Outline: *System Development and Programming with the ADSP-21160*

Course Name:	System Development and Programming with the ADSP-21160
Course Number:	ADST-120
Course Description:	This is a practical course with 'hands on' training using the latest VisualDSP++ software development tools. First the core elements of the processor, which includes the Computational Units, the Data Address Generators, and the Program Sequencer, are examined in detail along with the relevant assembly code instructions. A number of simulator labs help in understanding operation of the individual elements. Memory configuration (both internal and external) is discussed next. Advanced instructions are presented with a follow on lab on code optimization. The I/O peripherals, which include the SPORTS, Link Ports, and External Port, are discussed in detail along with DMA operation between these peripherals and internal memory. This section also deals with system booting and SBSRAM interfacing. Hardware development tools, such as evaluation boards and ICE's are introduced. Throughout the course, the various aspects of the software development process using the latest tools are discussed including setting up and building projects, assembly language programming, code debugging, simulation, tool support for code overlays and shared memory, and 'C' programming support.
Goals/Objectives:	The main course objective is to understand the architecture of the ADSP-21160 DSP sufficiently to enable DSP system designers to resolve hardware/software issues with their applications. Additional goals include gaining a thorough understanding of both assembly language programming and code development (including 'C' programming issues) with the latest software tools
Pre-requisites:	Previous embedded microprocessor background would be an asset (hardware and/or software)
Target Audience:	System Designers needing to make informed decisions on design tradeoffs, Hardware Designers needing to develop external interfaces, and Code Developers needing to know how to get the highest performance from their algorithms
Target Duration:	3.5 days

3 The ADSP-21K Family Core Architecture - Part I

3.1 The ADSP-21000 Family Core Internal Architecture

3.1.1 Registers and Data Types

3.1.2 Register types: UREG, SREG, DREG - overview

3.1.3 Register File

3.1.4 Native data types and data word alignment

3.1.5 Fixed point - 32 bit integer, fractional

3.1.6 Floating point - 32 bit single precision, 40 bit extended single precision

3.1.7 Simulator Exercise: registers exercise, basic simulator operation

3.2 ALU

3.2.1 Features

3.2.2 Instructions

3.2.3 Flags

3.2.4 Simulator Exercise: ALU operation

3.2.5 Mini-Quiz

3.3 Multiplier/MAC

3.3.1 Features

3.3.2 Instructions

3.3.3 Flags

3.3.4 Fractional and integer math

3.3.5 Simulator Exercise: MAC operation

3.3.6 Mini-Quiz