ARM 서버 기반 BlueField DPU 테스트베드 구축 사례 연구

곽재혁, 정기문
한국과학기술정보연구원
jhkwak@kisti.re.kr, kmjeong@kisti.re.kr

A Case Study on building BlueField DPU testbed on top of ARM servers

Jae-Hyuck Kwak, Kimoon Jeong
Korea Institute of Science and Technology Information

요 약

프로세서 성능 향상이 둔화되고 네트워크 성능과의 차이가 커짐에 따라 하이퍼스케일러를 중심으로 SmartNIC이라는 기술이 주목을 받고 있다. SmartNIC은 이기종 도메인 프로세서와 범용 코어를 결합하여 인프라 작업을 오프로딩하는 혁신적인 기술로 볼 수 있으며 연산 가속, 데이터 가속, 네트워크 가속 등 오프로딩 되는 기능에 따라서 다양한 형태로 활용될 수 있다. 본 논문에서는 ARM 서버 기반으로 엔비디아의 SmartNIC인 BlueField DPU 테스트베드를 구축한 사례를 제시하고 시사점을 도출하였다.

1. 서론

무어의 법칙과 데나드 스케일링의 종료 이후에 멀티코어 아키텍처는 프로세서 성능 향상을 이끌었지만 암달의 법칙에 따라 응용의 병렬 처리로 인한 성능 향상이 제한을 받으면서 좁은 범위의 기능을 전문으로 하는 도메인 특화 프로세서가 등장하였다. SmartNIC은 인프라 작업에 특화된 이기종 도메인 프로세서에 인프라 작업을 오프로딩하는 혁신적인 기술이며 연산 가속, 데이터 가속, 네트워크 가속 등 오프로딩되는 기능에 따라서 다양한 형태로 활용될 수 있다.

그림 1에서 보는 것처럼 네트워크 트래픽의 폭발적인 증가로 야기된 네트워크의 성능 향상은 프로세서의 성능 향상을 크게 앞지르고 있다.[1] 구글, 아마존, 마이크로소프트와 같은 하이퍼스케일러는 인프라 기능을 가속화하기 위해 자체 SmartNIC을 설계하고 있으며 인텔, 엔비디아, AMD와 같은 제조업체는 프로그래밍 가능한 도메인별 프로세서를 갖춘 SoC를 제공하는 등 광범위한 시장을 위한 SmartNIC 개발을 강조하고 있다.

본 연구에서는 ARM서버 기반으로 엔비디아의 SmartNIC인 BlueField DPU 테스트베드를 구축하였으며 구축 단계에서 발생하는 문제점을 중심으로 시사점을 도출하였다.

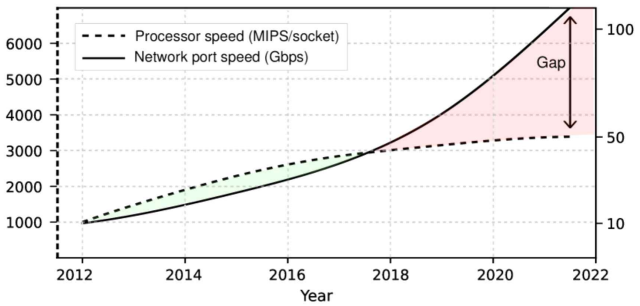

그림 1 네트워크 포트 성능과 프로세서 속도 비교

2. BlueField DPU 및 DOCA 소프트웨어 프레임워크

본 연구에서 사용한 BlueField-2 DPU는 64비트 ARM v8 A72 코어, Nvidia ConnectX-6 Dx 네트워크 어댑터, PCI Express 스위치를 하나의 칩으로 통합한 제품이다. 사용자는 ARMv8 멀티코어 프로세서와 ARM 소프트웨어 생태계를 활용하여 호스트 서버에서 실행되는 소프트웨어를 오프로딩할 수 있으며 ConnectX-6 Dx 오프로딩 컨트롤러는 RDMA 및 RDMA over Converged Ethernet(RoCE) 기술이 적용되어 NVMe over Fabrics(NVMe-oF)와 같은 네트워킹 및 스토리지 응용에 보다 향상된 성능을 제공할 수 있다. 그림 2는 BlueField-2 DPU의 구조를 보여주고 있다.[2]

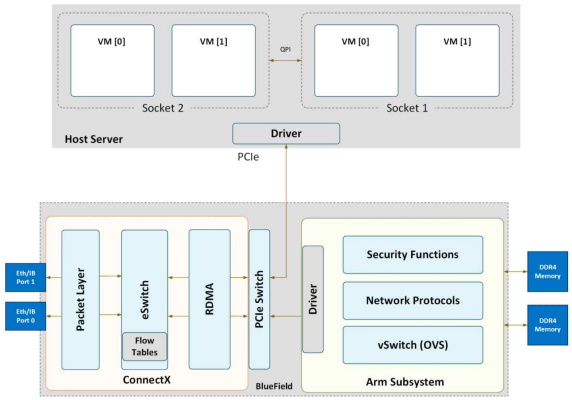

그림 2 BlueField-2 DPU 구조

DOCA 소프트웨어 프레임워크는 그림 3과 같이 BlueField DPU로 오프로딩되는 응용 프로그램과 서비스를 생성하고 관리하는 라이브러리와 필수적인 드라이버 및 도구를 제공한다.[3] DOCA 소프트웨어 프레임워크는 DOCA-Host 패키지와 BlueField 소프트웨어 번들로 나뉘는데 DOCA-Host 패키지는 BlueField DPU가 장착된 호스트 서버에 설치되는 라이브러리, 드라이버 등의 소프트웨어 패키지이며 BlueField 소프트웨어 번들은 BlueField DPU에 설치되는 소프트웨어 패키지로서 DOCA 런타임 드라이버와 라이브러리, 운영체제, 펌웨어 등을 포함한다.

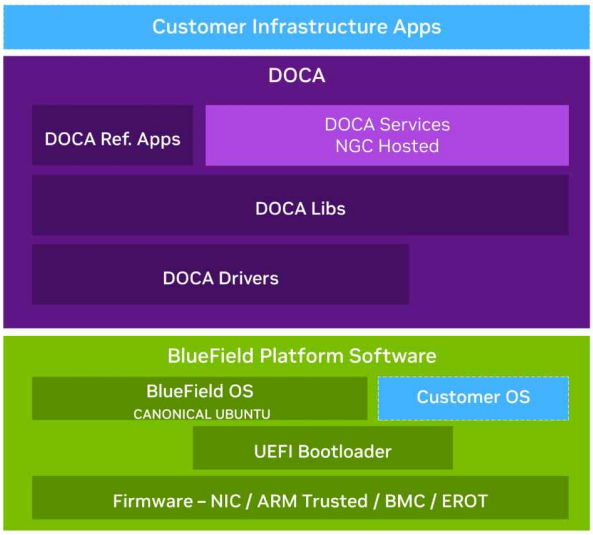

그림 3 DOCA 구조

3. BlueField DPU 테스트베드 구축

1) 호스트 서버 준비

본 연구에서 사용한 호스트 서버는 Gigabyte의 G242-P33 서버이다. G242-P33 서버는 그림 4에서 보여지는 것처럼 싱글 소켓 서버로 3GHz 속도로 동작하는 80코어의 Ampere Altra 프로세서를 장착하고 있으며 8개의 메모리 채널을 지원하는 3200MHz DDR4 메모리 16개를 장착하여 총 512GB의 메모리 용량을 가진다.[4] 후면의 PCIe 확장 슬롯은 x16 커넥터를 가지지만 PCIe 4.0 x8의 링크 속도만을 지원하는데 BlueField-2 DPU는 PCIe 4.0 x16의 링크 속도를 지원하므로 자동협상을 통해서 PCIe 4.0 x8의 링크 속도로 다운그레이드되었다. BlueField DPU를 장착할 호스트 서버의 PCIe 성능 사양은 사전에 확인하여 준비될 필요가 있다. 그림 5는 G242-P33 서버에 BlueField DPU를 장착하여 구성한 테스트베드의 모습이다.

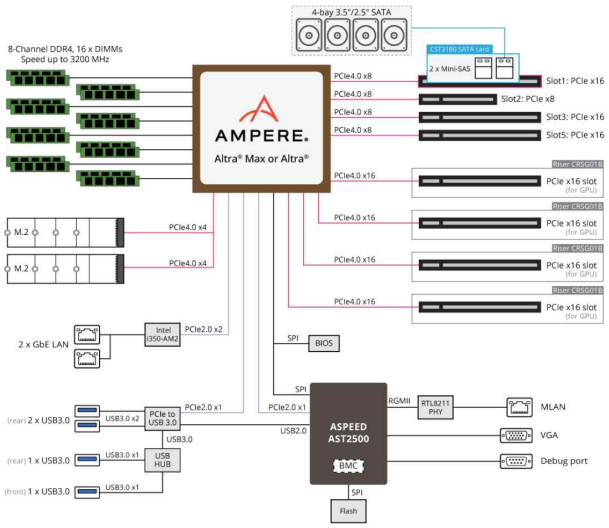

그림 4 G242-P33 서버 다이어그램

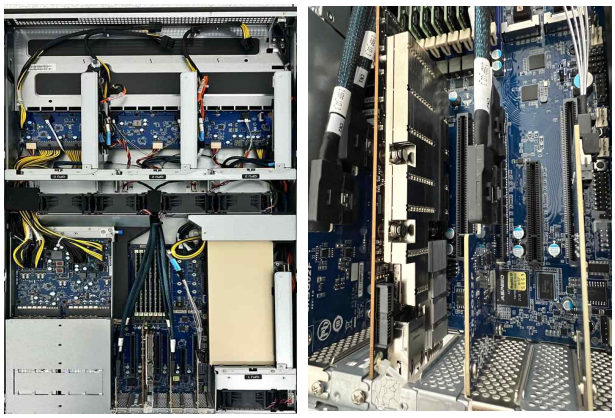

그림 5 BlueField DPU 테스트베드

2) 고성능 인터커넥트 구성

BlueField-2 DPU는 200Gb/s의 인피니밴드 혹은 이더넷을 지원한다. 인피니밴드와 이더넷 간의 변경은 mlxconfig 명령을 통해서 가능하다. /sbin/mlnx_bf_configure은 ib_umad 모듈이 로드될 때 실행되는데 이더넷을 사용하고 이더넷의

eSwitch 모드가 switchdev로 설정되어 있는 경우 리눅스 커널의 switchdev를 통해 Open VSwitch(OVS)를 자동으로 설정됨을 확인할 수 있었다. 인피니밴드를 사용하는 경우 호스트 서버에서 ibstat을 실행했을때 포트 상태가 Down인 문제가 있었는데 인피니밴드 스위치에 내장된 서브넷 관리자가 사용되어서 발생한 문제로 확인되었다. 인피니밴드 서브넷 관리자인 opensm은 호스트가 아닌 BlueField DPU에서 실행되어야 하며 네트워크에 문제가 있다면 인피니밴드 스위치에 내장된 서브넷 관리자가 실행중인 것은 아닌지, BlueField DPU에서 opensm이 실행되고 있는지를 확인할 필요가 있다.

3) 호스트 서버 운영체제 및 DOCA 소프트웨어 프레임워크 설치

DOCA-Host 패키지는 설치 유형에 따라서 지원하는 호스트 서버 운영체제와 커널 버전이 다르다. 표 2는 DOCA 설치 유형별로 지원 운영체제 및 아키텍처를 정리한 것이다. Redhat 계열보다는 Ubuntu 계열에서 더 잘 지원하는 것으로 보이며 본 연구에서 사용한 G242-P33 서버는 Ubuntu 24.04 설치 과정에서 알수 없는 문제가 발생하여 Ubuntu 22.04를 사용하였다. ARM 서버에서 운영체제 설치에 문제가 있다면 운영체제를 바꾸어서 설치해볼 필요가 있다.

표 2 DOCA 설치 유형별 지원 운영체제 및 아키텍처

운영체제	버전	아키텍처	doca-all	doca-networking	doca-ofed
RHEL /Rocky	8.6	aarch64			O
		x86	O	O	O
	8.7	aarch64			O
		x86			O
	8.8	aarch64	O	O	O
		x86	O	O	O
	8.9	aarch64	O	O	O
		x86	O	O	O
	9.0	aarch64			O
		x86			O
	9.1	aarch64			O
		x86	O	O	O
	9.2	aarch64			O
		x86			O
	9.3	aarch64			O
		x86			O
	9.4	aarch64		O	O
		x86		O	O
Ubuntu	20.04	aarch64			O
		x86	O	O	O
	22.04	aarch64	O	O	O
		x86	O	O	O
	24.04	aarch64	O	O	O
		x86	O	O	O

BlueField 소프트웨어 번들은 bfb-install을 통해 BlueField DPU의 전체 부트파티션과 펌웨어를 한번에 업데이트할 수 있다. 다만 DOCA 최신 버전의 경우 DOCA 초기 버전과의 차이로 인해 이미지 다운로드 과정에서 오류가 발생하였는데 이것은 DOCA의 이전 LTS버전을 BlueField DPU에 순차적으로 업그레이드하면 해결할 수 있었다. 또한, BlueField 소프트웨어 번들 설치 이후에 write counter to semaphore: Operation not permitted라는 에러가 발생하는 경우 BlueField DPU의 바이오스에서 SecureBoot를 비활성화하면 해결할 수 있었다.

4) 참조 응용 테스트

DOCA는 BlueField DPU를 이용하는 네트워킹 및 데이터 프로세싱을 프로그래밍할 수 있도록 라이브러리를 제공하여 프로세싱 파이프라인 혹은 다수의 DOCA라이브러리 기반 워크플로우 개발을 가능하게 한다. 표 3은 DOCA 소프트웨어 프레임워크에서 제공하는 몇가지 라이브러리를 나열한 것이다.

표 3 DOCA 라이브러리

라이브러리	기능
DOCA Commom	다양한 DOCA 라이브러리와 상호작용하는 전체적인 인터페이스를 제공
DOCA Flow	하드웨어 수준에서 범용 패킷 프로세싱 파이프라인을 생성
DOCA DMA	호스트와 DPU간의 메모리 영역 사이에 직접메모리접근(DMA)을 통한 데이터 복사
DOCA Comch	호스트와 DPU 간에 데이터 전달 채널을 생성
DOCA RDMA	CPU나 운영체제의 간섭없이 원격 머신의 메모리에 대한 직접적인 접근
DOCA GPUNetIO	CPU의 간섭 없이 네트워크 패킷의 실시간 GPU 처리
DOCA Compress	하드웨어 수준에서 호스트와 DPU 메모리 영역의 데이터를 압축 및 압축 해제

참조 응용 프로그램은 /opt/mellanox/doca/applications에 설치되어 있으며 meson과 ninja를 사용하여 빌드하고 실행할 수 있었다. 참조 응용 프로그램의 하나로 제공되는 Allreduce는 allreduce 연산을 DOCA라이브러리를 이용하여 참조 구현한 것으로 DPU가 allreduce 알고리즘을 수행하도록 오프로딩하는 경우 그림 8과 같은 구조로 실행된다.[5]

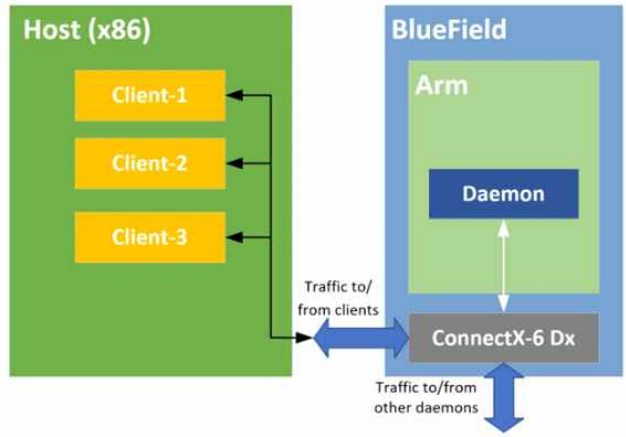

그림 8 allreduce 연산의 DPU 오프로딩

응용 프로그램의 기능을 DPU로 오프로딩하기 위해서는 Allreduce 참조 응용 구현에서 보는 것처럼 오프로딩된 기능을 DPU에서 데몬으로 구현하여 백그라운드로 실행하고 호스트의 클라이언트로부터 요청을 받아서 처리하는 방식으로 재작성 필요가 있다. 또한, Allreduce 참조 응용 프로그램은 DOCA의 최신 버전(2.8.0)에는 포함되지 않았는데 DOCA에서 제공하는 라이브러리가 계속해서 바뀌고 있는 것도 주의깊게 살펴볼 필요가 있다.

4. 결론 및 향후 계획

본 연구에서는 ARM 서버를 기반으로 BlueField DPU 테스트베드 구축 사례를 알아보고 구축 과정에서 발생한 문제점을 중심으로 시사점을 도출하였다. 정리하면 (1) BlueField DPU에서 사용되는 DOCA 소프트웨어 프레임워크는 운영체제와 커널 버전에 따라 지원되는 기능에 제한을 받으며 DOCA 설치 과정에서 순차적인 업그레이드가 필요하다는 것, (2) BlueField DPU에서 사용하는 네트워크 링크가 이더넷인 경우에만 OVS 브릿지가 자동으로 구성되고 인피니밴드의 경우 서브넷 매니저가 반드시 BlueField DPU에서 실행되어야 한다는 것, (3) BlueField DPU로 기능을 오프로딩하기 위해서는 호스트 서버의 클라이언트에서 처리되는 기능을 BlueField DPU의 데몬으로 구현하고 클라이언트가 데몬을 호출하도록 재작성할 필요가 있고 DOCA 버전업에 따른 DOCA 라이브러리 변화를 주의깊게 볼 필요가 있다는 것을 알수 있었다.

향후 계획으로는 병렬 연산이나 집합통신을 위한 보조 프로세서로서 병렬프로그래밍에서의 활용 가능성[6][7]과 I/O 작업을 효율화하는 보조 프로세서로서 병렬파일시스템에서의 활용 가능성[8][9]에 대해서 탐색해볼 예정이다.

※ 이 논문은 2024년도 한국과학기술정보연구원(KISTI)의 기본사업으로 수행된 연구입니다. (과제번호: (KISTI) K24L2M1C6, (NTIS) 2710018524)

참고문헌

[1] Elie F. Kfoury, et al, “A comprehensive survey on smartnics: Architectures, development models, applications, and research directions“, IEEE Access, 2024.

[2] NVIDIA BlueField BSP v4.8.0, https://docs.nvidia.com/networking/display/bluefieldbsp480

[3] DOCA Documentation v2.8.0, https://docs.nvidia.com/doca/sdk/index.html

[4] Gigabyte G242-P33, https://www.gigabyte.com/kr/Enterprise/GPU-Server/G242-P33-rev-100

[5] NVIDIA DOCA Allreduce Application Guide, https://docs.nvidia.com/doca/archive/2-7-0/nvidia+doca+allreduce+application+guide/index.html

[6] Muhammad Usman, et al, “DPU Offloading Programming with the OpenMP API”, Proceedings of the SC’23 Workshops of The International Conference on High Performance Computing, Network, Storage, and Analysis, 2023.

[7] Rhchard Graham, et al, “Optimizing Application Performance with BlueField: Accelerating Large-Message Blocking and Nonblocking Collective Operations”, ISC High Performance 2024 Research Paper Proceedings, Prometeus GmbH, 2024.

[8] Rob Davis, SmartNIC (DPU) Storage Solutions and Use Cases, SmartNICs Summit 2023, 2023.

[9] Peter-Jan Gootzen, “Dpfs: Dpu-powered file system virtualization”, Proceedings of the 16th ACM International Conference on Systems and Storage, 2023.

[10] Benjamin Michalowicz, et al, “Battle of the BlueFields: An In-Depth Comparison of the BlueField-2 and BlueField-3 SmartNICs“, 2023 IEEE Symposium on High-Performance Interconnects (HOTI), IEEE, 2023.