Technical Primer

Re-loadable Direct eXecution (RDX™) Architecture from Mistletoe Technologies

Given the ever-changing characteristics of today's networks – with all of the new protocols and formats that networks must support – having an adaptable architecture is essential to keeping equipment up-to-date and capable of supporting new features and advanced functionality. ASIC-based approaches offer a performance advantage, but are simply too fixed in implementation, limiting their usefulness. ASICs cannot easily adapt, for example, to accommodate new security or encapsulation protocols.

On the other hand, traditional programmable processors are based on the von Neumann architecture. In this architecture, programs are executed sequentially and performance is directly tied to the number of instructions a processor can perform each second. In order to double performance, it is necessary to double the clock speed/operating frequency of the processor. For many applications, scaling performance in this way is not feasible; moving from 100 Mbps performance to 1 Gbps security processing throughput requires a minimum 10X increase in operating frequency. Increasing a processor's clock rate by 10X is not a trivial exercise of engineering. In many cases, it simply isn't possible.

One alternative to significantly increasing clock speed is implementing multiple processors to process packets in parallel. Coordinating the action of multiple processors, however, increases the complexity of a system tremendously. In addition to processing packets at wire speed, the system must now efficiently route packets of varying length (and thus overall processing time) between multiple processors, manage contention of resources shared by these processors, and then bring back together all of the packets for forwarding. In many cases, the challenge of programming multiple processors can be as or more difficult to address than the actual packet processing task.

Direct Execution vs. Traditional von Neumann Processing

In order to achieve the increases in performance required to operate at today's higher link speeds, it is necessary to introduce a new processing architecture that marries the flexibility of von Neumann programming architecture with scalable parallel processing that is independent of operating frequency. This is what the Re-loadable Direct eXecution (RDX™) family of processors from Mistletoe offers.

From an abstract perspective, both von Neumann and RDX architectures handle packet processing in the same way: as a packet arrives, its various pieces are peeled off, identified, and then processed appropriately. The difference in overall performance, however, couldn't be more different. Where a von Neumann-based processor implements this process in programmable software, a Mistletoe RDX-based processor implements it in hardware governed by a programmable grammar-based hardware parser, leading to greater performance gains through hardware acceleration and syntax-directed processing.

Data-Driven Efficiency: In a von Neumann architecture, software determines what the processor should do next. As a packet begins to arrive, the processor must identify what type of packet it is, extract header information, process the data, and so forth. In order to identify, for example, what kind of packet has arrived, the processor must use a CASE structure (sometimes implemented as a series of IF-THEN statements). Each possible type of packet must be captured in the CASE structure, and the processor has to cycle through all of the options until it has successfully identified the packet.

In comparison, the Mistletoe RDX architecture is a data-driven flow. Once the packet type information is available, the parser locates the type in the appropriate grammar table. This search operation takes a single cycle and results in the parser launching the appropriate processing function in an available Direct Execution Engine (DEE). This search mechanism is implemented using ternary content addressable memory (TCAM) technology. As a result, the parser takes the same amount of time to search a grammar no matter how many entries it has.

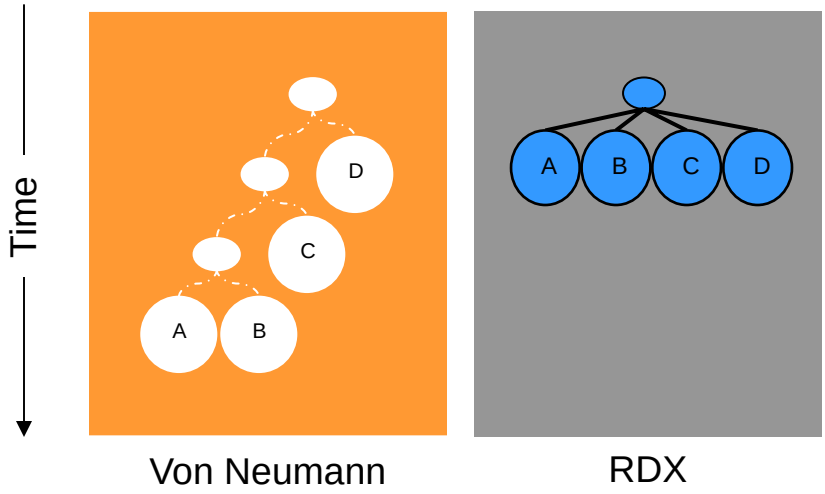

Figure 1. With Mistletoe RDX technology, data drives decisions rather than the traditional process of instructions driving decisions. Data-driven processing significantly improves performance and efficiency, while reducing latency, buffering, and power requirements.

CASE structures introduce a number of processing inefficiencies. Because it takes multiple processor cycles – range comparisons take twice as many – to evaluate each stage in a CASE structure, the more possible options, the longer the overall process can take. Depending upon the number of options, a CASE structure can require tens of cycles to find a match compared to a single cycle for an RDX parser. Additionally, the CASE structure utilizes branching instructions which flush the instruction pipeline once a match is found, adding further latency.

16-to-1 Parallel Processing: A von Neumann architecture can only execute one instruction at a time. For example, the processor must focus on the incoming packet to

identify an incoming packet segment. Once a segment has been identified, the processor must interrupt the identification flow to process the segment. Meanwhile, the rest of the packet continues to arrive and must be buffered until the processor can return its attention to the identification flow. Once the next segment is identified, the flow is once again interrupted to process the segment, and so on. Since one processor core must perform all segment identification and segment processing, the overall latency for processing the packet is maximized.

In contrast, the parallel processing of a Mistletoe RDX processor requires the parser to only identify a segment. Once the segment is identified, it is handed off to an independent DEE for processing, and the parser starts in on the next segment. For complex packet traffic, the 16 DEEs of an RDX processor enable parallel processing that easily provides more than 10X performance over von Neumann-based processors.

Simple Software Architecture: Because von Neumann-based architectures handle multiple tasks, including buffering new packets as they arrive and even potentially terminating TCP/IP, these systems require a fairly powerful operating system to manage switching between real-time tasks and application tasks. Multitasking increases overhead and design complexity. In the end, it can take thousands of instructions to process even the simplest packet.

Given that the parser is the heart of the RDX architecture, there is no need for multitasking or even an operating system since packets are parsed and processed as they arrive. RDX-based processors do not even need a kernel, nor do they utilize stacks, as these would serve no useful purpose in a data-driven architecture. This greatly simplifies how software is written and, by eliminating multitasking overhead, increases overall performance efficiency.

Zero Handoff Overhead: Once a von Neumann processor has identified a packet segment, it calls the appropriate processing function. In addition, it must handoff relevant packet information to the function, either by loading global variables or passing parameters over the stack. This extra handling of data incurs more than just cycle penalties because the extra load/stores associated with passing data increases latency due to memory access delays. High-performance von Neumann processors often execute faster than memory accesses occur, so the processor may need to stall to pass all of the relevant data.

In an RDX architecture, passing of relevant data is simplified as the parser only needs to

hand off a single pointer to a DEE and utilizes an optimized cache implementation. This increases efficiency by eliminating otherwise unnecessary load/store cycles and memory accesses.

Summary

Processors based on a von Neumann architecture can no longer provide the performance required for high performance network appliances. Relying on faster system clocks simply cannot increase processing performance enough to meet wire-speed constraints for 1 Gbps and 10 Gbps applications.

Mistletoe RDX provides extreme efficiency and high performance through a programmable, grammar-based hardware parser and 16 parallel DEEs which provide more than a 10X performance advantage over equivalent von Neumann processors. With a simple architecture that minimizes processing overhead married with embedded software, Mistletoe RDX makes it easier to deploy core appliance features in a straightforward manner. Mistletoe Technologies has made it possible to cost-effectively maintain high performance processing by implementing security functionality using RDX technology.