ASIC Cloud Trends

V.P.Sampath
Technical Architect
Adeptchips Private Limited, Bangalore

The bifurcation of computation industry into two modes such as the client and cloud. The client is the mobile SoC.Cloud is implemented by the accelerator. As transistors shrank, the necessary voltage and current; power is proportional to the area of the transistor. So, as the size of the transistors shrunk, and the voltage was reduced, circuits could operate at higher frequencies at the same power. As transistors get smaller, power density increases because these don't scale with size .These created a "Power Wall" that has limited practical processor frequency to around 4 GHz since 2006.

Need for ASIC Cloud

TCO improvement; vs TCO /NRE
TCO improvement determined by accelerator
TCO determined by scale of computation
NRE determined by ASIC design

Two for two rule

Moderate speed up with low NRE
Beats high speedup at high NRE

Build a model for NRE

Mask cost,ip license cost,labour cost, total cost, package NRE cost,Labour cost

Process node

Asic Process technology node from 250 mn to 16 nm gives us a range of:

256x in max accelerator size
15.5x in max tran freq
152 x in energy per op
28x in cost per ops
89x in mask cost

Verilog to TCO-optimized data rate

A joint specialized server and ASIC to optimize tco.Thermal option based on RCA properties, asic placement(DUCT layout),heat sink optimization(fins ,width,mask,depth),die size. voltage scaling is a first class optimization for TCO.

Core voltage increases from left to right.asic servers generally outputs best non asic in forms of tco

Wafer cost rise exponentially after 65nm jump on transistor to bigger wafers
Wafer diameter is 200mm until 180mm and 300 mm afterwards

Accelerator metric:

Energy efficiency (W per op/s)
Performance ($ per op/s)
Conventional trivial weighing
Energy delay product or energy product delay spread
Datacenter total cost of ownership as new mask

Cloud services are becoming increasingly globalized and data-center workloads are expanding exponentially. ASIC Clouds are not ASIC supercomputers that scale up problem sizes for a single tightly coupled computation. ASIC Cloud is an energy efficient, high-performance, specialized RCA (replicated compute accelerators) that is multiplied up by having multiple copies per ASIC, multiple ASICs per server, multiple servers per rack, and multiple racks per datacenter.

One of the primary goals of efficient data center infrastructure is to extract the maximum amount of compute capacity while expending the least power. At Face book, web servers must process an increasingly large number of requests simultaneously while responding to each individual request as quickly as possible. To keep up with this growing performance demand, they use processors with progressively more power over the past 10 years and redesigned web servers to pack more than twice the compute capacity in each rack while maintaining our rack power budget. This design provides a significant improvement in performance per watt over the previous generation-over-generation trajectory. With this system implemented, they achieve the same performance per watt today that would have otherwise required multiple new server generations.

Scaling Facebook's computing infrastructure to be as efficient and cost-effective as possible has been a consistent focus of our engineering efforts. The result was a one-processor server with lower-power CPUs, which worked better than the two-processor server for our web workload and is better suited overall to data center workloads. With this new system, not only were we able to avoid the flattening performance trajectory, but we could leapfrog the performance cadence we had been on for the past five server generations, as well. The system also operates within the same rack power budget, making our data centers more efficient than ever before. In Face book's cluster architecture, each cluster consists of more than 10,000 servers. Most of our user traffic comes through the front-end clusters, and web servers represent a major portion of the front-end cluster. These web servers run HHVM, an open-source virtual machine designed for executing programs written in Hack and PHP. HHVM uses a just-in-time compilation approach to achieve superior performance while maintaining the development flexibility that PHP provides. At a high level, this workload is simultaneously latency-sensitive and throughput-bound. Each web server needs to respond to a given user request quickly as well as serve requests from multiple users in parallel.

In CPU terms, we require good single-thread performance and excellent throughput with a large number of concurrent threads and architected this workload so that we can parallelize the requests using PHP's stateless execution model. There isn't much interaction between requests on a given web server, allowing us to efficiently scale out across a large number of machines.However, because we have a large code base and because each request accesses a large amount of data, the workload tends to be memory-bandwidth-bound and not memory-capacity-bound. The code footprint is large enough that we see pressure on the front end of the CPU (fetching and decoding instructions). This necessitates careful attention to the design of the caches in the processor.

Frequent instruction misses in the l-cache result in front-end stalls, which affect latency and instructions executed per second.web servers are heavily compute-bound and don't require much memory capacity, the two-socket servers we had in production had several limitations. The server has a QPI link that connects the processors, which created a NUMA problem. It also requires an accompanying chipset that required more power. We kept pushing the performance (and hence power) envelope. Intel provided us with 95W, then 115W, and now 120W-130W CPUs to hit our performance targets. Given an 11kW rack level power budget, pushing the limits of CPU power was not scalable, and we were not seeing performance keep up with increased power. Making a server-class CPU is difficult and something that we take for granted. Process transitions are also challenging, especially at such minuscule dimensions. We knew that Intel was solving a hard problem, but since our software was evolving at a rapid pace in parallel, we wanted to take this problem on as well and look at our system design through a new lens. There is a strong need for ASIC Clouds.Facebook runs face recognition on 2b pic/day.SIRI recognition speech has billion users..Youtube transcode to google for 500 hrs uploads per minutes.These incur high total cost of ownership for the provider.There is a need to reduce TCO.

Work requests from outside the datacenter will be distributed across these RCAs in a scale-out fashion. They target workloads comprising many independent but similar jobs.GPU and FPGA-based clouds have illustrated improvements in power and performance by accelerating compute-intensive workloads.ASIC-based clouds are a promising way to optimize the Total Cost of Ownership (TCO) of a given datacenter computation (e.g. YouTube transcoding) by reducing both energy

consumption and marginal computation cost. All system components can be customized for the application to minimize total cost of Ownership (TCO).Each ASIC interconnects its RCAs using a customized on-chip network. The ASIC's control plane unit also connects to this network and schedules incoming work from the ASIC's off-chip router onto the RCAs.Next, the packaged ASICs are arranged in lanes on a customized PCB and connected to a controller that bridges to the off-PCB interface .Specialized replicated compute accelerators (RCAs) are multiplied up by having multiple copies per application-specific integrated circuit (ASIC), multiple ASICs per server, multiple servers per rack, and multiple racks per datacenter.

Server controller can be a field-programmable gate array (FPGA), microcontroller, or a Xeon processor. The power delivery and cooling system are customized based on ASIC needs. If required, there would be DRAMs on the printed circuit board (PCB) as well. (PSU: power supply unit.)In some cases, DRAMs can connect directly to the ASICs. The controller can be implemented by an FPGA, a microcontroller, or a Xeon processor. It schedules remote procedure calls (RPCs) that come from the off-PCB interface on to the ASICs. Depending on the application, it can implement the non-acceleratable part of the workload or perform UDP/TCP-IP offload.

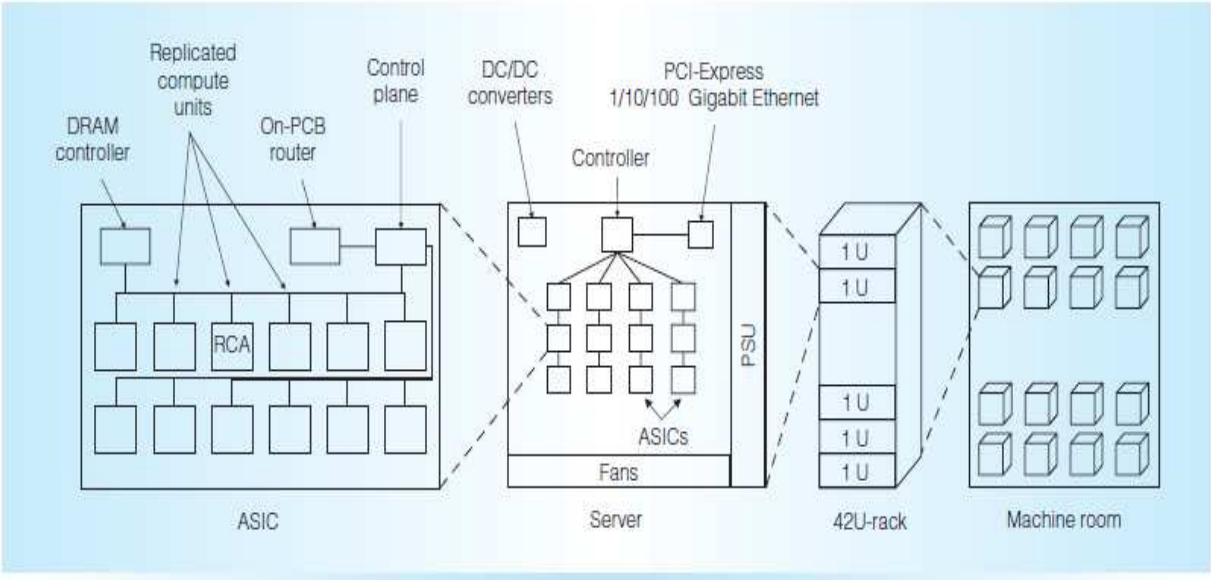

Figure 1.Architecture of an ASIC Cloud

The machine room at the datacenter has multiple 42U or 48U racks. Each rack is connected through high speed Ethernet to the external network.The racks have a central Ethernet switch (Top-of-Rack switch) which connects each of the server blades in the rack.

The server has a power supply unit (PSU) and cooling system. The off-PCB interface (10G Ethernet, PCI-e or other point-to-point links) delivers data to the server controller (FPGA or a micro-controller) which has access to an array of specialized ASICs.

Each of the ASIC consists of multiple accelerators called as the Replicated Compute Accelerators (RCA for brevity). The off-chip interface connects to the ASIC router and the controller which distributes the workload among the available RCAs through the internal network. Each ASIC may have on-chip clock generator or PLL, thermal sensor and power grid. In case of memory intensive applications, the DRAMs could be shared among RCAs with a memory controller on each ASIC.

Mahindra invites Indian startups to make a social network: Amid the Facebook data scandal, Billionaire businessman Anand Mahindra has invited "relevant" Indian startups that can make the country's own social networking company which is "widely owned&professionally managed&willingly regulated." "I'd like to see if I can assist with seed capital," the 62-year-old Mahindra Group Chairman added.

Google loses legal battle over Java, faces $9 billion fine: A US court has ruled that Google violated copyright law when it used computer software company Oracle's Java APIs to create the Android mobile operating system.