
<tr>
<td>15 November 2009</td>
<td>9.1</td>
<td>Updated for software version 9.1.</td>
</tr>
</tbody>
</table>

How to Contact Altera

For the most up-to-date information about Altera® products, refer to the following table.

<table>
<thead>
<tr>
<th>Contact 1</th>
<th>Contact Method</th>
<th>Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>Technical support</td>
<td>Website</td>
<td><a href="http://www.altera.com/support">www.altera.com/support</a></td>
</tr>
<tr>
<td>Technical training</td>
<td>Website</td>
<td><a href="http://www.altera.com/training">www.altera.com/training</a></td>
</tr>
<tr>
<td>Email</td>
<td><a href="http://www.altera.com/training">www.altera.com/training</a></td>
<td></td>
</tr>
<tr>
<td>Product literature</td>
<td>Website</td>
<td><a href="http://www.altera.com/literature">www.altera.com/literature</a></td>
</tr>
<tr>
<td>Non-technical support (General)</td>
<td>Email</td>
<td><a href="mailto:nacomp@altera.com">nacomp@altera.com</a></td>
</tr>
<tr>
<td>(Software Licensing)</td>
<td>Email</td>
<td><a href="mailto:authorization@altera.com">authorization@altera.com</a></td>
</tr>
</tbody>
</table>

Note to table:

(1) You can also contact your local Altera sales office or sales representative.

Typographic Conventions

This document uses the typographic conventions shown in the following table.

<table>
<thead>
<tr>
<th>Visual Cue</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Bold Type with Initial Capital Letters</strong></td>
<td>Indicates command names, dialog box titles, dialog box options, and other GUI labels. For example, <strong>Save As</strong> dialog box.</td>
</tr>
<tr>
<td><strong>Bold type</strong></td>
<td>Indicates directory names, project names, disk drive names, file names, file name extensions, and software utility names. For example, &lt;qdesigns&gt; directory, &lt;d: drive&gt;, and &lt;chiptrip.gdf&gt; file.</td>
</tr>
<tr>
<td><strong>Italic Type with Initial Capital Letters</strong></td>
<td>Indicates document titles. For example: <strong>AN 519: Stratix IV Design Guidelines</strong>.</td>
</tr>
<tr>
<td><strong>Italic type</strong></td>
<td>Indicates variables. For example, ( n + 1 ). Variable names are enclosed in angle brackets (&lt;&gt;). For example, &lt;file name&gt; and &lt;project name&gt;.pof file.</td>
</tr>
<tr>
<td><strong>Initial Capital Letters</strong></td>
<td>Indicates keyboard keys and menu names. For example, Delete key and the Options menu.</td>
</tr>
<tr>
<td>“Subheading Title”</td>
<td>Quotation marks indicate references to sections within a document and titles of Quartus II Help topics. For example, “Typographic Conventions.”</td>
</tr>
<tr>
<td>Visual Cue</td>
<td>Meaning</td>
</tr>
<tr>
<td>----------------------------</td>
<td>-------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Courier type</td>
<td>Indicates signal, port, register, bit, block, and primitive names. For example, data1, tdi, and input. Active-low signals are denoted by suffix n. Example: resetn. Indicates command line commands and anything that must be typed exactly as it appears. For example, c:\qdesigns\tutorial\chiptrip.gdf. Also indicates sections of an actual file, such as a Report File, references to parts of files (for example, the AHDL keyword SUBDESIGN), and logic function names (for example, TRI).</td>
</tr>
<tr>
<td>1., 2., 3., and a., b., c., and so on.</td>
<td>Numbered steps indicate a list of items when the sequence of the items is important, such as the steps listed in a procedure.</td>
</tr>
<tr>
<td>■ ■</td>
<td>The hand points to information that requires special attention.</td>
</tr>
<tr>
<td>!</td>
<td>A caution calls attention to a condition or possible situation that can damage or destroy the product or your work.</td>
</tr>
<tr>
<td>-</td>
<td>A warning calls attention to a condition or possible situation that can cause you injury.</td>
</tr>
<tr>
<td>-</td>
<td>The angled arrow instructs you to press the enter key.</td>
</tr>
<tr>
<td>❀</td>
<td>The feet direct you to more information about a particular topic.</td>
</tr>
</tbody>
</table>