NETRONOME

WHITE PAPER

NFP-4000 Theory of Operation

THE NETRONOME NFP-4000 DEVICE FAMILY IS PURPOSE-BUILT FOR SERVER-BASED NETWORKING APPLICATIONS.

CONTENTS

INTRODUCTION

The Netronome NFP-4000 device family is purpose-built for server-based networking applications. Server-based networking deployments have become mainstream in COTS servers and this includes cases where networking functions are implemented in-line, between the network port on PCIe server adapters and host applications, or virtual machines (VMs) and containers implemented in servers. Examples of networking functions include network virtualization, security, load balancing, quality of service, and telemetry. This also includes networking functions implemented in VMs (as in virtual network functions or VNFs).

The purpose of this document is to describe the theory of operation (TOO) of NFP-4000 family of Network Flow Processors (NFPs). The document is intended to complement the NFP-4000 product brief and Agilio OVS Software user guide.

This document begins with an overview of NFP-4000 family of devices, followed by architectural highlights that describe the key functional units within the device. The document then describes data flow operations utilizing the functional units, in both the ingress and egress directions. Key architectural features and differentiations related to delivering optimal price/performance for server-based networking functions are discussed.

NFP-4000 ARCHITECTURAL HIGHLIGHTS

The techniques that are used to deliver high performance in general purpose server processors are not the same as what is needed to efficiently process network traffic. This is because server CPU architectures are fundamentally designed and optimized for running data center applications, which by their nature are relatively long-lived programs that benefit from very high single threaded performance. High-speed network packet processing, on the other hand, requires context switching on the order of nanoseconds and requires very high degrees of processing parallelism to scale the performance.

To solve this problem, the NFP-4000 uses massively parallel processing and multithreading techniques to achieve high packet processing performance. NFP-4000 device incorporates a large pool of 32-bit processing elements referred to as Flow Processing Cores (FPCs). These FPCs are purpose-built and optimized for packet and flow-based processing.

THE PRIMARY USE CASE FOR THE NFP-4000 DEVICES ARE 10 AND 40GBE INTELLIGENT SERVER ADAPTERS (ISAS) FOR SDN CONTROLLED SERVER-BASED NETWORKING.

As packets are received from the network, an FPC thread picks up the packet and processes it. Additional threads are allocated to new packets as they arrive. NFP-4000 supports up to 60 FPCs and each of the FPCs supports 8 threads. Hence, the device will be able to process up to 480 packets simultaneously, and as a result deliver high performance and maximum efficiency. The large number of threads hides the effects of memory latency. There are also powerful hardware accelerators that offload mundane tasks from the FPCs, thus preserving valuable instruction cycles. Packet modifiers, statistics engines, load balancers, lookup engines, transaction engines, bulk memory engines, and traffic managers are some examples of hardware accelerators present in the device. Finally, the whole architecture is connected via a high performance distributed switching fabric that provides high bandwidth mesh connectivity between all the entities in the device.

The primary use case for the NFP-4000 devices are 10, 25 and 40GbE SmartNICs for SDN controlled server-based networking. The NFP-4000 is designed specifically for such a scenario to be used as an in-line coprocessor for x86 general purpose processors. The device is equipped with high performance PCIe interfaces that are capable of high-speed data/packet transfer between the host and the NFP. In addition to the hardware capability, programmable elements within the PCIe interface allow for additional optimizations through software control for packet transfers.

NFP-4000 FUNCTIONAL UNITS

Please refer to Figure 1 while reviewing this section. The functional units have been described in the left to right sequence.

The NFP-4000 is designed using a modular architectural approach that is composed of multiple islands with local instantiations of a distributed switching element. This innovative methodology allows for the architecture to scale up or down seamlessly, according to application needs, and also allows Netronome to deliver devices to market at different levels of scale in rapid succession. This section briefly describes the various islands and the distributed switching fabric entity that interconnects all the islands in a scalable fashion.

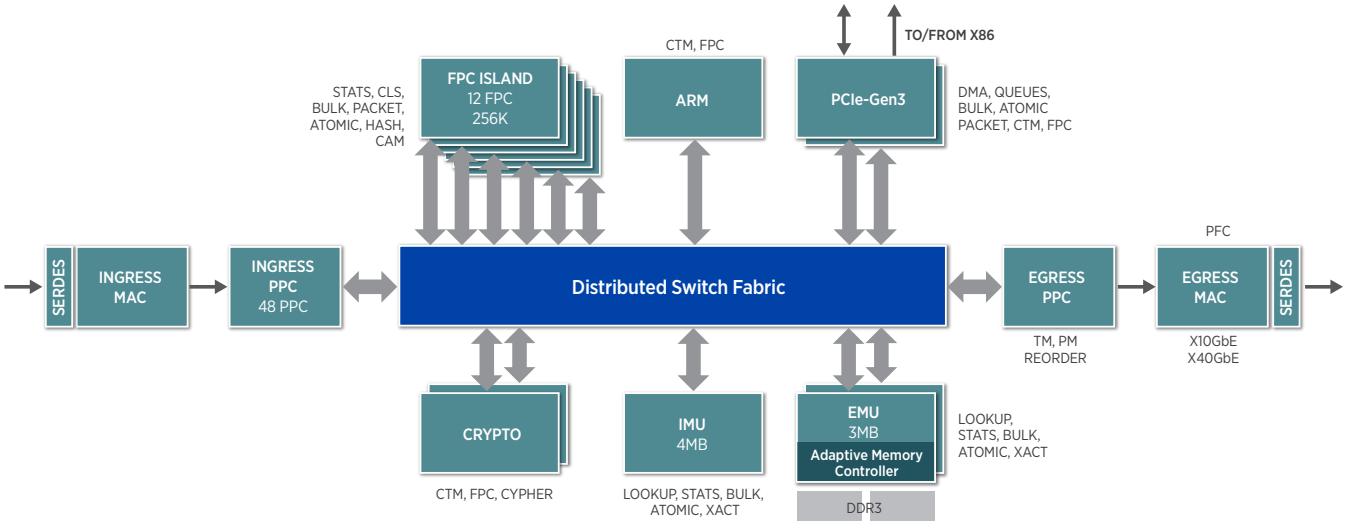

Figure 1. NFP-4000 Micro-architecture

MAC

The NFP-4000 MAC block includes support for standard x10GbE, x40GbE and x100GbE interfaces. The interfaces are natively capable of auto-negotiation with the attached platforms and can directly interface to standard optical, as well as direct attach copper, transceivers. The interfaces support datacenter bridging (DCB) requirements, including queuing structures and flow control requirements for Priority-based Flow Control (PFC) and Enhanced Transmission Selection (ETS) features. The MAC block time-stamps and performs all the required integrity checks on the packets and stores the packets into an intermediate buffer.

Packet Processing Core (PPC) Island

The Packet Processing Core island (PPC) is connected to an integrated MAC block providing standard network connectivity including 10GbE and 40GbE interfaces. In the Network-To-Host (Ingress) direction, the PPC island performs wire-speed flexible packet processing functions including packet header parsing, classification and load balancing to the rest of the functional units within NFP-4000 device. In the Host-To-Network (Egress) direction, the PPC supports hardware accelerators, including Packet Modification Engine, Traffic Manager and Reorder Engine. These accelerators are fully configurable and can be controlled via metadata appended to every packet.

Flow Processing Cores (FPCs)

As shown in Figure 1, the NFP-4000's 60 flow-processing cores (FPCs) are grouped in clusters and are also spread across multiple islands. The FPCs are 32bit custom cores capable of supporting eight threads each, which allow them to hide latencies typically associated with input/output or memory commands. Each FPC supports a large number General Purpose Registers and has access to a dedicated instruction and data memory.

As described in the NFP-4000 Software Data plane section, the FPCs could be programmed using either a "Functional Pipeline" or "Run To Completion" programming model. The ability to support multiple programming models provides the software architects tremendous flexibility to address a broad spectrum of applications with different processing and performance characteristics.

Crypto Security Acceleration

This unit supports a comprehensive portfolio of crypto algorithms. Data is fed into the crypto engines via the distributed switch fabric. This engine supports AES, 3DES, ARC4, Snow 3G, Kasumi, SHA 1, SHA-2, and MD5 bulk-encryption and hash algorithms. In addition to the cipher suite, FPCs are supported by the Crypto block for IPsec and SSL protocol processing at high performance in a flexible manner. NFP-4000 supports two crypto Acceleration islands that together deliver a high performance IPsec solution. Each crypto island on the NFP-4000 also supports four additional FPCs.

Arm Subsystem

The Arm functional unit is primarily used for configuration and initialization of the NFP-4000 device. A standard Linux operating system could be ported on the Arm subsystem to support applications in stand alone configurations. The Arm subsystem on the NFP-4000 also supports four additional FPCs for additional flow processing.

As shown in Figure 1, the Arm Island is directly connected to the switching fabric via a high-performance switching interface. This allows for efficient data transfers between the Arm island and the rest of the islands in the system. This feature allows applications to steer specific control packets from the network interfaces to be first classified by the PPC island and then delivered to the Arm island for exception path processing. The high performance datapath to and from the Arm island also allows for support of applications that require direct connectivity of a specific “out of band” network port to the Arm processor. In this scenario, the software on the Arm processors handles all the “out of band” management handshake and processing requirements.

External and Internal Memory Units (EMU & IMU)

The NFP-4000 supports multiple memory units organized in a hierarchical fashion. The Internal Memory Unit (IMU) supports 4MB of SRAM and the External Memory Unit (EMU) supports 3MB of SRAM. Each of these memory units supports purpose-built hardware accelerators to deliver wire-speed packet and flow processing capabilities. The EMU device could also be configured as a data cache for storing frequently used flow table entries while connected to high capacity external DDR3 memory. The NFP-4000 device supports one IMU and two EMUs. Each of these memory units also provides hardware acceleration support for accelerated lookup, load balancing, bulk and atomic transfers.

Adaptive Memory Controller

The NFP-4000 is also capable of supporting 8GB of external DDR3 memory. Given the large internal memory, external DDR3 is not required for the chip to operate. However, the use of external memory is recommended and provides the capability to scale the number of flows of any given application. The DDR3 memory in such a situation is used primarily to store flow table entries. The memory controller incorporates a large amount of internal memory that can adapt as a cache to store the most popular entries entirely via software control. The DDR3 could also be used as a packet buffer.

PCIe Gen3 Subsystem

The NFP-4000 has two PCIe Gen3 x8 interfaces. For most applications a single interface is used, but the device also supports direct simultaneous connectivity to two CPU sockets,

which can be useful for applications sensitive to NUMA effects. Each PCIe Gen3 interface is capable of supporting up to eight lanes simultaneously. Hardware capabilities for tasks similar to load Balancer, DMA, queuing, SR-IOV are natively supported by both the NFP-4000 PCIe Gen3 interfaces, resulting in applications being able to perform at wire-rate and low latencies.

Standards Compliant Scalable Ethernet Interfaces

The NFP-4000 device includes native support for standard 10G, 40G and 100G interfaces. The various device SKUs support the same footprint allowing customers to reuse one hardware design. The interfaces are natively capable of auto-negotiation with the northbound platforms and can directly interface to standard optical, as well as direct attach copper, transceivers. The interfaces support datacenter bridging (DCB) requirements including Priority-based Flow Control (PFC) and Enhanced Transmission Selection (ETS) features.

NETWORK-TO-HOST DATAPATH

This section discusses a "Day in the life of a packet " from Network to x86 for a generic application. Please refer to Figure 2 while reviewing this section.

Ingress MAC

In the "Network-To-Host" direction, the packet is received by the Ingress MAC. The MAC is capable of supporting 10GbE, 40GbE and 100GbE interfaces. The MAC block also natively supports queuing structures as well as handshaking required for PFC. After performing integrity checks on the packets, the MAC block stores the packets into an intermediate buffer. The packets can also be time-stamped by the MAC.

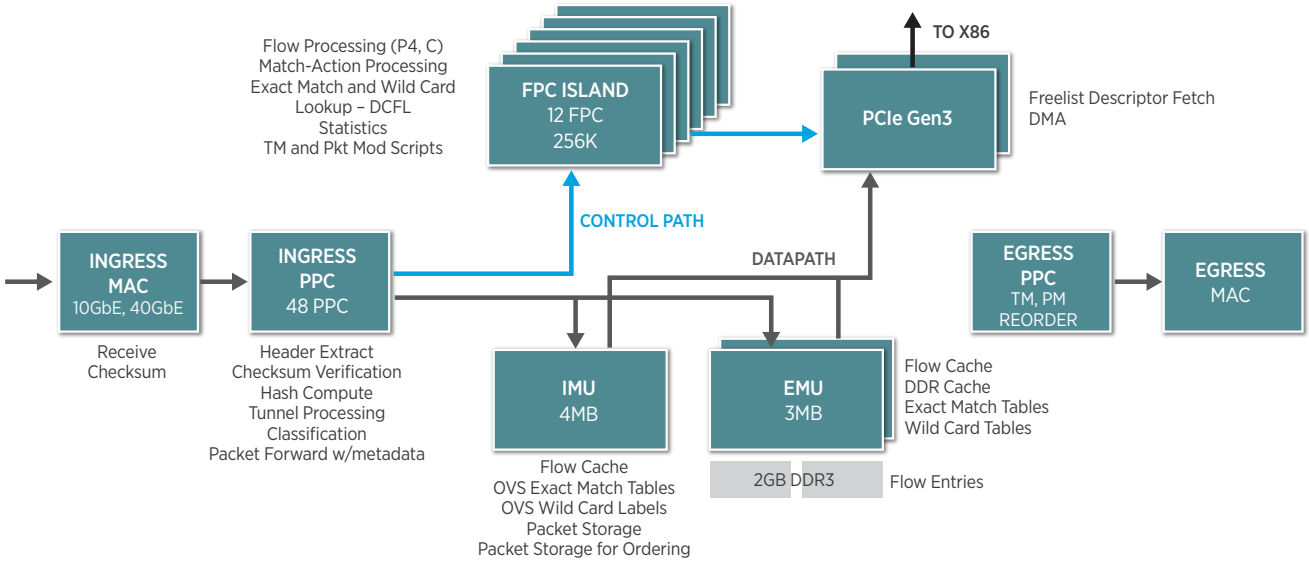

Figure 2. Network To Host Packet Flow

Ingress PPC Processing

The PPC Island directly interfaces to the MAC block and in the ingress direction supports Hardware Characterizer, 48 Packet Processing Cores and scatter/gather DMA capabilities under complete software control. A hardware characterizer within the PPC island parses through the packet. It then identifies L2/L3/L4 fields including VLAN Tags, MPLS, IPv4 and IPv6 as

well as proprietary headers. Following the identification process, the Hardware Characterizer packages the content into a metadata and delivers the packet along with the metadata to the 48 Packet Processing Cores. The Packet Processing Cores perform a fast lookup for additional classification and then they append a sequence number to the packet. The packet, along with the metadata, is delivered to the DMA engine within the PPC island. The DMA engine, under software control, has the capability to deliver the header of the packet and metadata to one destination and the payload to another destination. In the case of OVS implementation, the header and metadata are delivered to a Flow Processing Island, while the payload is forwarded to one of the memory units.

Flow Processing Island

The packet headers are delivered into the Cluster Target Memory (CTM) of the FPC island. The FPC island supports 12 FPCs and each FPC supports 8 threads. Under software control, a thread picks the next packet header from the CTM and generates an appropriate "key" for lookups. The particular thread performs multiple lookups based on the key to internal as well as external Flow Tables. Based on the lookup results, the FPC thread executes the required action. Forward, Drop, Add/Remove header, are all examples of actions. The FPCs can be programmed the C or P4 languages. However, in the case of OVS, the user drives the device through API.

PCIe Island

The PCIe island supports PCIe DMA capability as well as 4 FPCs. The FPCs within the PCIe island repackage the packet by first gathering the packet payload from the memory units and the modified packet header from the FPC island that was performing packet processing. The packets are then transferred to the host by the PCIe Island through the PCIeGen3 x8 interface.

HOST-TO-NETWORK DATAPATH

This section discusses a "Day in the life of a packet" from x86 to Network in the NFP-4000 for a generic application. Please refer to Figure 3 while reviewing this section.

PCIe Island

Packets from the host are received over the PCIe Gen3 x8 interface in the PCIe island. The packets are first stored in the internal memory within the PCIe island. The received packets are moved to the appropriate destinations including FPC islands, memory units or PPC island through the distributed switching fabric.

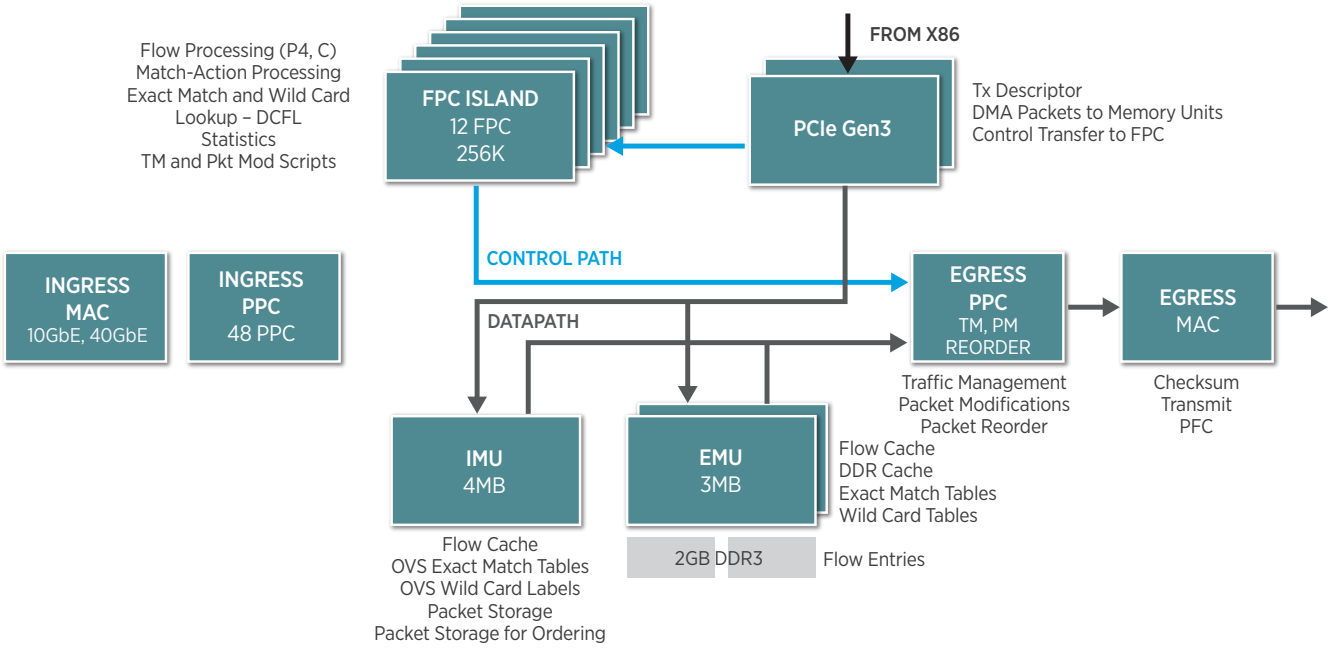

Figure 3. NFP-4000 Micro-architecture

FPC Island

The packets arrive at the CTM and a thread from the FPC island picks it up for packet processing. A key for the lookup is generated, using the header of the packet, and multiple lookups are performed to decide on an action. The FPC also generates and appends metadata to every packet. The metadata is used by packet modifier for modifying the packet headers.

Egress PPC

The Egress PPC is notified by a packet ready command from the FPC island indicating that it needs to transfer a packet. The packets are then transferred, reordered and stored into one of the traffic manager queues. The queue selection is based on the packet fields. The TM then sends the packet to the packet modifier (PM) block within the Egress PPC. The PM modification process is driven by either a static configuration, or via a packet modifier script appended to the packet by the FPC's earlier in the packet processing process. The modified packet is then stored in the Egress packet memory for transmission.

MAC

The MAC retrieves the packet from the Egress Packet Memory and performs an IPv4 or IPv6 checksum computation and insertion. The MAC also can optionally insert current timestamp information, and append the Frame Checksum Sequence (FCS). Finally the packets are transferred to the Network Interface for transmission.

NFP-4000 SOFTWARE DATA PLANE

So far we have described the NFP-4000 silicon level architecture and functionality. The actual data plane functionality in the system will ultimately be defined by the software that runs on the FPCs. There are two ways for customers to access this data plane functionality:

The easiest way to access NFP-4000 functionality which requires no direct programming

of the device is to use Netronome's Agilio Software. The Agilio Software provides all of the datapath code that runs on the NFP-4000 as a production-ready package. The current Agilio release supports a wide range of SmartNIC features, including OVS acceleration and offload, as well as common network virtualization tunneling protocol such as VXLAN, GRE, and MPLS. Feature updates are available with an ongoing support agreement. The Agilio datapath feature are controlled and accessed via APIs and/or SDN control mechanisms.

For more information on the Agilio Software features and functionality, see the Agilio Software Product Brief. For more information on the Agilio Software Architecture, see the Agilio Software Architecture White Paper.

If additional functionality beyond what is provided with the Agilio Software is needed, custom programming of the NFP-4000 can be enabled. There are two methods for customer-programming of the NFP-4000. The first is P4-based and is the simplest, allowing for high-level match-action syntax-based hardware-agnostic programming. This can be used to build a new networking datapath. The second model allows extensions to the Agilio OVS Software datapath using a sandbox extension method that supports both C and P4-based programming. For further details on programming the NFP-4000, see the document "Programming in C and P4".

NETRONOME

Netronome Systems, Inc.
2903 Bunker Hill Lane, Suite 150 Santa Clara, CA 95054
Tel: 408.496.0022 | Fax: 408.586.0002
www.netronome.com