Reference DESIGN

High-Performance 16-Point Complex FFT

April 8, 1999 Application Note

Xilinx, Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
FAX: +1 408-559-7114
Email: coregen@xilinx.com
URL: http://www.xilinx.com/ipcenter

Features

- High-performance 16-point complex FFT
- 16-bit complex input and output data
- 2's complement arithmetic
- 16-bit precision internal arithmetic
- Parallel architecture provides a new output sample on every clock
- Naturally ordered input and output data
- High performance and density guaranteed through Relational Placed Macro (RPM) mapping and placement technology

1 Functional Description

The xFFT16 fast Fourier transform (FFT) Core computes a 16-point complex FFT. The input data is a vector of 16 complex values represented as 16-bit 2's complement numbers – 16-bits for each of the real and imaginary component of a datum.

2 Theory of Operation

The discrete Fourier transform (DFT) $X(k), k = 0, \mathrm{K}, N-1$ of a sequence $x(n), n = 0, \mathrm{K}, N-1$ is defined as

$$X(k) = \sum_{n=0}^{N-1} x(n) e^{-jnk2\pi / N} \quad k = 0, \mathrm{K}, N-1 \qquad (1)$$

where N is the transform size and $j = \sqrt{-1}$. The *fast Fourier transform (FFT)* is a computationally efficient algorithm for computing a DFT.

The Xilinx 16-point transform engine employs a Cooley-Tukey radix-4 decimation-in-frequency (DIF) FFT [1] to compute the DFT of a complex sequence. In general, this algorithm requires the calculation of columns or *ranks* of radix-4 butterflies. These radix-4 butterflies are sometimes referred to as *dragonflies.* Each processing rank consists of $N/4$ dragonflies. For N=16 there are 2 dragonfly ranks, with each rank comprising 4 dragonflies.

The FFT processor input-data for the Core is a vector of 16 complex samples. The real and imaginary components of each sample are represented as 16-bit 2's complement numbers. The data input and output buffers are stored internally within the FPGA. The phase factors used in the

FFT calculation are generated within the Core. Like the input-data, the phase factors are kept to a precision of 16 bits.

3 Finite Word Length Considerations

3.1 Scaling

The radix-4 FFT algorithm processes an array of data by successive passes over the array. On each pass, the algorithm performs dragonflies, each dragonfly picking up four complex numbers and returning four complex numbers to the same addresses but in a different memory bank. The numbers returned to memory by the processor are larger than the numbers picked from memory. A strategy must be employed to accommodate this dynamic range expansion. A full explanation of scaling strategies and their implications is beyond the scope of this document, the reader is referred to several papers available in the open literature [2] [3] that discuss this topic.

The Xilinx 16-point FFT Core scales dragonfly results by a factor of 4 on each processing pass. The scaling results in the final output sequence being modified by the factor $1/16$. Formally, the output sequence $X'(k), k = 0,\mathrm{K}\ , N-1$ computed by the Core is defined in Eq. (2)

$$X'(k) = \frac{1}{16} X(k) = \frac{1}{N} \sum_{n=0}^{N-1} x(n) e^{-jnk2\pi/N} \qquad k = 0,\mathrm{K}\ , N-1 \tag{2}$$

4 Pinout

The Core symbol is shown in Figure 1.

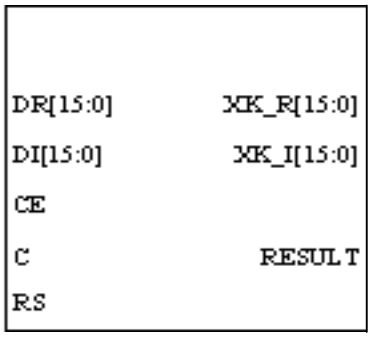

Figure 1: 16-point FFT symbol.

Table 1 defines the pin functionality.

Signal	Direction	Description	Signal	Direction	Description
C	Input	Clock input	START	Input	FFT start control (active high)
RS	Input	Master Reset (active high)			
CE	Input	Clock enable (active high)	RESULT	Output	Transform complete strobe (active high)
DR[15:0]	Input	Input databus – real component	XK_R[15:0]	Output	DFT result – real component
DI[15:0]	Input	Input databus – imaginary component	XK_I[15:0]	Output	DFT result – imaginary component

Table 1: 16-point FFT Core pin definitions.

5 Description

The 16-point FFT Core accepts naturally ordered data on the input buses *DR* and *DI* and performs a complex FFT. These buses are respectively the real and imaginary components of the input sequence. Data must be supplied in a continuous stream to the Core. An internal input data memory controller orders the data into blocks to be presented to the FFT processor. Each block is 16 samples in length. When a new block is assembled it is immediately passed on to the FFT engine. The calculation a complete FFT requires 16 clock cycles. This means that transforms are performed in a block-continuous fashion. The low processing latency allows data to be continuously streamed into the Core. The *START* signal is used to initiate the first transform only. Once *START* has been asserted to begin the first transform there is no need to assert it again – the Core will perform transforms continuously once it has been started. There is an initial 76 clock cycle latency from the time *START* is asserted to the availability of the first valid output sample. This is illustrated in Figure 2.

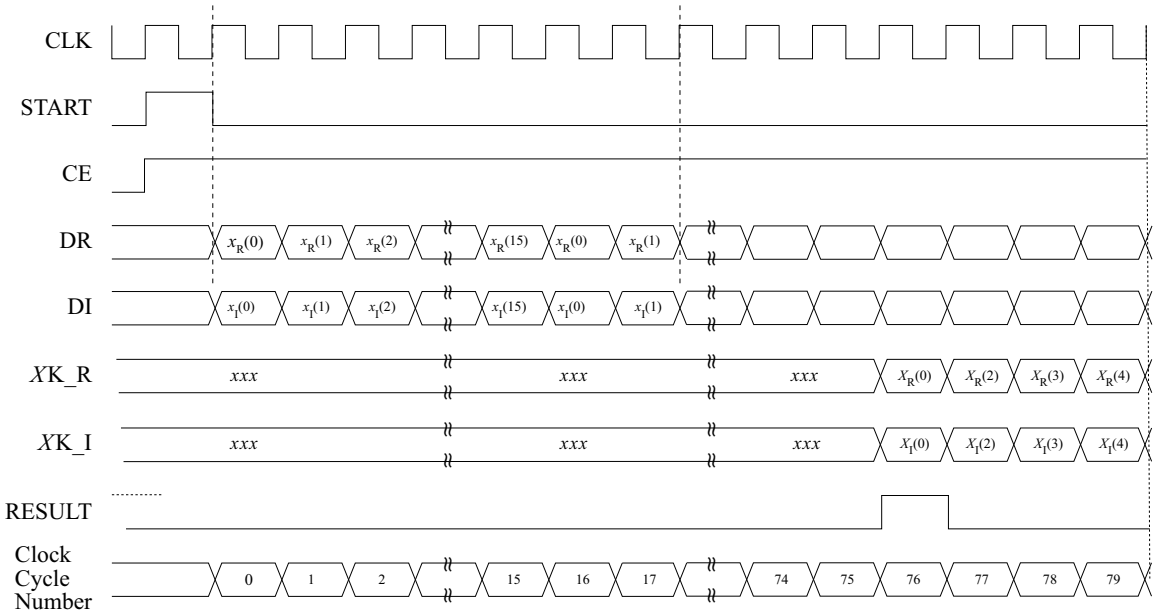

Figure 2: FFT core timing.

The user can elect to assert *START* at any time to re-synchronize the processor to the input data. However, this causes the internal pipeline to be flushed. The result is that each time *START* is applied, the *n* clock cycle startup latency is experienced.

Just as data is continuously streamed into the Core, DFT samples are also continuously streamed out of the Core on the *XK_R* and *XK_I* buses. These buses respectively provide the real and imaginary components of the complex output samples. The *RESULT* signal identifies the start of a transform output vector. Figure x shows the timing relationship between *RESULT* and the *XK_R* and *XK_I* buses. The DFT samples appear in natural order on the output buses starting with XK(0). This ordering is indicated in Figure 2.

5.1 Performance

The complete calculation of 1 16-point FFT requires 16 clock cycles. The transform execution time is

$$T_{\mathrm{FFT}} = \frac{16}{f_{\mathrm{CLK}}}$$

where f_{CLK} is the system clock frequency. Execution times for several clock frequencies are shown in Table 2.

Clock Frequency	52 MHz	58 MHz	60 MHz	74 MHz
FFT Execution Time	307.7 ns	275.9 ns	266.7 ns	216.2 ns

Table 2: FFT execution times for several values of system clock frequency.

5.2 Clock-Enable – *CE*

There are several issues involving the clock-enable *CE* pin that designers should be familiar with when developing systems with this core. *CE* is a high fan-out signal and should be presented to the Core via a low-skew clock buffer, either *BUFGP* or *BUFGS,* to achieve maximum operating frequency. Refer to the Xilinx 4000 series device data book for more information on these features. **The *CE* pin is a master clock enable for the entire Core**. When in the inactive state, all core operations are stalled until *CE* is re-asserted.

6 Core Resource Utilization

The 16-point FFT Core occupies 1520 CLBs. The geometry of the RPM requires it to be placed in a XC4062 or larger device.

7 Ordering Information

This macro is provided free of charge to all Xilinx customers. For additional information contact your local Xilinx sales representative, or e-mail requests to dsp@xilinx.com.

8 References

[1] J. W. Cooley and J. W. Tukey, ``An Algorithm for the Machine Calculation of Complex Fourier Series'',
Math. Comput., Vol. 10, pp. 297-301, April 1965.

[2] W. R. Knight and R. Kaiser, ``A Simple Fixed-Point Error Bound for the Fast Fourier Transform'', *IEEE Trans. Acoustics, Speech and Signal Proc.,* Vol. 27, No. 6, pp. 615-620, Dec. 1979.

[3] L. R. Rabiner and B. Gold, *Theory and Application of Digital Signal Processing,* Prentice-Hall Inc., Englewood Cliffs, New Jersey, 1975.

Xilinx Reference Design License

By using the accompanying Xilinx, Inc. Reference Designs (the “Designs”), you agree to the following terms and conditions. You may use the Designs solely in support of your use in developing designs for Xilinx programmable logic devices or Xilinx HardWire™ devices. Access to the Designs is provided only to purchasers of Xilinx programmable logic devices or Xilinx HardWire™ devices for the purposes set forth herein.

The Designs are provided by Xilinx solely for your reference, for use as-is or as a template to make your own working designs. The Designs may be incomplete, and Xilinx does not warrant that the Designs are completed, tested, or will work on their own without revisions. The success of any designs you complete using the Designs as a starting point is wholly dependent on your design efforts. As provided, Xilinx does not warrant that the Designs will provide any given functionality, and all verification must be completed by the customer.

Xilinx specifically disclaims any obligations for technical support and bug fixes, as well as any liability with respect to the Designs, and no contractual obligations are formed either directly or indirectly by use of the Designs. XILINX SHALL NOT BE LIABLE FOR ANY DAMAGES, INCLUDING WITHOUT LIMITATION DIRECT, INDIRECT, INCIDENTAL, SPECIAL, RELIANCE OR CONSEQUENTIAL DAMAGES ARISING FROM THE USE OF THE DESIGNS, EVEN IF XILINX HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Xilinx makes no representation that the Designs will provide the functionality you are looking for, or that they are appropriate for any given use. Xilinx does not warrant that the Designs are error-free, nor does Xilinx make any other representations or warranties, whether express or implied, including without limitation implied warranties of merchantability or fitness for a particular purpose. The Designs are not covered by any other license or agreement you may have with Xilinx.

The Designs are the copyrighted, confidential and proprietary information of Xilinx. You may not disclose, reproduce, transmit or otherwise copy the Designs by any means for any purpose not set forth in this license, without the prior written permission of Xilinx.

You agree that you will comply with all applicable governmental export rules and regulations, and that you will not export or reexport the Designs in any form without the appropriate government licenses.

XILINX, INC., 2100 Logic Drive, San Jose, California 95124