insideHPC

insideHPC Research Report

Are FPGAs the answer to the "Compute Gap?"

Written by Dan Olds, Gabriel Consulting Group

With the deluge of new data from new sources, it isn't surprising to find that data centers are running short on compute capacity. In this research report, we explore the world of accelerators, primarily FPGAs, to see if they're the right answer to fill the 'compute gap'.

Introduction

More than 50 billion devices will be connected to the internet by the year 2020. Fifty billion, that's a very big number. More than 200 billion sensors will be web enabled in the coming years. And more than 100,000 racks of systems (racks, not individual systems) will be sold this year. The "*anything* as a service" market will be worth more than $1 trillion in 2020.

One of the biggest challenges in IT today is getting enough compute resources to handle current processing tasks and keep up with future needs.

What all of these trends have in common is that they require increasing compute capabilities to become reality. In fact, one of the biggest challenges in IT today is getting enough compute resources to handle current processing tasks and keep up with future needs. This gap between what is available have today and what is needed in the future is a problem both in scientific computing (High Performance Computing or HPC) and commercial enterprise computing.

On the HPC side, there is always the need to model more complex interactions and develop models with more variables and higher accuracy. Even though HPC computing capability has increased considerably, at least judging from the Top500 list benchmarks, there is still significant unmet demand for more compute power.

Enterprises are facing much the same challenges, with the advent of Big Data giving them the ability to, for example, slice and dice customer purchasing patterns in a myriad of ways. But harnessing these actionable insights takes a lot more compute horsepower to run these analytical models, and a shortage of processing capability becomes a bottleneck from applying the real time decision making that global business imperatives demand.

Simply adding more of the same servers that are deployed in the existing data center, is one solution to the 'compute gap' problem. However, this method can result in further stretching system footprints and draw higher electrical demand to data centers that may be approaching their physical limits. Replacing traditional systems in the installed base with the Industry leading energy efficient performance of servers fueled by Intel® Xeon® processors with complementary accelerators depending on particular workload needs can help organizational agility while reducing operational costs (TCO).

Adding accelerators and inline processing to relatively freshly deployed (or new) servers is a good means to get more compute capacity without contributing to server sprawl or heavy electrical load.

Contents

An Array of Accelerators

There are several types of application accelerators to select from and Intel offers a wide variety of accelerated platforms. The best accelerator, from a raw performance standpoint, would be a custom ASIC designed for each of your most prevalent applications. However, this could be a very expensive undertaking, with set-up costs that can run millions of dollars for a single ASIC.

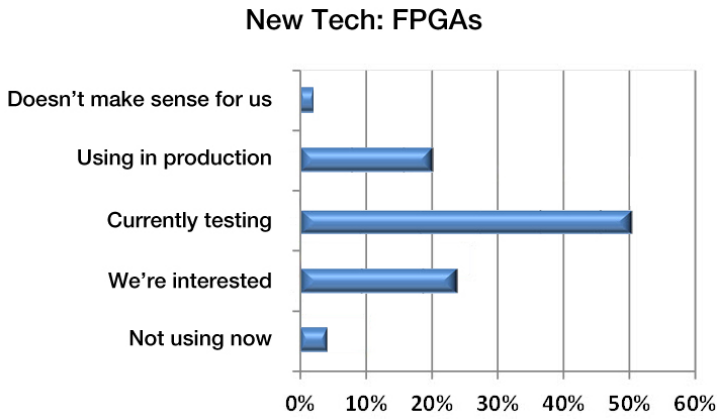
This may also result in a rigid infrastructure that is geared for only the applications you're running today – which means problems when applications invariably change tomorrow.

GPUs and FPGAs can vastly accelerate compute intensive workloads, while adding only a fraction of the power of an entire server to the data center electrical load. In fact, watt for watt, these alternatives are often more efficient than general purpose CPUs on many workloads.

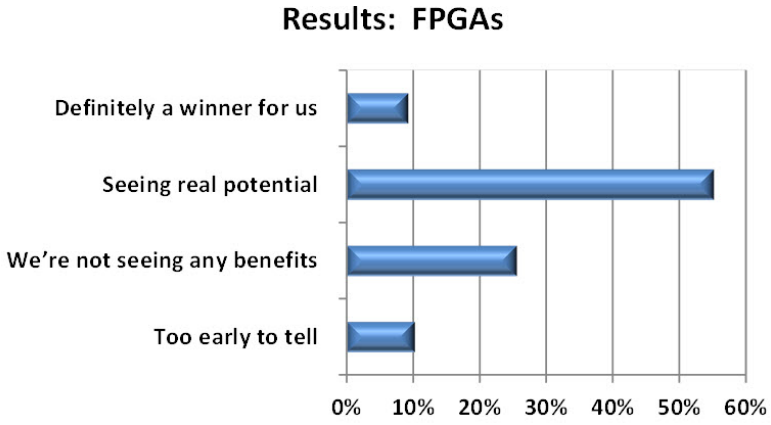
While GPUs can offer very fast numerical processing, they are a bit of a one trick pony when compared to FPGAs. GPUs can accelerate some compute workloads radically, but that's all they can do. FPGAs, on the other hand, have the flexibility to be used as compute accelerators, as additional high speed memory, or even as ultra-high bandwidth storage controllers. They can also be deployed as an offload high speed network adapter, or an inline pre and post processor.

Moreover, a FPGA can be programmed (and reprogrammed) to accelerate almost every computing workload, which is important, since not every application has the necessary CUDA versions that allow it run efficiently on GPUs. The reprogrammable FPGA can also be used on workloads that evolve over time, such as security workloads or before finalization of a standards.

The joint InsideHPC/Gabriel Consulting Group 2016 HPC & Large Enterprise Purchasing Sentiment survey showed that the vast majority of our respondents are either interested in or actively testing FPGAs in their data centers.



Of the respondents who are either testing or using FPGAs in production, more than 60% say that the technology is showing real potential or is definitely a winner in their data center.



5 Myths About FPGAs

MYTH NO. 1

FPGAs are only for embedded devices that are not useful for datacenter applications.

Fact: The rapid data processing capability of FPGAs have a tremendous potential to transform how datacenter deal with the massive influx of Big Data. In the past, FPGAs were used mostly in embedded devices and for ASIC testing purposes. Today, influential first-tier providers such as Microsoft Azure are deploying FPGAs for everyday processing chores. The recent flood of unstructured data from new sources, and the computing associated with processing this data, is spurring demand for FPGAs due to their ability to process data very quickly while using very little energy.

MYTH NO. 2

The increase in data is no big deal: current trends to scale out existing data-center architectures and move to software-defined data center networks and storage will more than handle it.

In fact, just scaling out existing data center resources, even with a move to newer infra-structure like software-defined networks, will not cope with the avalanche of data. We will have to upgrade every facet of the data center, adding hardware acceleration to servers, shifting more computational resources into the data-center networks, and reorganizing, and devoting more processing to, storage systems.

MYTH NO. 3

The perception that FPGAs are very challenging to program.

This was certainly true in the past; FPGA users needed the skills of hardware logic designers to configure the devices or write code in the difficult to master VHDL or Verilog languages.

It's a much different story today, because programming languages OpenCL and C/C++ can be used to program modern FPGAs. This can result in an average 5-6x less development time, which is quite substantial. In some cases the development time has even been cut by as much as 90%. Additionally, Intel is creating novel new approaches and tools for developing with FPGAs that make their development much more like traditional software development flows.

One example (admittedly non-HPC related) of how easy it is to program today's FPGAs concerns how a summer intern optimized GZIP using OpenCL. Whereas industry leading companies had previously coded GZIP on a FPGA using Hardware Description Language (HDL, the assembly language for FPGAs), over the course of several months, Intel showcased how one summer intern created the GZIP algorithm with OpenCL in just over a month.

The intern's initial performance of the OpenCL port was only 10% slower and used only 12% more resources — which isn't too bad for a student-level optimization effort, right? With a few additional tweaks by an optimization specialist, today the code operates at 20% better than the RTL version.

MYTH NO. 4

FPGAs are too expensive to make sense for general use.

When performance is factored in, FPGAs can actually be an economical option for accelerating many workloads in HPC and the enterprise as well. In cases when a server has to handle a mix of workloads (making ASICs impractical), FPGAs are a very energy-efficient solution, delivering more computations per joule than alternatives.

MYTH NO. 5

FPGAs are power hungry and generate an extraordinary amount of heat — making them a bad choice for today's server dense racks.

This really isn't the case. The amount of power that a FPGA uses is totally dependent on what it's doing at the time. At full utilization, the typical mid-range Intel FPGA in the data center uses anywhere from 25-75 watts, depending on the model and workload, with idle power levels much lower. Note: This range is for a sustained, not burst, workload — including complete data transfer.

FPGAs = Parallelism to the Max

The reason FPGAs can offer so much performance when compared other compute alternatives like GPUs is because FPGAs allow for much more parallelism. Systems configured only with CPUs or GPUs have fixed sets of computational units that are devoted to particular tasks which can't easily be changed by the user—it's locked in.

FPGAs are different because they have 10 to 100 times the number of computational units of CPUs or GPUs. They also have millions of reconfigurable logic elements, thousands of memory blocks, and thousands of DSP blocks, plus dozens of high speed transceivers and multiple configurable memory controllers. And some FPGAs (including the Intel FPGAs discussed below) have hardware dedicated to handling single precision floating point, which will give customers more performance while using less energy.

FPGAs have millions of reconfigurable logic elements, thousands of memory blocks, and thousands of DSP blocks, plus dozens of high speed transceivers and multiple configurable memory controllers.

Like building blocks, all of these components can be configured to exactly match the desired configuration for the application the customer wants to run. It's this ability to customize hardware that makes FPGAs so attractive for parallel applications.

A Closer Look at FPGAs: Intel

Intel is a leading FPGA provider actively pushing the limits of technology with its line of Cyclone, Arria and Stratix FPGA products. For example, its new Hyperflex architecture, with advanced retiming, pipelining, and optimization mechanisms, yields performance that is significantly higher than conventional FPGAs while using much less power.

A very hot area today is machine learning and other emerging elements of Artificial Intelligence. These types of applications, for example, teach machines how to recognize images or human speech.

One popular machine learning benchmark is AlexNet, a set of millions of images covering tens of thousands different image classifications. Systems run through these images, using algorithms to learn to recognize specific objects in the images.

Here's a comparison of how various platforms were able to perform on the AlexNet benchmark*. *(See table below).*

The Intel Arria was able to process 25.5 images/second/watt basis, and can complete the job using very little power when compared to other options. This chart also shows that the Arria FPGA can be run in half precision (FP16) configuration—nearly doubling throughput and significantly increasing performance per watt.

Platform	FPGA Power (watts)	Throughput (images/sec)	Throughput (images per watt)
Intel Arria 10-115 (FP32 @300MHz)	~35	575	16.4 images/sec/watt
Intel Arria 10-115 (FP16 @297MHz)	40	1,020	25.5 images/sec/watt
4x Intel Arria 10-115 (FP16 @297MHz)	160	4,080	25.5 images/sec/watt

Getting Started with FPGAs

Customers typically target a particular application that is giving them pain for acceleration. They often approach one of Intel's set of FPGA board partners and discuss the characteristics of the application they need to accelerate. At that point, the customer can leave it to the partner to customize the FPGA for the application or they can utilize the Intel FPGA developer kit and customize the application themselves.

Intel offers a large number of reference designs and programmable FPGA elements for a wide range of industries. For example, there are reference designs for genomics, big data analytics and artificial intelligence for HPC and data center applications. There is also a reference design for automotive manufacturing that incorporates Intel FPGAs to accelerate basic continuous wave frequency modulation for automotive radars, like those used in collision avoidance.

The company also offers a quite a few daughter cards that can be used in conjunction with their FPGAs to give applications more functionality. These include PCIe and HDMI cards to convey video, Gigabit Ethernet communication cards, camera link cards, and a host of others.

One of the most exciting future developments for FPGAs, and processing in general, is the potential for industry standard CPUs to be coupled with FPGAs. Intel has discussed plans to integrate its Xeon CPUs with their FPGA products, which would put incredible performance into one easy to implement package.

Intel has discussed plans to integrate its Xeon CPUs with their FPGA products, which would put incredible performance into one easy to implement package.

This combination would open up a whole new horizon of high performance for applications like image identification, encryption/decryption, security mechanisms like firewalls, and things like virtual switches with huge bandwidth and very low latency.

The future for FPGAs is very bright today. Any customer who is looking to increase application performance (and/or lower their power requirements) should take a close look at what FPGAs can do for them. These accelerators can fast-track any workload, and do it with excellent price/performance.

*Vanilla AlexNet Classification Implementation as specified by http://www.cs.toronto.edu/~fritz/absps/imagenet.pdf

Training Parameters taken from Caffe open-source Framework are 224x224x3 Input, 1000x1 Output

FP16 with Shared Block-Exponents

All compute layers (incl. Fully Connected) done on the FPGA except for Softmax

Arria 10-1150 FPGA, -1 Speed Grade on Altera PCIe DevKit with x72 DDR4 @ 1333 MHz

Largest mid-range part from newest family (1518 DSPs, 2800 M20Ks)

Power measured through on-board power monitor (FPGA POWER ONLY)

ACDS 16.1 Internal Builds + OpenCL SDK 16.1 Internal Build

Compute machine is an HP Z620 Workstation, Intel® Xeon® processor E5-1660 at 3.3 GHz with 32GB RAM. The Intel Xeon processor is not used for compute.