NETRONOME

WHITE PAPER

Applying Netronome Composable IP Blocks to Modern SoC Designs

EFFICIENT COPROCESSOR DESIGNS REQUIRE MORE COMPLETE AND SOPHISTICATED IP BLOCKS IN ALL AREAS – NETWORK AND HOST INTERFACES, INTERNAL AND EXTERNAL MEMORY, HARDWARE ACCELERATORS AND PROCESSING LOGIC – ALL WORKING TOGETHER..

CONTENTS

1. INTRODUCTION

The increasing rate of change in protocols and algorithms related to networking, security and machine learning, coupled with the slowing down of Moore's Law, have prompted growing interest in flexible domain-specific coprocessor SoCs. Efficient coprocessor designs require more complete and sophisticated IP blocks in all areas – network and host interfaces, internal and external memory, hardware accelerators and processing logic – all working together to enable the most efficient data movement. This document covers Netronome's IP portfolio related to such modern SoC development and explores extending these building blocks to other coprocessor applications.

Table 1 lists all significant Netronome IP blocks classified by their application in a SoC design. IP blocks sourced from third parties are marked as such. The table also highlights Netronome NFP products that use the IP and the process node with which the IP has been designed and verified. For each SoC function, Netronome complements technology-specific sourced IP with significant IP of its own to enhance the function's features, performance and efficiency. For example, a network interface is significantly enhanced by combining sourced Ethernet Phy and MAC IP with packet-oriented Netronome IP blocks such as the packet classifier, port manager and traffic manager. Similarly, basic technology-specific sourced memory controller IP is enhanced with queuing and processing engines from the Netronome IP portfolio. The functionality of the IP blocks listed is described briefly in Table 1 and detailed later on how Netronome has used those IP blocks in the design of its NFP products.

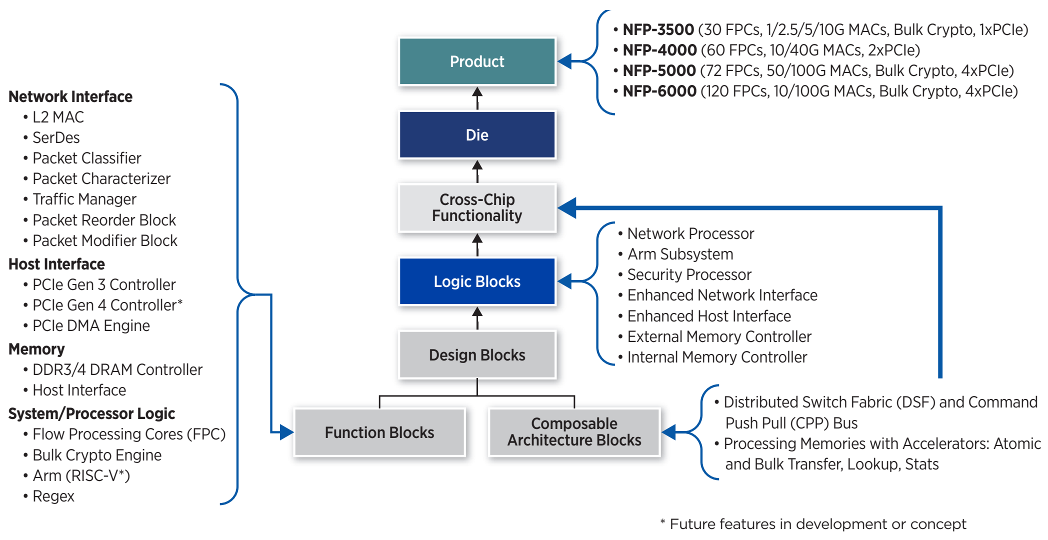
2. NETRONOME IP PORTFOLIO

IP TYPE	FUNCTION					
	NETWORK INTERFACE	MEMORY INTERFACE	APPLICATION LOGIC	HOST INTERFACE	INTERNAL BUS	PROCESS DESIGN KIT
Technology-Specific	SerDes: Process-specific hard IP 6.25Gb/s, 12.9Gb/s Ethernet MAC: 1G, 2.5G, 5G, 10G, 25G, 40G, 100G	Memory PHY Process-specific hard IP Memory compiler for single- and dual-port SRAM LP DDR4 controller 2400-3200 MHz		PCIe Gen3 PHY, controller (backwards compatible)		GPIO, PHYs, PLLs, Clocking, JTAG
Application-Oriented	Port Manager: Maps ports to internal resources Packet Classifier: n-tuple classifier Packet Modifier block: Modify packet fields on egress Packet Egress Reorder block: Preserves ingress packet arrival order Traffic Manager	Queue Engine: Hardware queue support for threading, work management Packet Stats Engine Atomic Engine: Atomic read/write Bulk Engine: Large memory transfer Hash/Trie Lookup Engine Last-level cache Memory Packet Engine: Packet transfer management	Arm 11 Flow Processing Cores Bulk Crypto	PCI DMA engine	Distributed Switch Fabric (DSF): Intra chip scalable bus Island Master Bridge (IMB) bus: Intra logic-block bus Bus agents: Interface	

TABLE 1 – List of field-proven Netronome and third-party IP blocks

3. APPLYING THE IP TO SOC DESIGNS

A SoC is built around a set of logic and memory elements that are instantiated multiple times in the design. Figure 1 builds a functional tree of the IP blocks as they are applied to the SoC architecture. The IP is classified as Design Blocks, Logic Blocks and Cross-Chip functions.



Product
Die
Cross-Chip Functionality
Logic Blocks
Design Blocks
Function Blocks
Composable Architecture Blocks
• NFP-3500 (30 FPCs, 1/2.5/5/10G MACs, Bulk Crypto, 1xPCIe)
• NFP-4000 (60 FPCs, 10/40G MACs, 2xPCIe)
• NFP-5000 (72 FPCs, 50/100G MACs, Bulk Crypto, 4xPCIe)
• NFP-6000 (120 FPCs, 10/100G MACs, Bulk Crypto, 4xPCIe)
• Network Processor
• Arm Subsystem
• Security Processor
• Enhanced Network Interface
• Enhanced Host Interface
• External Memory Controller
• Internal Memory Controller
• Distributed Switch Fabric (DSF) and Command Push Pull (CPP) Bus
• Processing Memories with Accelerators: Atomic and Bulk Transfer, Lookup, Stats
Network Interface
• L2 MAC
• SerDes
• Packet Classifier
• Packet Characterizer
• Traffic Manager
• Packet Reorder Block
• Packet Modifier Block
Host Interface
• PCIe Gen 3 Controller
• PCIe Gen 4 Controller*
• PCIe DMA Engine
Memory
• DDR3/4 DRAM Controller
• Host Interface
System/Processor Logic
• Flow Processing Cores (FPC)
• Bulk Crypto Engine
• Arm (RISC-V*)
• Regex
* Future features in development or concept

Figure 1. Functional tree of the IP blocks

3.1. Design Blocks

Design Blocks are the primitive IP elements in a SoC built using the Netronome IP portfolio. Figure 1 shows a list of the significant Design Blocks in the IP portfolio. There are two types of Design Blocks: Functional Blocks and Composable Architecture Blocks. Functional Blocks implement discrete functionality such as processing cores, MACs, SerDes, packet classifiers, traffic managers and PCIe host interface etc. They can also accelerate compute-intensive functions such as accelerators for packet classification and bulk crypto functions. Composable Architecture Blocks enable composition of cross-chip functions and are discussed in greater detail later.

3.2. Logic Blocks

Multiple instances of one or more Design Blocks are combined to form a Logic Block. A Logic Block implements a significant SoC-level function. For example, a networking SoC contains Logic Blocks for network I/O, host I/O, clusters of network processing cores with memory, etc. Functional Block IP components can be put together to create a Logic Block. For example, SerDes, L2 MACs and Traffic Manager may be used together to build a Network Interface Logic Block. Logic Blocks are closely tied to the physical design of the SoC as will be evident later in this document. Further description of Logic Block components is available in upcoming sections that describe the application of the IP to the NFP ASIC and its theory of operation.

3.3. Composable Architecture Blocks for Cross-Chip Functions

In SoCs built with Netronome IP, cross-chip functions are composed from multiple distributed Design Block instances. To build a cross-chip function, every Logic Block in a SoC will need one or more instances of a Composable Architecture Block contributing to that function. By building functions from distributed instances, ASICs can be scaled efficiently. The NFP contains two important cross-chip functions: (a) A Distributed Switch Fabric (DSF) and (b) Processing Memory.

For the DSF, each Logic Block in a Netronome SoC contains a Bus Agent Design Block. The Bus Agent provides an interface to Design Blocks in the Logic Block and direct interfaces only to near-neighbor Logic Blocks. A Logic Block is not directly connected to any Logic Block that is not its physical neighbor. Across Logic Blocks, these bus agents are coordinated to form a Distributed Switch Fabric (DSF) for cross-switch data transfers between any pair of Logic Blocks.

Processing Memory is the second example of a function formed from Composable Architecture Blocks. Processing Memory enables programmable cores to perform arithmetic and logic operations on data while it still resides in the memory, without requiring a physical transfer to the core. It is implemented by adding Processing Design Blocks at the interfaces to internal and external memory.

4. SCALING AND REUSE

The scalable architecture makes reuse simpler and reduces the time and cost to develop a custom SoC. The Design and Logic Blocks have been used to produce multiple NFP devices at varying aggregate line rates across fabrication process nodes. The IP can be reused in non-networking applications where data movement efficiency matters. Figure 2 captures the potential for reuse at varying levels of abstraction.

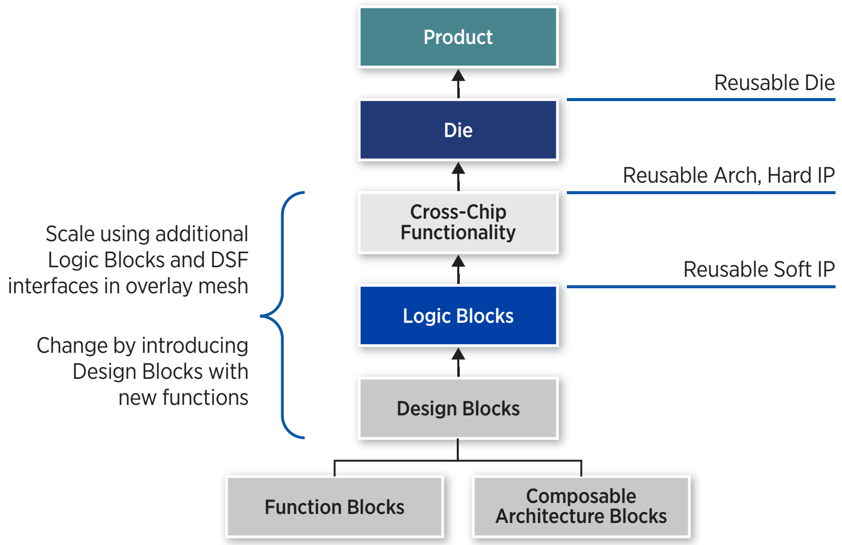

Product
Reusable Die
Die
Reusable Arch, Hard IP
Cross-Chip Functionality
Scale using additional Logic Blocks and DSF interfaces in overlay mesh
Reusable Soft IP
Logic Blocks
Change by introducing Design Blocks with new functions
Design Blocks
Function Blocks
Composable Architecture Blocks

Figure 2. Reusing, scaling, and changing functionality

4.1. Design Blocks

The NFP architecture can be upgraded and/or modified with new Design Blocks without impacting the functionality of other Logic Blocks. For example, custom cores based on a different instruction set, such as the open-source RISC-V processing cores, can be added to complement or replace the FPCs. New Design Blocks can be added to support functions like NVMe for disaggregated storage. Similarly, the Processing Memory Design Block can be extended to include compute processing for multiply-accumulate and external memory bulk initialization operations. These design blocks can be reused as soft IP in other IC designs.

4.2. Logic Blocks

NFP functionality is scaled and changed by changing the number of Logic Blocks and/or upgrading/creating new Logic Blocks. Figure 3 shows an example where some current Logic Blocks are replaced by newly-created third-party Coprocessor Logic Blocks (e.g., for learning, inferencing, storage, search, etc.). For example, in the Enhanced Host Interface Logic Block, the PCIe Design Block can be upgraded to support multi-host capability or a new generation of the PCIe specification to support higher line rates. The Security Processor Logic Block can be enhanced to provide hardware acceleration for new crypto ciphers.

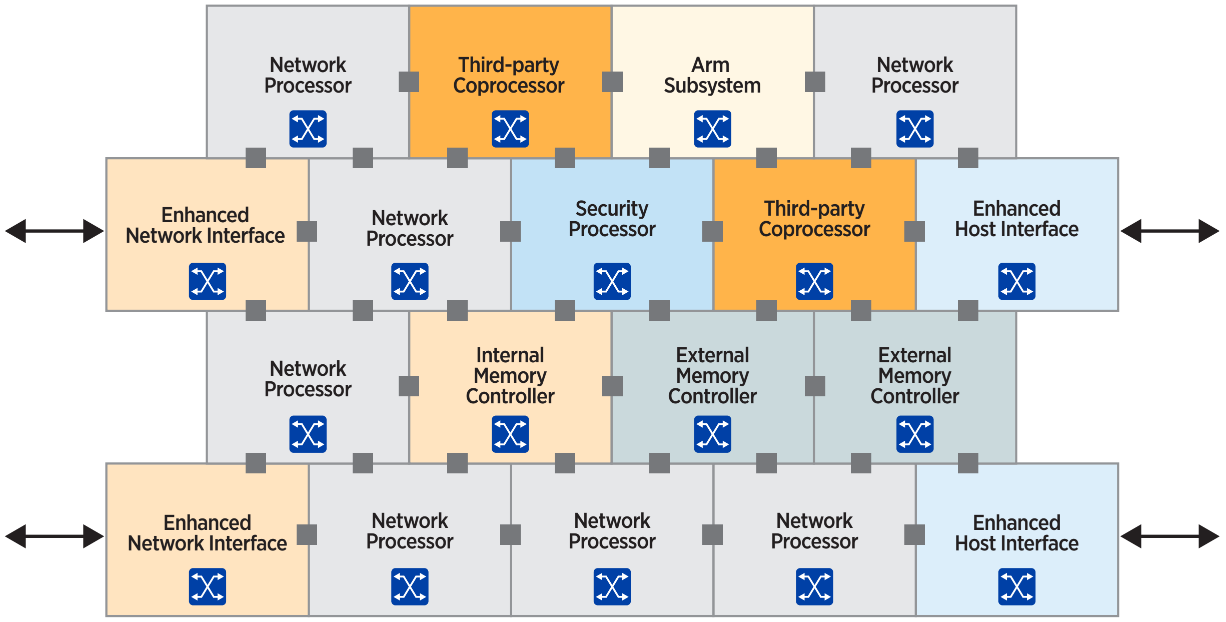


Figure 3. Adding third-party coprocessor IP Blocks

4.3. Scaling as Monolithic ICs or Disaggregated Chiplets

New silicon products can reuse NFP IP in several ways to reduce the time to tapeout a high-performance product. New products can use Design Blocks and Logic Blocks as soft IP. In this approach, the soft IP from the NFP will be incorporated into the RTL for the new product.

A new design can be based on the physical design of an existing NFP. A new product is designed as follows:

- Developers create new Logic Blocks, with Netronome and new Design Blocks. The new Logic Block includes DSF connectors for it to integrate with the rest of the IC.
- Physically lay out the new Logic Block next to and abutting an existing Logic Block or replace a Logic Block. The rest of the SoC is now reachable through the new Logic Block's immediate neighbors.

There is potential to enhance the switch fabric with die-to-die ultra-short reach (USR) SerDes connectivity options to enable scaling using multiple and disaggregated dies in an MCM, or Chiplet. NFP die can be used as Chiplets in an MCM.

5. CONCLUSION

The increasing importance of domain-specific accelerators is driving the need for more complete and sophisticated IP blocks for use in SoC designs. Coprocessors can be optimized to accelerate workloads such as networking and machine learning or inferencing, Netronome has developed an advanced portfolio of Design Block IPs that can be used in such SoCs. Netronome IP includes novel cross-chip IP to enable SoCs with a composable architecture that easily scales in performance and size. These blocks have been proven in multiple generations of Netronome NFP SoCs and popular process nodes. The architecture can be reused in new silicon devices to accelerate data-intensive applications such as machine learning and distributed data processing.

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