



UltraLong FFT IP Core (ULFFT)

November 3, 2008

Product Specification

Dillon Engineering, Inc.

4974 Lincoln Drive

Edina, MN

USA, 55436

Phone: 952.836.2413

Fax: 952.927.6514

E-mail: info@dilloneng.com

URL: www.dilloneng.com

Features

- UltraLong algorithm for performing continuous Fast Fourier Transforms (FFTs)
- For transform lengths that exceed on-FPGA memory capacity
 - Up to 4M points using QDR SRAM
 - Up to 64M points using DDR SDRAM
- Any-width fixed- or floating-point data
- Run-time selectable length
- Run-time selectable Forward/Inverse transform mode
- Continuous processing at rate up to Fmax (see Table 1).
 - Data rate of 200MSamples/sec in Virtex-5.
- Natural-order inputs and outputs
- Includes C/C++ bit-accurate model and data generator
 - Model also usable from MATLAB
- Includes Verilog testbench and run scripts

Core Facts	
Provided with Core	
Documentation	User Guide
Design File Formats	ISE Project with EDIF/NGC netlist, Verilog source available for extra cost
Constraints Files	.ucf constraints
Verification	Verilog Testbench, Test Vectors
Instantiation Templates	Verilog
Reference Designs & Application Notes	None
Additional Items	C/C++ Model
Simulation Tool Used	
Aldec Riviera 2008.06	
Support	
Support Provided by Dillon Engineering, Inc.	

Table 1: Example Implementation Statistics for Xilinx® Virtex®-5 SXT-2, Single Precision Float

FFT Length	Fmax (MHz)	External Memory Type ¹	# Mem Banks	Min Mem Size (ea.)	Slice FF ²	Slice LUT ²	IOB ³	BUFG	BRAM ⁴	DSP48E	DCM	Design Tools
2M	200	QDRII SRAM	3	4Mx32	43,705	48,831	474	5	90	385	1	ISE® 10.1.02
16M	175	DDR2 SDRAM	3	32Mx64	59,430	62,434	567	6	459	445	1	ISE® 10.1.02
64M	100	DDR2 SDRAM	3	128Mx64	63,643	66,365	567	6	483	485	1	ISE® 10.1.02
64M	50	DDR2 SDRAM	2	128Mx64	34,968	36,354	425	6	302	249	1	ISE® 10.1.02
64M	50	DDR2 SDRAM	3	256Mx32	61,161	63,716	474	6	282	485	1	ISE® 10.1.02
64M	25	DDR2 SDRAM	2	256Mx32	33,314	34,863	316	6	168	249	1	ISE® 10.1.02

Notes:

1) Assuming QDRII SRAM @ 200MHz, DDR2 SDRAM @ 300MHz.

2) Actual slice count dependent on percentage of unrelated logic – see Mapping Report File for details

3) Assuming all core I/Os and clocks are routed off-chip.

4) Indicates maximum BRAM usage. Substituting distributed RAM and/or built-in FIFOs reduces BRAM count.

November 3, 2008

© 2008 Xilinx, Inc. All rights reserved. XILINX, the Xilinx Logo, and other designated brands included herein are trademarks of Xilinx, Inc.

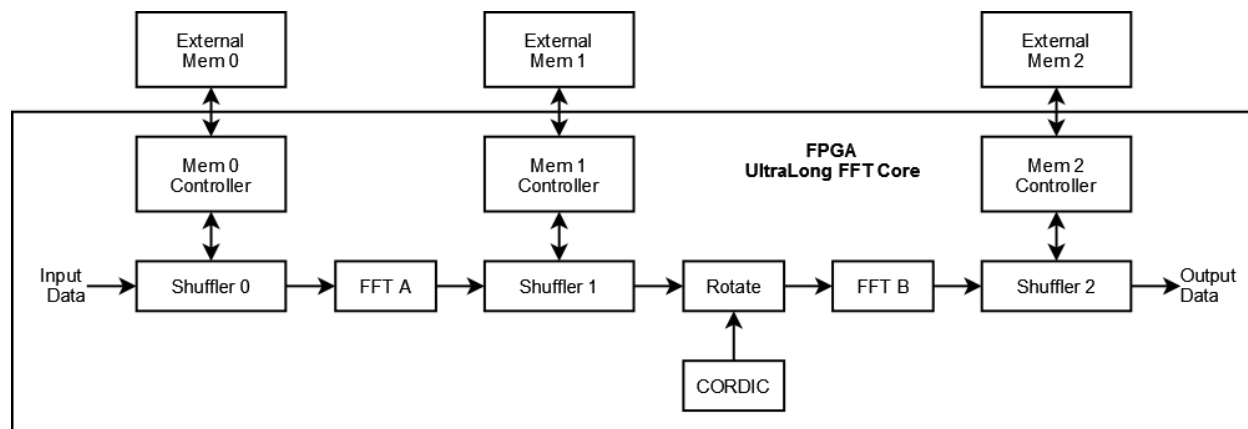


Figure 1: UltraLong FFT Block Diagram

General Description

Dillon Engineering's UltraLong FFT IP Core uses an efficient Fast Fourier Transform (FFT) algorithm to provide multimillion-point discrete transforms on data frames or continuous data streams. This structure utilizes state-of-the-art off-chip memory technology and N1- and N2-length pipelined radix-2 FFT engines with an additional rotation stage to perform $N=N_1 \times N_2$ transform lengths, from 1K to 64M points. The core is available with any width fixed or floating point data. The UltraLong IP Core is easily targeted to current Xilinx FPGA devices and various external memory types.

Functional Description

Shuffler

The Shuffler blocks transpose the N-length data into N1- and N2-length row and column orders, also handling different memory access burst requirements.

Memory Controllers

The Memory Controller blocks write and read data to and from external memories. Memory technologies include QDR and DDR SRAMs and DRAMs supported by the Xilinx Memory Interface Generator.

FFTA and B

The FFT blocks use radix-2 pipelined FFT engines to perform continuous N1- and N2-length transforms.

CORDIC/Rotate

The CORDIC generates complex coordinate multipliers as required by the Rotate block in the UltraLong FFT algorithm.

UltraLong Algorithm

The $N=N_1 \times N_2$ UltraLong Discrete Fourier Transform algorithm follows the form:

$$X[k_1 N_2 + k_2] = \sum_{n_1=0}^{N_1-1} \left[e^{-j \frac{2\pi n_1 k_2}{N}} \left(\sum_{n_2=0}^{N_2-1} x[n_2 N_1 + n_1] e^{-j \frac{2\pi n_2 k_2}{N_2}} \right) \right] e^{-j \frac{2\pi n_1 k_1}{N_1}}$$

Applications

The UltraLong FFT IP Core is useful in High Performance Embedded Computing (HPEC) applications which require continuous Digital Signal Processing (DSP) at high sample rates and long transforms. FFT hardware acceleration or co-processing is often a goal of scientific algorithms used in High Performance Computing (HPC). End applications and markets include radar, sonar, spectral analysis, acoustics, and telecommunications.

Core Modifications

The IP Core is available in netlist or parameterized source code and is customized to support the following:

- Netlist builds for current Xilinx Virtex-5 and Virtex-4 devices. FFT length and speed depend on chip resources, speed grade, and external memory interfaces.
 - Standard builds use 3 independent external memory banks as shown in Figure 1.
 - Alternative builds use 2 or 1 external memory banks and share a single FFT/Rotation engine. The same maximum FFT length applies with any option, but 3 banks support higher continuous data rates compared with 2 or 1 banks.
 - Further customized builds are available to support high-rate (to 200MSps) combined with long-length (to 64M points). These designs require DDR SDRAM for transpose storage and additional QDR SRAM for transpose burst cache.
- Per-transform length selectable in powers-of-2 from 2^{min} to 2^{max} points, with definable $min \geq 10$ and $min \leq max \leq 26$.
- IEEE-754 floating point math operations use Xilinx Coregen floating point cores, which are built separately using Coregen. Thus all trade-offs between speed, number of pipeline stages, DSP48/Mult macro usage, single-, double- or custom-precision float, etc., can be supported.
- Any width fixed-point math operators can be used in lieu of floating point, with options for scaling, rounding and saturation modes, all matched bit-accurate with the C/C++-model. Contact Dillon Engineering for more details.

Core I/O Signals

The core signal I/O have not been fixed to specific device pins to provide flexibility for interfacing with user logic. Descriptions of all standard signal I/O are provided in Table 2, 2a and 2b.

Table 2: Core I/O Signals.

Signal	Signal Direction	Description
CLK	Input	Clock Input. For QDR SRAM designs, this is the source clock for both the datapath logic and the memories. For DDR SDRAM designs, this is the datapath logic clock.
CLK200	Input	200MHz reference clock used by QDR and DDR delay control.
CLK0	Output	(QDR SRAM designs only). This is the CLK0 output of the DCM, matched to CLK.
CLK_REF	Input	(DDR SDRAM designs only). This is the source clock for the SDRAMs.
RST_N	Input	Active-low asynchronous reset. Resets all control logic.

UltraLong FFT IP Core

DIR	Input	Transform mode select. 0 = Forward FFT, 1 = Inverse FFT.
SEL[4:0]	Input	Transform length select. Valid range is from 5'd10 (indicating transform length of 1K) up to the maximum length supported by the build (e.g. 5'd26 for a transform length of 64M).
INIT_DONE	Output	When active, indicates all external memories have completed the PHY init sequence.
SYNC_IN	Input	Input sync strobe. Indicates to the core to begin processing i_data on the following clock cycle.
A[63:0]	Input	Input data. Complex data of the form R + iQ, where R is contained in bits 63:32 and Q is contained in bits 31:0, each a single-precision floating point number.
SYNC_OUT	Output	Output sync strobe. Indicates the core is sending processed o_data beginning on the following clock cycle.
X[63:0]	Output	Output data. Complex data of the form R + iQ, where R is contained in bits 63:32 and Q is contained in bits 31:0, each a single-precision floating point number.

Table 2a: Example I/O for each QDRII SRAM interface, using 2 components of 4Mx18 each.

Signal	Signal Direction	Description
QDRx_CQ[1:0]	Input	QDR read clock
QDRx_CQ_N[1:0]	Input	QDR read clock
QDRx_Q[35:0]	Input	QDR memory read data
QDRx_C[1:0]	Output	QDR read source clock
QDRx_C_N[1:0]	Output	QDR read source clock
QDRx_K[1:0]	Output	QDR write clock
QDRx_K_N[1:0]	Output	QDR write clock
QDRx_SA[19:0]	Output	QDR memory address
QDRx_D[35:0]	Output	QDR memory write data
QDRx_BW_N[3:0]	Output	QDR memory byte enable
QDRx_DOFF_N	Output	QDR DLL disable
QDRx_R_N	Output	QDR read enable
QDRx_W_N	Output	QDR write enable

Table 2b: Example I/O for each DDR2 SDRAM interface, using 4 components of 256Mx8 each.

Signal	Signal Direction	Description
DDRx_CK[3:0]	Output	DDR clock
DDRx_CK_N[3:0]	Output	DDR clock
DDRx_A[14:0]	Output	DDR row/col address
DDRx_BA[2:0]	Output	DDR bank address
DDRx_RAS_N	Output	DDR row select
DDRx_CAS_N	Output	DDR col select
DDRx_WE_N	Output	DDR write enable
DDRx_CS_N	Output	DDR chip select
DDRx_CKE	Output	DDR clock enable
DDRx_ODT[3:0]	Output	DDR on-die termination
DDRx_DM[3:0]	Output	DDR data mask
DDRx_DQS[3:0]	InOut	DDR dqs strobe
DDRx_DQS_N[3:0]	InOut	DDR dqs strobe
DDRx_DQ[31:0]	InOut	DDR data

Critical Signal Descriptions

All data interface and internal operation of the core is synchronous to CLK. Simple SYNC strobes are used on the input and output interfaces to signal that data is valid on the following clock cycle. An active SYNC coinciding with the last data point thus indicates back-to-back transforms. A SYNC_IN strobe active while the core is already inputting data is ignored. Tying SYNC_IN active will signal the core to perform continuous transforms, and SYNC_OUT will strobe as normal to frame the output data.

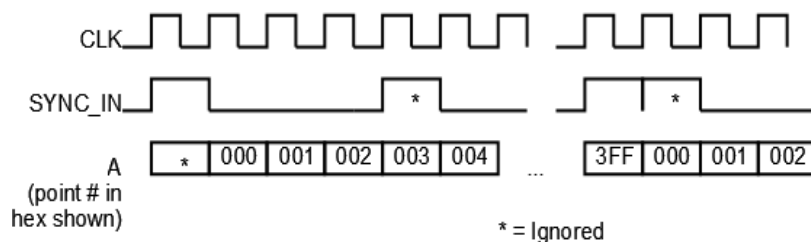


Figure 2: Interface Input Timing, 1K-Length Back-to-Back Transforms

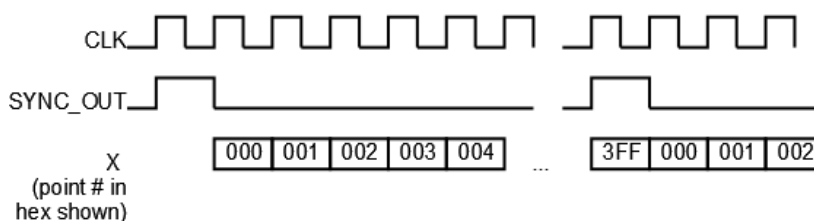


Figure 3: Interface Output Timing, 1K-Length Back-to-Back Transforms

The DIR and SEL configuration inputs are by default selectable per-transform, but must be stable starting with SYNC_IN active and must not be changed until the transformed data has been completely emptied from the core (i.e. 2^m clocks after the corresponding SYNC_OUT).

The user design must wait until INIT_DONE is active before inputting data.

Core Assumptions

Following SYNC_IN, the initial transform has a start-up latency dependent on the three transpose operations and the FFT pipeline latencies for the length of the transform. The core provides continuous processing at steady state, though the SYNC IN to OUT latencies may vary slightly due to internal pipeline alignment and memory interface interruptions such as refresh.

Verification Methods

The core is verified to be bit-accurate with the C/C++ data model under all supported lengths, modes, throughputs and data format, using a rigorous simulation suite of directed and random data. Our model development is evaluated in terms of SQNR with a double-precision floating point software FFT implementation.

Dillon Engineering's FFT IP Cores have been proven over the years in many Xilinx designs.

Ordering Information

This product is available directly from Dillon Engineering, Inc. Please contact Dillon Engineering for pricing and additional information about this product using the contact information on the front page of this datasheet.

Visit www.dilloneng.com/fft_ip to see all of Dillon Engineering's FFT IP offerings, including:

- Pipelined FFTs (single point per clock cycle, fixed or floating point)
- Parallel Butterfly FFTs (continuous FFTs at multiple points per clock cycle)
- Full Parallel FFTs (extremely fast rates, up to 25GSamples/sec)
- 2D FFTs (Two-dimensional transform for image processing)
- Mixed Radix FFTs (for non-power of 2 FFT lengths)

Related Information

Xilinx Programmable Logic

For information on Xilinx programmable logic or development system software, contact your local Xilinx sales office, or:

Xilinx, Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
URL: www.xilinx.com