In Search Of The (10-Gbps Chip) Holy Grail

Michael Weingarten and Bart Stuck

To make the next bandwidth leap, we'll first need progress in components.

Michael Weingarten (mikew@signallake.com) and Bart Stuck (barts@signallake.com) are managing directors of Signal Lake, a venture capital fund (Westport CT and Boston MA). Weingarten also is managing director of Telecom Advisory Services at Monitor Group (Cambridge MA), a leading international strategy consulting firm. Signal Lake is an investor in Terago, one of the chip companies mentioned in this article.

As those of you who have seen the movie "Indiana Jones and the Last Crusade" know, the Holy Grail was the chalice used by Jesus at the Last Supper. In medieval Europe, it was believed that anyone drinking from the Grail would gain immortal youth. More than a few knights went "holy grailing"—wandering around in search of the elusive goblet—and the term has joined the lexicon as a synonym for the ultimate high risk/high reward venture. We think this aptly describes the search for 10-Gbps (and follow-on 40-Gbps) datacom/telecom chips.

The Upside Opportunity

The need for 10-Gbps and faster chips is obvious. DWDM, OC-192 packet over SONET (POS) and 10-Gig Ethernet all work at link speeds of 10-Gbps. State-of-the-art optical transmission soon will move to 40 Gbps. To handle these bit-streams efficiently at the edge of the optical network, we need electronics that work at these speeds.

Unfortunately, such chips do not exist. When you read announcements about 10-Gbps chips, they generally refer to upstream PMD/PHY/framer opto-electronic functions (see Figure 1 for a block diagram of a typical line card using 10-Gbps chips, and Table 1 for a description of chip element functionality). These involve the conversion of optical flows into electrical flows and breaking up the flows into discrete packets. That's not child's play, but it's relatively simple compared to routing and switching individual packets and/or flows at high speeds, not to mention the ability to incorporate next-generation QOS, VPN and MPLS functionality.

Unfortunately, 2.5 Gbps is the state-of-the-art in dataplane, control plane and switching, with most chips running no faster than 1 Gbps. We all can read announcements about "true" 10-Gbps chips, but virtually all of it has been just that—announcements, not volume shipments.

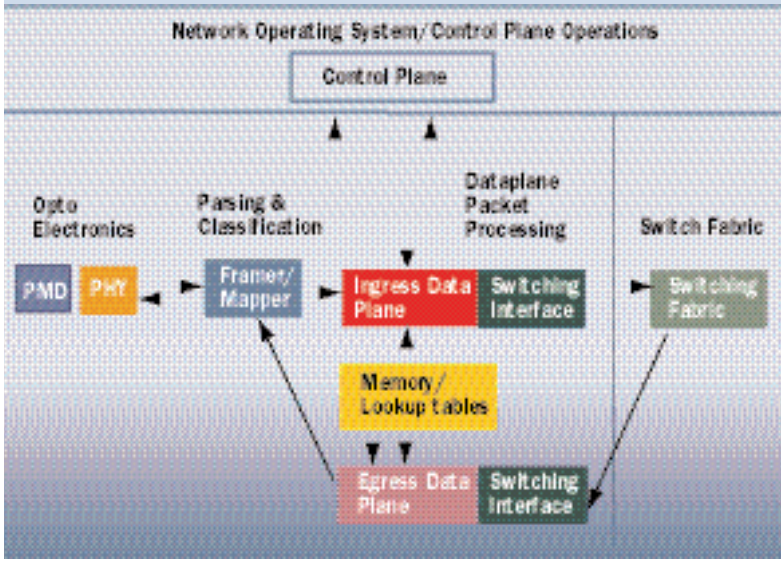
The lack of 10-Gbps speeds is particularly critical in the dataplane. This is where all packets must be classified, policed, shaped and routed before going into the switching fabric to be switched to the appropriate destination. If you want 10-Gbps throughput, you must have a 10-Gbps dataplane.

Making dataplane matters more complex, there is a fundamentally different logic for ingress versus egress processing. (Ingress refers to packets coming into your router from outside your network; egress refers to packets from your own network that you are forwarding out). Egress processing is the simpler of the two. You receive relatively homogenous packets from your own edge devices (which you presumably have some control over, and therefore greater uniformity), so egress traffic management (i.e., classification and scheduling) is relatively straightforward.

The really difficult task is ingress processing, where you receive a non-homogenous set of packet types and lengths from a variety of external sources. All of those packets must then go through parsing, classification, packet editing, metering, policing and admission control. So, arguably, there are two related but separate dataplane design issues—ingress and egress—neither of which are yet resolved so as to provide 10-Gbps throughput.

Two other elements, the control plane and switching fabric, are important but less critical than the dataplane. The control plane, in contrast to the dataplane, does not need to operate at

FIGURE 1 10 Gigabit Line Card



Using multiple, lower-speed chips would be five times the cost of using 10-Gbps chips

TABLE 1 Chip Element Functionality

Topology Element [Optical input/output]	Functionality
PMD (Physical Media Device)	Physical interconnection with optical fiber; conversion from optical to electrical signal.
PHY (Physical Layer Device)	Clock regeneration; multiplexing/demultiplexing into several parallel electrical bit-streams. PMD and PHY are often consolidated into a single chip, with framer/mappers at slower speeds.
Framer/Mapper	Framer segments electrical bit-streams into packets; mapping converts data from one protocol to another.
Data Plane	Routing of the data stream: Parses, classifies and processes the bit-stream in a manner that allows packets to be switched in the switching fabric. Handles QOS functionality such as policing/metering, marking/coloring, and admission control/dropping. At 10 Gbps speeds, there are separate logic paths for ingress and egress.
Memory	External memory that works with data plane chips, e.g., CAM (content addressable memory for routing lookup tables).
Fabric Interface	Allows seamless integration between the data plane and the switching fabric. Can be done within the dataplane chip or via separate custom chip or FPGA.
Switching Fabric	Actual crossbar switching functionality.
Network Operating System/ Control Plane Operations	Overall network control plane and operating functionality, interfacing with all other chip components

10 Gbps to ensure 10-Gbps line rates. That's because the control plane typically only handles exception packets—i.e., those whose non-standard characteristics require special processing; for example, unusual packet destinations that aren't in lookup table memory. Since in a real-world environment, one out of five to 10 packets requires special handling, the control path only needs to operate at 1–2 Gbps.

To have 10-Gbps system throughput, the switching fabric also needs to operate at 10 Gbps, but the dataplane remains, arguably, the single most difficult 10-Gbps chip element. Its functions are more complex and involve intensive interaction with the memories, control plane and switching fabric. We therefore will focus on dataplane chip development in this article.

Current Workarounds

In the absence of 10-Gbps dataplane chips, the fallback is to use mux/demux techniques and move to parallel processing—4×2.5 Gbps or even 10×1 Gbps. However, if you choose these alternatives, you suffer from 4–10 times the processing streams. Aside from the non-trivial complexities of making these streams interface seamlessly, designing a system with 4–10 times the chips generally means a 4–10-times reduction in port density, a 4–10-times increase in required shelf space as well as power/heat consumption, and a cost penalty of 2–5 times (assuming that one 10-Gbps chip would sell for twice the price of a 2.5-Gbps chip).

Therefore, if we assume that an OC-192 card using 1-Gbps chips would require about 50 chips (10 chips to handle each of the five main functions in Figure 1), an OC-192 board would require a footprint approximately 15″×20″. At a unit power consumption of 2 watts per chip, the board would consume 100 watts. At a unit chip cost of $200, the full board would cost $10,000 (the selling price to systems vendors would be 3–4 times this; retail price to end users would be 9–12 times this, which would make the final price $90,000–$120,000).

In contrast, consider the comparative economics for an OC-192 board using 10-Gbps chips. Such a card would have five chips (one each for control plane, ingress dataplane, egress dataplane, external memory and switching fabric), would require a footprint of 15″×5″ and have about 30 watts of power dissipation per board, at a manufacturer's total cost of $2,000—one-fifth the cost of the example above.

So, clearly, routers can be designed with 10-Gbps input/output ports using lower-speed chips. The problem is that it will cost a lot more money than with 10-Gbps chips, it will be much bigger than a breadbasket and you'd be able to fry eggs on it. To see ubiquitous broadband any time soon, it's going to take 10- and then 40-Gbps dataplane and switching fabric chips.

The Market Understands This

Market analysts understand this. That is why leading Wall Street analysts are forecasting very high growth and substantial market size for OC-192 chips—even in the current Nasdaq depression.

Chip companies and entrepreneurs also understand this. This is why there have been lots of 10-Gbps startups. Table 2, (p. 48) lists 22 chip startups acquired during 1999–2001 by major chip companies and system suppliers for a total cost of $19 billion. Each company was sold for very high prices ($300 million+), despite the fact that the acquirers were buying very small companies (35 engineers) with no working product or even the prospect of a product within a year. Even in the current depressed environment, Cisco's recent $150 million acquisition of the 55 employees of AuroraNetics cost $2.72 million per employee for a one-year-old company.

The Great Chance Of Failure

As previously noted, part of the upside is based on the fact that no one so far has succeeded in developing 10-Gbps datapath and switching fabric chips. Indeed, there have been some expensive, painful blowouts. Even where players are still alive and kicking, there have been numerous delay announcements.

In our opinion, the fundamental driver behind the failures to date is the players' inability to understand important architectural design issues and to make appropriate choices. If so, the solution is to make the appropriate choices, and the players with the best chances at winning will be those who make these choices soonest.

Underestimating The Dataplane Problem

The first issue has to do with underestimating the difficulty in passing a realistic, 10-Gbps traffic stream through the dataplane, combined with overestimating the ability of certain technologies to handle those streams.

It's relatively easy to create chips that can process uniform 40-byte packets with no header exceptions. It's much more difficult to handle a flow of asynchronous packet sizes with lots of classification problems. When you do that, you get what chip engineers call "bubbles"—bottlenecks in the process flow in which packets back up. Unless your chip design takes these real world considerations into account, you will never create commercial products.

Broadly speaking, there are three basic approaches to designing datapath chips. The first is to design application specific integrated circuits (ASICs), in which the necessary programming is embedded in the chips. To handle exception processing, these datapath ASICs will communicate with control path chips.

The basic advantage of ASICs is that by hard-coding the appropriate software in silicon, they are inherently faster than software-based microprocessors. The disadvantage is that their embedded software is less adaptable; it takes time-consuming and expensive foundry "re-spins" to accommodate individual customer needs or to update requirements. To deal with this, companies developing ASICs typically adopt a "configurable" approach, in which certain logic elements of the chip incorporate downloadable micro-code software that can be changed/customized.

The second approach is to use network processor chips (NPs). NPs are general-purpose arrays of microprocessor cores that are software programmable. With the appropriate programming, one can program a single NP that can handle datapath as well as control path functionality. The advantages of this approach include:

- Having both datapath and control path in the same chip obviates the need to pass data across two or more chips.
- The greater programmability of NPs makes it easier for individual systems vendors to customize chip functionality for their particular needs.

The third approach is to develop customized multiprocessing chips, which use arrays similar to those in NPs, but with custom processors that are designed to perform certain specific networking functions. In some ways, these arrays are a hybrid of microprocessor and ASIC functionality on a single chip.

Which of these are appropriate solutions to running datapaths at 10 Gbps and 40 Gbps? In the past few years, major players such as IBM, Intel, Broadcom, AMCC (via MMC), Motorola (via C-Port) and Vitesse (via SiTera) made very large bets on network processors. While 2–3 years ago, NP speeds only ran at 300–400 Mbps, believers in Moore's Law figured that with a few doublings, much higher speeds were possible, but ignored the fact that going from 400 Mbps to 10 Gbps entails more than four doublings!

Since NPs therefore were expected to reach the magic 10-Gbps dataplane mark—eventually—they were seen by their proponents as the appropriate architecture choice because you could integrate dataplane and control plane on a single chip for much lower system cost, while retaining maximum software reprogramming flexibility. Just as general-purpose microprocessors won the battle in the PC arena over ASICs, the winning NP vendor would become the Intel of networking devices and have a market cap in the tens of billions. Or so the logic went.

Unfortunately, using NPs in the dataplane for high-speed applications has proven to be a gigantic bust. The fastest commercially available NPs now run at 2.4 Gbps, but even this is only a published benchmark. When you run one of these chips against a realistic test flow, the real speeds are substantially slower. Furthermore, if you really take advantage of the customization capabilities, throughput speed drops dramatically—due to bubbles in the process flow.

So net-net, we are skeptical that NPs will be the solution for 10/40-Gbps datapath throughput. This does not mean that NPs won't be critically important for the control path, since this involves much slower—1–2 Gbps—data flows. It also does

Network processors will be more important for the control path than the datapath

TABLE 2 1999-2001 Chip Company Acquisitions

Fabless IC Company	Acquirer	Acquisition Price	Date Announced	Market Focus
Stratum One	Cisco	$450 M	6/29/99	Traffic Management
Abrizzio	PMC-Sierra	$400 M	8/24/99	Switch Fabric
Agere	Lucent	$415 M	1/20/00	Switch Fabric
Growth Networks	Cisco	$355 M	2/16/00	Switch Fabric
C-Port	Motorola	$430 M	2/22/00	Network Processor
Extreme Packet	PMC-Sierra	$415 M	3/3/00	Traffic Management
AANetcom	PMC-Sierra	$890 M	3/3/00	Switch Fabric
GBPSA A/S	Intel	$1,250 B	3/15/00	SONET/SDH, Ethernet
Basis	Intel	$450 M	3/21/00	Network Processor
Orologic	Vitesse	$450 M	3/27/00	Traffic Management
YuniNetworks	AMCC	$241 M	4/20/00	Switch Fabric
SiTera	Vitesse	$750 M	4/20/00	Network Processor
HotRail	Conexant	$394 M	6/28/00	Switch Fabric
Quantum Effect Devices	PMC-Sierra	$2,300 B	7/12/00	Network Processor
Silicon Spice	Broadcom	$1,200 B	8/12/00	Network Processor
NewPort	Broadcom	$1,240 B	8/15/00	SONET/SDH, Ethernet
MMC Networks	AMCC	$4,500 B	8/28/00	Network Processor
SwitchOn	PMC-Sierra	$450 M	9/26/00	Traffic Management
Allayer	Broadcom	$274 M	10/17/00	Gbps Ethernet
SiByte	Broadcom	$2,040 B	11/7/00	Network Processor
Lara Networks	Cypress	$225 M	6/8/01	Network Processor
AuroraNetics	Cisco	$150 M	7/11/01	Gbps Ethernet

Source: Company Press Releases, Signal Lake

not mean that NPs won't eventually get to higher speeds, particularly as chip-trace spacing—the spacing between copper "wires" on a chip—decreases from .18 microns to .13 and .11. The closer the spacing, the greater the chip performance per square mm.

However, it's likely that other solutions will get there first, since they too can take advantage of trace improvements. Configurable ASICs are our top choice.

If you design an ASIC optimized for throughput with embedded logic, but which can send exception packets to the control path for special processing, it is inherently easier to reach 10-Gbps speeds with ASICs than with NPs.

The knock against ASICs is that they are insufficiently reconfigurable. But, ASICs with downloadable microcode enable some degree of customization and, when you come down to it, there aren't that many features that you really need in the datapath anyway, beyond a relatively standardized IETF set of QOS, VPN and MPLS features. To the extent that you want to program lots of customized application-layer switching features, these will probably be coded at lower speeds farther out in the periphery of the network.

What about custom multiprocessors? To the extent that some of the custom multiprocessor architecture is executed in a manner that mimics the advantages of ASICs—putting the functional equivalent of microprocessors and ASICs (with the advantages of both) on one chip, as opposed to simply having an array of custom microprocessor cores—this approach could work.

However, even with this alternative, you need to crawl (albeit very fast) before you can walk. It's easier to separate ASIC from microprocessor functionality for first-generation 10-Gbps chipsets, rather than trying to get there in one giant step. While doing it all together might ultimately be a strong solution, the degree of complexity is greater and therefore the risk of failure is higher.

The Need For System Integration

Developing high-speed datapath chips is crucial, but it's also irrelevant if the datapath doesn't integrate with the rest of the system. Handling networking information is a multichip process flow, in which the pieceparts need to work together. In particular, the dataplane needs to interface closely with the control plane for exception processing. It also needs to be able to interface seamlessly with the switching fabric with respect to such issues as queuing, flow control and speedup.

These interfacing issues are not trivial, and are much more complicated than simply matching the interface speed between chips. If you don't spend a great deal of effort in treating the control plane, dataplane and switching fabric as a system, your chips may not work at all, may not work under a number of conditions and/or experience bubbles that degrade performance substantially.

In this context, the real issue is not who announces that they have sent their first chip to foundry, or even that they have umpteen design wins with people like Cisco, Nortel and Juniper. The winner in 10-Gbps will be the player developing chips *that work together as a system.* This does not necessarily mean that the winner needs to develop every single chip in the chipset, although the ability to do this may be a competitive advantage. It does mean that each chip and, in particular, the dataplane and control plane, must work together as though they were developed jointly.

The need for a systems approach also means that the winning chip developer will not be the player with the best and brightest ASIC electrical engineers. That's necessary but not sufficient. Instead, a successful chip company will have as many software engineers as ASIC designers—possibly more—many of whom will have substantial experience in systems design.

The datapath must not only be fast—it must integrate with the rest of the system

It also will be characterized by extensive software libraries and reference designs that allow systems integrators to develop their system-level products quickly. Indeed, the chip designer that has figured out all the inter-chip issues might be in the position to provide complete board-level solutions ready for system integrator software customization, increasing its relative value-add at the expense of systems integrators.

Where these system skills do not exist, one will tend to see vendors arguing that they use standard interfaces and that their devices are ready to be custom-programmed by their system integrator customers. This argument ignores several things:

- The vendor likely has ignored interface issues in its haste to get chips out and underestimated the difficulty in doing the systems interfaces.
- The system integrator, not knowing the detailed logic flow of the chip it is purchasing, may not know enough to program the interfaces in a way that avoids bubbles.
- Designing these interfaces will almost certainly, substantially delay time to market.

Even a successful software interface in this scenario—i.e., an interface between chips that don't work together "gluelessly"—probably will need to be implemented via field programmable gate array (FPGA) chips interfacing between the dataplane and control plane, and between the dataplane and switching fabric. Unfortunately, FPGAs running at 10 Gbps cost $1,000 each, so the added software "glue" comes with a real economic penalty.

Another danger sign is when a chip company acquires a control plane from one startup, a dataplane from another and a switching fabric from a third, then announces it has a "complete" solution. Simply owning one of everything doesn't mean having a complete system.

One Additional Driver: Reduced Chipset Count

Developing the first 10-Gbps systems solution is great. However, over the longer term, the real winner will be the one able to put the greatest amount of functionality on a single chip at any point.

The NP approach combining dataplane and control plane in one chip would have been nice, but it isn't workable for now. Instead, we are seeing solutions with different chips for control plane, ingress dataplane and egress dataplane. Even within a single function such as egress dataplane, we are seeing solutions that require a two-chip set. That's great, and it may be a good way to get to market quickly. However, at big bucks per chip, the cost of the extra chips quickly adds up, particularly at the telecom service-provider level with two successive markups.

If a player can do ingress or egress dataplane in a single chip, that's better than two. Even better, if a player could do ingress and egress dataplane in a single chip, which no one has announced to date, it would allow much higher port densities and much lower systems costs. So, part of the competitive game will be reduction of the number of chips in the chipset, and the elimination of expensive FGPA interfaces.

And of course, this whole process will repeat itself with 40 Gbps…

Conclusion

The good news for *BCR* readers: Unlike the Holy Grail, which was an ultimately futile effort, 10-Gbps chips should show up in commercial quantities sometime in 2002. This will have profound effects for the economics of 10-Gbps Ethernet and other high bandwidth applications□

Companies Mentioned In This Article

Agere (www.agere.com)
AMCC (www.amcc.com)
Broadcom (www.broadcom.com)
Cisco (www.cisco.com)
Cypress Semiconductor (www.cypress.com)
IBM (www.ibm.com)
Intel (www.intel.com)
Lucent (www.lucent.com)
Motorola (www.motorola.com)
PMC Sierra (www.pmcsierra.com)
Terago (www.terago.com)
Vitesse (www.vitesse.com)