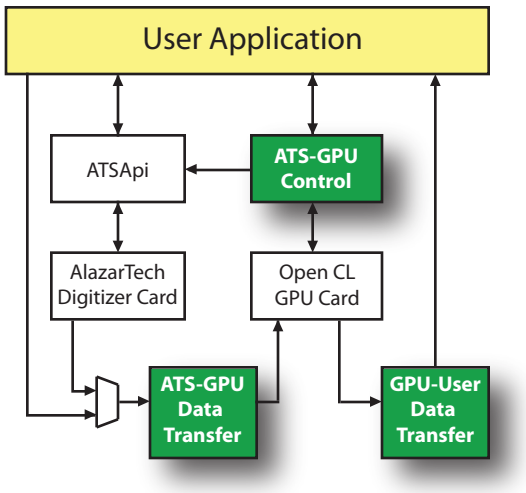
ATS-GPU
Real Time Signal Processing Software

- Use GPU to perform signal processing on acquired data
- Supports GPUs with Open CL drivers
- Designed to work with AlazarTech PCI Express waveform digitizers
- 2048 point FFTs at up to 235 KHz trigger repeat rate for one channel
- Gapless, streaming 1 Million point FFTs at up to 460 MS/s for one channel
- Includes sample programs in C/C++, LabVIEW and MATLAB
- Compatible with 32/64 bit Windows 7/XP



Product	GPU Compatibility	Operating System	Throughput to GPU	Example Programs	FFT Length	Max. FFTs Per Second	FFT Type
ATS-GPU	Open CL v1.0	Windows 7, Vista & XP 32/64 bit	Up to 1.6 GB/s	C/C++, LabVIEW, MATLAB	2048 4096 1 Million	Up to 235,000 Up to 118,000 Up to 460	Floating point with windowing

Overview

ATS-GPU is a software framework developed by AlazarTech to allow users to do real-time data transfer from its PCI Express waveform digitizers to an Open CL compatible Graphical Processing Unit (GPU) at rates up to 1.6 GB/s.

Modern GPUs include very powerful processing units and a very high speed graphical memory bus. This combination makes them perfectly suited for signal processing applications.

Unfortunately, GPUs do not allow other hardware devices, such as waveform digitizers, to DMA data directly to the GPU's on-board memory.

This forces users to manually copy data from the buffer returned by the waveform digitizer to the GPU. This copying process is relatively slow and causes the overall data throughput to be drastically reduced.

ATS-GPU solves this problem by transfering data to and from the GPU using highly optimized software routines implemented at the kernel level of the operating system and assisted by hardware.

The floating point FFT routines built into ATS-GPU have also been optimized to provide the maximum number of FFTs per second. Kernel code running on the GPU can do zero-padding, apply a windowing function, do a floating point FFT, calculate the amplitude and convert the result to a log scale.

It is also possible for users to supply their own data to the GPU for signal processing. In this mode, overall throughput is dependent on the rate at which users can supply data to the GPU.

ATS-GPU also includes source code of the software framework required to transfer data from a waveform digitizer to a GPU and from the GPU to user application. Users can use this framework to create their own GPU-based analysis function.

Typical ATS-GPU Application

A typical user application that uses ATS-GPU consists of the following minimum sections:

1) Setup waveform digitizer hardware parameters
   This includes input range, coupling and impedance, trigger parameters etc.

2) Setup waveform digitizer data transfer method and download GPU kernel code
   This involves selecting whether the waveform generator will capture a gapless, continuous stream of data or a triggered acquisition.
   Most imaging applications use No Pre Trigger (NPT) triggered acquisition, whereas most radio applications use continuous streaming

3) Start data capture
   User can initialize other system components before starting the capture

4) Wait for result buffer(s) to be ready

5) Consume result buffer
   For highest performance, make sure data consumption is faster than the rate at which result buffers are supplied by ATS-GPU.

6) Repeat steps 4 and 5
   Repeat until all result buffers have been consumed or the application has to be closed

ATS-GPU Data Flow

ATS-GPU includes Windows driver code and matching GPU kernel code that allows acquired data to be transferred to the GPU at very high speed.

Similarly, once the GPU is finished processing the data, ATS-GPU delivers the result in data buffers that a typical user application can access.

All these details are masked from the user of ATS-GPU, resulting in a high performance system that is easy to program.

Performance Dependencies

Since the host CPU is involved in moving data to and from the GPU and in scheduling GPU kernels, CPU speed and motherboard's memory bandwidth can have a significant impact on the overall performance.

For example, when operated in a relatively old Dell T7400 machine that uses an Intel E5405 2 GHz CPU and DDR2 memory, a combination of ATS9350 and Asus GTX560 was able to do 2048 point FFTs at a rate of 140 KHz. On an Asus P6T7 motherboard that uses an Intel i7-990X 3.4 GHz CPU and DDR3 memory, the same hardware and software was able to go up to 235 KHz.

The type of GPU used can also have an impact on the maximum performance. For example, on the same motherboard, an NVIDIA Tesla C2070 GPU provided 13% better performance at the cost of twice the power consumption (1000 Watts) and ten times the price.

Complexity of the kernel code running on the GPU can have a significant impact on the overall performance. Users should optimize their code to take advantage of the GPU's high speed memory.

The programming environment can also have an impact on the overall performance. C/C++ offers the highest performance, MATLAB can achieve virtually the same performance as C/C++, but LabVIEW suffers a 20% performance penalty due to extra memcpy commands required.

Computer Power Supply

GPUs are power hungry. Even the low cost models such as Asus GTX560 consume 500 Watts of power. As such, users must make sure their computer's power supply has sufficient capacity.

ATS-GPU FFT

ATS-GPU comes with a built-in GPU-based, very high speed floating point FFT capability for data acquired by AlazarTech's PCI Express waveform digitizers or for user-supplied data.

Users can use sample programs in C/C++, LabVIEW or MATLAB to setup the waveform digitizer parameters, setup FFT parameters in the GPU, do the acquisition, and receive the FFT result buffer.

Waveform digitizer data is transferred to the GPU in a buffer that will contain many records. This number, RecordsPerBuffer, is specified by the user. Users should make sure that they choose this number such that the buffer size is in the order of 1 MByte or larger. Smaller buffers can reduce overall data throughput.

For software validation purposes, ATS-GPU FFT allows the GPU to operate on user-supplied data. It should be noted that the overall throughput may be significantly reduced.

If the number of samples per record is not a power-of-2, ATS-GPU FFT will perform zero-padding to the next power of 2. It will then apply a windowing function, do a single-precision floating point FFT, calculate the amplitude and convert the result to logarithmic values. This data will then be returned to the user application.

A P6T7 machine using an Intel i7-990X 3.4 GHz CPU, DDR3 memory and Asus GTX560 GPU had these benchmarks:

FFT Length	Single Channel Max.		Dual Channel Max.	
	NPT Trigger Repeat Rate	Continuous Sample Rate	NPT Trigger Repeat Rate	Continuous Sample Rate
1024	450 KHz	500 MS/s	330 KHz	340 MS/s
1536*	308 KHz		192 KHz	
2048	235 KHz	500 MS/s	172 KHz	350 MS/s
3072*	158 KHz		81 KHz	
4096	118 KHz	500 MS/s	74 KHz	325 MS/s
6144*	76 KHz		38 KHz	
8192	60 KHz	500 MS/s	35 KHz	320 MS/s

* Zero-padded to the next power of 2

The standard ATS-GPU is shipped as compiled code (a DLL for Windows). If users want access to the FFT source code, a license must be purchased separately.

Very Long FFTs

For some applications, it is necessary to perform a very long FFT, e.g. as long as one million points.

Even if a waveform digitizer has an on-board FPGA, such very long FFTs do not fit inside an FPGA due to resource limitations of the FPGA.

ATS-GPU is fully capable of calculating such very long FFTs. Our benchmarks using Intel i7-990X CPU and Asus ATX570 GPU have shown that ATS-GPU is capable of doing 460 one million point FFTs per second in single channel mode (keep up with sample rate of up to 460 MS/s) and 520 one million point FFTs per second in dual channel mode (keep up with sample rate of up to 260 MS/s).

Host CPU vs. ATS-GPU for FFT Calculation

ATS-GPU includes sample programs that allow a user to compare the performance of host CPU-based FFT calculation versus ATS-GPU based calculation.

For basic FFT processing, users can run a sample program that shows how fast can an FFT be calculated using the FFTW algorithm running on the host CPU. Benchmarks done on a 3.4 GHz CPU showed that a 2048 point FFT can be done in 5.13 microseconds. This does not include the time for transfer from a waveform digitizer, conversion to complex float, windowing and logarithmic conversion of the output.

Subsequently, they can run a sample program that shows how fast the same FFT can be calculated using ATS-GPU that runs on the GPU plugged into the computer. Benchmarks done using Asus ATX-560 and the same 3.4 GHz CPU showed that a 2048 point FFT (including conversion to complex float, windowing and logarithmic conversion of the output) can be done in 2.84 microseconds.

Users can also run sample programs that not only benchmark raw FFT processing, but also include acquisition and transfer of data by an AlazarTech PCI Express waveform digitizer. Benchmarks done under the same conditions listed above showed a processing time of 28.57 microseconds per FFT for the host CPU and 2.84 microseconds per FFT for ATS-GPU.

Compatible GPUs

ATS-GPU has been designed to be compatible with all GPUs that have Open CL v1.0 drivers. Testing was done using Asus GTX560 and NVIDIA Tesla C2070.

ATS-GPU
Real Time Signal Processing Software

It should be noted that ATS-GPU supports only one GPU. If you have multiple GPUs installed in your computer, ATS-GPU will let you select one of them for use.

Compatible Waveform Digitizers
All AlazarTech PCI Express waveform digitizers are compatible with ATS-GPU. Only single-board configurations are supported at this time.

AlazarTech's PCI bus waveform digitizers are not supported, as the host CPU is more than capable of handling data rates generated by PCI bus boards.

ATS-GPU cannot directly be interfaced with non-AlazarTech waveform digitizers. However, users can always capture data from non-AlazarTech digitizers and pass it to the GPU using the software validation data path.

Note that this will probably not provide optimal throughput. Also note that AlazarTech will not support this type of software development.

Using ATS-GPU on File Based Data
In many circumstances, users have previously captured raw data on file that they would like to process using a GPU.

ATS-GPU provides example programs in C/C++, MATLAB and LabVIEW that show how to read data from file, format it according to the GPU's requirements and transfer it to a GPU for FFT processing. This is done using the software validation datapath.

It should be noted that the rate at which FFTs can be calculated using these example programs will be limited by the speed of the disk drive in the end user's computer.

Example Programs
A number of example programs are supplied with ATS-GPU.

Example programs are classified according to programming languages supported: C/C++, MATLAB and LabVIEW.

For any of the programming languages, each AlazarTech PCI Express Waveform Digitizer has its own set of sample programs. This makes it very easy for programmers to use an appropriate example program as a starting point for their design without having to worry about specific board capabilities, as that has already been taken care of.

Using CUDA with ATS-GPU
Some customers may have existing CUDA based GPU code and may want to reuse this code with ATS-GPU.

Theoretically, it should be possible to mix Open CL and CUDA kernel code on a GPU, but this has not been tested by AlazarTech and no support will be provided by AlazarTech for any such development.

Software Licensing Policy
Users are allowed to freely distribute ATS-GPU dynamically linked library (DLL) as long as there is an AlazarTech PCI Express waveform digitizer present in the same computer. If an AlazarTech PCI Express waveform digitizer is not present in the computer, users must purchase a separate license for each computer on which ATS-GPU is installed.

For source code version of ATS-GPU, licensing policy depends on whether the user has made significant changes to the functionality of ATS-GPU or not.

If significant changes have been made to the functionality of ATS-GPU, user is free to distribute a compiled DLL for the modified program as he or she pleases. If no significant changes have been made to the functionality of ATS-GPU, user is only allowed to distribute a compiled DLL in computers in which an AlazarTech PCI Express waveform digitizer is present.

In no case is the user allowed to distribute or share the source code of ATS-GPU with other users.

Annual Subscription
ATS-GPU is a constantly evolving product, with new functionality being added on a regular basis.

The purchase of a license of ATS-GPU automatically allows customers to download updates from AlazarTech web site for a period of 12 months from the date of purchase.

Customers who want to receive updates of new releases beyond this 12 month period should purchase an annual subscription.

ATS-GPU for Linux
ATS-GPU is currently available for Windows only. A Linux version will be available in the future.

ORDERING INFORMATION

ATS-GPU for Windows	ATSGPU-WIN
ATS-GPU Annual Subscription	ATSGPU-ANN
ATS-GPU FFT Kernel Source Code	ATSGPU-FFT

Manufactured By:

Alazar Technologies, Inc.

6600 TRANS-CANADA HIGHWAY, SUITE 310
POINTE-CLAIRE, QC, CANADA H9R 4S2

TOLL FREE: 1-877-7-ALAZAR OR 1-877-725-2927
TEL: (514) 426-4899 FAX: (514) 426-2723

E-MAIL: info@alazartech.com