Efficient Sum-of-Product Based Constant Multiplication for FFT Applications

Joseph Wanstrath and Travis LaCour
Vinay Karkala, Graduate Student
Dr. Sunil Khatri, Professor

Introduction

Abstract

•N-point Fast Fourier Transform (FFT) in VLSI
- –Requires multiplication of data by one of N constant terms.
  - –Multiple Constant Multiplication (MCM)
- –Past implementations have used multiple cascaded adders.
  - –Results in larger delay and area

•We perform MCM for FFT using a Sum-of-Product (SOP) based realization
- –Minimize delay and area
- –Utilize time-based compression
- –Final addition is executed by a hybrid adder

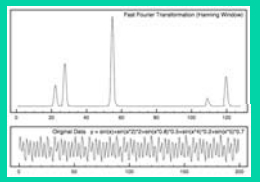
•Compared with best known technique/paper
- –Lower area for high-resolution FFTs
- –Lower delay for all compared FFTs

Background

•The Fast Fourier Transform (FFT)
- –Calculates the frequency components of a signal

•Uses:
- –Digital Signal Processing
- –Circuit Testing (Oscilloscopes)
- –Wireless Internet (Wi-Fi) Systems
- –Voice Recognition Systems

•How to Calculate the Transform
- –Multiply the input by N constants
  - sin(N), 0 < N < 360°
- –Calculate the sum of the products



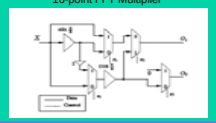
$$X(k) = \sum_{n=0}^{N-1} x(n) \cdot \left(\cos\left(\frac{2 \cdot \pi \cdot k}{N}\right) - j \cdot \sin\left(\frac{2 \cdot \pi \cdot k}{N}\right) \right)$$

Previous Work

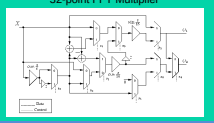
Fahad Qureshi and Oscar Gustafsson. (2009). *Low-Complexity Reconfigurable Complex Constant Multiplication for FFTs.* Linkoping, Sweden. IEEE Xplore.

•Uses trigonometric identities to reduce the number of multiplications and additions
•Multiplexors create necessary coefficients
•Requires several adder circuits to create one output term
•The architecture must be rebuilt for any increase in the number of coefficients

16-point FFT Multiplier

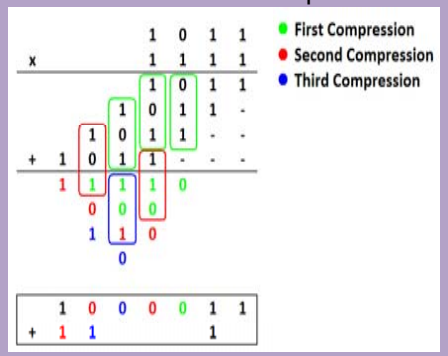


32-point FFT Multiplier

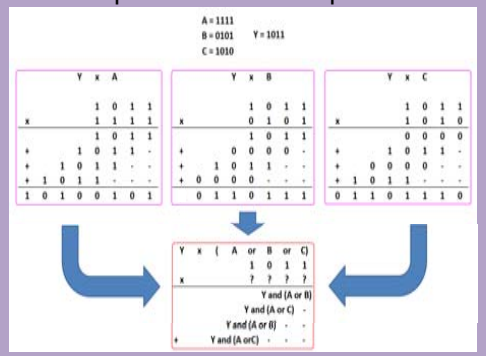


Our Approach

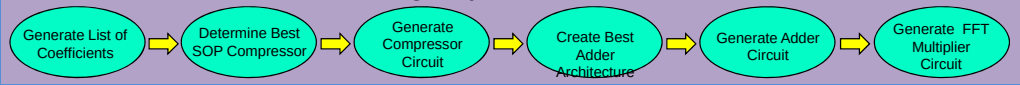
Sum-of-Products Compression



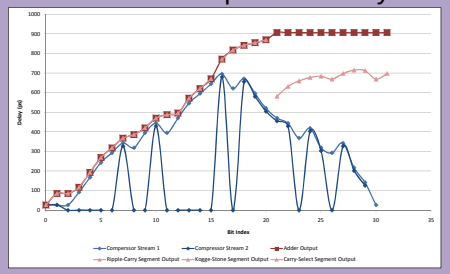
Multiple Constant Multiplication



Logic Synthesis Flow



Adder and Compressor Delays



Hybrid Adder

•Ripple-Carry Adder (Least Significant Bits)
- – Advantages
  - – Little hardware necessary
  - –We have the time to allow propagation of "carry" bits
- – Disadvantages
  - –Linear increase in delay for each bit

•Kogge-Stone Adder (Middle Bits)
- – Advantages
  - – Fastest adder structure available
  - –Minimizes the penalty at the bits with the highest delays from the compressor
- – Disadvantages
  - – Hardware intensive architecture, large amount of wiring

• Carry-Select Adder (Most Significant Bits)
- – Advantages
  - – Generates output before "carry-in" bit is ready", no critical delay penalty
  - – Ensures that the middle-bits determine critical delay of the hybrid adder
- – Disadvantages
  - –Contains 2 Kogge-Stone Adder blocks, very hardware intensive

Results

Experimental Setup

•Experimental data was calculated using 65nm technology libraries
•Calculated delay based on capacitance load at each logic gate

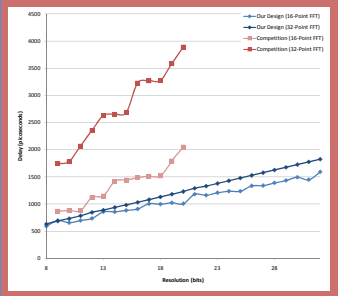
Our Design
• Delay and Area were calculated exactly based on the circuits used in our architecture
• All circuit elements were considered when calculating Area

Comparative Design
• Calculations represent a lower-bound conservative estimate of the architecture
• Delay considers only the number of adders claimed in the research
  – Ignores multiplexors
• Area only considers the number of adders claimed
  – Ignores multiplexors and other adders used in the architecture

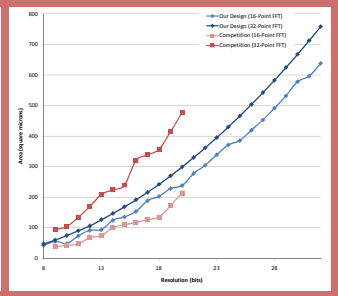
Future Work

•Include decoder analysis to determine Partial Product Terms for coefficients
• Implement and compare architectures on a FPGA to verify results
• Consider the use of 4:3 SOP Compression algorithm

Area Comparison



Delay Comparison



Conclusion

•Our design shows significant improvement in delay of the multiplier circuit
- – 30% delay improvement when used for a 16-Point FFT
- – 60% delay improvement when used for a 32-Point FFT

• Our design shows improvement in area for larger values of N
- – 33% larger area when used for a 16-Point FFT
- –35% area improvement when used for a 32-Point FFT

• Our SOP based algorithm is only dependent on bit-size
- – Architecture can be changed to different values of N without changing circuit
- – The architecture can be scaled to larger bit sizes with no change in structure

References

[1] Fahad Qureshi and Oscar Gustafsson. (2009). *Low-Complexity Reconfigurable Complex Constant Multiplication for FFTs.* Linkoping, Sweden. IEEE Xplore.
[2] Das, Sabyasachi. *Design Automation Techniques for Datapath Circuits* [Ph.D thesis]. Boulder: University of Colorado at Boulder; 2007.
[3] Parent, D.W. "Kogge Stone Adder Logic Verification." Electrical Engineering Lecture –Design of CMOS Digital Integrated Circuits. 2008. San Jose State University.
[4] Weste, Neil H.E., David Harris, Ayan Banerjee. 2005. *CMOS VLSI Design: A Circuits and Systems Perspective, 3rd Edition.* Delhi, India: Pearson Education.

Department of Electrical and Computer Engineering
Texas A&M University
College Station, TX 77843-3128

Electrical Engineering Research Applications to Homeland Security
National Science Foundation Research Experiences for Undergraduates (NSF REU)