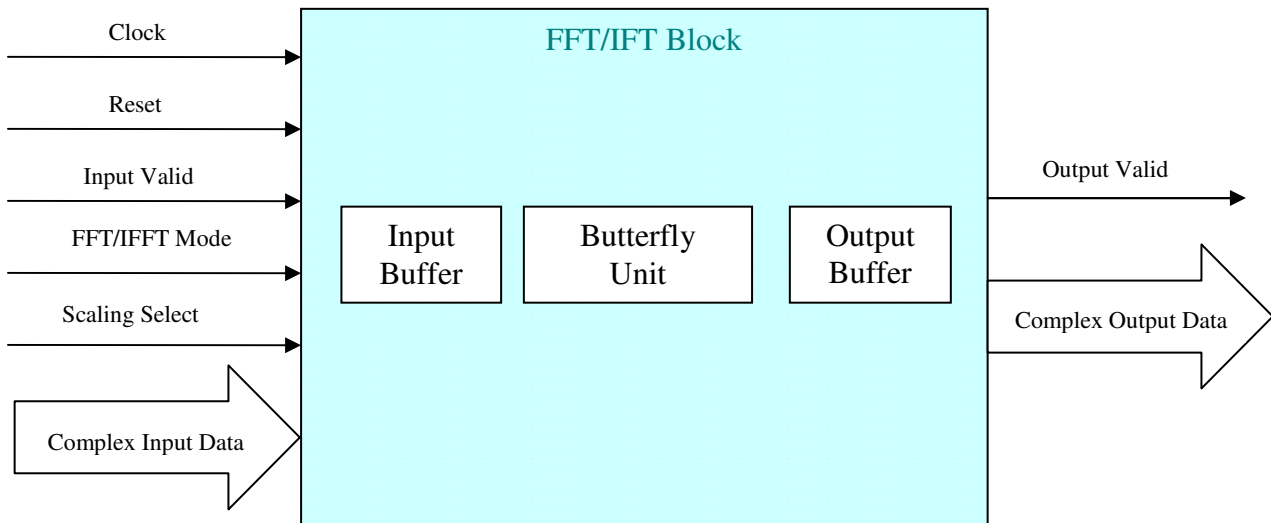# N-Point FFT/IFFT Core

Features

- Parameterisable FFT block size
- Parameterisable input signal width
- Parameterisable internal scaling type (unscaled, scaled on every stage, optimised scaling)
- Programmable input and output word lengths and internal precision
- Area efficient design
- Silicon verified in multiple devices
- Optimized for WLAN (802.11, 802.16), DVB and other OFDM standards

Deliverables

- Netlist or synthesisable RTL source code in VHDL.
- Comprehensive verification test bench and vectors in VHDL
- Integration documentation and user guide

Overview

The Fast Fourier Transform (FFT) is an efficient algorithm for computing the Discrete Fourier Transform (DFT). This Intellectual Property core was designed to offer very fast transform times while keeping the resource utilization to a minimum. Our implementation is a radix-2 architecture. This core is written in VHDL, capable of being used on any FPGA/ASIC architecture.

The FFT/IFFT core is able to perform an N-point FFT/IFFT in approximately 2.4xN clock cycles. It implements a double buffered input stage allowing a block to be processed while another block is being written in.

Signal scaling is controllable by the Scaling Select configuration parameter. It can be either unscaled, scaled by 2 on every stage or optimised scaling.

This core is implemented using a mix of parallel and serial processing for efficient resource utilisation and fast transform times.

Performance[1]

The following is the resource utilisation summary on a Spartan-3E part for 16bits input data and scaling on every stage.

N	Slices	Block RAMs	18X18 Multipliers	Flip Flops
128	1590	14	8	1970
256	1630	14	8	2000
512	1680	14	8	2030
1024	1720	14	8	2060
2048	1830	20	8	2100

Note 1: Resource utilisation as reported by Xilinx ISE synthesiser. Utilisation may vary depending on application. Core clock rate depends on application.