**IP Core Generation Report for lab5**

**Summary**

|  |  |
| --- | --- |
| IP core name | Pitch\_Shi\_ip |
| IP core version | 1.0 |
| IP core folder | [hdl\_prj\ipcore\Pitch\_Shi\_ip\_v1\_0](matlab:uiopen('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\*.*')) |
| Target platform | Generic Altera Platform |
| Target tool | Altera QUARTUS II |
| Target language | VHDL |
| Model | [lab5](matlab:Simulink.ID.hilite('lab5')) |
| Model version | 1.79 |
| HDL Coder version | 3.11 |
| IP core generated on | 24-Apr-2018 13:36:54 |
| IP core generated for | [Pitch\_Shift](matlab:Simulink.ID.hilite('lab5:52')) |

**Target Interface Configuration**

You chose the following target interface configuration for [lab5](matlab:Simulink.ID.hilite('lab5')) :   
  
Processor/FPGA synchronization mode: **Free running**   
  
Target platform interface table:

| **Port Name** | **Port Type** | **Data Type** | **Target Platform Interfaces** | **Bit Range / Address / FPGA Pin** |
| --- | --- | --- | --- | --- |
| [Data\_in](matlab:Simulink.ID.hilite('lab5:53')) | Inport | sfix28\_En24 | External Port |  |
| [Frequency\_Shift](matlab:Simulink.ID.hilite('lab5:54')) | Inport | int16 | External Port |  |
| [Output](matlab:Simulink.ID.hilite('lab5:55')) | Outport | sfix28\_En24 | External Port |  |

**Register Address Mapping**

The following AXI4 bus accessible registers were generated for this IP core:

| **Register Name** | **Address Offset** | **Description** |
| --- | --- | --- |
| IPCore\_Reset | 0x0 | write 0x1 to bit 0 to reset IP core |
| IPCore\_Enable | 0x4 | enabled (by default) when bit 0 is 0x1 |
| IPCore\_Timestamp | 0x8 | contains unique IP timestamp (yymmddHHMM): 1804241336 |

The register address mapping is also in the following C header file for you to use when programming the processor:   
[include\Pitch\_Shi\_ip\_addr.h](matlab:edit('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\include\Pitch_Shi_ip_addr.h'))   
The IP core name is appended to the register names to avoid name conflicts.

**IP Core User Guide**

**Theory of Operation**   
  
This IP core is designed to be connected to an embedded processor with an **AXI4 interface.**The processor acts as master, and the IP core acts as slave. By accessing the generated registers via the AXI4 interface, the processor can control the IP core, and read and write data from and to the IP core.   
  
For example, to reset the IP core, write 0x1 to the bit 0 of IPCore\_Reset register. To enable or disable the IP core, write 0x1 or 0x0 to the IPCore\_Enable register. To access the data ports of the MATLAB/Simulink algorithm, read or write to the associated data registers.   
  
   
  
This IP core also support the **External Port**interface. To connect the external ports to the FPGA external IO pins, add FPGA pin assignment constraints in the Altera Qsys environment.   
  
**Processor/FPGA Synchronization**   
  
The **Free running**mode means there is no explicit synchronization between embedded processor software execution (SW) and the IP core (HW). SW and HW runs independently. The data written from the processor to IP core takes effect immediately, and the data read from the IP core is the latest data available on the IP core output ports.   
  
   
  
**Altera Qsys Environment Integration**   
  
This IP Core is generated for the Altera Qsys environment. The following steps are an example showing how to integrate the generated IP core into Altera Qsys environment:   
  
1. Copy the IP core folder into the "ip" folder in your Altera Qsys project folder. If there is no folder named "ip", create one. This step adds the IP core into the Qsys project user library.   
2. In the Qsys project, find the IP core in the user library and add the IP core to the design.   
3. Connect the S\_AXI port of the IP core to the embedded processor's AXI master port.   
4. Connect the clock and reset ports of the IP core to the global clock and reset signals.   
5. Assign a base address for the IP core.   
6. Connect external ports and add FPGA pin assignment constraints.   
7. Generate FPGA bitstream and download the bitstream to target device.   
  
If you are targeting Intel SoC hardware supported by HDL Coder Support Package for Intel SoC Devices, you can select the board you are using in the Target platform option in the Set Target > Set Target Device and Synthesis Tool task. You can then use Embedded System Integration tasks in HDL Workflow Advisor to help you integrate the generated IP core into Altera Qsys environment.

**IP Core File List**

The IP core folder is located at:   
[hdl\_prj\ipcore\Pitch\_Shi\_ip\_v1\_0](matlab:uiopen('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\*.*'))   
Following files are generated under this folder:   
  
**IP core definition files**   
[Pitch\_Shi\_ip\_hw.tcl](matlab:edit('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\Pitch_Shi_ip_hw.tcl'))   
  
**IP core report**   
[doc\lab5\_ip\_core\_report.html](matlab:web('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\doc\lab5_ip_core_report.html'))   
  
**IP core HDL source files**   
[hdl\Pitch\_Shi\_ip\_src\_Pitch\_Shift\_pkg.vhd](matlab:edit('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\hdl\Pitch_Shi_ip_src_Pitch_Shift_pkg.vhd'))   
[hdl\Pitch\_Shi\_ip\_src\_Latch\_5.vhd](matlab:edit('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\hdl\Pitch_Shi_ip_src_Latch_5.vhd'))   
[hdl\Pitch\_Shi\_ip\_src\_Subsystem.vhd](matlab:edit('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\hdl\Pitch_Shi_ip_src_Subsystem.vhd'))   
[hdl\Pitch\_Shi\_ip\_src\_FFT\_Pulse\_gen.vhd](matlab:edit('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\hdl\Pitch_Shi_ip_src_FFT_Pulse_gen.vhd'))   
[hdl\Pitch\_Shi\_ip\_src\_Hanning\_window.vhd](matlab:edit('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\hdl\Pitch_Shi_ip_src_Hanning_window.vhd'))   
[hdl\Pitch\_Shi\_ip\_src\_Read\_Address\_en.vhd](matlab:edit('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\hdl\Pitch_Shi_ip_src_Read_Address_en.vhd'))   
[hdl\Pitch\_Shi\_ip\_src\_Read\_Addr\_gen.vhd](matlab:edit('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\hdl\Pitch_Shi_ip_src_Read_Addr_gen.vhd'))   
[hdl\Pitch\_Shi\_ip\_src\_DualRateDualPortRAM\_generic.vhd](matlab:edit('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\hdl\Pitch_Shi_ip_src_DualRateDualPortRAM_generic.vhd'))   
[hdl\Pitch\_Shi\_ip\_src\_Addr\_offset.vhd](matlab:edit('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\hdl\Pitch_Shi_ip_src_Addr_offset.vhd'))   
[hdl\Pitch\_Shi\_ip\_src\_Subsystem\_block.vhd](matlab:edit('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\hdl\Pitch_Shi_ip_src_Subsystem_block.vhd'))   
[hdl\Pitch\_Shi\_ip\_src\_TWDLROM.vhd](matlab:edit('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\hdl\Pitch_Shi_ip_src_TWDLROM.vhd'))   
[hdl\Pitch\_Shi\_ip\_src\_MINRESRX2FFT\_CTRL.vhd](matlab:edit('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\hdl\Pitch_Shi_ip_src_MINRESRX2FFT_CTRL.vhd'))   
[hdl\Pitch\_Shi\_ip\_src\_SimpleDualPortRAM\_generic.vhd](matlab:edit('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\hdl\Pitch_Shi_ip_src_SimpleDualPortRAM_generic.vhd'))   
[hdl\Pitch\_Shi\_ip\_src\_MINRESRX2FFT\_MEMORY.vhd](matlab:edit('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\hdl\Pitch_Shi_ip_src_MINRESRX2FFT_MEMORY.vhd'))   
[hdl\Pitch\_Shi\_ip\_src\_MINRESRX2FFT\_BTFSEL.vhd](matlab:edit('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\hdl\Pitch_Shi_ip_src_MINRESRX2FFT_BTFSEL.vhd'))   
[hdl\Pitch\_Shi\_ip\_src\_Complex4Multiply.vhd](matlab:edit('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\hdl\Pitch_Shi_ip_src_Complex4Multiply.vhd'))   
[hdl\Pitch\_Shi\_ip\_src\_MINRESRX2\_BUTTERFLY.vhd](matlab:edit('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\hdl\Pitch_Shi_ip_src_MINRESRX2_BUTTERFLY.vhd'))   
[hdl\Pitch\_Shi\_ip\_src\_MINRESRX2FFT\_MEMSEL.vhd](matlab:edit('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\hdl\Pitch_Shi_ip_src_MINRESRX2FFT_MEMSEL.vhd'))   
[hdl\Pitch\_Shi\_ip\_src\_MINRESRX2FFT\_OUTMux.vhd](matlab:edit('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\hdl\Pitch_Shi_ip_src_MINRESRX2FFT_OUTMux.vhd'))   
[hdl\Pitch\_Shi\_ip\_src\_FFT\_HDL\_Optimized.vhd](matlab:edit('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\hdl\Pitch_Shi_ip_src_FFT_HDL_Optimized.vhd'))   
[hdl\Pitch\_Shi\_ip\_src\_Analysis.vhd](matlab:edit('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\hdl\Pitch_Shi_ip_src_Analysis.vhd'))   
[hdl\Pitch\_Shi\_ip\_src\_MATLAB\_Function.vhd](matlab:edit('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\hdl\Pitch_Shi_ip_src_MATLAB_Function.vhd'))   
[hdl\Pitch\_Shi\_ip\_src\_Frequency\_Processing.vhd](matlab:edit('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\hdl\Pitch_Shi_ip_src_Frequency_Processing.vhd'))   
[hdl\Pitch\_Shi\_ip\_src\_Hanning\_window\_block.vhd](matlab:edit('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\hdl\Pitch_Shi_ip_src_Hanning_window_block.vhd'))   
[hdl\Pitch\_Shi\_ip\_src\_Chart.vhd](matlab:edit('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\hdl\Pitch_Shi_ip_src_Chart.vhd'))   
[hdl\Pitch\_Shi\_ip\_src\_Chart1.vhd](matlab:edit('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\hdl\Pitch_Shi_ip_src_Chart1.vhd'))   
[hdl\Pitch\_Shi\_ip\_src\_HDL\_FIFO.vhd](matlab:edit('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\hdl\Pitch_Shi_ip_src_HDL_FIFO.vhd'))   
[hdl\Pitch\_Shi\_ip\_src\_HDL\_FIFO1.vhd](matlab:edit('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\hdl\Pitch_Shi_ip_src_HDL_FIFO1.vhd'))   
[hdl\Pitch\_Shi\_ip\_src\_HDL\_FIFO2.vhd](matlab:edit('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\hdl\Pitch_Shi_ip_src_HDL_FIFO2.vhd'))   
[hdl\Pitch\_Shi\_ip\_src\_HDL\_FIFO3.vhd](matlab:edit('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\hdl\Pitch_Shi_ip_src_HDL_FIFO3.vhd'))   
[hdl\Pitch\_Shi\_ip\_src\_MATLAB\_Function\_block.vhd](matlab:edit('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\hdl\Pitch_Shi_ip_src_MATLAB_Function_block.vhd'))   
[hdl\Pitch\_Shi\_ip\_src\_Overlap\_and\_Add.vhd](matlab:edit('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\hdl\Pitch_Shi_ip_src_Overlap_and_Add.vhd'))   
[hdl\Pitch\_Shi\_ip\_src\_TWDLROM\_block.vhd](matlab:edit('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\hdl\Pitch_Shi_ip_src_TWDLROM_block.vhd'))   
[hdl\Pitch\_Shi\_ip\_src\_MINRESRX2FFT\_CTRL\_block.vhd](matlab:edit('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\hdl\Pitch_Shi_ip_src_MINRESRX2FFT_CTRL_block.vhd'))   
[hdl\Pitch\_Shi\_ip\_src\_MINRESRX2FFT\_MEMORY\_block.vhd](matlab:edit('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\hdl\Pitch_Shi_ip_src_MINRESRX2FFT_MEMORY_block.vhd'))   
[hdl\Pitch\_Shi\_ip\_src\_MINRESRX2FFT\_BTFSEL\_block.vhd](matlab:edit('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\hdl\Pitch_Shi_ip_src_MINRESRX2FFT_BTFSEL_block.vhd'))   
[hdl\Pitch\_Shi\_ip\_src\_Complex4Multiply\_block.vhd](matlab:edit('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\hdl\Pitch_Shi_ip_src_Complex4Multiply_block.vhd'))   
[hdl\Pitch\_Shi\_ip\_src\_MINRESRX2\_BUTTERFLY\_block.vhd](matlab:edit('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\hdl\Pitch_Shi_ip_src_MINRESRX2_BUTTERFLY_block.vhd'))   
[hdl\Pitch\_Shi\_ip\_src\_MINRESRX2FFT\_MEMSEL\_block.vhd](matlab:edit('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\hdl\Pitch_Shi_ip_src_MINRESRX2FFT_MEMSEL_block.vhd'))   
[hdl\Pitch\_Shi\_ip\_src\_MINRESRX2FFT\_OUTMux\_block.vhd](matlab:edit('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\hdl\Pitch_Shi_ip_src_MINRESRX2FFT_OUTMux_block.vhd'))   
[hdl\Pitch\_Shi\_ip\_src\_IFFT\_HDL\_Optimized.vhd](matlab:edit('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\hdl\Pitch_Shi_ip_src_IFFT_HDL_Optimized.vhd'))   
[hdl\Pitch\_Shi\_ip\_src\_Synthesis.vhd](matlab:edit('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\hdl\Pitch_Shi_ip_src_Synthesis.vhd'))   
[hdl\Pitch\_Shi\_ip\_src\_Pitch\_Shift\_tc.vhd](matlab:edit('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\hdl\Pitch_Shi_ip_src_Pitch_Shift_tc.vhd'))   
[hdl\Pitch\_Shi\_ip\_src\_Pitch\_Shift.vhd](matlab:edit('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\hdl\Pitch_Shi_ip_src_Pitch_Shift.vhd'))   
[hdl\Pitch\_Shi\_ip\_dut.vhd](matlab:edit('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\hdl\Pitch_Shi_ip_dut.vhd'))   
[hdl\Pitch\_Shi\_ip\_SimpleDualPortRAM\_generic.vhd](matlab:edit('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\hdl\Pitch_Shi_ip_SimpleDualPortRAM_generic.vhd'))   
[hdl\Pitch\_Shi\_ip\_rdfifo\_data\_classic.vhd](matlab:edit('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\hdl\Pitch_Shi_ip_rdfifo_data_classic.vhd'))   
[hdl\Pitch\_Shi\_ip\_rdfifo\_data.vhd](matlab:edit('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\hdl\Pitch_Shi_ip_rdfifo_data.vhd'))   
[hdl\Pitch\_Shi\_ip\_SimpleDualPortRAM\_singlebit.vhd](matlab:edit('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\hdl\Pitch_Shi_ip_SimpleDualPortRAM_singlebit.vhd'))   
[hdl\Pitch\_Shi\_ip\_rdfifo\_last\_classic.vhd](matlab:edit('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\hdl\Pitch_Shi_ip_rdfifo_last_classic.vhd'))   
[hdl\Pitch\_Shi\_ip\_rdfifo\_last.vhd](matlab:edit('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\hdl\Pitch_Shi_ip_rdfifo_last.vhd'))   
[hdl\Pitch\_Shi\_ip\_rdfifo\_rid\_classic.vhd](matlab:edit('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\hdl\Pitch_Shi_ip_rdfifo_rid_classic.vhd'))   
[hdl\Pitch\_Shi\_ip\_rdfifo\_rid.vhd](matlab:edit('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\hdl\Pitch_Shi_ip_rdfifo_rid.vhd'))   
[hdl\Pitch\_Shi\_ip\_axi4\_module.vhd](matlab:edit('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\hdl\Pitch_Shi_ip_axi4_module.vhd'))   
[hdl\Pitch\_Shi\_ip\_addr\_decoder.vhd](matlab:edit('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\hdl\Pitch_Shi_ip_addr_decoder.vhd'))   
[hdl\Pitch\_Shi\_ip\_axi4.vhd](matlab:edit('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\hdl\Pitch_Shi_ip_axi4.vhd'))   
[hdl\Pitch\_Shi\_ip.vhd](matlab:edit('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\hdl\Pitch_Shi_ip.vhd'))   
  
**IP core C header file**   
[include\Pitch\_Shi\_ip\_addr.h](matlab:edit('hdl_prj\ipcore\Pitch_Shi_ip_v1_0\include\Pitch_Shi_ip_addr.h'))

Code Interface Report for lab5

**Input ports**

|  |  |  |
| --- | --- | --- |
| **Port Name** | **Datatype** | **Bits** |
| clk | boolean | 1 |
| reset | boolean | 1 |
| clk\_enable | boolean | 1 |
| *[Data\_in](matlab:set_param('lab5',%20'hiliteAncestors',%20'none');%20Simulink.ID.hilite('lab5:53'))* | sfix28\_En24 | 28 |
| [*Frequency\_Shift*](matlab:set_param('lab5',%20'hiliteAncestors',%20'none');%20Simulink.ID.hilite('lab5:54')) | int16 | 16 |

**Output ports**

|  |  |  |
| --- | --- | --- |
| **Port Name** | **Datatype** | **Bits** |
| ce\_out | boolean | 1 |
| [*Output*](matlab:set_param('lab5',%20'hiliteAncestors',%20'none');%20Simulink.ID.hilite('lab5:55')) | sfix28\_En24 | 28 |

Generic Resource Report for lab5

**Summary**

|  |  |
| --- | --- |
| Multipliers | 10 |
| Adders/Subtractors | 18008 |
| Registers | 13887 |
| Total 1-Bit Registers | 244658 |
| RAMs | 17 |
| Multiplexers | 2777 |
| I/O Bits | 76 |
| Static Shift operators | 8 |
| Dynamic Shift operators | 0 |

**Detailed Report**

##### Report for Multiply-instantiated Identical Subsystems 0 : *[Pitch\_Shift](matlab:set_param('lab5',%20'hiliteAncestors',%20'none');%20Simulink.ID.hilite('lab5:52'))*

##### Number of I/O Bits (76)

[+] Number of Input Bits: 47   
[+] Number of Output Bits: 29 

##### Report for Multiply-instantiated Identical Subsystems 1 : [*Synthesis*](matlab:set_param('lab5',%20'hiliteAncestors',%20'none');%20Simulink.ID.hilite('lab5:196'))

##### Multipliers (4)

28x28-bit Multiply : 4 

##### Adders/Subtractors (46)

9x9-bit Adder : 14   
57x57-bit Adder : 1   
58x58-bit Adder : 2   
10x10-bit Adder : 11   
4x4-bit Adder : 1   
2x2-bit Adder : 1   
3x3-bit Adder : 1   
22x22-bit Subtractor : 3   
12x12-bit Subtractor : 2   
9x9-bit Subtractor : 1   
29-bit Subtractor : 6   
57x57-bit Subtractor : 1   
58x58-bit Subtractor : 2 

##### Registers (173)

1-bit Register : 53   
9-bit Register : 19   
3-bit Register : 3   
10-bit Register : 3   
7-bit Register : 1   
28-bit Register : 69   
56-bit Register : 12   
57-bit Register : 2   
58-bit Register : 8   
4-bit Register : 2   
2-bit Register : 1 

##### RAMs (6)

512x28-bit RAM : 6 

##### Multiplexers (64)

1-bit 7-to-1 Multiplexer : 1   
3-bit 6-to-1 Multiplexer : 1   
7-bit 7-to-1 Multiplexer : 1   
9-bit 3-to-1 Multiplexer : 6   
9-bit 5-to-1 Multiplexer : 1   
28-bit 10-to-1 Multiplexer : 4   
9-bit 2-to-1 Multiplexer : 5   
28-bit 3-to-1 Multiplexer : 8   
9-bit 12-to-1 Multiplexer : 1   
28-bit 4-to-1 Multiplexer : 6   
9-bit 4-to-1 Multiplexer : 3   
1-bit 3-to-1 Multiplexer : 2   
1-bit 2-to-1 Multiplexer : 1   
4-bit 24-to-1 Multiplexer : 1   
1-bit 6-to-1 Multiplexer : 1   
4-bit 6-to-1 Multiplexer : 1   
10-bit 12-to-1 Multiplexer : 1   
9-bit 9-to-1 Multiplexer : 1   
3-bit 5-to-1 Multiplexer : 1   
1-bit 21-to-1 Multiplexer : 1   
1-bit 18-to-1 Multiplexer : 2   
1-bit 17-to-1 Multiplexer : 3   
28-bit 13-to-1 Multiplexer : 2   
1-bit 5-to-1 Multiplexer : 2   
1-bit 10-to-1 Multiplexer : 2   
10-bit 7-to-1 Multiplexer : 1   
2-bit 4-to-1 Multiplexer : 1   
1-bit 9-to-1 Multiplexer : 1   
1-bit 8-to-1 Multiplexer : 1   
1-bit 4-to-1 Multiplexer : 1   
28-bit 2-to-1 Multiplexer : 1 

##### Number of Shift operators (4)

Static Shift Right : 4 

##### Report for Multiply-instantiated Identical Subsystems 1 : *[Overlap\_and\_Add](matlab:set_param('lab5',%20'hiliteAncestors',%20'none');%20Simulink.ID.hilite('lab5:209'))*

##### Multipliers (1)

[+] 28x16-bit Multiply : 1 

##### Adders/Subtractors (21)

[+] 11x11-bit Adder : 9   
[+] 2x2-bit Adder : 1   
10x10-bit Adder : 8   
[+] 46x46-bit Adder : 3 

##### Registers (209)

28-bit Register : 2   
[+] 11-bit Register : 7   
16-bit Register : 1   
44-bit Register : 38   
[+] 2-bit Register : 1   
1-bit Register : 144   
10-bit Register : 16 

##### RAMs (4)

1024x44-bit RAM : 4 

##### Multiplexers (53)

[+] 11-bit 3-to-1 Multiplexer : 5   
1-bit 2-to-1 Multiplexer : 24   
10-bit 2-to-1 Multiplexer : 16   
44-bit 2-to-1 Multiplexer : 8 

##### Report for Multiply-instantiated Identical Subsystems 1 : [*MATLAB Function*](matlab:set_param('lab5',%20'hiliteAncestors',%20'none');%20Simulink.ID.hilite('lab5:243'))

##### Registers (1024)

2-bit Register : 1024 

##### Multiplexers (4)

[+] 1-bit 5-to-1 Multiplexer : 4 

##### Report for Multiply-instantiated Identical User-defined blocks 3 : [*Chart3*](matlab:set_param('lab5',%20'hiliteAncestors',%20'none');%20Simulink.ID.hilite('lab5:242')), [*Chart2*](matlab:set_param('lab5',%20'hiliteAncestors',%20'none');%20Simulink.ID.hilite('lab5:241')), [*Chart1*](matlab:set_param('lab5',%20'hiliteAncestors',%20'none');%20Simulink.ID.hilite('lab5:240'))

##### Registers (3)

3-bit Register : 3 

##### Multiplexers (6)

3-bit 6-to-1 Multiplexer : 3   
1-bit 9-to-1 Multiplexer : 3 

##### Report for Multiply-instantiated Identical User-defined blocks 1 : [*Chart*](matlab:set_param('lab5',%20'hiliteAncestors',%20'none');%20Simulink.ID.hilite('lab5:239'))

##### Registers (1)

3-bit Register : 1 

##### Multiplexers (2)

3-bit 6-to-1 Multiplexer : 1   
1-bit 9-to-1 Multiplexer : 1 

##### Report for Multiply-instantiated Identical Subsystems 1 : *[Hanning\_window](matlab:set_param('lab5',%20'hiliteAncestors',%20'none');%20Simulink.ID.hilite('lab5:223'))*

##### Registers (2)

[+] 16-bit Register : 2 

##### Multiplexers (1)

[+] 11-bit 3-to-1 Multiplexer : 1 

##### Report for Multiply-instantiated Identical Subsystems 1 : [*Frequency Processing*](matlab:set_param('lab5',%20'hiliteAncestors',%20'none');%20Simulink.ID.hilite('lab5:263'))

##### Adders/Subtractors (4)

11x11-bit Adder : 2   
10x10-bit Adder : 2 

##### Registers (12283)

16-bit Register : 2   
11-bit Register : 4   
[+] 28-bit Register : 8182   
10-bit Register : 4   
[+] 1-bit Register : 4091 

##### Multiplexers (18)

[+] 1-bit 2-to-1 Multiplexer : 9   
11-bit 2-to-1 Multiplexer : 2   
10-bit 2-to-1 Multiplexer : 2   
[+] 28-bit 2-to-1 Multiplexer : 2   
[+] 28-bit 2047-to-1 Multiplexer : 2   
[+] 1-bit 2047-to-1 Multiplexer : 1 

##### Report for Multiply-instantiated Identical User-defined blocks 1 : [*MATLAB Function*](matlab:set_param('lab5',%20'hiliteAncestors',%20'none');%20Simulink.ID.hilite('lab5:274'))

##### Adders/Subtractors (17885)

32x32-bit Adder : 8176   
32x32-bit Subtractor : 9198   
17x17-bit Subtractor : 511 

##### Multiplexers (2557)

28-bit 1544-to-1 Multiplexer : 2   
16-bit 3-to-1 Multiplexer : 2555 

##### Report for Multiply-instantiated Identical Subsystems 1 : [*Analysis*](matlab:set_param('lab5',%20'hiliteAncestors',%20'none');%20Simulink.ID.hilite('lab5:42'))

##### Multipliers (4)

28x28-bit Multiply : 4 

##### Adders/Subtractors (46)

9x9-bit Adder : 14   
57x57-bit Adder : 1   
58x58-bit Adder : 2   
10x10-bit Adder : 11   
4x4-bit Adder : 1   
2x2-bit Adder : 1   
3x3-bit Adder : 1   
22x22-bit Subtractor : 3   
12x12-bit Subtractor : 2   
9x9-bit Subtractor : 1   
29-bit Subtractor : 6   
57x57-bit Subtractor : 1   
58x58-bit Subtractor : 2 

##### Registers (175)

1-bit Register : 56   
9-bit Register : 19   
3-bit Register : 3   
10-bit Register : 3   
7-bit Register : 1   
28-bit Register : 68   
56-bit Register : 12   
57-bit Register : 2   
58-bit Register : 8   
4-bit Register : 2   
2-bit Register : 1 

##### RAMs (6)

512x28-bit RAM : 6 

##### Multiplexers (63)

1-bit 7-to-1 Multiplexer : 1   
3-bit 6-to-1 Multiplexer : 1   
7-bit 7-to-1 Multiplexer : 1   
9-bit 3-to-1 Multiplexer : 6   
9-bit 5-to-1 Multiplexer : 1   
28-bit 10-to-1 Multiplexer : 4   
9-bit 2-to-1 Multiplexer : 5   
28-bit 3-to-1 Multiplexer : 8   
9-bit 12-to-1 Multiplexer : 1   
28-bit 4-to-1 Multiplexer : 6   
9-bit 4-to-1 Multiplexer : 3   
1-bit 3-to-1 Multiplexer : 2   
1-bit 2-to-1 Multiplexer : 1   
4-bit 24-to-1 Multiplexer : 1   
1-bit 6-to-1 Multiplexer : 1   
4-bit 6-to-1 Multiplexer : 1   
10-bit 12-to-1 Multiplexer : 1   
9-bit 9-to-1 Multiplexer : 1   
3-bit 5-to-1 Multiplexer : 1   
1-bit 21-to-1 Multiplexer : 1   
1-bit 18-to-1 Multiplexer : 2   
1-bit 17-to-1 Multiplexer : 3   
28-bit 13-to-1 Multiplexer : 2   
1-bit 5-to-1 Multiplexer : 2   
1-bit 10-to-1 Multiplexer : 2   
10-bit 7-to-1 Multiplexer : 1   
2-bit 4-to-1 Multiplexer : 1   
1-bit 9-to-1 Multiplexer : 1   
1-bit 8-to-1 Multiplexer : 1   
1-bit 4-to-1 Multiplexer : 1 

##### Number of Shift operators (4)

Static Shift Right : 4 

##### Report for Multiply-instantiated Identical Subsystems 1 : [*Subsystem*](matlab:set_param('lab5',%20'hiliteAncestors',%20'none');%20Simulink.ID.hilite('lab5:184'))

##### Multipliers (1)

[+] 28x16-bit Multiply : 1 

##### Adders/Subtractors (1)

[+] 11x11-bit Adder : 1 

##### Registers (7)

[+] 11-bit Register : 1   
[+] 1-bit Register : 2   
28-bit Register : 2   
16-bit Register : 1   
44-bit Register : 1 

##### RAMs (1)

[+] 2048x28-bit RAM : 1 

##### Multiplexers (2)

1-bit 2-to-1 Multiplexer : 2 

##### Report for Multiply-instantiated Identical Subsystems 1 : *[Addr\_offset](matlab:set_param('lab5',%20'hiliteAncestors',%20'none');%20Simulink.ID.hilite('lab5:283'))*

##### Adders/Subtractors (1)

[+] 11x11-bit Adder : 1 

##### Report for Multiply-instantiated Identical Subsystems 1 : *[Read\_Addr\_gen](matlab:set_param('lab5',%20'hiliteAncestors',%20'none');%20Simulink.ID.hilite('lab5:86'))*

##### Adders/Subtractors (1)

[+] 11x11-bit Adder : 1 

##### Registers (2)

[+] 11-bit Register : 2 

##### Multiplexers (1)

[+] 11-bit 2-to-1 Multiplexer : 1 

##### Report for Multiply-instantiated Identical Subsystems 1 : *[Read\_Address\_en](matlab:set_param('lab5',%20'hiliteAncestors',%20'none');%20Simulink.ID.hilite('lab5:162'))*

##### Adders/Subtractors (2)

[+] 11x11-bit Adder : 2 

##### Registers (4)

[+] 11-bit Register : 4 

##### Multiplexers (2)

[+] 11-bit 3-to-1 Multiplexer : 1   
1-bit 2-to-1 Multiplexer : 1 

##### Report for Multiply-instantiated Identical Subsystems 1 : *[Hanning\_window](matlab:set_param('lab5',%20'hiliteAncestors',%20'none');%20Simulink.ID.hilite('lab5:180'))*

##### Registers (2)

[+] 16-bit Register : 2 

##### Multiplexers (1)

[+] 11-bit 3-to-1 Multiplexer : 1 

##### Report for Multiply-instantiated Identical Subsystems 1 : [*Subsystem*](matlab:set_param('lab5',%20'hiliteAncestors',%20'none');%20Simulink.ID.hilite('lab5:68'))

##### Multiplexers (1)

[+] 1-bit 2-to-1 Multiplexer : 1 

##### Report for Multiply-instantiated Identical Subsystems 1 : [*Latch 5*](matlab:set_param('lab5',%20'hiliteAncestors',%20'none');%20Simulink.ID.hilite('lab5:151'))

##### Adders/Subtractors (1)

[+] 5x5-bit Adder : 1 

##### Registers (2)

[+] 5-bit Register : 2 

##### Multiplexers (2)

[+] 5-bit 2-to-1 Multiplexer : 1   
[+] 1-bit 2-to-1 Multiplexer : 1