DSP for FPGAs

Training Objectives

This three-day course will review DSP fundamentals from the perspective of implementation within the FPGA fabric. Particular emphasis will be given to highlighting the cost, with respect to both resources and performance, associated with the implementation of various DSP techniques and algorithms. Topics include:

- Introduction to FPGA hardware and technology for DSP applications
- DSP fixed-point arithmetic
- Signal flow graph techniques
- HDL code generation for FPGAs
- Fast Fourier Transform (FFT) Implementation
- Design and implementation of FIR, IIR and CIC filters
- CORDIC algorithm
- Design and implementation of adaptive algorithms such as LMS and QR algorithm
- Techniques for synchronisation and digital communications timing recovery

Prerequisites

MATLAB® Fundamentals and *Simulink® for System and Algorithm Modeling*

Products

- MATLAB
- Simulink
- DSP System Toolbox™
- Signal Processing Toolbox™
- Communications System Toolbox™
- HDL Coder
- MATLAB Coder
- Fixed Point Designer™

Course Outline

Day 1 of 3

Introduction to DSP FPGA Hardware (1.0 hrs)

Objective: Provide introduction to DSP and FPGA. Understand general FPGA architecture and why FPGAs are uniquely suited to the implementation of DSP algorithms.

- From discrete logic to FPGAs -some history!
- The generic DSP system
- DSP cores and processors review
- Custom and semi-custom ASICs
- System-on-chip (SOC)
- FPGA flexibility and functionality
- FPGAs vs Programmable DSPs

Linear Systems DSP Algorithm Review (0.5 hrs)

Objective: Review fundamental concepts of sampling theorem, quantization, Fourier analysis and digital filter design.

- Aliasing and reconstruction filters
- Sampling rates and wordlengths
- Z-domain notation and fundamental analysis
- FIR and IIR filters
- Digital filter design and specification
- Oversampling techniques (sigma delta)

FPGA Technology (1.0 hrs)

Objective: Explore different Xilinx FPGA families and architectures. Provide introduction to Spartan 3 and Virtex-5 FPGAs.

- The FPGA technology roadmap
- Clocking rates, data rates and sample rates
- FPGA memory and registers
- Input/output blocks and requirements
- Bits, Slices and Configurable Logic Blocks
- Comparable MIPs Performance Ratings
- FPGA Families and Sources

FPGA elements for DSP algorithms (1.5 hrs)

Objective: Understand DSP slices, clocking resources and power consumption.

- Building delay lines and Shift Registers
- Use of RAM (memory) on FPGAs
- Serial to Parallel and Parallel to serial
- Multiplexors for channel selection
- Full adders, carry logic, and adder trees
- Multipliers: Shift and Add; ROM based
- Efficient multiplier implementation

DSP Arithmetic Essentials (1.0 hrs)

Objective: Understand fixed point binary arithmetic. Map arithmetic operations to Xilinx FPGA hardware.

- 2's complement fixed point arithmetic
- Fundamental adders and multiplier arrays
- Division and square root arrays….not so easy!
- Wordlength issues and Fixed point arithmetic
- Saturate and wraparound
- Overflow and underflow

Signal Flow Graph (SFG) Techniques (2.0 hrs)

Objective: Review the representation of DSP algorithms using signal flow graph. Use the Cut Set method to improve timing performance. Implement parallel and serial FIR filters.

- DSP/Digital Filter Signal Flow Graphs
- Latency, delays and "anti-delays"!
- Re-timing: Cut-set and delay scaling
- The transpose FIR
- Pipelining and multichannel architectures
- SFG topologies for FPGAs

Day 2 of 3

Frequency Domain Processing (1.0 hrs)

Objective: Discuss the theory and FPGA implementation of the Fast Fourier Transform.

- DFT, FFT and IFFT
- FFT FPGA architectures
- FFT wordlength growth and accuracy

Multirate Signal Processing for FPGAs (4.0 hrs)

Objective: Develop polyphase structure for efficient implementation of multirate filters. Use CIC filter for interpolation and decimation.

- Upsampling and interpolation filters
- Downsampling and decimation filters
- Efficient arithmetic for FIR implementation
- Integrators and differentiators
- Half-band, moving average and comb filters
- Cascade Integrator Comb (CIC) Filters (Hogenauer)
- Efficient arithmetic for IIR Filtering

CORDIC Techniques (2.0 hrs)

Objective: Introduce CORDIC algorithm for calculation of various trigonometric functions.

- CORDIC rotation mode and vector mode
- Compute cosine and sine function
- Compute vector magnitude and angle
- Architecture for FPGA implementation

Day 3 of 3

Adaptive DSP Algorithms and Applications (2.0 hrs)

Objective: Introduce LMS algorithm in adaptive signal processing. Illustrate QR algorithm as a Recursive Least Squares (RLS) technique and why it is particularly suited to FPGA implementation.

- Adaptive applications (equalisation, beamforming)
- LMS Algorithms and parallel implementation
- Non-canonical LMS algorithms
- Linear algebra; solving linear systems of equations
- The QR algorithm for adaptive signal processing
- QR processing requirements and numerical issues

DSP Enabled Communications and FPGAs (3.0 hrs)

Objective: Review quadrature modulation and pulse-shaping. Discuss implementation of numerically controlled oscillators.

- Quaternary Phase Shift Keying (QPSK)
- Transmit/Receive Filters - Root Raised Cosine
- Undersampling and Digital Downconversion
- Direct digital upconversion
- Digital IF stages (and fs/4 Systems)
- Numerically controlled oscillators (NCO)

- Design partitioning for FPGAs

Timing and Synchronisation Issues (2.0 hrs)

Objective: Cover symbol timing recovery, carrier phase recovery, carrier frequency recovery and frame synchronization.

- Carrier recovery, squaring and Costas loops, PLLs
- Phase rotations; Sampling rate conversions
- Symbol timing recovery, early/late gate detection
- Delay locked loop timing and synchronisation