Design of a 16-Point Winograd Fast Fourier Transform Algorithm Integrated Circuit System

T. S. Wailes, W. D. O'Connor, and M. A. Mehalic
Department of Electrical and Computer Engineering
Air Force Institute of Technology
Wright-Patterson Air Force Base, OH, 45433-7765

Abstract

This paper describes the use of VHDL in the specification, design, and development of a large-scale project that includes several custom ASICs, standard digital components, board-level design, and bus interfacing. The common mechanism for the design was the use of VHDL in system testing, behavioral descriptions, structural descriptions, and synthesis. The use of VHDL allowed the project to be completed in one-fifth of the time used for previous methods.

1 Introduction

1.1 Background

The Discrete Fourier Transform (DFT) is widely used to provide frequency analysis of time-domain signals. Computation of the DFT, however, is a complex mathematical process. To achieve the computational speeds necessary, a hardware implementation is often necessary.

The Cooley-Tukey Fast Fourier Transform (FFT) is a high-speed algorithm which implements the Fourier Transform [1]. Although it probably is the most commonly used method, it requires the calculation of $O(Nlog(N))$ multiplications. Fortunately, other algorithms exist to calculate the DFT, including the Winograd algorithm that requires more total arithmetic operations but only $O(N)$ multiplication operations [2]. Since multiplications take longer to compute than addition or subtraction operations, the Winograd algorithm can be used to compute an DFT faster than the traditional FFT algorithm.

The Winograd DFT algorithm is described by the following equation [2]:

$$X = CDAx.$$

In this equation, x is a vector containing the time-domain samples, X is a vector containing the frequency-domain samples, and C,D, and A are the Winograd DFT processing matrices. The A matrix, known as the pre-addition matrix, consists of only (+/-) 1 and 0 [3]. It can be implemented as successive addition and subtraction operations performed on the incoming data points, in a pre-determined order. The D matrix, known as the multiplication matrix, is a diagonal matrix. It can be implemented by multiplying the results obtained from the pre-addition matrix by constant coefficients. The C matrix, or post-addition matrix, is similar to the pre-addition matrix in that its elements consist of (+/-) 1 and 0. The Fourier transform is obtained from the results of post-addition matrix operations.

1.2 WFTA Architecture

While a hardware (vs. software) implementation provides faster DFT computation, flexibility in the size of the DFT calculated is lost. Fortunately, a specific digital signal processing system using the DFT will usually require a specific point or sample size. A point size refers to the total number of data points calculated in a DFT. Consider, for example, a system which senses the voltage of a signal at time t. Sixteen voltage samples are collected and denoted $s(t)$, $s(t+1)$, $s(t+2)$, ..., $s(t+15)$. A 16-point Winograd Fourier Transform Algorithm Integrated Circuit (WFTA16 IC) can be used to provide the DFT of the 16 time-domain samples. If the point size of the DFT desired is known, the C,D and A matrices of the Winograd algorithm can be calculated ahead of time [3]. The resulting arithmetic operations, then, can be hard-coded onto a silicon integrated circuit (IC). This technique produces an IC which is dedicated to computing a specific point-sized DFT. While the IC is only capable of one specific function, it computes this function at a high speed. However, DFTs

of arbitrary size can also be computed by using the Good-Thomas Prime Factor Algorithm (PFA) to combine several WFTA IC processors [4].

WFTA ICs can be combined in a system to provide larger transforms as follows: a WFTA system can complete a Fourier transform of point size equal to each of the processor's point sizes multiplied together. A 4080-point transform, then, would require a 15-point WFTA processor (WFTA15), 16-point WFTA processor (WFTA16), and 17-point WFTA processor (WFTA17) since 4080 = 15*16*17.

1.3 The Use of VHDL

In previous work on this project, VHDL had been used occassionally to try to document the operation of the system. However, the attempts were unsuccessful due to the lack of conviction on the use of VHDL. For this instance, it was decided that VHDL would be used throughout. Using the CAD tools, VHDL would be leveraged as much as possible. This meant the use of top-down design, behavioral VHDL and synthesis for random logic such as control units, structural VHDL and synthesis for regular structures such as data paths, and board-level simulation of standard parts using VHDL. In addition, the use of VHDL allowed some flexibility into a usually inflexible hardware implementation of a DFT algorithm. With structural and behavioral VHDL, different size transforms could be synthesized and arrayed in a matter of days instead of months.

2 Design Methodologies and Tools

WFTA IC and system architectures have been examined in detail since 1985. Originally, the research goal was to design a high-speed pipelined 4080-point DFT system capable of producing transforms in approximately 100 μs [3]. This task involved the design of a 15-point, a 16-point, and 17-point processor, as well as a circuit board.

The initial IC layouts contained various features designed to increase operational reliability, provide extensive observability and controllability, and to provide high precision. These included single error correction and double error detection circuitry, special test buses to permit the isolation of individual components of the WFTA IC, and rounding circuitry to obtain the highest possible precision. Unfortunately, these features also added significant design complexity.

Due to problems encountered while designing the large scale 4080-point system, a 16-point system was designed by Herron in 1992 as a proof of concept [5]. Though specified and simulated in VHSIC Hardware Description Language (VHDL), the physical board was not completed. During this initial design phase of the WFTA ICs, the following design methodology was used:

1. WFTA microcells were first developed in Magic. Magic is a public-domain, Manhattan-style layout development tool and part of the Berkeley Computer-Aided Design (CAD) tool set developed by the University of California at Berkeley (UCB). The microcells used were the low-level WFTA cells, such as adders/subtractors, serial-to-parallel converters, etc.. Many of the cells were built with transmission gates and dynamic logic. This led to fast, compact cells which were difficult to simulate.

2. The low-level cells were then simulated using ESIM (ESIM is a switch level simulator, also developed and distributed by UCB). Alteration of the simulation files was necessary for some cells to permit simulation. Unfortunately, this step was prone to create invalid simulation results.

3. The low-level cells were then interconnected to form WFTA components (Parallel-In / Serial-Out Component (PISO), Multiplication Matrix, etc.) for a specific WFTA IC processor.

4. The WFTA components were simulated using ESIM, if possible.

5. These components were then manually interconnected to implement a specific point-size WFTA processor.

6. If possible, the entire WFTA IC was simulated using ESIM. For an IC of this size, simulation is a difficult task.

7. Finally, the IC was fabricated through the MOSIS fabrication service in a 1.2 μm Complementary Metal-Oxide-Semiconductor (CMOS) process.

The WFTA ICs built using the above bottom-up design procedure were difficult to simulate and validate resulting in multiple failures. The primary focus was always on the speed and compactness of the end circuit, not its functional correctness. Therefore, the research effort was unsuccessful. Because of the lack of success, a simplified approach was attempted for this

research that emphasized building a simple yet fully functional circuit using a top down approach [9]. The steps used are summarized below:

1. A simplified WFTA16 processor was described at the behavioral level with VHDL, using the equations provided in [4] and the operational method described in [6]. It was then simulated to validate its operation.
2. The WFTA microcells were described and simulated in behavioral VHDL according to [6].
3. The microcells were then combined to form structural models of WFTA16 components and the processor itself. These structural models were simulated to validate operation.
4. Using the Synopsys Design Analyzer, the subcomponents were synthesized directly into a structural representation. This structural representation was synthesized to a Magic-compatible CMOS layout representation via the Lager/Octools automated layout generator. If a component could not be synthesized, it was custom designed.
5. The layout representation for each sub-component was simulated using ESIM. This simulation was compared with the VHDL simulation for the same sub-component.
6. The sub-component layout representations were connected together in Magic to form a WFTA16 IC.
7. The layout representation of the WFTA16 was simulated using ESIM. The results of this simulation was compared against the structural WFTA16 VHDL simulation.
8. The layout representation was sent out to MOSIS for fabrication in a 2.0 μm CMOS process.

The end product using this approach was not perfect, however it was close enough to allow a complete WFTA system test with minor corrections to the output obtained. This result was no accident; on the contrary, it was the deliberate application of top-down design techniques that permitted this project's success. We now document the WFTA16 IC fabricated and tested on a VMEbus prototyping board.

The biggest problem was not the complexity of the circuits involved, but the integration of the design tools. A mix of commercial and public-domain university tools was used. These included the Synopsys tools, the Meta-Software HSPICE simulator, and the Berkeley Magic layout editor and ESIM switch-level simulator. Testing was performed on an Integrated Measurement Systems XL-100+ using the TestVIEW software. Unfortunately, the only common link between all these tools were translation programs written in-house, and VHDL. Specifically, the netlist from the Synopsys tools had to be translated into a format for the Octtools/Lager place-and-route software. This translator was written in-house. Test vectors were transfered in ASCII format between the VHDL simulator, ESIM, and TestVIEW. VHDL routines were written to generate the ASCII files in the appropriate format so that the VHDL vectors could be used in the switch-level simulator and the IC tester.

Another difficulty turned out to be the library characterization. The Synopsys tool suite does not have a library for the Lager standard cells. Therefore, this library had to be characterized and compiled in-house. Due to the magnitude of the effort, not all cells were characterized and put into the design library, and those that were characterized had some errors that were discovered later. In addition, VHDL descriptions had to be written for all the library cells. Even wtih these problems, however, the overall process worked.

3 WFTA16 Integrated Circuit

3.1 WFTA16 IC Architecture

The full 16-point complex DFT processor consists of a real and imaginary section, and these sections are identical. Each contains its own parallel-in/serial-out component(PISO), pre-addition matrix, multiplication array, post-addition matrix, and serial-in/parallel-out component(SIPO). The WFTA16 operates in a bit-serial, pipelined fashion. For this research project, however, the real and imaginary sections were contained on separate ICs. A WFTA16 IC block diagram is shown in Figure 1. Two WFTA16 ICs, then, will be needed to compute a complex-valued DFT.

The control circuitry provides the signals necessary to regulate WFTA16 data flow. It also generates the clock signals necessary to clock data on and off chip and through the WFTA16. The control circuit is a simple 32-state finite state machine, cycling through the same control signals every 32 clock pulses. During the 32-clock cycles, 16 input data words are clocked into the WFTA16 and 16 output data words are clocked out of the WFTA16. It receives the off-chip clock signal, as well as three input scaling bits,

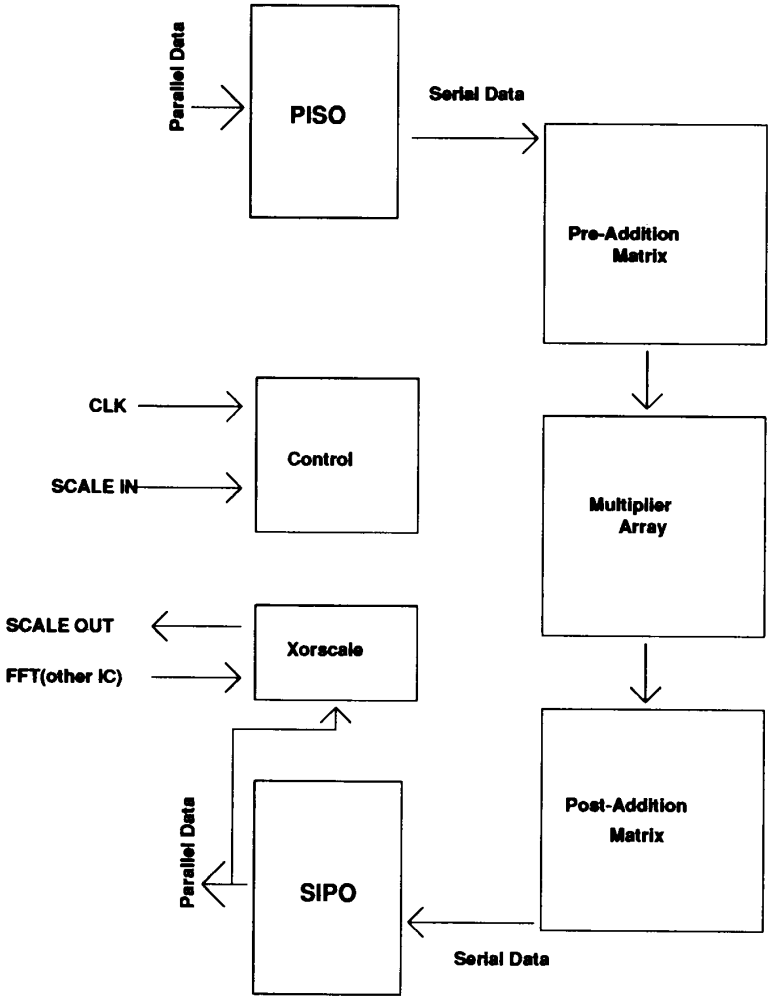

Figure 1: The simplified WFTA16 processor floorplan.

as inputs. At the onset of every cycle, the input scale bits are valid and are latched. Some of the output pulse signal timing depends upon the input scaling bits. The clock signals used on chip are purposely isolated from the off-chip input clock. To ensure synchronous operation, they are generated and latched by the internal control circuitry. This was necessary to reduce internal clock skew, which can cause a problem in the edge-triggered environment chosen for this implementation.

Once the sixteen 16-bit words have been clocked on chip in parallel, these values are latched into the serial register cells. After sending several zeroes into the pre-addition matrix, the 16 serial streams of input data are clocked into the pre-addition matrix, LSB first. The MSB of each serial stream is then sign extended until the next set of 16 inputs is latched from parallel to serial. This process of zero-filling and sign extension causes the 16-bit input words to be extended to 32 bits for the WFTA internal data path [6]. The total number of sign extends and zero fills depends upon the input scaling factor. At the completion of the sign extension the newly-formed 32-bit words are clocked into the pre-addition matrix.

The implementation of the Winograd Transformation is accomplished by serially shifting the data through multiple stages within the pre-addition, multiplier, and post-addition matrices. Each stage is reset by a special control pulse that proceeds the actual data to be transformed. Bit serial adder/subtractor cells perform the needed addition and subtraction functions within the pre and post addition matrices. Special bit-serial Booth multiplier cells perform the required multiplication of the input by constants. A special delay cell is used when multiplication by one is required. Finally, the operation of the post-addition matrix requires communication between the imaginary and real WFTA16 ICs. Certain stages within the post-addition matrix calculations for the real/imaginary WFTA IC produce variables used in the next stage of the imaginary/real WFTA16 IC.

The SIPO operates similarly to the PISO. In fact, the same basic subcells form each component. However, they are used differently. The SIPO converts a serial data stream to a parallel data stream. When 16 valid output DFT values have been clocked into the SIPO, these values are latched from the serial to the parallel path. The serial data, which is clocked in from the post-addition matrix, is in 32-bit format. Because only the sixteen high-order bits of this data are used, the present implementation truncates the 16 least significant bits. Parallel output data is then clocked off-chip out of the SIPO.

The scaling circuitry, called the Xorscale component, communicates information needed to increase the overall signal-to-noise ratio. This information allows the output of one WFTA IC to be properly scaled at the inputs to the next stage of computation, allowing maximum precision of the overall WFTA transform.

3.2 VHDL Design Process

The design of the WFTA16 IC involved many individual steps. However, they all involved the use of VHDL. Specifically, structural VHDL, behavioral VHDL, and VHDL test benches were used throughout. The test benches performed exhaustive simulation on small circuits (less than 4 inputs), and simulated selected functional vectors on larger circuits that were aggregations of smaller circuits. As an example, the next section will give details of the procedure used to design the WFTA16 pre-addition matrix. Although this validation method was used for all the WFTA16 subcells and components, they will not all be described here.

The pre-addition matrix is used to generate the Winograd coefficients necessary to implement a particular DFT [3]. To model the I/O characteristics of the cells required, a behavioral VHDL description of each was written. The only major difference between the cells described by this research effort and the cells in the WFTA standard cell library is the following: the VHDL descriptions assume only one clock, while the previously-designed WFTA standard cells use a ϕ_1/ϕ_2 clocking scheme. A single clock was used to maintain compatibility with the Lager Library, which uses a single-clock negative edge-triggered register. Despite this difference, the the functions of the WFTA subcells and the VHDL descriptions of those cells are identical.

Consider, for example, the addsub cell [6]. It provides both bit serial addition and subtraction functions, with a carry and borrow bit that can be reset. A VHDL behavioral description was used to represent this component. The addition and subtraction components were described by two combinational processes, one performing the addition and the other perfroming the subtraction, while the sequential components were described by a third process which waited on a clock. A synchronous reset was also implemented in this process. Functions were computed locally using variables, while the processes communicated via signals.

Once the addsub behavioral code was designed and tested (using the VHDL debugger), it was given a symbol using SGE. The addsub and the various other

subcell components were then wired together according to the WFTA16 equations from Taylor to form the WFTA16 pre-addition matrix schematic [4]. The pre-addition matrix schematic was then tested at this point using the VHDL debugger. Thus, behavioral VHDL synthesis was combined with schematic capture and structural VHDL netlists. Using the Synopsys Design Compiler, the pre-addition matrix was synthesized as a single component. The Design Compiler maps the pre-addition matrix to a library of pre-defined gates (such as two-input NAND gates, an edge-triggered D flip-flop, etc.). This structural description of the pre-addition matrix was then mapped to a layout via the Octtools/Lager software. This software produces a Magic layout and an ESIM file. Using the ESIM file, the pre-addition matrix transistor layout was tested with the same test vectors that were used to test the VHDL schematic description.

All of the WFTA16 components, including the control unit, were designed in the manner described above. However, some of the components produced by the Octtools/Lager software were too large for practical use. These components were re-designed using a partial custom process. The Magic standard-cell gates from the Octtools/Lager library were wired together to build the layout representation for component subcells. These subcells were wired together to form the desired component. This process was referred to as 'partial custom' because gate layouts (*i.e.* NAND gates, D-flip flops, etc.) were already designed in Magic. The partial-custom version was simulated in ESIM and then compared with the fully synthesized layout simulation to ensure correct functionality. In the WFTA16 which was fabricated, the pre-addition matrix, the post-addition matrix, the control circuitry, and the scaling circuitry were fully-synthesized. The PISO, the SIPO, and the multiplier array were designed in the partial-custom method.

The WFTA16 layout-representation components were wired together in Magic to form the layout of the WFTA16. The WFTA16 was simulated in ESIM against the same test vectors used in the VHDL simulation. Once correct results were obtained, the WFTA16 was sent out to MOSIS for fabrication.

4 Bank-Switching Memory Design

In a pipelined multi-WFTA processor environment, WFTA processors read and write data to and from one another. Since WFTA processors read and write data during each clock cycle, these operations occur simultaneously. It would seem that one WFTA processor could directly read output data from another. However, the order of the output data has to be re-arranged before it is written to the next WFTA processor according to the prime factor algorithm (PFA) [4]. The PFA controller manipulates the order of data from one processor to the next according to the PFA algorithm. The BSM provides data storage between processors such that the order of the data points can be changed. This step is needed when using multiple WFTA-processors to obtain an DFT equal to the product of their individual point sizes, as described above. A BSM is not necessary, then, in a single-processor system. However, it was included in the 16-point DFT system to prove the concept of integrating a BSM with a multi-processor WFTA-based architecture.

The BSM was described behaviorally in VHDL. The storage elements were treated as an array of variables. The control unit for the memory was described as a process within the behavioral description. After the VHDL simulations were shown to work, the process describing the control unit was separated into a component, and the component was synthesized and laid out using the automated tools. It was then combined with the semi-custom layout of the memory array.

Since the BSM is an application-specific IC, it requires an on-chip control circuit. The controller is essentially an address generator and was specified as a simple VHDL behavioral description. Basically, it provides a four-bit counter which cycles through the sixteen addresses needed. This controller was synthesized into a layout representation via the Synopsys Design Analyzer and the Lager/Octtool layout generation tools. The VHDL description for the controller was combined with a description of the RAM memory banks to provide a complete behavioral description of the 16-word BSM. This behavioral was then simulated to ensure correct operation.

After the BSM IC controller and static RAM memory bank cells were created, the Magic design tool was used to wire them together. Next, the entire BSM IC was tested via HSPICE simulation to ensure proper operation. Finally, the 16-word BSM was fabricated in a 2 μm CMOS process through the MOSIS fabrication service in a 64PC46X68 MOSIS standard frame package. As in the case of the WFTA16, all non-synthesized components were partial-custom designed using Lager/Octtool standard cell gates optimized for a 2 μm CMOS process.

5 16-Point Transform DFT board

The 16-point board was designed using the same general methodology as the WFTA 16, with a heavy emphasis on VHDL simulation, validation, and documentation. Thus, the following general steps were used:

1. The Bank-Switching Memory (BSM) VHDL behavioral model and layout representation were designed. As described above, this layout was partially custom, partially synthesized.
2. The DFT board controller, which is needed to interface the 16-point board with the VMEbus, was described behaviorally in VHDL.
3. Symbols for the BSM, the WFTA16, and the board controller were created using the Synopsys Graphical Environment (SGE). These symbols, along with symbols for some standard transistor-transistor logic (TTL) components, were schematically connected to form the 16-point DFT board.
4. Using the VHDL debugger, the DFT system schematic was simulated. Most of the de-bugging was focused upon the board controller behavioral description.
5. Once simulation was correct, a wiring netlist was obtained using the Hierarchy Navigator Tool available via the SGE tools. This netlist was used to wire-wrap the VMEbus-compatible DFT board.

The architecture designed in this research supports pipelining. However, the VMEbus does not support a dual data bus (one for reading data, one for writing data). Only one 32-bit data bus is available. It can be used for either writing to or reading from a slave board during any given cycle. Therefore, since the VMEbus does not support simultaneous reads/writes, the 16-point processor board will not operate in a pipelined fashion. Instead, the board was designed to complete one 16-point complex DFT at a time.

6 Board Architecture

The 16-point DFT board was designed using the WFTA16 IC, the bank-switching memory (BSM) IC, and a FPGA board controller IC. Various transistor-transistor logic (TTL) parts, discrete components, and a VME board were also used to support this project. The board layout was based heavily upon the A24:D16:D08(E0) VME slave memory example shown in [7], extended for 32-bit data transfers. The data path originates from the VME bus, originally feeding the input BSM ICs while the tri-state buffers are in high-impedance mode. Input data is then fed into the WFTA16 processors, after which output DFT data is fed into the output BSM ICs. The output data is then input to the tri-state buffers, which are enabled to drive the VME bus with the output data. The tri-state buffers were used to prevent the output BSM and the VME host processor from writing to the VME data bus simultaneously.

Each component on the board was represented by a behavioral VHDL process, allowing the 16-point system to be simulated. In the case of the board controller, the behavioral VHDL process was, for all practical purposes, the actual IC. The board controller was synthesized, using the Synopsys Design Analyzer, into a gate-level schematic. This schematic was then implemented on a Xilinx XC4003A FPGA, providing a direct mapping from behavioral VHDL to physical IC.

6.1 16-point DFT Board Controller

The 16-point DFT board controller was designed to support a VME interface to the WFTA16 IC processor. Therefore, it does not support a pipelined interface. However, since a re-programmable FPGA was used, its configuration PROM can be replaced. The new configuration PROM can be programmed to support pipelined operation, if needed.

The controller operates according to four separate, sequential cycles. Each cycle is specified by a separate sequential process contained in the board controller VHDL behavioral model. These cycles are:

1. VME host processor writes 16 real/imaginary data points to input BSM ICs
2. Real/imaginary input BSM ICs write to real/imaginary WFTA16 ICs
3. Real/imaginary WFTA16 ICs write to real/imaginary output BSM ICs
4. VME host processor reads 16 real/imaginary outputs from real/imaginary output BSM ICs

6.2 Programming the FPGA

The Xilinx 4003APG120 Logic Configuration Array (LCA) was programmed according to a VHDL behav-

ioral description. The steps necessary to program the FPGA are listed below.

1. A valid, synthesizable VHDL description was obtained. This VHDL description was simulated fully before attempting to go any further.

2. Using the Synopsys Design Analyzer, the VHDL description was synthesized into a structural representation, and then mapped to a Xilinx-compatible FPGA gate library. At this point, a Xilinx part is specified.

3. The design files were transferred to the Xilinx XACT system for implementation. This process is explained in Xilinx reference manuals [8].

7 Results

7.1 WFTA16 IC Results

Thirty-two WFTA16 ICs were received from the MOSIS fabrication service. These 2.0 μm CMOS ICs were fabricated in a 84P79X92 MOSIS standard frame. They were bonded to a 84-pin pin grid array (PGA) package. Testing was accomplished via the IMS ASIC Verification System (tester) in the VLSI Laboratory. Yield obtained was 53%, and maximum operating speed was 15 MHz.

Two main tests were conducted, these being a 1 Hz sine wave input test and a DC signal test. For each test, sixteen input points were required. The sixteen inputs of the DC test were all set to the maximum allowable positive input value for this test (7FFF in hex, or 1.0 in decimal). This should produce an impulse function for the output of the DFT. This impulse function should contain the maximum allowable output value (7FFF in hex, or 16.0 in decimal) at a frequency of zero (output value #1) and zeros for all other outputs. The expected DFT results were obtained from Matlab, a data-analysis software tool, for a 16-point DFT.

The control output vectors from the IC testing matched VHDL and ESIM results exactly. However, the expected DFT output for the WFTA16 IC in the DC test did not match the actual output. Instead, the output result was shifted one binary point to the right; the MSB of the output data word was invalid. By shifting the output data word one place to the left, and zero-filling the LSB, the output result would have been correct. One-bit of precision, however, will be lost.

Clock skew in the WFTA16 IC is the expected cause of the errors present in the output DFT. In fact, a race between the PISO component serial-path and zero-fill section is suspected, due to the errors present in the output DFT. The PISO serial path receives one signal as its master clock; this signal drives 256 edge-triggered flip-flops. The zero-fill section, however, receives a different signal as its clock; this signal drives about 1750 edge-triggered flip-flops. Due to the large load difference between the two clock signals, the PISO serial path LSB sometimes over-writes the zero-fill section.

Problems were the result of incomplete library characterization. The Lager standard-cell library is a public-domain library typically used by universities. Only some of the components have been functionally characterized and integrated into the Synopsys Design Compiler. The timing and load characterization was not complete at the time of this project. Therefore, gate loading and drive characteristics were not fully implemented in the library. The addition of buffer cells, which are in the Lager library, to the Design Compiler library, as well as better characterization of the capabilities of the cells already in the library, should have averted the clock skew problem.

7.2 BSM Results

Twelve BSM ICs were received from the MOSIS fabrication service. They were bonded into a 65-pin PGA package. Testing was accomplished via the IMS tester in the VLSI Laboratory. Yield obtained was 100%, and test results matched the VHDL and HSPICE simulations. Maximum operating frequency was found to be 25 MHz. Only on-board address generation was tested, as this was the part required for proper operation of the BSM to support requirements of the the 16-point DFT board.

7.3 VMEbus DFT Board Results

A series of functional test vectors was developed using the VHDL simulation and knowledge of the operation of the board. Due to the fact that the simulator platform had no access to the chassis housing the board, the test vectors had to be transfered manually. This limited the number of test vectors that practically could be run, and caused a heavy reliance on the accuarcy of the simulator for detailed testing and debugging. The following tests were performed to assess board performance.

- Real positive DC test. The maximum positive value was written to the real section of the board.

This test vector was also applied to the WFTA16. Imaginary inputs were set to zero.

- Real negative DC test. The maximum negative value was written to the real section of the board. Imaginary inputs were set to zero.

- Real square wave test. A square wave (maximum positive value for 4 inputs, zero for the remaining 12 inputs) was written to the real section of the DFT board. Imaginary inputs were set to zero.

The DC test was designed to test the WFTA16 IC independently, and ensure the DFT board functioned correctly. The square wave test assessed whether inter-IC communication between the real and imaginary post-addition matrices was functioning correctly, since the DFT of a square wave contains both real and imaginary values.

When tested with Block Fill and Memory Display commands, the VMEbus-compatible DFT board functioned as specified in VHDL. It is evident that the WFTA16 IC functioned differently on the 16-point DFT board than it did on the IMS tester. The clock-skew related errors seem to be greatly diminished on the DFT board. In the real positive DC test, as well as the real square wave test, no errors occur. The expected output, obtained from VHDL simulations, matches the actual output values. However, an error does occur with the real negative DC test. It is suspected that superior power conditioning and a reduced clock rate (purposely only 3.6 MHz) on the DFT board is responsible for the decrease in clock skew errors within the WFTA16. On the IMS tester 84-pin PGA board, there is only one $10\mu F$ capacitor between Vdd and GND. On the DFT board, there is one $1\mu F$ capacitor between Vdd and GND. However, there is also one 0.1 μF capacitor for each of the four Vdd/GND pairs of the WFTA16 IC. Although clock skew errors have been decreased on the DFT board, it still presents a problem.

8 Accomplishments

Complete top-down VHDL documentation and validation was the most successful aspect of this research effort, in that all simulations of subcells, ICs, and board-level descriptions were found to have correct results. Other WFTA ICs can be structurally defined using the same subcells developed during this research. This will speed up the development process, as well as standardizing all WFTA ICs.

The VLSI design phase of this project was strongly driven by VHDL documentation. The VLSI circuits should be seen only as an implementation of the VHDL specifications. While the VHDL specifications are more or less permanent, the VLSI implementation of those specifications may change. Changes may be implemented due to another engineer's design philosophy, or new fabrication technologies.

The VHDL design efforts also provided a direct mapping from behavioral VHDL to physical IC in many cases resulting in a great savings of design time. Consider, for example, the VME board controller. It was programmed entirely onto a Xilinx FPGA via a behavioral description, requiring almost no physical design work.

The 16-point DFT board was designed to be VHDL-documented at all levels. This provided two main advantages:

1. The board-level description was compatible with IC VHDL descriptions. Therefore, simulation of the board at all levels was possible before physically constructing it.

2. The controller was generated and programmed onto an FPGA via behavioral VHDL. In the past, the control logic was designed using standard TTL components. This process is time consuming and requires a large amount of area on a board. On the 16-point DFT board, the FPGA controller is contained on a 120-pin PGA socket. The only other circuitry required is two on-board PROMs to program the FPGA, a delay line, and two TTL flip-flops. The greatest advantage is that, if an error is found, the behavioral VHDL description can be re-synthesized. The corrected structural description can then be programmed onto a new PROM. By replacing the old PROM, the corrected controller will be implemented. This is a highly efficient and inexpensive way to correct errors or make upgrades.

The DFT board was shown to function as specified in VHDL. Although the board functioned correctly, the WFTA16 IC still exhibited clock skew errors for certain test vectors, as was explained earlier in this paper.

9 Conclusions

The WFTA16 based research recently completed was discussed and assessed in this paper. The VHDL documentation and validation accomplished was the

most important aspect of the research, as it provided a method of WFTA design and simulation via standard parts. Although functional, the VLSI components designed to support this research effort still need to be improved; areas of improvement include clock skew reduction, speed enhancement, and smaller power consumption. In the area of board design, the use of a FPGA as a re-programmable controller proved to be an especially efficient design method.

Acknowledgements

This work was supported by the Department of the Air Force, Rome Laboratory (RL/OCTS), Griffiss Air Force Base, NY.

References

[1] J. Cooley and J. Tukey, "An Algorithm for the Machine Computation of Complex Fourier Series," *Mathematics of Compututation*, vol. 19, pp. 297–301, November 1965.

[2] S. Winograd, "On Computing the Discrete Fourier Transform," *Mathematics of Compututation*, pp. 175–199, January 1978.

[3] K. W. Scribner, "Redesign, Simulation, and Development of a High-Speed, 4080-Point Winograd Fast Fourier Transform Processor," Master's thesis, AFIT/GCE/ENG/91D-08, Air Force Institute of Technology (AU), 1991.

[4] K. Taylor, "Architecture and Numerical Accuracy of High Speed DFT Processing Systems," Master's thesis, AFIT/GE/ENG/85D-47, Air Force Institute of Technology (AU), 1985.

[5] J. Herron, "Design and Development of a High-Speed Winograd Fast Fourier Transform Processor Board," Master's thesis, AFIT/GCE/ENG/92D-05, Air Force Institute of Technology (AU), 1992.

[6] P. W. Coutee, "Arithmetic Circuitry for High Speed VLSI Winograd Fourier Transform Processor," Master's thesis, AFIT/GE/ENG/85D-11, Air Force Institute of Technology (AU), 1985.

[7] W. D. Peterson, *The VMEbus Handbook*. 10229 N. Scottsdale Road, Suite B. Scottsdale, AZ 85253: VFEA International Trade Association, 3 ed., 1993.

[8] Xilinx, 2100 Logic Drive, San Jose, CA 95124, *XACT Development System*, 1993.

[9] W. D. O'Connor, "Design of a 16-Point Winograd Fast Fourier Transform Algorithm Integrated Circuit for use in a 16-Point VMEbus-Compatible FFT Board," Master's thesis, AFIT/GE/ENG/93D-30, Air Force Institute of Technology (AETC), 1993.