Chapter 1. Standard Blockset
Revision History ................................................................. 1–1
Errata ....................................................................................... 1–1
Error During Compilation ....................................................... 1–2
MegaCore Function Blocks Fail ............................................... 1–2
DSP Builder Launches Incorrect Version ................................. 1–3
Generating FIR Compiler MegaCore Function Creates Errors .... 1–3
SignalTap II and HIL Design Flows Fail with Development Boards 1–3
State Machine Editor Block Fails to Integrate ......................... 1–4
HDL Import Block Fails to Import a RAM Initialization File ....... 1–5
MATLAB Runs Out of Java Virtual Machine Heap Memory ...... 1–5
VCD Sink Block Does Not Generate HDL Files on Linux ......... 1–5
MATLAB Hangs If You Press Enter in Signal Compiler Device Box 1–6
Additional Project Assignments Required for User Libraries ...... 1–6
Dual-Clock FIFO Simulation Does Not Match ModelSim ........... 1–7

Chapter 2. Advanced Blockset
Revision History ................................................................. 2–1
Errata ....................................................................................... 2–1
Incorrect Logic Created ........................................................... 2–2
Excessive Register Usage ....................................................... 2–2
Loop Block Incorrect Behavior ................................................. 2–3
MATLAB Supported Versions .................................................. 2–3
Simulation Fails ................................................................. 2–4
DSP Builder Advanced Blockset Reports an Incorrect Version Number 2–4
ATB Generation Does Not Follow Links out of Subsystems ....... 2–5
Incorrect VHDL if Single Simulink Import Block is Connected to Multiple ChannelIn Ports 2–5
Bad Port Indexing for Complex Data on ChannelOut and ChannelIn 2–6
Driving Simulink Scopes With Complex Signals Gives Errors 2–6
No Support for 1-Input Mode .................................................... 2–7
Imaginary Output from Complex to Real Imaginary Gives Internal Errors 2–7
Multiplier Optimizations Can Cause Signal Width Mismatch in Synthesis 2–8
Latency Not Balanced for ModelBus Blocks Outside Primitive Subsystem 2–8
Saturation Limitation When using the Scale Block .................... 2–9
Very High Hard Multiplier Threshold Does Not Force the Use of Logic 2–9
No Forward Flow Control in Primitive Subsystems ................. 2–10
Cannot Import a Combined Blockset Design into SOPC Builder 2–10
Incorrect Simulation Results if Word Length Greater Than 64 bits 2–10
Compilation Error if Unused Channel Out Port is Terminated .... 2–11

Additional Information
Document Revision History ..................................................... Info–1
How to Contact Altera .......................................................... Info–1
Typographic Conventions ..................................................... Info–1
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.

<table>
<thead>
<tr>
<th>Version</th>
<th>Date</th>
<th>Revision</th>
</tr>
</thead>
<tbody>
<tr>
<td>11.0</td>
<td>April 2011</td>
<td>Added Virtual Pins block.</td>
</tr>
<tr>
<td>10.1</td>
<td>December 2010</td>
<td>Maintenance release.</td>
</tr>
<tr>
<td>10.0</td>
<td>July 2010</td>
<td>Maintenance release.</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 v11.0, v10.1, and v10.1.

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

<table>
<thead>
<tr>
<th>Added or Updated</th>
<th>Issue</th>
<th>Affected Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>15 May 11</td>
<td>Error During Compilation</td>
<td>11.0 10.1 10.0</td>
</tr>
<tr>
<td></td>
<td>MegaCore Function Blocks Fail</td>
<td></td>
</tr>
<tr>
<td>15 Dec 10</td>
<td>DSP Builder Launches Incorrect Version</td>
<td>10.1 10.0 Fixed</td>
</tr>
<tr>
<td>15 Jul 10</td>
<td>Generating FIR Compiler MegaCore Function Creates Errors</td>
<td>11.0 10.1 10.0</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>01 Apr 09</td>
<td>MATLAB Runs Out of Java Virtual Machine Heap Memory</td>
<td>11.0 10.1 10.0</td>
</tr>
<tr>
<td>15 Mar 09</td>
<td>VCD Sink Block Does Not Generate HDL Files on Linux</td>
<td>11.0 10.1 10.0</td>
</tr>
<tr>
<td>01 Nov 08</td>
<td>MATLAB Hangs If You Press Enter in Signal Compiler Device Box</td>
<td>11.0 10.1 10.0</td>
</tr>
<tr>
<td>15 May 08</td>
<td>Additional Project Assignments Required for User Libraries</td>
<td>11.0 10.1 10.0</td>
</tr>
<tr>
<td>01 May 07</td>
<td>Dual-Clock FIFO Simulation Does Not Match ModelSim</td>
<td>11.0 10.1 10.0</td>
</tr>
</tbody>
</table>
Error During Compilation

You may see the following error during compilation:

Error: DSP block output WYSIWYG primitive

"adl_db_GN:autoinst|adl_db_GN_adl_db_ADL:adl_db_adl_0|adl_db_GN_adl_db_AD
b_ADl_FDL:adl_db_adl_fdl_0|adl_db_GN_adl_db_ADl_FDL_PPF_lM:adl_db_adl_fdl
_fdl_ppf_im_0|alt_dspbuilder_multiply_add_stratix_iii_GNXSQ42PSZ:ppf_tap
_4_7|almult_add:u1|mult_add_946g:auto_generated|mac_out5"

has CHAININ port that must be driven by the dataout port of a DSP
block output WYSIWYG primitive with the OPERATION_MODE value set to
ACCUMULATOR_CHAIN_OUT or TWO_LEVEL_ADDER_CHAIN_OUT File:

Y:/SmxFadingSimulator/hw/FPGAs/DSP_Kernel/Build/systems/adl/FPGA/DB/ad
l_db_dspbuilder/db/mult_add_946g.tdf

Line: 36

Affected Configurations

This issue affects all configurations that contain a chain of Stratix III DSP blocks.

Design Impact

Design compilation fails with errors at the end of synthesis flow in the Quartus II
software.

Workaround

This issue has no workaround.

Solution Status

This issue will be fixed in a future version of the DSP Builder standard blockset.

MegaCore Function Blocks Fail

You may see erratic behavior with the MegaCore function blocks, which may be
because the DSP Builder installation path contains spaces.

Affected Configurations

DSP Builder designs that contain any MegaCore functions.

Design Impact

You cannot generates instances with MegaCore functions.

Workaround

Ensure that you do not use spaces in the DSP Builder installation directory.

Solution Status

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

If MATLAB points to incorrect DSP Builder libraries or gives errors, the startup.m file may contain incorrect paths, which have a higher precedence than the new paths.

Affected Configurations

This issue affects Linux OS.

Design Impact

MATLAB points to old DSP Builder libraries if available, or gives errors if the previous version of DSP Builder is removed.

Workaround

To work around this issue, manually remove all old path settings from startup.m.

Solution Status

This issue is fixed in DSP Builder v10.1.

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 unparameterized (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.

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.

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.
2. Advanced Blockset

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.

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>11.0</td>
<td>April 2011</td>
<td>Maintenance release.</td>
</tr>
</tbody>
</table>
| 10.1    | December 2010 | ■ Added new blocks  
|         |             | ■ Added new design examples                    |
| 10.0    | July 2010  | Added new blocks                              |

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 v11.0, v10.1, and 10.0.

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 (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 May 11</td>
<td>Incorrect Logic Created</td>
<td>11.0  10.1  10.0</td>
</tr>
<tr>
<td>15 Dec 10</td>
<td>Excessive Register Usage</td>
<td>Fixed  11.0  10.1</td>
</tr>
<tr>
<td></td>
<td>Loop Block Incorrect Behavior</td>
<td>—     Fixed  11.0</td>
</tr>
<tr>
<td>15 Jul 10</td>
<td>MATLAB Supported Versions</td>
<td>—     Fixed  11.0</td>
</tr>
<tr>
<td></td>
<td>Simulation Fails</td>
<td>—     Fixed  11.0</td>
</tr>
<tr>
<td>15 Nov 09</td>
<td>DSP Builder Advanced Blockset Reports an Incorrect Version Number</td>
<td>Fixed  11.0  10.1  10.0</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>Fixed  11.0  10.1  10.0</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>—     Fixed  11.0</td>
</tr>
<tr>
<td></td>
<td>Imaginary Output from Complex to Real Imaginary Gives Internal Errors</td>
<td></td>
</tr>
<tr>
<td>15 May 09</td>
<td>Multiplier Optimizations Can Cause Signal Width Mismatch in Synthesis</td>
<td></td>
</tr>
</tbody>
</table>
Table 2–2. DSP Builder Advanced 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>01 Nov 08</td>
<td>Incorrect Logic Created If you use folding on a subsystem with multipliers larger than that natively supported (for example wider than 18 × 18 on Cyclone families and larger than 36 × 36 on Stratix families) illegal logic may be created.</td>
<td>✔ ✔ ✔</td>
</tr>
<tr>
<td></td>
<td>Affected Configurations This issue affects all designs that use folding.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Design Impact The design may fail.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Workaround To work around the issue, set the workspace variable DSBAFeatures.OptimizeLargeMults = false for the HDL generation of the model in such cases. For example, on the File menu, point to Model Properties, click on the Callbacks tab, click InitFcn, and type in the following function: DSPBAFeatures.OptimizeLargeMults = false; Alternatively, on the File menu, point to Model Properties, click on the Callbacks tab, click StopFcn, and type in the following function: DSPBAFeatures.OptimizeLargeMults = true;</td>
<td>✔ ✔ ✔</td>
</tr>
<tr>
<td>15 May 08</td>
<td>Excessive Register Usage DSP Builder incorrectly sets the following attribute, when generating some floating-point designs: -name AUTO_SHIFT_REGISTER_RECOGNITION OFF</td>
<td>✔ ✔ ✔</td>
</tr>
<tr>
<td></td>
<td>Affected Configurations This issue affects all floating-point designs.</td>
<td></td>
</tr>
</tbody>
</table>

Incorrect Logic Created

If you use folding on a subsystem with multipliers larger than that natively supported (for example wider than 18 × 18 on Cyclone families and larger than 36 × 36 on Stratix families) illegal logic may be created.

Affected Configurations

This issue affects all designs that use folding.

Design Impact

The design may fail.

Workaround

To work around the issue, set the workspace variable DSBAFeatures.OptimizeLargeMults = false for the HDL generation of the model in such cases. For example, on the File menu, point to Model Properties, click on the Callbacks tab, click InitFcn, and type in the following function:

```
DSPBAFeatures.OptimizeLargeMults = false;
```

Alternatively, on the File menu, point to Model Properties, click on the Callbacks tab, click StopFcn, and type in the following function:

```
DSPBAFeatures.OptimizeLargeMults = true;
```

Solution Status

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

Excessive Register Usage

DSP Builder incorrectly sets the following attribute, when generating some floating-point designs:

```
-name AUTO_SHIFT_REGISTER_RECOGNITION OFF
```

Affected Configurations

This issue affects all floating-point designs.
Design Impact
DSP Builder does not optimize register chains within the VHDL floating-point libraries to MLAB-based structures, which causes excessive register use.

Workaround
To work around this issue, change OFF to ON with floating-point designs.

Solution Status
This issue is fixed in the DSP Builder advanced blockset v11.0.

Loop Block Incorrect Behavior
The Loop block shows incorrect behavior with a mismatch between Simulink and VHDL if it receives a go while the block is disabled. Also it may not reset the running flag.

Affected Configurations
This issue affects all configurations.

Design Impact
You cannot implement a Loop block in your design.

Workaround
This issue has no workaround.

Solution Status
This issue is fixed in DSP Builder v10.1.

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 is fixed in DSP Builder v10.1.
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 is fixed in DSP Builder v10.1.

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 is fixed in the DSP Builder advanced blockset v11.0.

**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 mode.
```

**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 is fixed in the DSP Builder advanced blockset v11.0.

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:

```
Altera DSP Builder Advanced Internal Restriction: 'in[i].m_inWidth <= WIDE_MAX_DATAWIDTH'. Error in block 'rrh_duc/device/DUC/carrier_duc/CarrierEnableGainSum/Or_rsrd_fix' in .\hw_fu.cpp:1221.
```

**Workaround**

This issue has no workaround.

**Solution Status**

This issue is fixed in DSP Builder v10.1.

**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.

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 a 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.

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.

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.

**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.
This chapter provides additional information about the document and Altera.

**Document Revision History**

The following table shows the revision history for this document.

<table>
<thead>
<tr>
<th>Date</th>
<th>Version</th>
<th>Changes</th>
</tr>
</thead>
<tbody>
<tr>
<td>15 May 2011</td>
<td>11.0</td>
<td>Updated for software version 11.0.</td>
</tr>
<tr>
<td>15 December 2010</td>
<td>10.1</td>
<td>Updated for software version 10.1.</td>
</tr>
<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**

To locate 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></td>
<td>Email</td>
<td><a href="mailto:custrain@altera.com">custrain@altera.com</a></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</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**

The following table shows the typographic conventions this document uses.

<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>Indicate command names, dialog box titles, dialog box options, and other GUI labels. For example, <strong>Save As</strong> dialog box. For GUI elements, capitalization matches the GUI.</td>
</tr>
<tr>
<td><strong>Bold</strong></td>
<td>Indicates directory names, project names, disk drive names, file names, file name extensions, software utility names, and GUI labels. For example, <strong>\qdesigns</strong> directory, <strong>D:</strong> drive, and <strong>chiptrip.gdf</strong> file.</td>
</tr>
<tr>
<td><strong>Italic Type with Initial Capital Letters</strong></td>
<td>Indicate document titles. For example, <strong>Stratix IV Design Guidelines</strong>.</td>
</tr>
<tr>
<td>Visual Cue</td>
<td>Meaning</td>
</tr>
<tr>
<td>--------------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td><em>italic type</em></td>
<td>Indicates variables. For example, ( n + 1 ). Variable names are enclosed in angle brackets (&lt; &gt;). For example, <code>&lt;file name&gt;</code> and <code>&lt;project name&gt;.pof</code> file.</td>
</tr>
<tr>
<td>Initial Capital Letters</td>
<td>Indicate keyboard keys and menu names. For example, the 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>Courier type</td>
<td>Indicates signal, port, register, bit, block, and primitive names. For example, <code>data1</code>, <code>tdi</code>, and <code>input</code>. The suffix <code>n</code> denotes an active-low signal. For example, <code>resetn</code>.</td>
</tr>
<tr>
<td></td>
<td>Indicates command line commands and anything that must be typed exactly as it appears. For example, <code>c:\qdesigns\tutorial\chiptrip.gdf</code>.</td>
</tr>
<tr>
<td></td>
<td>Also indicates sections of an actual file, such as a Report File, references to parts of files (for example, the AHDL keyword <code>SUBDESIGN</code>), and logic function names (for example, <code>TRI</code>).</td>
</tr>
<tr>
<td>➷</td>
<td>An angled arrow instructs you to press the Enter key.</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>Bullets indicate a list of items when the sequence of the items is not important.</td>
</tr>
<tr>
<td>❮</td>
<td>The hand points to information that requires special attention.</td>
</tr>
<tr>
<td>❔</td>
<td>A question mark directs you to a software help system with related information.</td>
</tr>
<tr>
<td>❘</td>
<td>The feet direct you to another document or website with related information.</td>
</tr>
<tr>
<td>CAUTION</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>WARNING</td>
<td>A warning calls attention to a condition or possible situation that can cause you injury.</td>
</tr>
<tr>
<td>❮</td>
<td>The envelope links to the Email Subscription Management Center page of the Altera website, where you can sign up to receive update notifications for Altera documents.</td>
</tr>
</tbody>
</table>