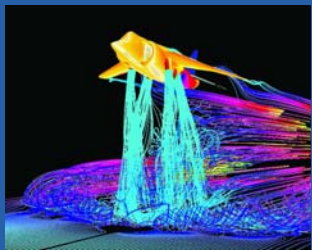
NALLATECH

High Performance Embedded Computing using FPGAs

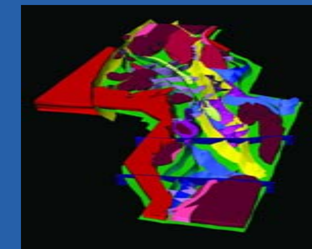
The Professional Leader in Innovative, Low Risk and High Performance FPGA Processing Solutions for Customers' Challenging Applications

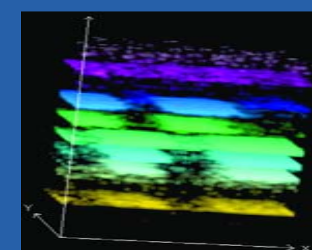
Microprocessors to FPGAs



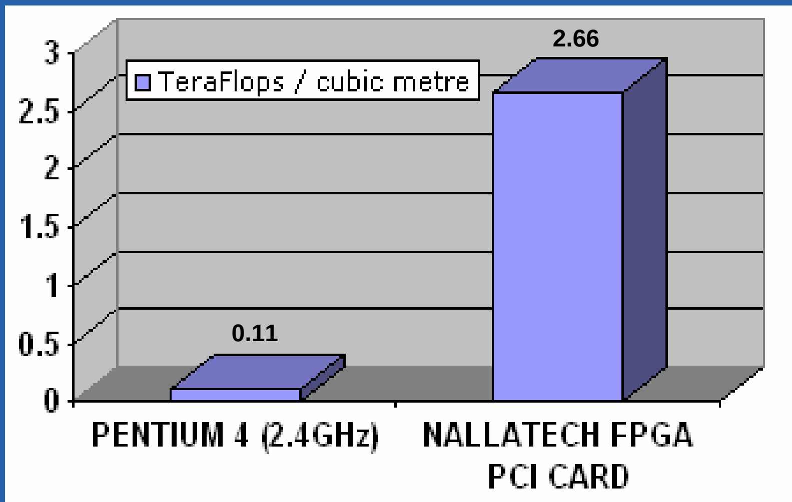
Nallatech have over 10 years FPGA system experience

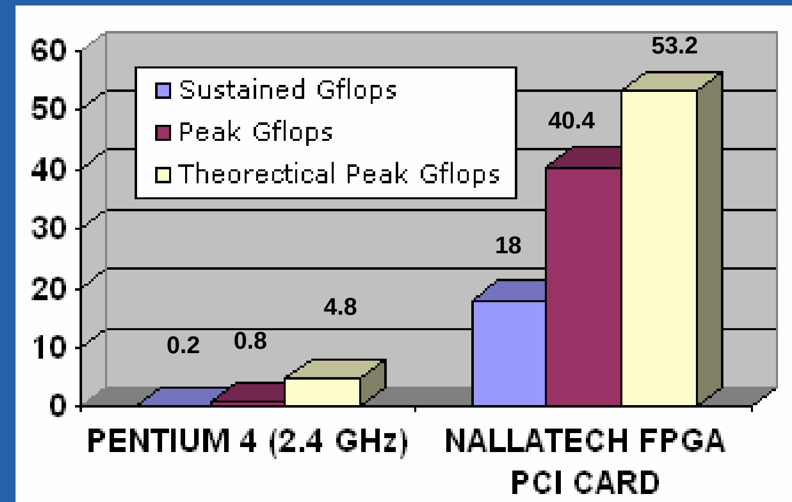
- ☑ High performance
- ☑ Flexibility
- ☑ Low latency
- ☑ Low cost
- ☑ Low power consumption
- ☑ Scalability
- ☑ Obsolescence mitigation
- ☑ Easily upgradeable
- ☑ SWAP

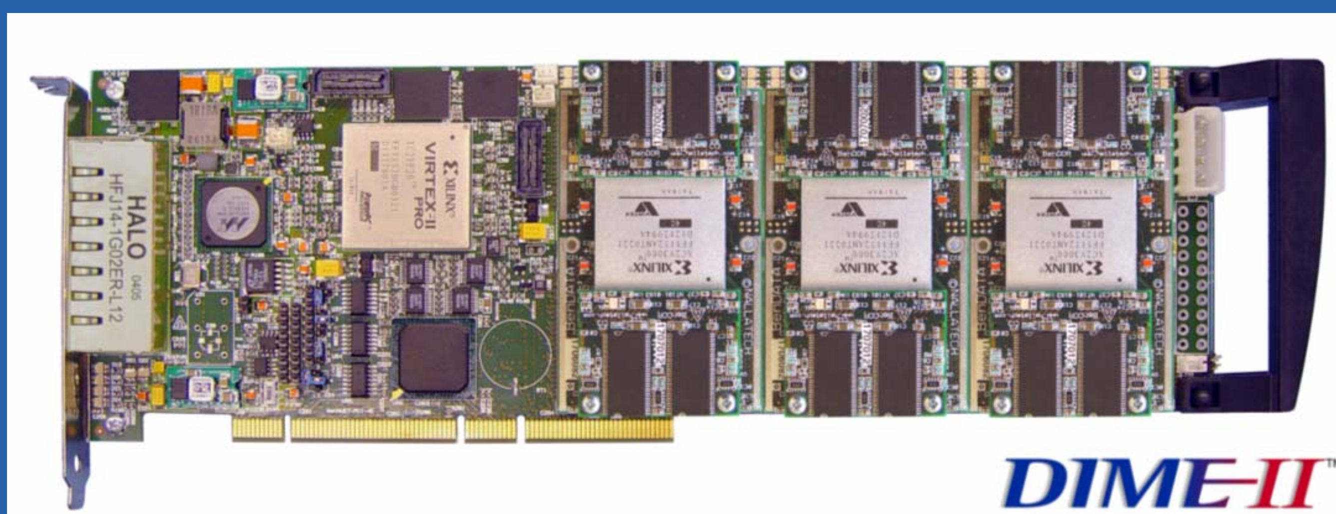




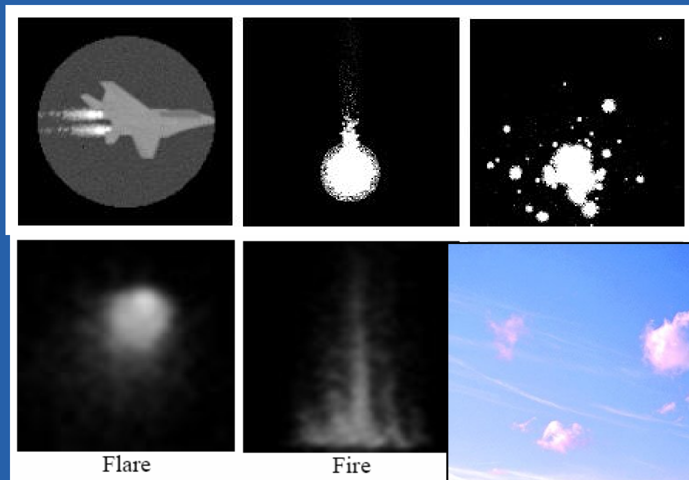
FPGA System Performance



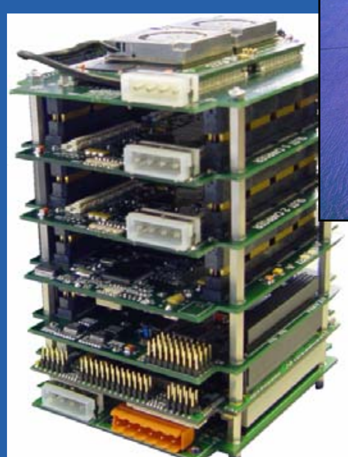


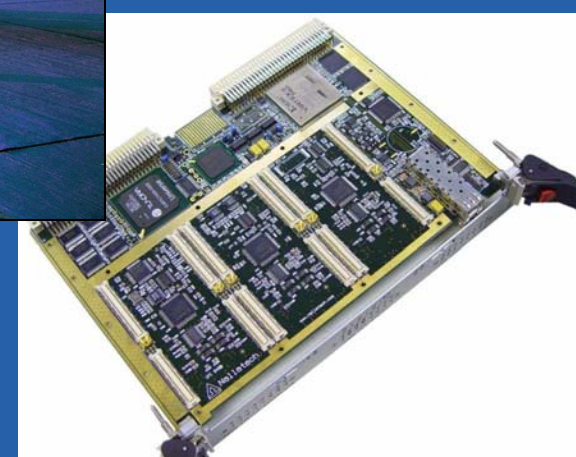


HWIL



HPEC





Aerospace & Defense

NALLATECH

Thank You for your attention

Web: www.nallatech.com
Tel : (410) 552 3352
Toll Free: 1-877-44-NALLA
Fax : (410) 552 3354

High Performance Embedded Computing using Field Programmable Gate Arrays

Craig Petrie, Malachy Devlin and Keith Regester
1010 Liberty Road, Suite 103, Eldersburg, MD. 21784
www.nallatech.com

Abstract

Ongoing commercial and technology needs are driving increasingly demanding performance requirements for new embedded processing systems. Often, these systems have utilised general-purpose processors, with the processing capability of each consecutive generation of device increasing in accordance with Moore's Law - providing scientists and engineers with the ability to process large amounts of sensed data and solve computationally intensive problems in a number of end-markets.

With sensor systems of the future expected to increase in both functionality and performance while adhering to constraints such as size, weight and power (SWAP), system engineers now need to consider alternative processing technologies such as Field Programmable Gate Arrays (FPGAs) in order to meet the requirements of high performance embedded systems operating in real time. This paper discusses the SWAP and performance advantages of FPGAs in embedded applications, with emphasis on the implementation of IEEE-754 floating point arithmetic and example HPEC applications using Nallatech Inc COTS FPGA products.

***Keywords**: FPGA, COTS, SWAP, Low Latency, Real Time, Scalability, Embedded Systems, High Performance, Architecture, IEEE Floating Point.*

Computing Challenge

Since the early days of computing, it has always been the aim to process large quantities of data in the minimum time with minimum levels of power consumption. Technical and economic considerations have often meant that this has been carried out in a less than optimum way, using general purpose processing devices and high level programming languages. Although not particularly efficient, this approach has allowed applications to be developed within reasonable timescales and budgets using readily available technologies. Over the years, Moore's Law has promised processing performance improvements that have allowed larger and larger computing challenges to be undertaken. Supercomputers have gradually evolved into large multi-processor distributed systems capable of tackling a range of demanding computing problems.

Embedded Processing

Whereas supercomputers regularly fill complete rooms or buildings, this approach to computing does not lend itself well to embedded applications where systems are restricted by size, weight and power (SWAP) constraints. High performance embedded computing (HPEC) systems typically consist of a number of different processing elements, combining to process sensor data into a form suitable for use by other attached systems and controllers. This processing chain can be divided into four distinct stages; front end processing in close proximity to the sensor, secondary data processing, system control and secure communications or data storage [1].

In many cases, general purpose processors are not suitable for such systems, and so embedded computing applications have pursued an alternative approach to computing – one in which well defined problems are tackled using highly-customizable processing technologies capable of solving a limited range of problems quickly and efficiently. The use of technologies such as ASICs, RISC and DSP processors have helped alleviate the SWAP problems inherent in many HPEC applications, however by no means represent a final solution. Advances in sensor technologies available now and predicted over the next few years are providing engineers with the ability to implement sophisticated hi-fidelity embedded processing systems featuring increased bandwidths, functionality and data rates. As the aggregate throughput requirements for embedded systems shatters the GOPS barrier, these types of processing technologies become unsuitable since performances can only be met by concatenating processing blocks in a pipeline architecture.

Size, Weight and Power (SWAP)

This incremental approach to boosting system performance has limitations, and is often an unacceptable solution, particularly for SWAP constrained applications. These design considerations represent significant challenges to the system engineer, particularly when the application is intended to be used as part of an embedded system operating in real time, where system complexity is often compounded by environmental conditions such as temperature and vibration.

Assuming that these problems are overcome, or at least manageable, closed loop designs relying upon real time operation can be significantly affected by excessively long system latencies. Compromises are possible through reductions in data rate and sensor resolution; ironically, these compromises can lead to a requirement for more compute intensive algorithms to overcome the fact that the fidelity of data is now reduced. Ultimately, these sacrifices only contribute to overall system performance degradation.

Competing Technologies

Silicon and power efficient ASIC solutions have been used extensively in high volume and defense HPEC applications of the past; however this solution has become prohibitively expensive, with development times and NRE costs far greater than competing technologies. Maintaining and upgrading ASIC based systems is expensive and the technology refresh rate, introduces other complications such as obsolescence. This has limited the technology choice to the selection of RISC and DSP processors from vendors such as Texas Instruments, Analog Devices, IBM and Motorola. Unfortunately, these devices do not always satisfy the performance requirements of HPEC applications, nor do they resolve the fundamental performance limitations of latency and memory bandwidth associated with microprocessor architectures within scalable systems. Attention has therefore turned towards the development and utilisation of new types of processing architectures such as Field Programmable Gate Arrays (FPGAs), which are able to alleviate many of the economic and technical challenges associated with HPEC systems.

Various benchmarking exercises have been undertaken over the last few years comparing RISC, DSP and FPGAs for a number of different application examples such as Viterbi Encoding, Fast Fourier Transforms and FIR digital filters. This work has demonstrated that families of FPGAs such as the Virtex-II and Virtex-II Pro from Xilinx are able to compete, and in many cases, out-perform multi-gigahertz processors [2]. It is important to realise that for many types of computing problems, memory bandwidth rather than processor speed is the main system bottleneck.

Processing Challenges

Take for example, the implementation of a typical image processing operation such as convolution. This involves a matrix multiplication resulting in several multiplications per image pixel. Using a modest 5x5 window and a conventional DSP processor, a latency of several clock cycles per pixel is added to the algorithmic latency since the overall performance of the processor is limited by the number of multiplications that can be performed in parallel and scheduling of memory accesses. On-chip processor memory is usually too small to buffer a full high resolution image frame, and so external memory read and writes are required to complete a single calculation. This I/O performance bottleneck is pronounced due to the typical single port for memory access on processors and becomes a significant problem when larger windows or multi-frame algorithm are implemented. Higher frame rates and resolutions which are used to achieve better imaging accuracies further escalates this problem.

Table 1 provides a comparison of the external bandwidth and peak floating-point performance for the TigerSHARC ADSP-TS202S DSP processor, the Motorola MPC7455 PowerPC, and the Xilinx XC2VPX70 FPGA.

	External bandwidth [GBytes/sec]	Peak floating point performance [GFlops/sec]
XC2VPX70 Xilinx FPGA	**70**	**27**
MPC7455 PowerPC	**1.064**	**8**
TigerSHARC ADSP-TS202S	**5**	**3.6**

Table 1 – Performance Comparison [3, 4, 5]

The comparison clearly shows that the FPGA has the highest theoretical performance capabilities of the 3 devices. The 992 pins of general purpose I/O in addition to the 20 Multi-Gigabit Transceivers (MGTs) operating at 10.3125 Gbits/sec make the FPGA an attractive solution for many of the computing challenges associated with HPEC systems. The reconfigurable array of logic blocks, memories and multipliers provided within FPGAs offer a high performance hardware architecture ideal for building parallel processing pipelines operating at hundreds of MHz. FPGAs featuring up to 10Mbits of embedded Block RAM operating at 300MHz provide an on-chip memory bandwidth of several Tbits/sec.

Take, for example, the new Virtex-4 range of FPGAs from Xilinx. Manufactured using the latest 90nm processing technology, they provide users with 500MHz XtremeDSP slices delivering an aggregate DSP performance of 256GigaMACs per second [6]. High accuracy Digital Clock Managers, reconfigurable synchronous dual-port static BRAM and FIFOs provide the necessary clock management and memory resource required to implement high performance algorithms. Continuing the trend set by the previous generation of Virtex-II Pro FPGAs, the Virtex-4 features multiple 32-bit RISC PowerPC processors delivering an excess of 1300 Dhrystone MIPS [6].

As a result, FPGAs are now being used as effective DSP engines. Although today's DSP processors boast high levels of performance, they can't compete against FPGAs for specialised computing. FPGAs can be configured with a custom hardware design, implementing control logic in the hardware, saving precious clock cycles per calculation, reducing component count and increasing reliability. Innovations and state of the art silicon processing techniques have dramatically improved the functionality and capability of the FPGA over the last 6 years, allowing them to be used in a wide variety of applications typically dominated by microprocessors or expensive and inflexible ASICs.

The importance of FPGA COTS Products

One of the difficulties for engineers and scientists wanting to use FPGA technology to help improve the performance of their applications has been the availability of flexible, scalable COTS products supporting the latest FPGAs and design tools.

There have been a number of modular standards used over the last few years that have supported new generations of processing technologies, including FPGAs; however they have limitations when used in real time processing applications. Firstly there are those based around specific microprocessors, for example the TIM-40 from Texas Instruments and SHARCPAC from Analog Devices. The main difficulty with this category is that the system engineer has to constrain the capability of supported FPGAs in order to emulate a microprocessor interface - restricting the superior I/O bandwidth of the FPGA.

Secondly, there are the microprocessor neutral module standards. One of the most popular is the PCI Mezzanine Card (PMC). Unfortunately, this is still principally designed with microprocessor-based systems in mind. It is also, perhaps more seriously, based around a non-deterministic bus communications system with variable latency. This again implies constraining the FPGA to a less than optimum solution, with the added complication that the bus behaviour will alter when the system is changed. In addition, significant parts of the FPGA real estate must be dedicated to handling the non-determinism. Within a real-time system it is critical that bandwidths and latency can be guaranteed. Using this type of module means, practically, that this creates a more complex requirements specification and adds risk to the system design and integration success.

Nallatech DIME-II Architecture

In order to address these problems, and present a processing platform that truly exploits the strengths of FPGA technology, Nallatech has developed a range of COTS plug and play motherboards and modules supporting the latest Virtex-II and Virtex-II Pro FPGAs from Xilinx.

Nallatech motherboards are available in VME, PCI, cPCI, and PC104*plus* form factors, allowing system designs to be easily ported from one form factor to another. This capability permits the rapid development and deployment of sophisticated embedded processing systems – saving money and reducing time to market. Due to the modular nature of the architecture which is based on COTS allows the deployment of new technologies such as the Xilinx Virtex-4 FPGA without having to redesign the supporting motherboards or systems.

The high-performance modular architecture is based on the open DIME-II standard that incorporates system level intelligence features such as temperature and voltage monitoring, and a module to motherboard bandwidth of up to 8 GBytes/sec (over 15 times the theoretical maximum performance of 64 bit / 66 MHz PMC).

Figure 1 is a photograph of a Nallatech PC104plus COTS motherboards featuring 3 DIME-II expansion slots supporting different families of processing modules.

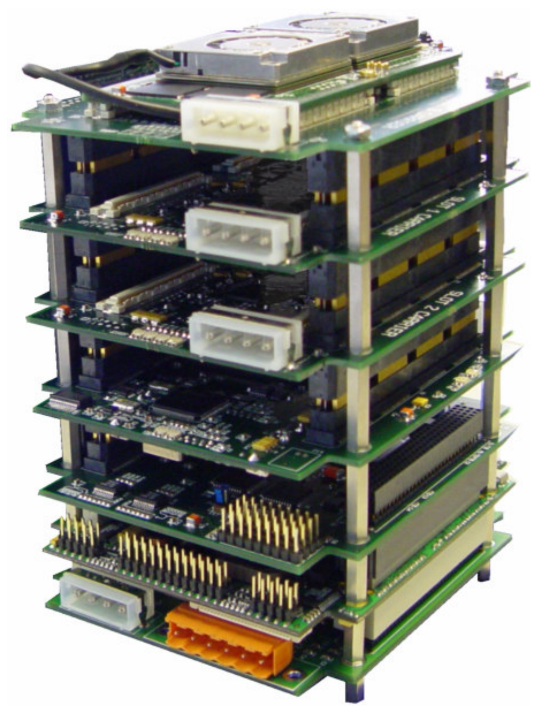


Figure 1

PC104*plus* carrier card capable of supporting up to 7 Xilinx Virtex-II and Virtex-II Pro FPGAs [7]

Floating Point Calculations using FPGAs

FPGA vendors such as Xilinx and Altera have aggressively developed the capabilities of FPGA devices over the last decade. Originally, FPGAs were used for little more than glue logic, however the introduction of fast carry chains, multipliers, memories and embedded processors within the FPGA fabric itself has transformed the FPGA into a high performance hardware architecture ideal for building processing pipelines. Semiconductor technology advances driving Moore's Law are expected to yield FPGA devices with a factor of three to eight times more peak floating point performance than comparable microprocessors [8].

As a result, engineers and computer scientists have begun using the FPGA to perform compute-intense algorithms using IEEE-754 single and double precision floating point formats. Before FPGA devices such as the Virtex-II and Virtex-II Pro, the only way to process these operations was to write software routines that executed relatively slowly compared to dedicated hardware implementations. This has been a problem for many of today's FPGA-based HPEC applications, which rely upon the use of floating point for a variety of scientific operations including square root, logarithm, exponential and trigonometric functions. Fortunately, a number of FPGA optimised floating point libraries are now available, allowing the implementation of complex algorithms, real-time graphics and control systems. This now means that floating-point units no longer represent the bottleneck in computationally intensive designs. Designers forced to use fixed-point implementations in the past can now avoid the errors introduced by fixed-point quantization that accumulates over time when performing iterative algorithms.

FPGA Example Design – N Body Problem

In order to demonstrate the potential power of FPGA-based computing, an N Body gravity simulation was implemented using Nallatech's own optimised IEEE-754 floating point cores on a Nallatech PCI motherboard populated with 4 Xilinx V2V6000 FPGAs shown in Figure 2.

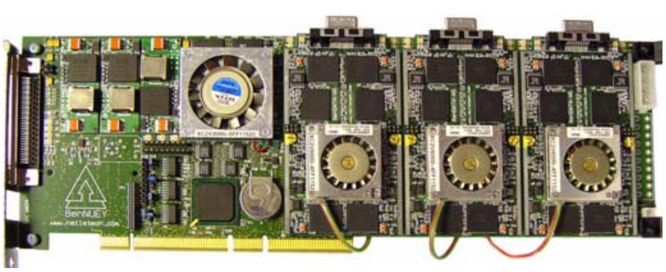


Figure 2 – Nallatech FPGA PCI Card

An N Body gravity model implemented on the FPGA system calculated the gravitational force applied by all other bodies for each individual body. This force was then used to update each body position and velocity, with the result displayed on screen for the user. Almost all the computation time was spent calculating the forces since the number of interactions is proportional to the square of the number of bodies.

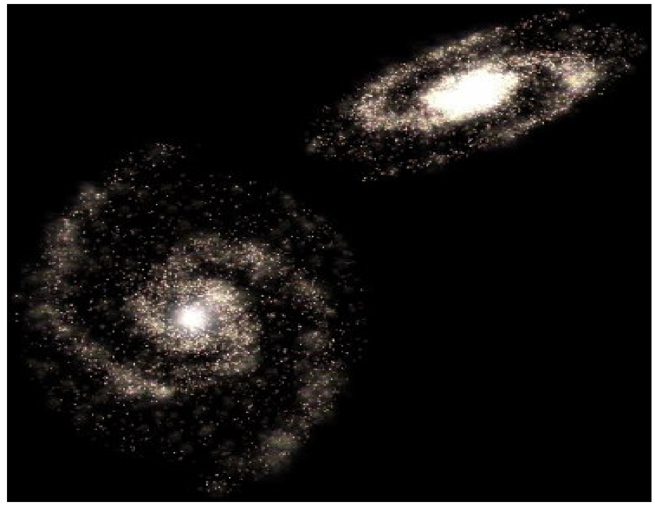


Figure 3 – N Body Gravity Simulation

This example design demonstrates the true potential of FPGAs for high performance applications. By pipelining the multiple force calculations, the simulation was massively accelerated – performing more than 400 floating point calculations in parallel, resulting in a sustained performance of 18 GFlops. When the same calculations were executed on the Host 2.4GHz Pentium 4 processor, the performance level dropped to 0.2 GFlops.

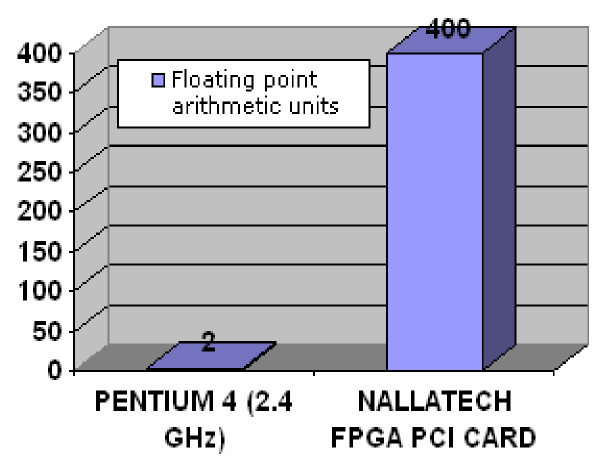


Figure 4 – Floating-point arithmetic units

Figure 5 illustrates the floor plan of one of the XC2V6000 FPGAs used in this example design. The rectangular blocks represent the different floating-point arithmetic units implemented on the FPGA. The floating point cores used in this application were written using VHDL and were floor planned onto the target FPGA in order to make efficient use of the FPGA resources and to achieve operating frequencies approaching 150MHz.

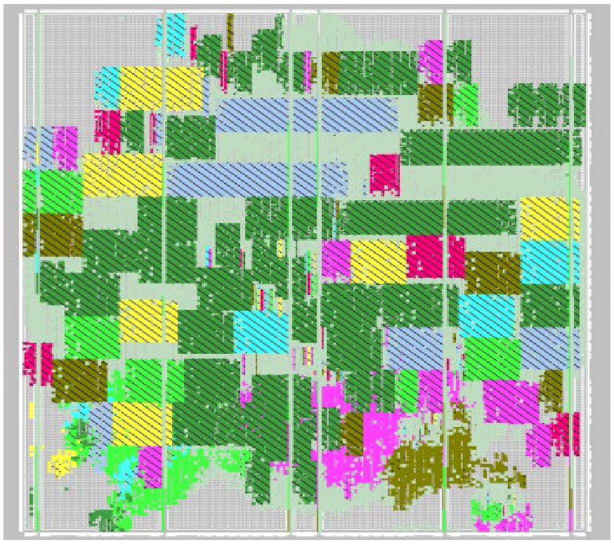


Figure 5 – FPGA floor plan

The results summarised in figures 6 and 7 clearly demonstrate that the FPGA based processing system is far superior in terms of computational performance and power consumption than the Pentium 4 processor.

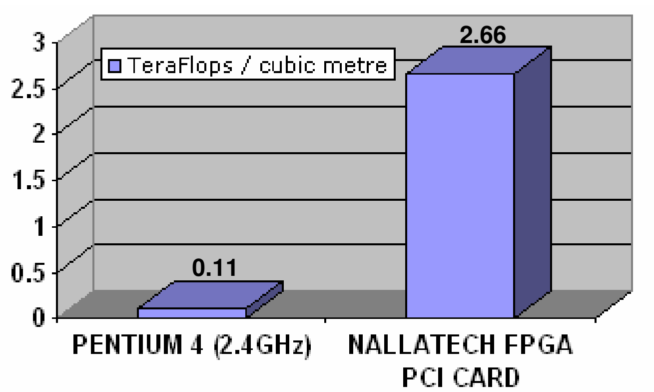


Figure 6 – Theoretical peak volume performance

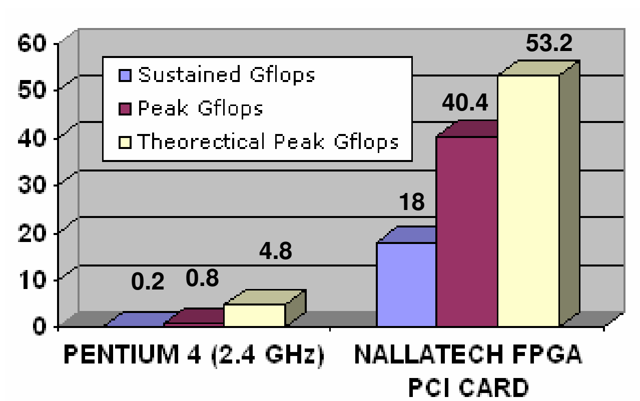


Figure 7 – Relative performance of gravity model

The use of reconfigurable resources resulted in a x90 improvement in performance in the same physical volume. This means that, in theory, one 2U box could replace the processing power of ten 42" high racks. Also, the power consumption per GFlop of computing performance was approximately 200 times better. This equates to a superior price-performance figure using the FPGA system compared to conventional microprocessors.

This performance improvement could be further improved by porting the design to a system consisting of the new Virtex-II Pro based "BenBLUE-III" DIME-II module illustrated in figure 8.

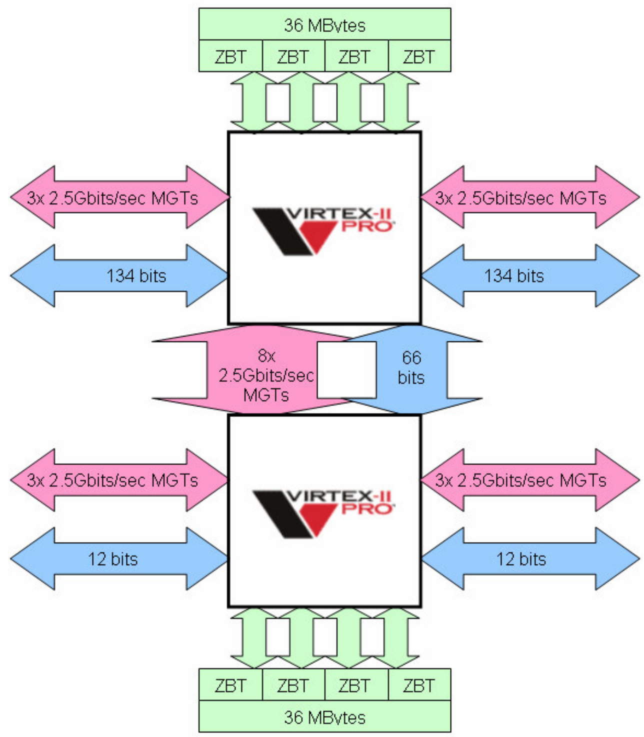


Figure 8 – BenBLUE-III functional diagram

This particular example can be applied to a number of applications such as molecular modelling, however, it does raise the question; can this be demonstrated successfully for HPEC applications?

HPEC Applications Using FPGA COTS Products

Nallatech have been involved in high performance embedded processing using FPGAs for over 10 years. To date, Nallatech products have been utilised in a range of embedded applications including Unmanned Aerial Vehicles (UAVs), Missile simulation and detection, real time image processing, RADAR, Navigation and Satellite systems.

Prior to the release of the Virtex series of FPGAs from Xilinx in 1998, the primary solution that Nallatech used was based on parallel processing system with FPGAs as co-processors. As described previously the processors were the bottleneck in the system but pre-Virtex the FPGA did not have the capacity to manage the system and implement complex algorithms. The FPGA landscape is now very different. In the following examples FPGAs have been used in place of conventional processors because the SWAP constraints, real time processing and latency requirements are simply too demanding to be satisfied by single or multiple microprocessors.

Example 1
Real Time Iterative Processing

A recent HPEC application using Nallatech FPGA COTS products involved the real time processing of high-resolution video onboard Unmanned Aerial Vehicles (UAVs). The application was designed as an onboard processing system to perform detect and avoid functionality.

The nature of the application, the severe SWAP constraints and the computationally intensive algorithms required to achieve a successful mission required the use of FPGA COTS products providing extremely high performance densities. The application's iterative processing design was implemented on a Nallatech PCI carrier card, with the hardware acceleration achieved using 3 Nallatech "BenBLUE-II" DIME-II modules, consisting of two Xilinx 2V6000 FPGAs and ZBT memory. The floating point operations required to process the real time video stream were used throughout the 7 FPGAs, resulting in a high performance, low latency system processing data at 36 GFlops/sec.

As well as providing ultra high performance density, the UAV application also benefited from reduced support infrastructure costs, lower total cost of ownership and significantly reduced risk since the hardware used was based upon COTS products [11].

Example 2
Image Generation for Real Time Particle Models

The real time processing capabilities of FPGA COTS products were demonstrated in a defense related HPEC application involving a HWIL image generation system for real time particle models. Nallatech COTS hardware featuring Xilinx FPGAs were used to model thousands of individual particles at frame rates over 100Hz and a resolution of 1024x1024 in order to create highly realistic signatures in terms of spatial dynamics and IR signature [10]. Particle models are ideal for simulating dynamic objects such as flares, exhaust plumes, fires and explosions.

Plumes and flares are objects commonly associated with weapon IR scene generation. Their behaviour exhibit large random elements and therefore polygonal modelling is not applicable. The parallel processing properties of FPGAs were ideal in this example because the particle models exhibited high levels of parallelism.

Mathematically, the application needed to calculate a number of parameters related to individual particles, including initial velocity, random velocity, radius, initial temperature, random temperature, thermal decay, drag factor, turbulence and particle generation rate.

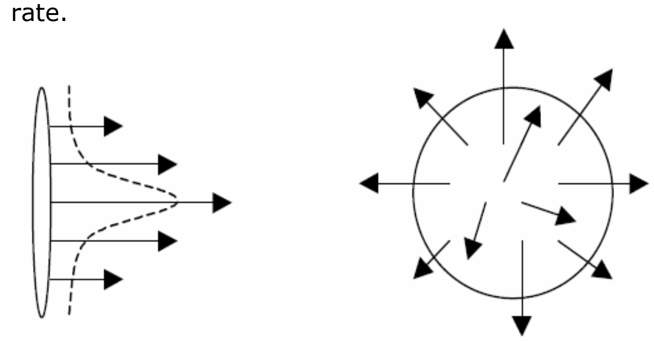


Figure 9 – Examples of particle flow [10]

Gaseous objects could then be modelled as a system of individual particles each contributing to the whole. The behaviour of each particle is dependent upon thermal and aerodynamic forces, and so by varying the reaction of particles to these forces it is possible to simulate forms such as plumes and flares.

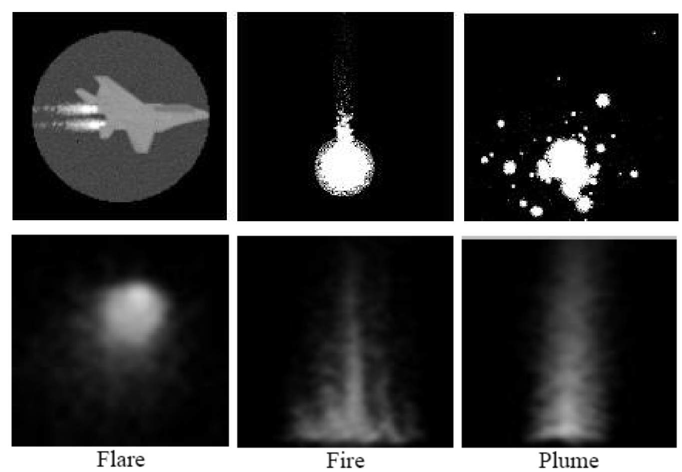


Figure 10 – Particle generated images [10]

The aim of this study was to improve the performance of particle models by using the inherent parallelism of hardware. Previously developed particle models have consisted of hundreds of particles describing an overall system. Particle models are repetitive in nature and suitable for development on an FPGA. By using hardware to perform repetitive algorithms the particle system can be expanded from hundreds of particles to thousands. This is primarily due to the ability of hardware to perform multiple tasks in parallel. Increasing the number of particles increased the fidelity of the model.

The real time HWIL application demonstrated that FPGA designs are capable of performing simple graphical models such as particle methods with impressive results. More complicated processing can be performed by linking multiple FPGA modules in parallel. The increase in particles in this particular example significantly improved the representation of particle models such as chaotic plumes and fires. It has also allowed true full screen coverage in real-time allowing simulation of large area countermeasures with guaranteed frame rate.

The implementation used the Nallatech DIME architecture which facilitated an infra-structure amenable to FPGAs, offering plug and play support, flexibility and scalability. This ensured that the application work carried out could be easily transposed to other variants of systems which may offer more capacity or interface to different video channels.

Example 3
Real Time Data Processing and Storage

The use of FPGAs for solving SWAP constrained HPEC systems was put to the test recently during the design of a complex multi-board, real time image processing system with a mass storage interface. The application called for the system to be deployed on a commercial aircraft operating at high altitude, with a high-resolution camera being used to capture the effects of atmospheric turbulence. This raw data was to be processed, formatted and stored for later analysis.

The intention was to upgrade the system at a later date and use the high-resolution video data to drive a decision engine that would control the aircraft's avionic systems. This would allow for a smoother flight and better fuel efficiency. The size, weight and power constraints imposed by the operating environment immediately ruled out the use of certain types of form factors and technologies. The computing power required to process the high-resolution data in real time would have translated into multiple server racks of conventional CPUs – an impractical solution in this case.

The Nallatech BenNUEY-PC104+ solution was selected as the main processing platform for the system. The PC104*plus* form factor satisfied the physical and mechanical constraints of the application, while the scalability and flexibility of the high bandwidth DIME-II architecture allowed the system to be tailored through the support of plug-and-play DIME-II COTS modules.

In order to provide as large a dataset as possible to aid the turbulence analysis in the laboratory after the flight, a mass storage interface was used to store the captured video data and telemetry. The data was time-stamped and partitioned so that engineers analysing the data offline could select a specific time interval from the duration of the flight and read it back for analysis. A graphical user interface utilising the Nallatech Field Upgradeable Software Environment (FUSE) API was used to handle the configuration of the multiple FPGAs in the system in addition to the monitoring of system temperature and voltage values. Power consumption and management were as important in this application as any other embedded system operating in difficult and uncontrollable environments.

The BenNUEY-PC104+ carrier card was used to format the processed image data from the camera, with fast access ZBT SRAM memory used to buffer the data while it was serialised and transmitted over high speed LVDS links to a bank of 4 SCSI hard drives – providing a total storage capacity of one Terabyte.

Figure 11 – HPEC application using Nallatech FPGA COTS products [9]

The flexibility of the FPGAs allowed sections of the design to be optimised without physically altering the hardware, while the availability of the spare DIME-II module slots on the BenNUEY-PC104+ offered the customer the option of scaling the system to support additional SCSI disks.

The availability of Nallatech COTS products coupled with system level tools such as DIMEtalk, allowed these example HPEC applications to be designed, implemented, tested and deployed within a matter of months, and at relatively low cost. An equivalent system based upon microprocessors would have struggled to operate in real time, would have required multiple rack PCs and consumed greater amounts of power. A silicon and power efficient ASIC solution would have resulted in lengthy development timescales and significant NRE costs.

Conclusions

With sensor systems of the future expected to increase in both functionality and performance, engineers developing next generation embedded systems are relying more and more upon alternative processing technologies such as FPGAs in order to meet user requirements.

Embedded processing applications aiming to process real time data are struggling to achieve suitable performance levels using traditional microprocessor architectures given strict constraints of size, weight and power. These factors are severely restricting the choice of technology for the system engineer tasked with delivering the end product.

FPGA solutions using COTS products are quickly becoming the only realistic solution for such applications, and provide a viable economic path from prototyping to production. The ability to implement and compress advanced signal processing algorithms in small COTS form factor platforms will allow embedded processing systems to be built with multi-mode functionality quicker and cheaper than ever before, leading to higher levels of integration and more reliable systems.

The key advantages of FPGAs in these applications are increases in per-device performance, device I/O and reduced per-device power consumption – i.e. improved performance density. However, the design of FPGA-based systems can be more demanding than the microprocessor-based solutions. COTS FPGA hardware solutions and high-level FPGA algorithm design tools, such as those provided by Nallatech, have improved this position, helping designers to reduce development times, cost and risk when developing high performance embedded systems.

References

[1] *"Improved Availability & Reduced Life Cycle Costs of Military Avionics Systems" – D. J. Shand, D. Denning. Nallatech Ltd. February 2004.*

[2] *BDTI – FPGA/DSP Tackles Telecom Apps - http://www.bdti.com/articles/info_eet0207fpga.htm#Figure%201. August 2004.*

[3] *Xilinx FPGA Product Tables, August 2004. http://www.xilinx.com/products/tables/fpga.htm*

[4] *"Embedded Processing and DSP – TigerSHARC Processor". 500 MHz TigerSHARC Processor with 12 Mbit on-chip embedded DRAM. Analog Devices, August 2004. http://www.analog.com/processors/epProductPage/0,2542,ADSP%252DTS202S,00.html*

[5] *COTS Portable / Mobile – "Using the TigerSHARC DSP for Low-Power, High Performance Applications". Tim Wheatley, Systems Manager, Transtech DSP UK. December 2002*

[6] *Xilinx Virtex- 4: One family, multiple platforms. August 2004. [2] "Xilinx Delivers Virtex-4 FPGAs" – http://www.xilinx.com/prs_rls/silicon_vir/0480v4shipment.htm July 2004*

[7] *"BenNUEY-PC104+ Product Flyer" - http://www.nallatech.com/solutions/products/embedded_systems/dime2/rc_systems/bennuey-pc104/ July 2004.*

[8] *"FPGAs vs. CPUs: Trends in peak floating point performance. K. D. Underwood. February 2003.*

[9] *"FPGA Computing Provides Superior Performance Density for UAV Applications" – C. Sanderson. Nallatech Ltd. June 2003.*

[10] *"High Frame Rate, Low Latency hardware-in-the-Loop Image Generation – An illustration of the Particle Method and DIME". A. Cantle, M. Devlin, E. Lord and R. Chamberlain. Nallatech Ltd, September 2000.*

[11] *"Using FPGAs for Image Processing on Airborne Applications" – M. Devlin, C. Petrie, N. Harold, C. Sanderson, D. Stark. Nallatech Ltd. July 2004.*

NALLATECH

Nallatech is the world's leading supplier of high performance FPGA solutions. A comprehensive product range delivers the highest performance-density, maximum scalability and most complete solutions available on the market today. Nallatech offers a unique high-speed architecture, combined with support for industry standard PCI, cPCI, PC/104*plus* and VME, along with software support for the most popular operating systems and programming languages. Nallatech provides flexible, easy to implement solutions for Data, Communications, Digital Signal, and Image Processing. Customers benefit from lower costs, reduced power consumption and improved performance in end markets including Telecommunications Infrastructure, Aerospace and Defense, Imaging, and Scientific Computing. www.nallatech.com