확장 가능한 HLS 기반 딥러닝 가속 하드웨어 개발

(Development of Scalable HLS based Deep Learning Accelerator)

권 용 인*, 유 미 선, 박 제 만, 이 제 민, 김 태 호
한국전자통신연구원
(Yongin Kwon, Misun Yu, Jeman Park, Jemin Lee, Taeho Kim)
(Electronics and Telecommunications Research Institute)

Abstract : Rapid growth of deep learning technology brings various opportunities for accurate and fast inference models. Recently, new neural operations and algorithms appear for better chances so that FPGAs become popular for time-to-market. There are tons of FPGA chips available and they all have different specifications and configurations. A design implemented by HLS can fit very well with one FPGA product but cannot be synthesized for others or cannot utilize enough resources in the chips. In this paper, we introduce Multi-EVTA which consists multiple EVTA, an HLS based hardware of deep learning accelerator. As a result, we improve the hardware resource utilization by 250% and the performance of ResNet18v1 by 100% on ZCU102 board.

Keywords : FPGA, VTA, HLS, ISA, Deep Learning

Ⅰ. 서 론

빠른 딥러닝 기술의 발달에 따라 새로운 형태의 딥러닝 연산과 모델이 지속적으로 늘어나고 있다. 이러한 기술 발달 속도에 대응하기 위해 FPGA (Field Programmable Gate Array)를 통한 딥러닝 가속 하드웨어의 개발이 점점 대중화되고 있고, 다양한 딥러닝 가속 하드웨어를 지원하는 컴파일러와 런타임 소프트웨어 기술 또한 그 중요성이 커지고 있다. 특히 High-level Synthesis (HLS)를 활용한 딥러닝 하드웨어의 구현은 고수준 언어를 사용하여 하드웨어를 구현할 수 있어 개발 시간을 단축 시키며 대상으로 하는 FPGA 하드웨어에 따라 컴파일러를 통한 최적의 Register Transfer Language (RTL) 변환을 지원한다.

개발 및 상용으로 사용 가능한 FPGA 하드웨어의 종류와 사양은 매우 다양하다. Xilinx사의 Zynq 멀티 프로세서 SoC 제품군들에 따르면, 현재 판매중인 제품의 수는 30종 이상이다. HLS를 통한 하드웨어 기술은 FPGA 종류에 따라 자원 부족에 의해 RTL 생성이 불가하거나 자원 활용률이 매우 떨어져 FPGA 사양 대비 성능이 떨어질 수 있다. 따라서 각 FPGA에 맞게 HLS 개발을 별도로 이루어져야 함은 하드웨어 개발 뿐만 아니라 개발된 하드웨어를 구동하기 위한 소프트웨어 라이브러리나 런타임 개발에도 매우 많은 시간 지연을 초래한다.

본 논문에서는 FPGA 종류에 상관없이 동일한 HLS 기반 하드웨어를 사용하되, 그 수를 조정하는 방법을 사용하여 FPGA 자원 활용률을 높이고 딥러닝 연산 성능 또한 향상됨을 보여준다

Ⅱ. 본 론

FPGA 칩은 미리 정의된 다양한 자원들로 구성되어 있고, 프로그래밍을 통하여 이 자원들의 디지털 회로를 연결한다. FPGA의 자원은 크게 CLB (Configurable Logic Block), IOB(Input Output Block), Programmable Interconnect, 고정 함수로직, 메모리 등으로 이루어져 있고, 표 1의 예와 같이 이들의 크기와 개수는 제품에 따라 매우 다르다.

*Corresponding Author (yongin.kwon@etri.re.kr)
권용인, 유미선, 박제만, 이제민, 김태호: 한국전자통신연구원

※ 이 논문은 2021년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원을 받아 수행된 연구임 (No.2018-0-00769,인공지능 시스템을 위한 뉴로모픽 컴퓨팅 SW 플랫폼 기술 개발)

표 1. Xilinx사 Artix UltraScale+ FPGA 제품 별 자원 비교
Table 1. Comparison of resources for different Xilinx's Artix UltraScale+ FPGA products

	AU10P	AU15P	AU20P	AU25P
Logic Cells(K)	96	170	238	308
Flip-Flops(K)	88	156	218	282
LUTs(K)	44	78	109	141
BRAM(Mb)	3.5	144	200	300

EVTA (Extended Versatile Tensor Accelerator)[1]는 오픈소스 딥러닝 컴파일러인 TVM[2]의 한 컴포넌트로 개발된 VTA(Versatile Tensor Accelerator)[3]의 하드웨어를 확장하여 개발한 인스트럭션 세트 기반의 아키텍처로 기본 구조는 그림 2와 같다.

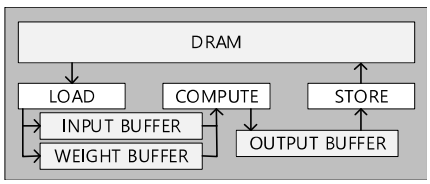

그림 2. EVTA 하드웨어 아키텍처
Fig. 2. The hardware architecture of EVTA

EVTA 하드웨어의 기본 형태는 PYNQ-Z1/Z2의 XC7Z02칩에서 구현이 가능한 수준이며, 그림 3과 같이 하드웨어에 따라 연산기나 버퍼를 늘려 구현할 수 있다. 이 경우, 늘어난 연산기와 버퍼를 충분히 활용할 수 있도록 하기 위해 런타임 라이브러리를 수정해야 한다.

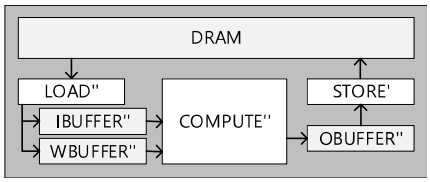

그림 3. 변형된 EVTA 하드웨어 아키텍처
Fig. 3. The hardware architecture of modified EVTA

본 연구에서는 변형된 EVTA에 맞게 런타임 라이브러리를 수정해야하는 문제를 해결하기 위해 Multi-EVTA를 구현하였다. Multi-EVTA는 동종의 EVTA를 FPGA 자원이 허용하는 만큼의 수만큼 구성하여 그림4와 같이 DRAM을 공유하는 형태로 구현된다. 동일한 하드웨어 IP를 사용하여 그 수만 설정 하기 때문에 FPGA 종류에 따른 빠른 최적화가 가능하다. 실행을 위한 런타임 라이브러리 또한 구성하는 EVTA의 하드웨어 어드레스만 설정해 주면 자동으로 빌드 되어 추가적인 구현을 필요로 하지 않는다.

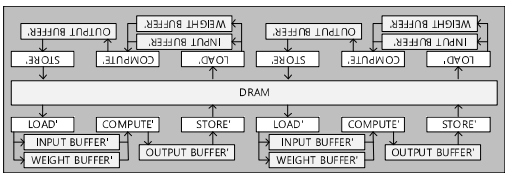

그림 4. Quad EVTA 하드웨어 아키텍처
Fig. 4. The hardware architecture of Quad EVTA

Ⅲ. 실험 및 분석

1. 실험 환경

본 논문의 실험에 사용한 FPGA 보드는 Zynq UltraScale+ MPSoC ZCU102로, Xilinx사의 Zynq UltraScale+ XCZU9EG-2FFVB1156 MPSoC 칩이 탑재되어있다. 해당 칩에는 Quad-core ARM Cortex-A53 CPU, LPDDR4 4GB를 구성하고 있으며, 가용한 FPGA 자원으로는 2,520개의 DSP Slice, 32.1Mb의 Block RAM, 600K의 System Logic Cell등이 있다.

Xilinx사의 Vivado HLS 도구를 사용하여 EVTA HLS 코드를 비트스트림 합성을 하였으며 Vivado에서 제공해주는 합성 도구를 이용하여 Single EVTA와 Quad EVTA를 합성하였다.

2. 실험 결과

표 2는 Single EVTA와 Quad EVTA의 FPGA 자원 활용률의 차이를 보여준다. 실험 결과, Quad EVTA의 Flip-Flops와 LUTs의 경우 경우 활용률이 4배 이상이 되었고 BRAM의 경우 최적화가 잘 되어 2.6배가 되었다.

표 3은 하드웨어 구성에 맞는 설정에 따라 런타임 라이브러리를 통한 Resnet18[4] 모델의 수행 성능을 보여준다. Quad EVTA를 통한 실행의 경우 네 개의 쓰레드를 생성하여 각 쓰레드가 서로 다른

입력 이미지를 처리하도록 하였다. 그 결과 Quad EVTA의 경우 Single EVTA보다 약 2배의 처리율 향상을 보여준다.

표 2. Single EVTA와 Quad EVTA의 자원 활용률 비교
Table 2. Comparison of resources utilization for Single EVTA and Quad EVTA

	Single EVTA(%)	Quad EVTA(%)
Flip-Flops	8	51
LUTs	11	54
BRAM	15	39

표 3. Single EVTA와 Quad EVTA의 Resnet18 처리성능
Table 3. Comparison of Resnet18's throughput for Single EVTA and Quad EVTA

	Single EVTA	Quad EVTA
Images per Second	11.4	22.9

IV. 결 론

본 논문에서는 HLS로 구현된 인스트럭션 세트 기반 딥러닝 가속 하드웨어인 EVTA를 FPGA자원에 따라 구성을 쉽게 변경하고 런타임을 통해 동작이 가능케 하여 빠른 하드웨어/소프트웨어 최적화를 가능하게 하였다. 향후, 다양한 FPGA칩을 대상으로 활용하여 처리율 향상 뿐만 아니라 지연시간을 단축할 수 있는 런타임 개발에 대한 연구가 필요하다.

References

[1] Yongin Kwon, et al. HLS 기반 딥러닝 가속 하드웨어의 ISA 확장을 통한 성능 향상, 대한임베디드공학회 학술대회(추계), 2020
[2] Chen, Tianqi, et al. "TVM: An automated end-toend optimizing compiler for deep learning.", OSDI. 2018.
[3] VTA, https://tvm.apach.org/vta
[4] https://github.com/onnx/models