Datasheet: Variable FFT

Features

- ***High Throughput***: obtained from high clock rates (>500MHz using 65nm technology) and novel algorithms
- ***FFT size***: any user chosen set of power-of-two or non-power-of-two sizes chosen at run-time (examples here are for LTE/WiMax OFDMA using 128/256/512/1024/2048 points)
- ***Programmability***: Finite-state-machine control circuitry for matching circuit/application functionality and I/O interface.
- ***Dynamic Range***: combined block floating point and floating point architecture means smaller word lengths can be used for post-processing operations such as equalization (~6db/bit).
- ***Scalability***: array based architecture means arbitrarily higher throughputs are obtained by increasing array size
- ***Power***: array interconnects are entirely local, reducing parasitic routing capacitance to keep power dissipation low and clock speed high
- ***Implementation FPGA***: Centar's DFT circuit can be used in any FPGA fabric containing embedded multipliers and memories.
- ***Data I/O*****:** Streaming, normal order I/O with fixed-point 2’s complement input words

Options

- Cyclic prefix insertion for LTE or WiMax implementations
- Fixed point word input lengths (2’s complement)
- Output format
  - Fixed-point
  - Block floating-point
  - Floating-point

Algorithm

The transform computation is based on a new formulation[1] of the discreet Fourier transform (DFT), different than any other FFT implementation, which decomposes it into structured sets of small, matrix-based DFTs. In particular the locality, simplicity and regularity of the processing core keeps interconnect delays lower than cell delays, leading to clock speeds that can approach the FPGA fabric limitations, e.g., "worst case" Fmax speeds >500MHz in 65nm FPGA technology. Short critical-path lengths with less delay in cells than interconnects also lower power dissipation. Additionally, a novel "base-4" algorithm reduces the number of cycles needed per FFT to less than the transform size value *N*. Because the circuit has a "memory based" architecture[2], it is programmable so that a range of transform sizes (even non-powers of

[1] J. Greg Nash, "Computationally Efficient Systolic Array for Computing the Discreet Fourier Transform, IEEE Trans. Signal Processing, Vol. 53, No.12, December 2005, pp.4640-4641.
[2] J. Greg Nash, “High-Throughput Programmable Systolic Array
FFT Architecture and FPGA Implementations”, Presented at the 2014 International Conference on Computing, Networking and Communications (ICNC), Honolulu, HI, Feb 2014.

2) can be performed on the same array given adequate memory resources. (Data provided here applies to LTE transform size requirements). Finally, it includes a low overhead hybrid floating-point feature that increases dynamic range for a given fixed-point word size.

Architecture

The architecture consists of two small arrays of pipelined, fine-grained, locally connected, simple processing elements, each containing a complex adder, a few registers and multiplexors. By cycling data through this array in a programmable way any size transform can be supported as well as desired variations from the standard DFT calculation.

Scaling

Word growth during computation is handled automatically using a combination of block floating point (BFP) and floating point (FP) features that provide a much higher dynamic range than other fixed-point FFT circuits with the same input word length. A measure of dynamic range in decibels is the ratio of the magnitude of the largest FFT coefficient and the largest round-off noise value for "single tone" real inputs (random frequency and phase):

	LTE/WiMax OFDMA Transform Sizes				
	128	256	512	1024	2048
Mean	103	105	103	104	105
Std. Dev.	2.7	2.3	4.2	2.2	2.0
Maximum	117	111	115	111	111
Minimum	92	95	96	96	99

Dynamic Range (db) (2000 FFT blocks with 16-bit inputs)

Typically, circuits show almost 6db/bit of dynamic range as defined above.

Signal-to-Quantization-Noise-Ratio (SQNR)

Unlike traditional pipelined FFTs, all additions are performed at full precision so that the round-off errors occur only in the twiddle multiplication steps. Consequently, the SQNR is much higher than found in other FFT architectures for a given input bit length.

	LTE/WiMax OFDMA Transform Sizes				
	128	256	512	1024	2048
Mean	87	87	82	83	81
Std. Dev.	1.6	1.4	0.83	0.84	0.62
Maximum	91	90	85	85	83
Minimum	83	83	79	80	79

SQNR (db) (2000 FFT blocks with 16-bit inputs)

Precision

Using random data, accuracy corresponding to the SQNR results above can be seen from the error summary table below:

Transform Size	128	256	512	1024	2048
Mean	7.63E-05	8.44E-05	1.42E-04	0.000134	0.0001641
Std Dev	0.0002007	0.000111	0.000277	0.000171	0.0005115
Max	0.089456	0.026465	1.82E-01	0.11264	0.62231
Min	0	0	7.96E-08	5.41E-08	2.88E-08

Performance and Resources

Here performance and resource usage data is provided on a 2048-point, 16-bit streaming variable FFT example (123/256/512/1024/2048) for LTE/WiMax. The circuit was compiled using Altera's software tools (Quartus II) and a Stratix III EP3SE50F484C2 FPGA. The TimeQuest static timing analyzer was used to determine maximum clock frequencies (Fmax) at 1.1V and 85C (worst case settings).

	No Cyclic Prefix	Cyclic Prefix
Technology	65nm	65nm
FPGA	Stratix III	Stratix III
Architecture	SA	SA
ALMs/slices	4522	4517
Memory (K)	290	290
Memory (M9K)	42	42
Mults (18-bits)	33	33
Sample Rate (MHz)	510	501
clock (Fmax, MHz)	510	501

In the Table above the adaptive logic module (ALM) is the basic unit of a Stratix III FPGA (one 8-input LUT, two registers plus other logic). Comparison with Xilinx Virtex 5 devices can be made by noting that two M9K memories are equivalent to a Xilinx BRAM and that an ALM is equivalent to between 1.2 and 1.8 LEs, since benchmark studies show 1 ALM=1.2 LEs (Xilinx white paper WP284 v1.0, December 19, 2007) and 1 ALM=1.8 LEs (Altera white paper), respectively.

The memory size (Kbits) indicates the total used memory in the M9Ks and is a measure of how fully utilized they are. Considerable memory savings are possible if streaming operation is not necessary and data can be provided in out-of-order sequence.

Operation at 500MHz has been verified using an Altera Stratix III development kit, which included an EP3SL150F1152C2 FPGA. (The Fmax values were based on the best of ~20 seeds for each circuit).

Device Family Support

Altera Device Families Supported

- Stratix
- Aria

- Cyclone
- Hardcopy

Xilinx Device Families Supported

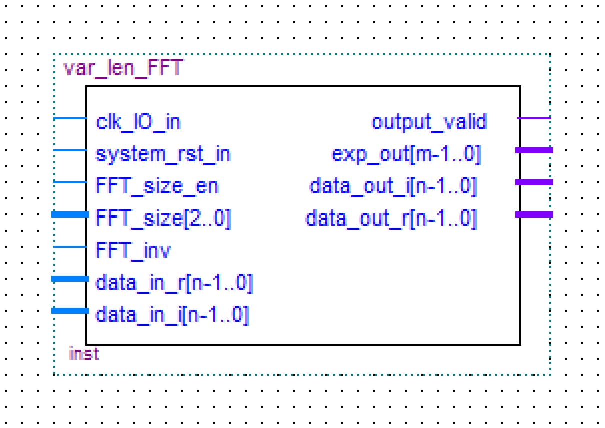
- Virtex 4-7
- Spartan
- Artix-7

Deliverables

- Netlist (e.g., for Altera FPGAs a *.qxp file for synthesis or a *.vo file for simulation)
- Synthesis constraints (e.g., for Altera FPGA's an *.sdc file)
- Modelsim Testbench (*.vo file for DFT circuit plus verilog testbench for control). Matlab verification utilities also available.
- Altera Stratix III FPGA board development kit testbench
- Matlab behavioral bit-accurate model (p-code)
- Documentation

Pin-outs (no cyclic prefix)

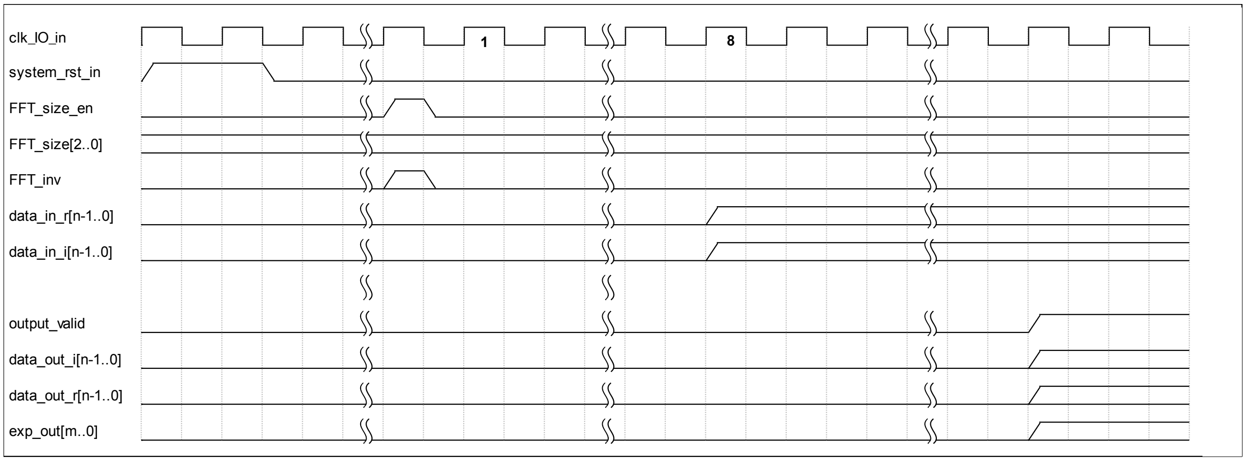
A symbol list corresponding to the pin-outs shown below are provided in the accompanying table. These apply to the "nominal variable" FFT circuit. Depending upon the desired interfaces, some signals could change.



Name	Signal	Description
clk_IO_in	clock	Can be either a low frequency board oscillator clock output in which case circuit clock is derived from a PLL or actual data I/O clock
system_rst_in	control	Resets circuit; active high
FFT_size	unsigned	Specifies transform size
FFT_size_en	control	Registers FFT_size value; active high
FFT_inv	control	High->forward; low->inverse
data_in_r/i	n-bit signed	Real and imaginary inputs
output_valid	control	High during data output
exp_out	m-bit unsigned	Exponents (one per real/imag data pair)
data_out_r/i	n-bit signed	Real and imaginary outputs

Timing Diagram (no cyclic prefix)

A variety of timing possibilities exist depending upon the desired interface. That shown here is applicable to a streaming, normal order input/output scheme.

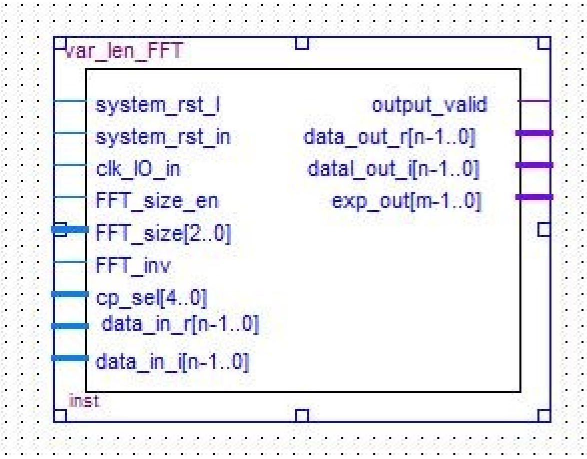


At any time after system_rst_in goes low, FFT_size_en is used to latch both the transform size and the direction of the transform. Following this, the circuit expects to see continuous data appearing on the 8th subsequent cycle. Valid data out occurs when output_valid goes high. For streaming operation output is continuous from this point on.

At any time the FFT transform size and/or direction can be changed by reasserting FFT_size_en for one cycle according to the diagram above.

Pin-outs (with cyclic prefix)

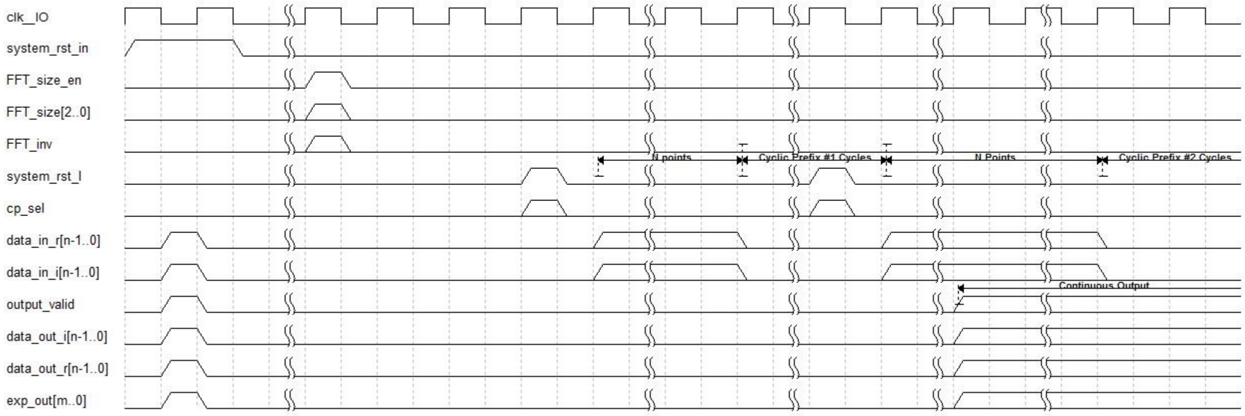
A symbol list corresponding to the pin-outs shown below are provided in the accompanying table. These apply to the "nominal variable" FFT circuit. Depending upon the desired interfaces, some signals could change.



Name	Type	Category	Description
clk_IO_in	clock	input	Can be either a low frequency board oscillator clock output in which case circuit clock is derived from a PLL or actual data I/O clock
system_rst_in	control	input	Resets circuit; active high
system_rst_I	control	input	Data input follows active high
FFT_size[2..0]	unsigned	input	Specifies transform size
FFT_size_en	control	input	Registers FFT_size value; active high
FFT_inv	control	input	High->forward; low->inverse
cp_sel[4..0]	control	input	Specifies cyclic prefix to be used (see table below)
data_in_r/i	n-bit signed	input	Real and imaginary inputs
output_valid	control	output	High during data output
exp_out	m-bit unsigned	output	Exponents (one per real/imag data pair)
data_out_r/i	n-bit signed	output	Real and imaginary outputs

Timing Diagram (with cyclic prefix)

A variety of timing possibilities exist depending upon the desired interface. That shown here is applicable to a normal order input/output scheme. Note that input is not streaming, but output is streaming, due to the use of cyclic prefixes..



At any time after system_rst_in goes low, FFT_size_en is used to latch both the transform size and the direction of the transform. There is a gap in between each block of N inputs, equal to the cyclic prefix time. Valid data out occurs when output_valid goes high. For streaming operation output is continuous from this point on.

At any time the FFT transform size and/or direction can be changed by reasserting FFT_size_en for one cycle according to the diagram above.