Implementation of Xilinx DDC Core on SDR-3000

Application Notes

Revision 1.0

April 2004

Table of Content

1 Introduction

1.1 General

A fundamental part of many communications systems is Digital Down Conversion (DDC). Digital radio receivers often have fast ADC converters delivering vast amounts of data; but in many cases, the signal of interest represents a small proportion of that bandwidth. A DDC allows the rest of that data to be discarded, allowing more intensive processing to be performed on the signal of interest. [1]

The DDC is typically located at the front-end of the signal processing conditioning chain, close to the A/D, and is usually required to support high-sample rate processing in the region of 100+ mega-samples-per-second. [2]

DDCs are used extensively in wireless and wireline communication systems. Some typical applications that employ DDCs are

- Software defined radios (SDRs)
- Digital receivers
- Cable modems
- BPSK, QPSK and QAM demodulators
- Spread spectrum communication systems
- CDMA2000 and 3G Basestations

1.2 Scope

In this document, the operation of generating a DDC core using the Xilinx CORE Generator is described in detail. The FPGA and software components developed to integrate the Xilinx DDC core into the SDR3001 system are also described. The example provided follows the data path below:

TM1 ADC0 -> SAND 0 -> PPC7410 Node A

Analog data is first sampled at 80MHz by the TM module. The sampled data is then transferred to SAND 0 where the Xilinx DDC core is implemented. The DDC core decimates the data by 64 before sending it to the PPC7410 Node A processor where the data is captured in a file.

1.3 Required software

- Tornado2.2.1
- Xilinx ISE 6.1.03
- SDR1.53 release

1.4 Required hardware

- Signal generator
- SDR3001 system

2 Xilinx DDC Core

2.1 Concept

The Xilinx DDC Core accepts an input signal sampled at high rate (~100MHz), down converts a desired frequency band-of-interest to baseband and adjusts the sample rate by a factor that ranges from 4 to 1048512 [2]. The Xilinx DDC Core consists of 4 stages: DDS, CIC, CFIR, and PFIR. The parameters implemented for each stage are presented in the sections below.

The table below lists general DDC parameters used.

Parameter	Value
System Clock Rate	80 MHz
Input Sample Rate	80 MHz
Input Data Width	14
Output Result Width	16

2.1.1 DDS (Direct Digital Synthesizer)

By choosing the frequency resolution (Δf) to be 0.2Hz, the number of bits ($B_{\theta(n)}$)used in the phase accumulator is then derived to be 29 bits (range 0x1 – 0x1FFF FFFF) based on equation 5 in [2].

As indicated from equation 1 in [2], the output frequency (f_{out}) is then a function of the sampling frequency (f_s), the number of bits used in the phase accumulator ($B_{\theta(n)}$), and the phase increment value ($\Delta\theta$). The output frequency in Hertz is defined as

$$f_{out} = f_s * \Delta\Phi / 2^{B\theta(n)} \text{ Hz [2]}$$

The table below lists parameters used for the DDS stage.

Parameter	Value
Spurious Free Dynamic Range	40 dB
Frequency Resolution	0.2 Hz
Frequency	Programmable
Phase Angle	None
Mixer Output Width	16

2.1.2 CIC (Cascade Integration/Comb Filter)

A 4-stage CIC filter is selected with a decimation rate of 32.

The gain is the square root of the decimation rate in the CIC Filter. Since the decimation value is set to 32 in the CIC Filter, the gain should be around 6 db. The register mapping for the gain is as follows:

Coarse Gain Value	Gain (dB)
0	0
1	6
2	12
3	18
4	24
5	30
6	36
7	42

Note: The Xilinx DDC core specifications state that the DDC core allows the decimation rate to be programmable but if you actually enable this feature then no data comes out of the core and you never get a valid rdy strobe. Therefore the current core included does not have a programmable decimation rate. Xilinx is aware of this problem and working on it currently.

The table below lists parameters used for the CIC stage.

Parameter	Value
Number of Stages	4
Differential Delay	1
Decimation Rate	32 (Fixed)
CIC Output Width	24
Gain Value	16 (Programmable)
CIC Gain Output Width	16

2.1.3 CFIR (Compensation Filter)

CFIR is not implemented in this example.

2.1.4 PFIR (Programmable Filter)

A symmetric 65 tap PFIR filter is implemented and it results in a decimation rate of 2. The coefficient for the PFIR filter is generated using the Weighted Least Squares design with Matlab. The parameters used to generate the filter coefficient are shown in the table below.

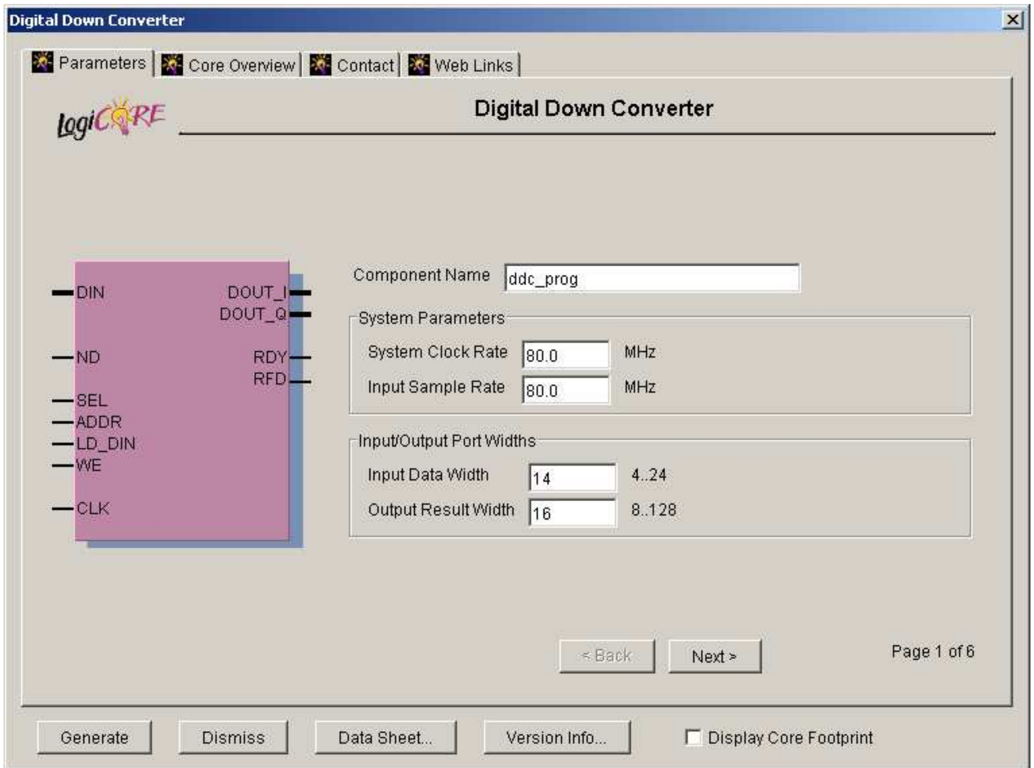
Parameter	Value
Sampling Frequency	80 MHz
Cutoff Frequency	5.5 MHz
Passband Ripple	0.1 dB
Stopband Ripple	-80 dB
Number of taps	65

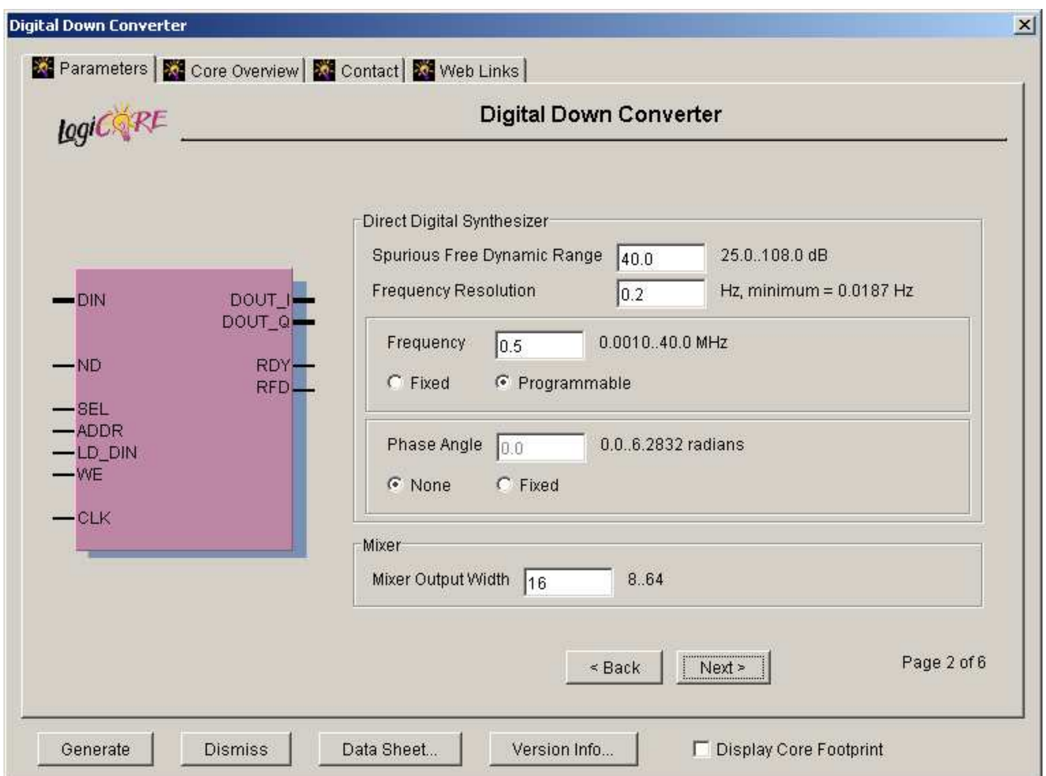
The table below lists parameters used for the PFIR stage.

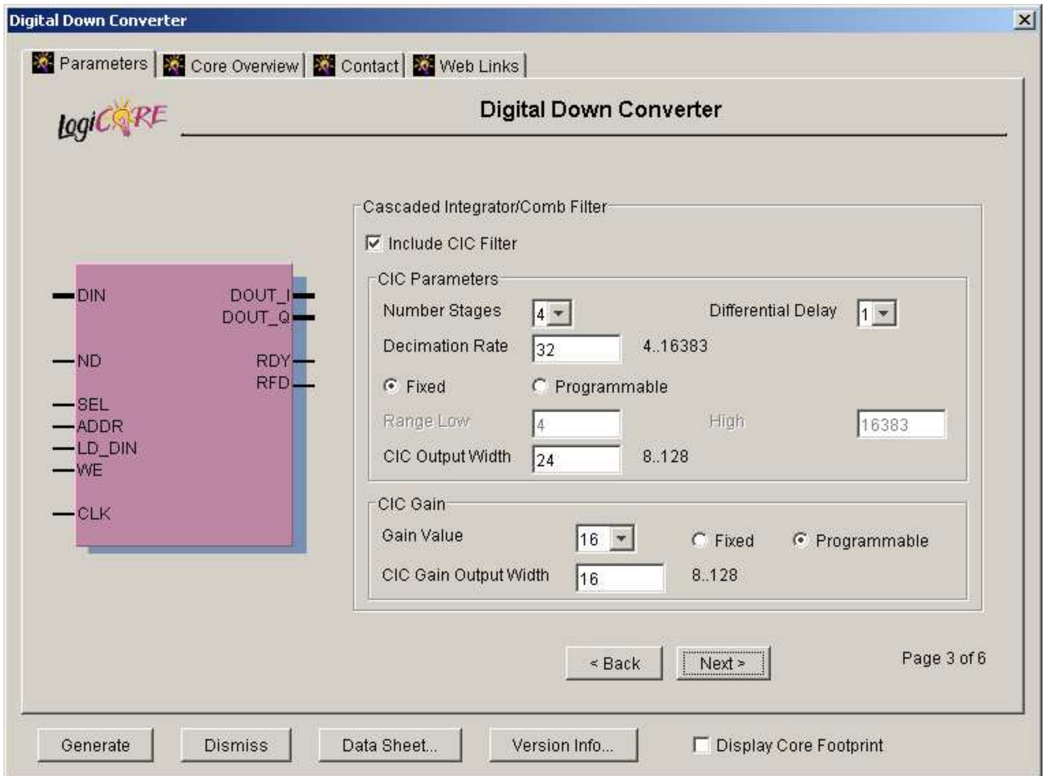
Parameter	Value
Filter Type	Decimation
Decimation Rate	2
Response	Symmetric
Result Precision	16
Filter Length	65
Coefficient Precision	16
Coefficient Filename	fir65.coe

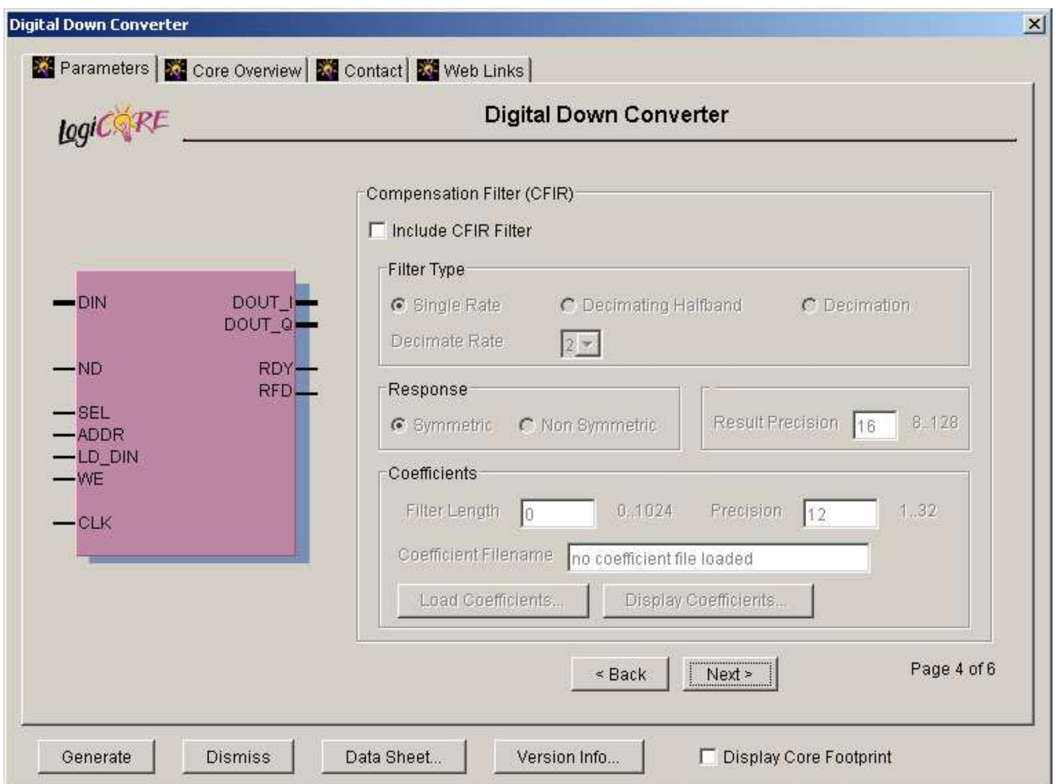
2.2 Generation of the core

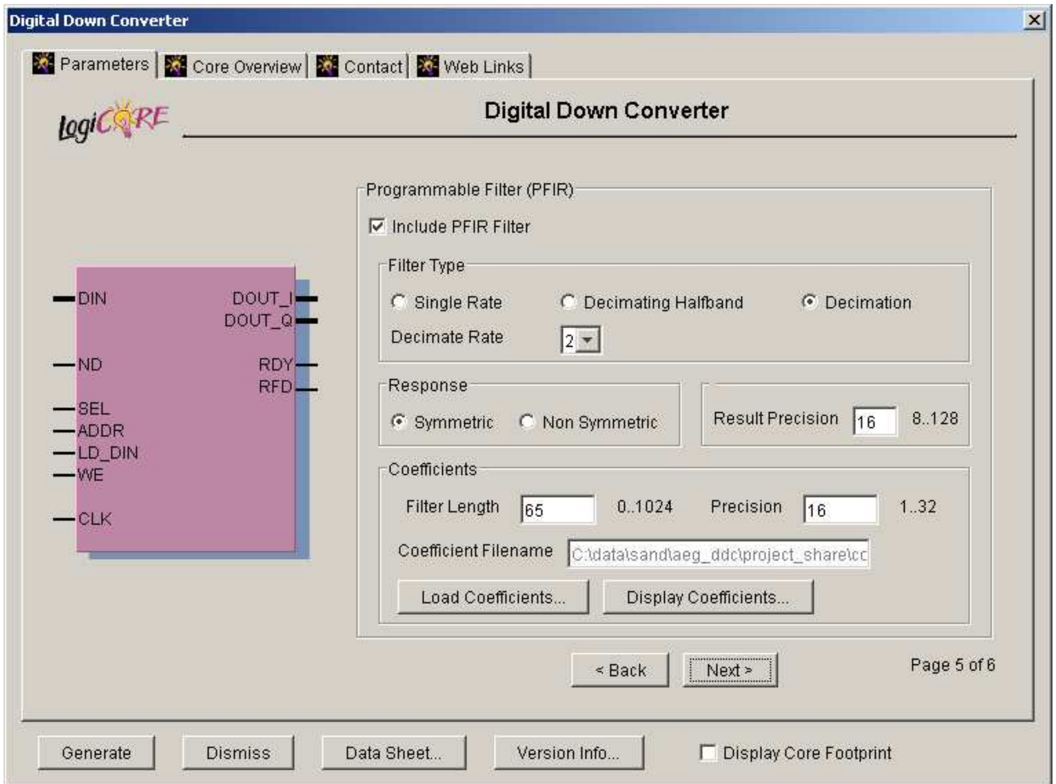
The DDC core can be generated following the parameters listed above and the screen shots below using Xilinx CORE Generator.

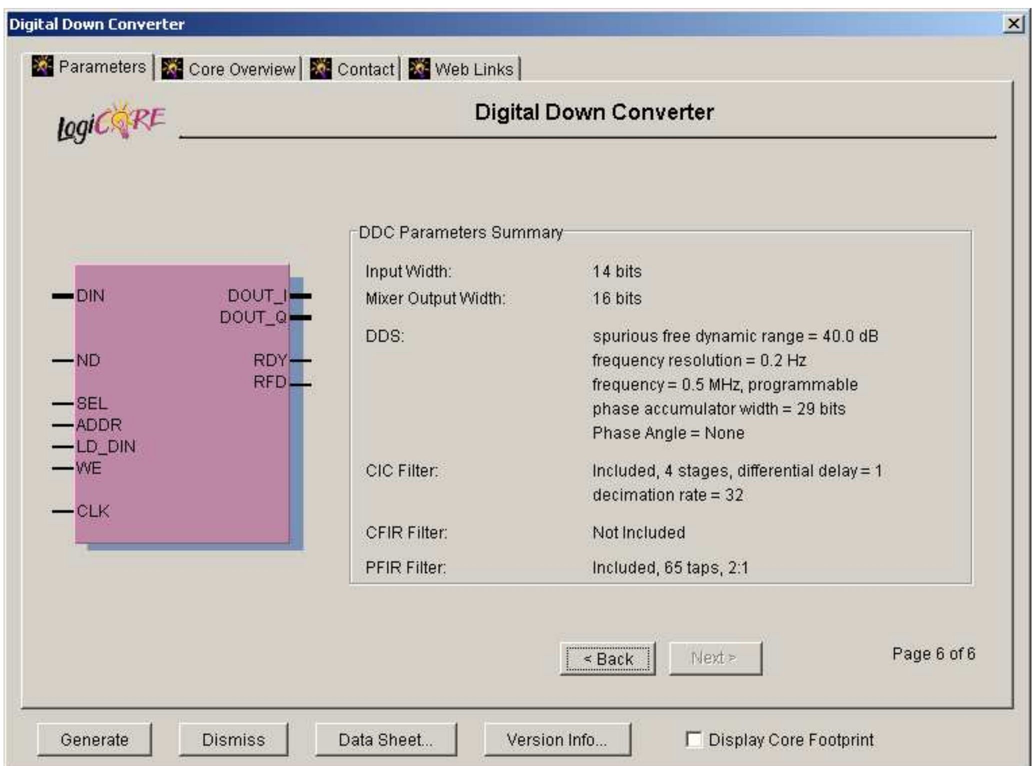

Digital Down Converter
Parameters
Core Overview
Contact
Web Links
LogiCORE
Digital Down Converter
DIN
DOUT_I
DOUT_Q
ND
RDY
RFD
SEL
ADDR
LD_DIN
WE
CLK
Component Name
ddc_prog
System Parameters
System Clock Rate
80.0
MHz
Input Sample Rate
80.0
MHz
Input/Output Port Widths
Input Data Width
14
4..24
Output Result Width
16
8..128
< Back
Next >
Page 1 of 6
Generate
Dismiss
Data Sheet...
Version Info...
Display Core Footprint


Digital Down Converter
Parameters
Core Overview
Contact
Web Links
LogiCORE
Digital Down Converter
DIN
DOUT_I
DOUT_Q
ND
RDY
RFD
SEL
ADDR
LD_DIN
WE
CLK
Direct Digital Synthesizer
Spurious Free Dynamic Range
40.0
25.0..108.0 dB
Frequency Resolution
0.2
Hz, minimum = 0.0187 Hz
Frequency
0.5
0.0010..40.0 MHz
Fixed
Programmable
Phase Angle
0.0
0.0..6.2832 radians
None
Fixed
Mixer
Mixer Output Width
16
8..64
< Back
Next >
Page 2 of 6
Generate
Dismiss
Data Sheet...
Version Info...
Display Core Footprint


Digital Down Converter
Parameters
Core Overview
Contact
Web Links
Digital Down Converter
DIN
DOUT_I
DOUT_Q
ND
RDY
RFD
SEL
ADDR
LD_DIN
WE
CLK
Cascaded Integrator/Comb Filter
Include CIC Filter
CIC Parameters
Number Stages
4
Differential Delay
1
Decimation Rate
32
4..16383
Fixed
Programmable
Range Low
4
High
16383
CIC Output Width
24
8..128
CIC Gain
Gain Value
16
Fixed
Programmable
CIC Gain Output Width
16
8..128
< Back
Next >
Page 3 of 6
Generate
Dismiss
Data Sheet...
Version Info...
Display Core Footprint

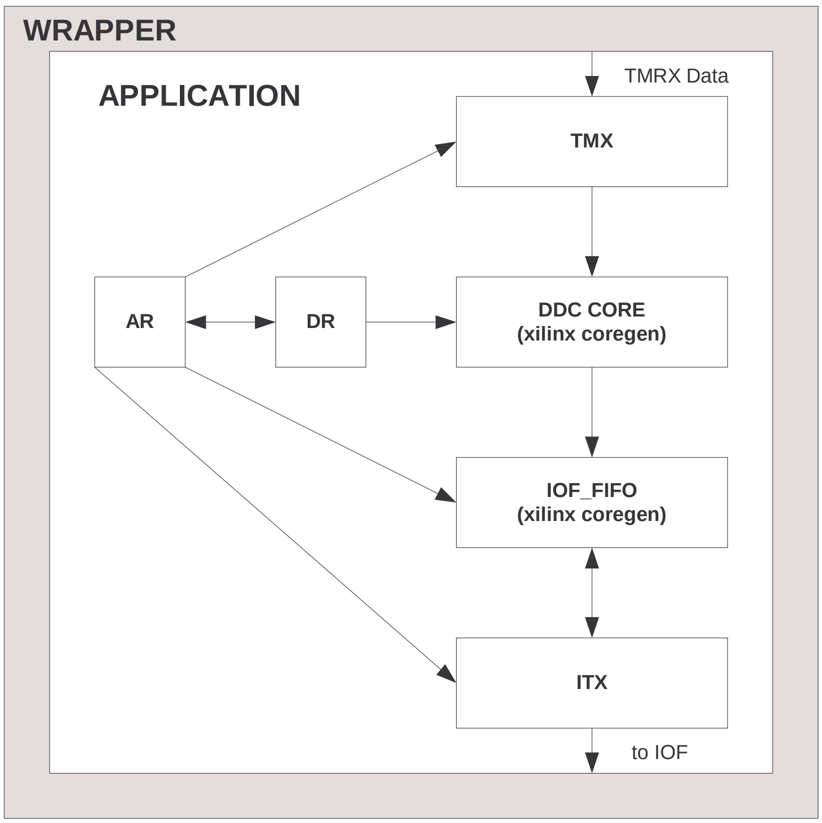

Digital Down Converter
Parameters
Core Overview
Contact
Web Links
Digital Down Converter
DIN
DOUT_I
DOUT_Q
ND
RDY
RFD
SEL
ADDR
LD_DIN
WE
CLK
Compensation Filter (CFIR)
Include CFIR Filter
Filter Type
Single Rate
Decimating Halfband
Decimation
Decimate Rate
2
Response
Symmetric
Non Symmetric
Result Precision
16
8..128
Coefficients
Filter Length
0
0..1024
Precision
12
1..32
Coefficient Filename
no coefficient file loaded
Load Coefficients...
Display Coefficients...
< Back
Next >
Page 4 of 6
Generate
Dismiss
Data Sheet...
Version Info...
Display Core Footprint


Digital Down Converter
Parameters
Core Overview
Contact
Web Links
Digital Down Converter
DIN
DOUT_I
DOUT_Q
ND
RDY
RFD
SEL
ADDR
LD_DIN
WE
CLK
Programmable Filter (PFIR)
Include PFIR Filter
Filter Type
Single Rate
Decimating Halfband
Decimation
Decimate Rate
2
Response
Symmetric
Non Symmetric
Result Precision
16
8..128
Coefficients
Filter Length
65
0..1024
Precision
16
1..32
Coefficient Filename
C:\data\sand\aeg_ddc\project_share\cc
Load Coefficients...
Display Coefficients...
< Back
Next >
Page 5 of 6
Generate
Dismiss
Data Sheet...
Version Info...
Display Core Footprint


Digital Down Converter
Parameters
Core Overview
Contact
Web Links
Digital Down Converter
DIN
DOUT_I
DOUT_Q
ND
RDY
RFD
SEL
ADDR
LD_DIN
WE
CLK
DDC Parameters Summary
Input Width:
14 bits
Mixer Output Width:
16 bits
DDS:
spurious free dynamic range = 40.0 dB
frequency resolution = 0.2 Hz
frequency = 0.5 MHz, programmable
phase accumulator width = 29 bits
Phase Angle = None
CIC Filter:
Included, 4 stages, differential delay = 1
decimation rate = 32
CFIR Filter:
Not Included
PFIR Filter:
Included, 65 taps, 2:1
< Back
Next >
Page 6 of 6
Generate
Dismiss
Data Sheet...
Version Info...
Display Core Footprint

3 Integration of the DDC core

In order to integrate the generated DDC core into a SAND application that receives data from the TM module and transmits the decimated data to the IO Framer, the interface blocks shown in the diagram below between the DDC core and the wrapper were created.



AR	Application Register Block
IOF_FIFO	IO Framer FIFO Block
ITX	IO Framer Transmit Block
TRX	Transition Module Receive Block
DR	DDC Read Write Register Block

3.1 AR – Application Register Block

The application register block interfaces to the DDC core, the Power PC Interface, IO Framer transmit block, and the transition module receive block.

Note: Refer to the SAND Application Development guide for details on how to use the PPC Interface.

This block provides the PPC access to the following registers:

IO Framer Transmit Header (Read/Write)

Address: 0x1000

This register allows the user to select the header to be sent to the IO Framer.

Bit	Description
[31:0]	These bits are routed to ar_itx_header (Default 32’b0)

Transition Module Received Valid Data Count (Read Only)

Address: 0x1008

This register allows the user to read the number of valid data segments that have been received from the transition module.

Bit	Description
[31:16]	Reserved (Default ‘0’)
[15:0]	The value of the number of valid data received

DDC Decimation Rate Register Address (Write Only)

Address: 0x100C

Bit	Description
[31:30]	Reserved (Default ‘0’)
[29: 0]	The decimation rate (valid range 0x4 – 0x3FF)

DDC Phase Increment Register Address (Write Only)

Address: 0x1010

Bit Description

[31:29] Reserved (Default '0')

[28: 0] The phase increment value (valid range 0x1 – 0x1FFF_FFFF)

Output frequency = ((80*10^6)*Phase increment)/(2^29)

DDC Gain Value Address (Write Only)

Address: 0x1014

Bit Description

[31:30] Reserved (Default '0')

[29: 0] The phase increment (valid range 0 - 7)

```
Mapping
0 – 0 db
1 – 6 db
2 – 12 db
3 – 18 db
4 – 24 db
5 – 30 db
6 – 36 db
7 – 42 db
```

Note: Refer to the Xilinx Digital Down Converter V1.0 product specification for more details regarding CIC gain settings.

Port Map

Signal	Size	I/O	Description
clk	1	input	25 MHz PPC Clock
reset	1	input	Reset (synchronous to clk)
ppc_reg_en	1	input	Register enable (qualifies rd/wr signals)
io_pr_ppc_reg_read	1	input	Register read signal
io_pr_ppc_reg_write	1	input	Register write signal
io_pr_ppc_addr	[15:2]	input	Address

io_pr_ppc_data	[31:0]	input	Write Data
reg_read_data	[31:0]	output	Read Data
pr_ap_ab_data_good_count	[15:0]	input	Number of valid 64 bit data words from TM
ar_dr_write	1	output	Register write signal for the DDC core
ar_dr_addr	[4:0]	output	Register address for the DDC core
ar_dr_data	[31:0]	output	Register write data for the DDC core
ar_dr_done	1	input	DDC Core finished with the command
ar_itx_header	[31:0]	output	Header to use when writing to the IO Framer

3.2 FIF0_32x256– IO Framer FIFO Block

This block is written with data from the Transition Module Receiver block and its data is read by the IO Framer transmit block.

The FIFO is 256 words deep and each word is 32 bits long. It is an asynchronous FIFO with the writes synchronized to an 80 MHz clock and the reads synchronized to a 100 MHz clock.

If the full signal ever goes high then there will be an interrupt sent to the wrapper (via the application block).

Port Map

Signal	Size	I/O	Description
ainit	1	input	Reset (synchronous to clk)
din	[31:0]	input	Data into fifo
wr_clk	1	input	80 MHz Sample Clock
wr_en	1	input	Write request
rd_clk	1	input	100 MHz Sample Clock
rd_en	1	input	Read request

dout	[31:0]	output	Data out of fifo
full	1	output	FIFO full
empty	1	output	Not used/left open
wr_count	[7:0]	output	Number of word in FIFO synched to wr_clk
rd_count	[7:0]	output	Number of word in FIFO synched to rd_clk

3.3 ITX – IO Framer Transmit Block

This block reads from the FIFO block and sends the header (ar_itx_header) followed by 15 data words and asserts the write signal to the IO Framer block.

Note that the IO Framer requires 16 word packets. The value of ff_itx_rd_count must be greater then 16 to ensure that there are enough words in the FIFO before starting to transfer the words to the IO Framer interface.

Due to the asynchronous nature of the FIFO, the rd_count and wr_count will not necessarily hold the exact same value at a given point in time. They can be up to 3 cycles off.

Note: Refer to the SAND Application Development Guide for details on how to use the IO Framer Transmit Interface.

Port Map

Signal	Size	I/O	Description
clk	1	input	80 MHz Sample Clock
reset	1	input	Reset (synchronous to clk)
pr_ap_f0_rpf	1	input	Indicates that the IO Framer is ready for a packet
ap_pr_f0_write	1	output	Outgoing data to the IO Framer interface
ap_pr_f0_data	[31:0]	output	Write strobe for the IO Framer interface
ap_pr_f0_data_valid	1	output	Indicates to the IO Framer that the data is valid
ar_itx_header	[31:0]	input	Header to be used when writing to the IO

			Framer
itx_ff_rden	1	output	Read request to the FIFO
ff_itx_rd_count	1	input	Number of words in the FIFO
ff_itx_data	[31:0]	input	Data read from the FIFO

3.4 TRX – Transition Module Receive Block

The transition module receive block receives 32 bit data from the transition module and sends bits 15:2 (14 bits) of the incoming data (LSBs padded with 0's) to the DDC core block. If the ddc_trx_rfd signal ever goes low this is an error condition and an interrupt will be set which is handled at the application level.

Port Map

Signal	Size	I/O	Description
clk	1	input	80 MHz Sample Clock
reset	1	input	Reset (synchronous to clk)
pr_ap_a_data_good	1	input	Data from the Transition Module is valid
pr_ap_a_rx_data	[32:0]	input	The data from the transition module
ddc_trx_rfd	1	input	Indicates that the core is ready for data
trx_ddc_din	[13:0]	output	Data for the DDC core
trx_ddc_nd	1	output	Signals the core that there is new data
trx_rfd_intn	1	output	Indicated that the DDC core is not ready for data which is an error
pr_ap_a_data_good_cnt	[15:0]	output	The number of valid data segments received. (This counter will wrap)

3.5 DR – DDC Read Write Register Block

The DR block receives the address and data from the AR block and a write strobe when it writes to the DDC Core registers. All of the signals are synchronized to the 80 MHz clock and dr_ar_done is asserted when the request is complete.

Port Map

Signal	Size	I/O	Description
clk	1	input	80 MHz Sample Clock
reset	1	input	Reset (synchronous to clk)
ar_dr_write	1	input	Write strobe received from the AR block
ar_dr_addr	[4:0]	input	Address of the DDC register to write to
ar_dr_data	[31:0]	input	Data to write to the DDC Core register
dr_ddc_we	1	output	Write register request to the DDC core
dr_ddc_ldin	[31:0]	output	Data to write to a DDC core register
dr_ddc_addr	[4:0]	output	Address of the register in the DDC core
dr_ddc_sel	1	output	Select the DDC core

4 Software components

This example illustrates the usage of the DDC core within the SDR3001 system. Analog data coming into the TM module is sampled at 80MSPS and passed to the PRO3100 board. The SAND device (SAND 0) that implements the DDC core decimates the data before sending it to the PPC7410 processor on the PRO3500 where the received data is stored in a file.

4.1 PRO3100_405 Software Component

DDC_download_fpga

This function obtains a system handle and loads the FPGA programming files (bit files) into the appropriate SAND devices.

DDC_Config

The DDC_Config function configures the TM1 Flexfabric IO Interface FPGA so that the SAND devices receive data from the TM module. This function also configures SAND 0 with the appropriate IO Framer transmit header, phase increment value, as well as the CIC output gain. It also sets up a channel to transfer the decimated data from SAND 0 to the PPC7410 processor A on the PRO3500 board.

4.2 PRO3500_7410 Software Component

DDC_pro3500

This function sets up a channel to receive the data from SAND 0 and stores it in a file.

5 Result

Connect the signal generator to the RX0 connector on the TM module and run the example. Based on the input sampling rate of 80MSPS and DDC decimation rate of 64 (32 from the CIC stage and 2 from the PFIR stage), the sampling rate out of the DDC core should be 1.25MSPS.

In Figure 1, a 510KHz 1Vpp sine wave is applied from the signal generator. Based on the phase increment value (ΔΦ) of 0x333333, the output frequency of 500KHz is generated from the DDS; therefore resulting in 10KHz I and Q signals. Based on the 1.25MSPS sampling rate, there should be 125 samples per period.

In Figure 2, a 2010KHz 1Vpp sine wave is applied from the signal generator. Based on the phase increment value (ΔΦ) of 0xCCCCCC, the output frequency of 2000KHz is generated from the DDS; therefore resulting in 10KHz I and Q signals. Based on the 1.25MSPS sampling rate, there should be 125 samples per period.

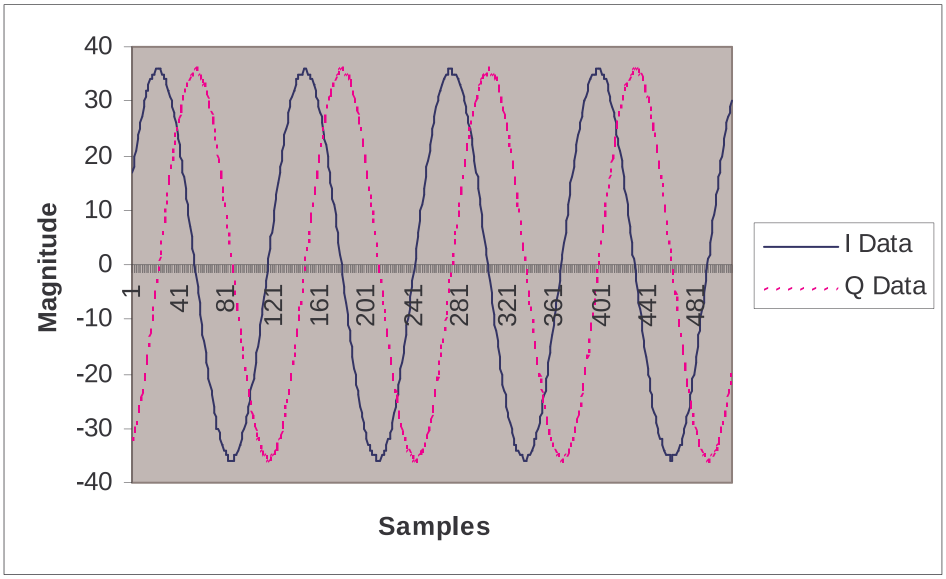


Figure 1: I and Q data (f_{in} 510 kHz, ΔΦ = 0x333333, gain = 1 (6db))

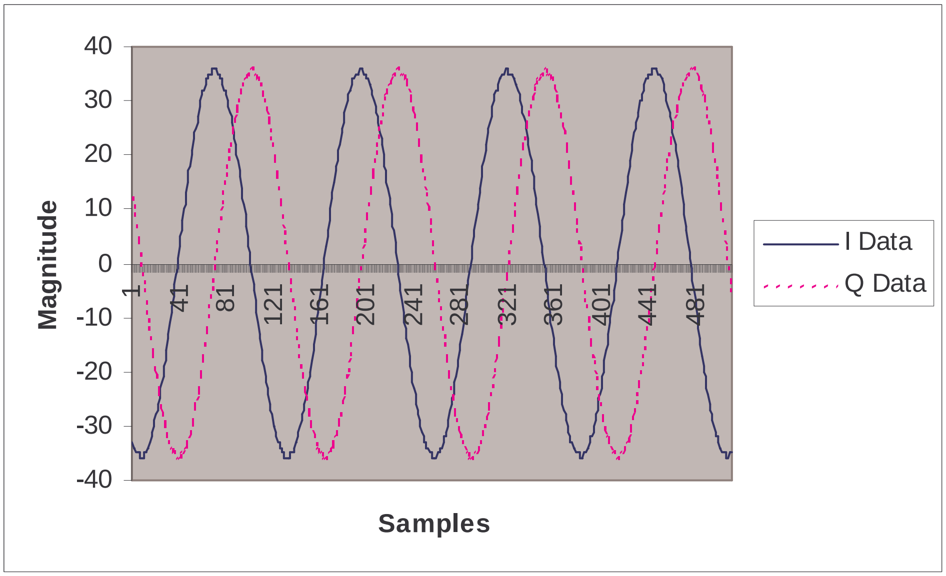


Figure 2: I and Q data (f_{in} 2010 kHz, $\Delta\Phi$ = 0xcccccc, gain = 1 (6db))

6 Reference

[1] http://www.hunteng.co.uk/info/ddctheory.htm Digital Down Converter (DDC) Theory

[2] Xilinx Product Specification; Digital Down Converter V1.0

[2] Spectrum Signal Processing Inc. DDC Project Xilinx Core Development Technical Note

[3] Spectrum Signal Processing Inc. DDC Project Firmware Detailed Design