

# Zynq SoC에서 재구성 가능한 하드웨어 가속기를 지원하는 멀티쓰레딩 시스템 설계

## Multi-threaded system to support reconfigurable hardware accelerators on Zynq SoC

신 현 준\*, 이 주 흥\*\*★

Hyeon-Jun Shin\*, Joo-Heung Lee\*\*★

### Abstract

In this paper, we propose a multi-threading system to support reconfigurable hardware accelerators on Zynq SoC. We implement high-performance JPEG decoder with reconfigurable 2D IDCT hardware accelerators to achieve maximum performance available on the platform. In this system, up to four reconfigurable hardware accelerators synchronized with SW threads can be dynamically reconfigured to provide adaptive computing capabilities according to the given image resolution and the compression ratio. JPEG decoding is operated using images with resolutions 480p, 720p, 1080p at the compression ratio of 7:1-109:1. We show that significant performance improvements are achieved as the image resolution or the compression ratio increase. For 1080p resolution, the performance improvement is up to 79.11 times with throughput speed of 99 fps at the compression ratio 17:1.

### 요 약

본 논문에서는 Zynq SoC 환경에서 재구성 가능한 하드웨어 가속기를 지원하는 멀티쓰레딩 시스템을 제안한다. 압축된 정지 영상의 픽셀 데이터를 복원하는 고성능 JPEG 디코더를 구현하고 2D-IDCT 함수를 재구성 가능한 하드웨어 가속기로 설계하여 성능을 검증한다. 구현된 시스템에서 최대 4개의 재구성 가능한 하드웨어 가속기는 소프트웨어 쓰레드와 동기화되어 연산을 수행할 수 있으며 이미지 해상도와 압축률에 따라 다른 성능 향상을 보인다. 1080p 해상도 영상의 경우 17:1의 압축률에서 최대 79.11배의 성능 향상과 99fps의 throughput 속도를 보여준다.

*Key words* : HW/SW co-design, Reconfigurable hardware accelerator, Multi-threaded system, Image processing, JPEG

### 1. 서론

일반적으로 범용 마이크로프로세서(CPU)를 활용한 소프트웨어 설계는 이식성이 좋고 유연하여 엔지니어의 생산성을 증대시킬 수 있다. 또한 FPGA

(Field Programmable Gate Array)를 활용한 하드웨어 설계는 파이프 라이닝과 병렬 연산을 통하여 높은 연산량과 에너지 효율을 제공한다. CPU와 FPGA가 결합된 이기종 내장형 시스템에서는 이러한 소프트웨어와 하드웨어 설계의 장점을 결합한

\* Associate Research Engineer, Chips&Media

\*\* Professor, Dept. of Electronic and Electrical Engineering, Hongik University

★ Corresponding author

E-mail : joolee@hongik.ac.kr Tel : +82-44-860-2544

Manuscript received Mar. 6, 2020; revised Mar. 19, 2020; accepted Mar. 23, 2020.

This is an Open-Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

HW/SW co-design platform으로 응용 어플리케이션의 성능을 향상시킬 수 있다[1-2]. 하지만 영상 신호처리 응용분야와 같이 실시간으로 입력되는 영상의 데이터 특성에 따라 연산량이 변화하는 응용 어플리케이션 설계에서 조건에 따라 하드웨어 자원이 불필요하게 낭비되는 결과를 가져올 수 있다[3]. 이를 보완하기 위한 재구성 가능 하드웨어를 지원하는 시스템 설계에서 하드웨어 가속기 모듈과 소프트웨어 모듈 간의 통신 및 동기화, 스케줄링 기능이 필요하나 이는 설계의 복잡성을 증가시켜 엔지니어의 생산성을 제한하고 있다.

본 논문에서는 FPGA와 Cortex-A9 dual-core processor가 탑재된 Zynq-7000 SoC 플랫폼을 활용하여  $\mu\text{C}/\text{OS-II}$  환경에서 재구성 가능한 하드웨어 가속기를 지원하는 멀티쓰레딩 시스템을 제안한다. 소프트웨어로 압축된 이미지를 복원하는 고성능 JPEG 디코더를 구현하고, 재구성 가능한 하드웨어를 설계하여 FPGA의 자원을 효율적으로 사용할 수 있도록 한다. 실시간으로 입력되는 영상의 특성에 따라 필요한 하드웨어 가속기를 재구성하여 FPGA 자원을 소프트웨어의 쓰레드와 같이 사용할 수 있다. HW thread는 JPEG decoding 처리 과정에서 많은 연산을 필요로 하는  $8 \times 8$  2D-IDCT(Two Dimensional Inverse Discrete Cosine Transform)를 재구성 가능 하드웨어 가속기로 구현하여 실제 성능을 검증한다.

## II. 본론

### 1. Partial Reconfiguration

일반적으로 FPGA는 bitstream 파일을 통해 프로그래밍 되면 실시간으로 수정할 수 없다는 단점이 있다. 이를 보완하여 설계의 유연성을 향상시킨 기능이 PR(Partial Reconfiguration)이다. 그림 1과 같이 RP(Reconfigurable Partition)를 사전에 지정하여 Static 영역과 분리한다. Static 영역은 FPGA가 처음 다운로드 될 때 구성되는 영역이며, RP는 partial bitstream 파일을 통해 실시간 재구성이 가능한 영역이다. RP가 재구성되는 동안 Static 영역은 이로 인한 영향을 받지 않기 때문에 어플리케이션의 중단 없이 FPGA를 실시간으로 재구성 할 수 있다. PR을 활용하면 필요한 하드웨어 가속기를 재구성하여 사용하고 나머지는 비활성화 하여 FPGA

의 리소스를 효율적으로 사용할 수 있다[4-5]. Zynq SoC Platform은 PR을 지원하기 위해 2개의 인터페이스를 지원한다. ICAP(Internal Configuration Access Port)을 통해 PL이 스스로 재구성하거나 PCAP(Processor Configuration Access Port)을 통해 PS가 PL 영역을 재구성 할 수 있다. 본 논문에서는 최대 100MHz의 동작주파수에서 3.2Gbps의 속도를 지원하는 ICAP을 활용하여 PR을 진행하였다[6].

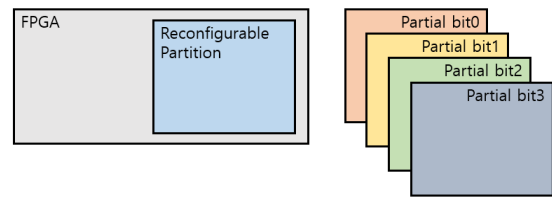


Fig. 1. Partial Reconfiguration.  
그림 1. 부분 재구성

### 2. Hardware Multi-threaded System

재구성 가능한 하드웨어를 지원하기 위한 방법들에서 multi-threaded programming 모델을 지원하는 ReconOS는 소프트웨어와 하드웨어에서 실행되는 기능을 지원하기 위한 OS 서비스와 하드웨어 가속기를 통합하기 위한 인터페이스를 그림 2와 같이 제공하고 있다[7].

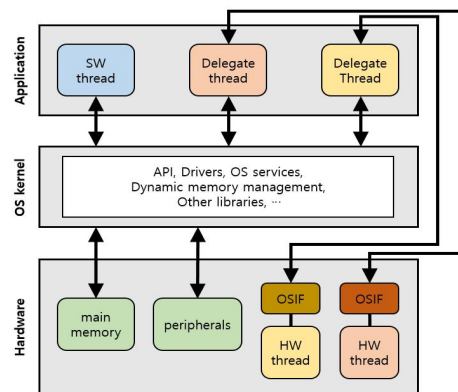


Fig. 2. Architecture of ReconOS to support hardware multi-threaded system.

그림 2. 하드웨어 멀티쓰레딩 시스템을 지원하기 위한 ReconOS의 구조

이는 hardware thread를 지원하여 세마포어, 뮤텝스 등의 OS 메커니즘을 사용할 수 있도록 기능을 확장하여 재구성 가능한 시스템에서 이식성을 개선한다. 본 논문에서는 제안된 모델을 바탕으로

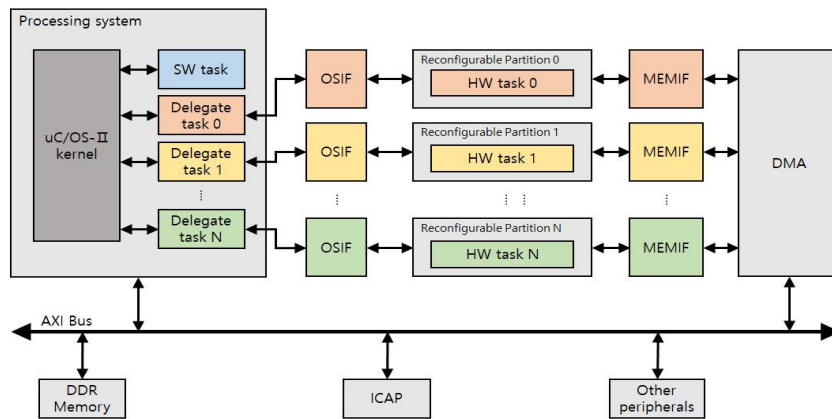


Fig. 3. System architecture to support multi-tasking on  $\mu$ C/OS-II.  
 그림 3.  $\mu$ C/OS-II에서 멀티태스킹을 지원하기 위한 시스템 아키텍처

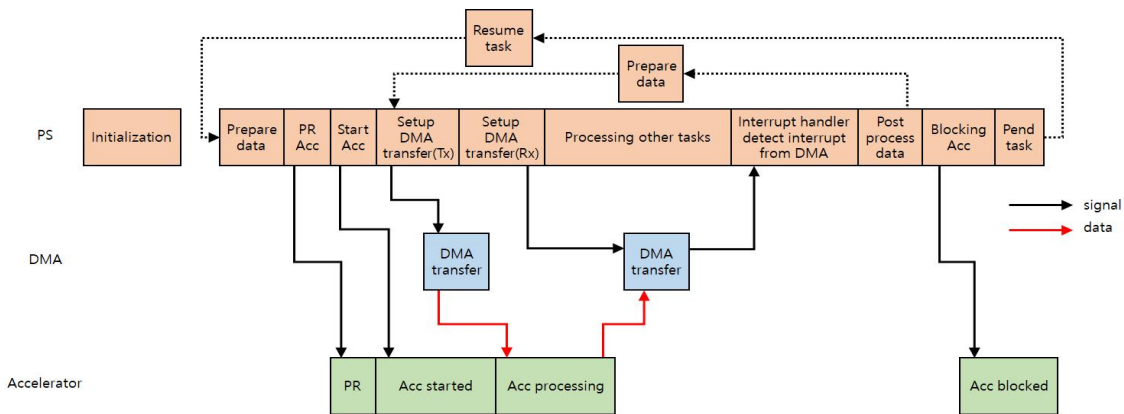


Fig. 4. Data processing sequence.  
 그림 4. 데이터 처리 과정

Zynq SoC 플랫폼 및  $\mu$ C/OS-II 운영체제를 활용하여 실시간으로 부분 재구성이 가능한 hardware multi-threaded system을 구현한다. 전체 시스템 아키텍처는 그림 3과 같다. 어플리케이션은 SW task와 delegate task로 나누어져 있는데 SW task는 소프트웨어 처리를 담당하며 delegate task는 커널과 HW task간의 인터페이스를 담당한다. PL 영역은 FPGA로 구성되어 있으며 여러 개의 RP로 나누어져 있다. 각각의 RP는 하나의 HW task를 의미하며 OSIF 모듈은 delegate task로부터 데이터를 받아 하드웨어 가속기를 제어한다. 실시간으로 이루어지는 PR은 ICAP이 담당하며 AMBA Bus를 통해 PS로부터 PR 요청을 받아 해당하는 RP를 실시간으로 재구성한다. 각 RM(Reconfigurable Module)은 MEMIF를 통해 데이터를 전송할 수 있고 처리된 데이터는 DMA에 의해 DDR Memory에 저장된다. 데이터 처리 순서는 그림 4와 같다. PS는 처리

할 데이터가 준비되면 하드웨어 가속기를 재구성한다. Delegate task가 OSIF 모듈을 통하여 하드웨어 가속기를 준비시키고 DMA가 DDR Memory의 데이터에 접근할 수 있도록 설정한다. 하드웨어 가속기가 데이터 연산을 수행하는 동안 delegate task는 대기상태가 되기 때문에 PS는 스케줄링을 통해 다른 task를 처리할 수 있다. DMA를 사용하여 연산이 끝난 데이터의 저장이 완료되면 delegate task를 준비 상태로 변경하여 운영체제가 스케줄링하도록 한다. 필요한 전체 데이터 처리가 끝나면 delegate task는 blank bitstream을 다운로드하여 하드웨어 가속기를 제거하고 대기상태가 되어 더 이상 스케줄링 되지 않는다.

**3. Case Study : 고성능 JPEG decoder**

본 논문에서는 압축된 정지 영상으로부터 픽셀 데이터를 복원하는 고성능 JPEG decoder를 설계하

여 SW design 및 HW/SW co-design 플랫폼에서의 성능을 검증한다.

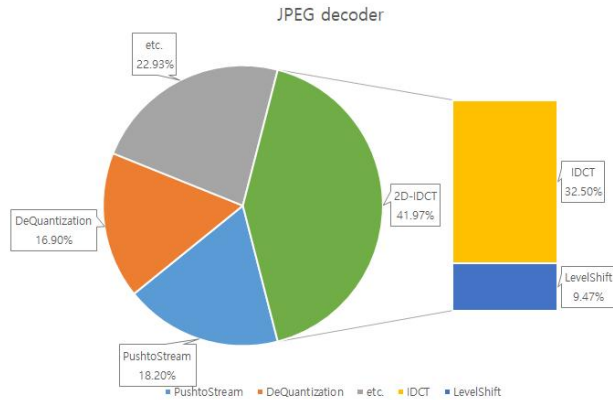


Fig. 5. The percentage of JPEG decoder functions in SW design.

그림 5. SW design에서 JPEG 디코더 함수의 처리 시간 분포

그림 5는 SW design에서 주요 함수의 처리시간 분포를 나타낸 것이다. SW design으로 구현된 JPEG decoding 어플리케이션에서 1080p 해상도의 이미지를 처리할 경우 inverse quantization을 수행하는 DeQuantization 함수는 16.90%, 복원된 픽셀 데이터를 DDR Memory로 전송하는 PushtoStream 함수는 18.20%, IDCT 연산을 수행하는 IDCT함수는 32.50%, 역변환 과정을 거친 픽셀 데이터를 DC shifting하는 LevelShift 함수는 9.47%로, LevelShift를 포함하는 2D-IDCT 연산은 전체 처리시간의 약 42%를 차지한다. 그러므로 HW/SW co-design으로 구현 시 HW task에서 RM으로 활용하기 위한 재구성 가능 하드웨어 가속기로 IDCT 및 LevelShift 함수를 선택하여 구현 및 검증한다. IDCT는 영상 압축 및 복원에 사용되는 알고리즘으로 입력 받는 주파수 계수들을 코사인 역변환 연산 과정을 거쳐 픽셀 데이터를 생성해내는 연산량이 높은 블록이다. 그림 6은 제안된 아키텍처를 활용하여 구현한 2D-IDCT 모듈의 내부 구조를 나타낸 것이다[8]. 하드웨어 가속기로 전달되는 데이터는 주파수 계수의 값, 해당 위치정보( $m,n$ ), 블록정보로 구성되어 있다( $m,n$ ). 정보를 이용하여 DCT Transform Matrix에 미리 저장된 코사인 데이터를 load하고 Multiplier와 PE Array를 이용하여 역변환 연산을 수행한다. Block 내의 마지막 데이터가 처리될 때까지 PE Array는 이전의 데이터를 연산하며 축적

하고, 모든 non-zero DCT 계수들의 연산이 끝나면 처리 데이터는 Output Buffer를 통해 Output IF로 출력된다. Input IF, Output IF 모듈은 AXI Bus를 통해 DMA와 데이터를 송수신한다. DMA는 HP 포트를 통해 외부 SDRAM DDR Memory로 데이터를 전송하여 소프트웨어가 이후 과정을 처리하도록 한다.

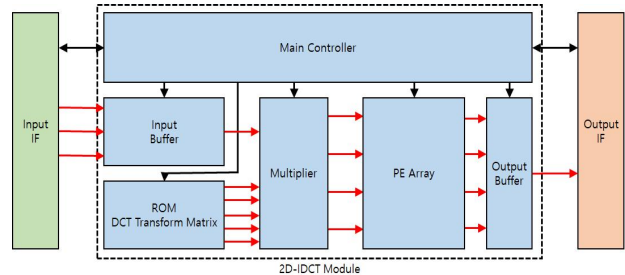


Fig. 6. 2D-IDCT Architecture.

그림 6. 2D-IDCT 아키텍처

그림 7은 AMBA Bus에 연결된 2D-IDCT 모듈의 인터페이스 다이어그램을 나타낸다. PS에서 DDR Memory에 데이터를 저장하고 AXI4-lite Bus를 통해 DMA의 레지스터를 설정하면 DMA는 DDR Memory로부터 데이터를 읽어 AXI4-stream Bus를 통해 Input FIFO에 전송한다. Input IF 모듈은 AXI4-stream Bus의 데이터를 2D-IDCT 모듈의 입력 데이터 형식으로 분할하여 전송하며 Output IF 모듈은 2D-IDCT 모듈의 출력 데이터를 AXI4-stream Bus의 데이터 형식으로 변환하여 전송한다. 해당 데이터의 연산이 끝나면 DMA는 PS로 interrupt 신호를 전송한다. Interrupt를 수신한 PS는 하드웨어 가속기의 동작 또는 제거를 결정한다. PR을 활용한 시스템 아키텍처는 그림 8과 같다.

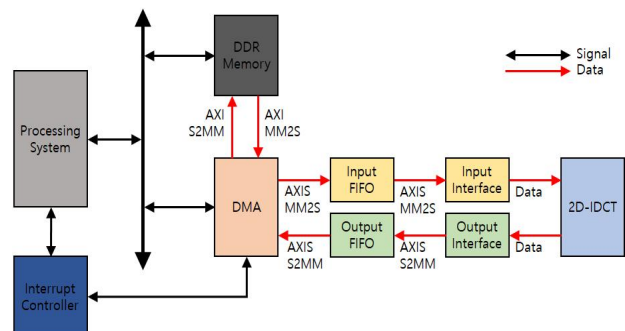


Fig. 7. System interface diagram.

그림 7. 시스템 인터페이스 다이어그램

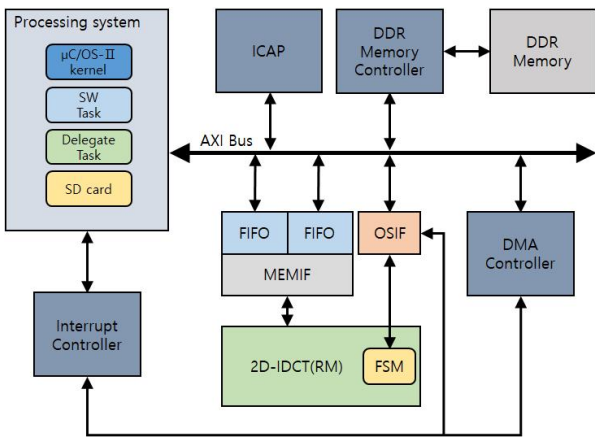


Fig. 8. System architecture with partial reconfiguration.  
그림 8. PR을 활용한 시스템 아키텍처

외부 메모리인 SD card에는 Boot를 위한 First Stage Boot Loader(FSBL), PL의 Static Logic을 다운로드하기 위한 full bitstream, 각각의 RM의 Logic을 다운로드하기 위한 partial bitstream, 처리할 영상 데이터가 저장되어 있다. FSBL에 의해 PS가 Boot되고 Static Logic이 다운로드 된 후 어플리케이션이 실행되면 운영체제는 partial bitstream 데이터와 영상 데이터를 DDR Memory에 저장하고 decoding을 진행한다. 처리과정에서 PR이 진행된다면 ICAP은 DDR Memory의 partial bitstream 데이터를 PL의 재구성 가능 영역에 다운로드 한다. PR이 진행되는 동안 Static Logic은 영향을 받지 않는다.

4. Experimental Results

본 논문에서는 480p, 720p, 1080p 등 다양한 해상도를 가지는 영상을 활용하여 고성능 JPEG 디코더를 최대 4개의 재구성 가능 하드웨어 가속기를 활용할 수 있는 hardware multi-threaded system으로 구현하고 검증하였다. 그림 9는 구현한 system의 이미지 처리 방식을 나타낸다.

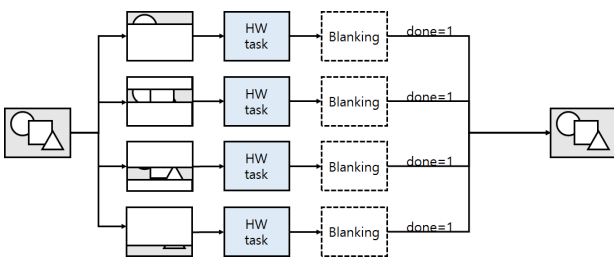


Fig. 9. Image processing sequence.  
그림 9. 이미지 처리 방식

전체 데이터 처리량은 영상 데이터의 non-zero DCT 계수의 개수에 영향을 받기 때문에 실시간 처리 시 HW task의 수가 유동적으로 변화하며 데이터를 처리하여야 한다. 소프트웨어는 이미지의 해상도, 압축률 등의 데이터 특성에 따라 필요한 HW task의 개수를 결정하고 생성한다. DDR Memory에 저장된 이미지는 생성한 hardware task 개수만큼 분할되고, 분할된 데이터는 각각의 하드웨어 가속기에 의해 연산된다. 연산이 완료된 hardware task는 완료 flag를 설정한 후 blank bitstream에 의해 제거되고 복원된 데이터는 DDR Memory에서 하나의 이미지로 재결합된다.

표 1은 구현된 Static Logic 및 2D-IDCT module의 리소스 사용률을 나타낸다. 4개의 하드웨어 가속기가 모두 동작할 경우 약 44.96%의 하드웨어 리소스를 사용하는 것을 확인하였다. 전체 bitstream의 크기는 4,045,663 Bytes이고 각각의 partial bitstream의 크기는 417,380 Bytes로 약 9.69배의 차이를 확인할 수 있으며 하나의 partial bitstream 로직 구성 시간으로 약 1.046ms 소모하는 것을 확인하였다.

Table 1. Resources utilization for Static Logic and 2D-IDCT module.

표 1. Static Logic과 2D-IDCT 모듈의 리소스 사용률

Resource	Available	Static Logic		2D-IDCT module	
		Utilization	%	Utilization	%
LUT	53200	14492	27.24	2357	4.43
LUTRAM	17400	1332	7.66	-	-
FF	106400	19961	18.76	3483	3.27
BRAM	140	30	21.43	4	2.86
DSP	220	-	-	5	2.27

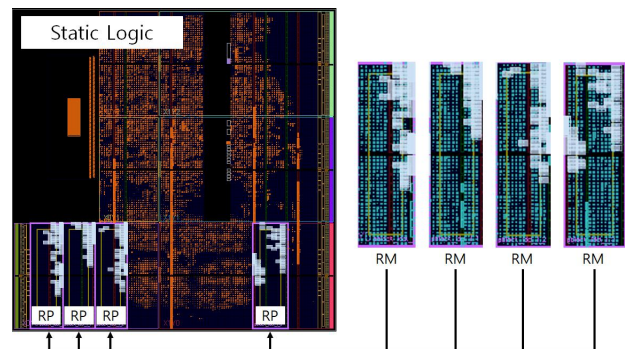


Fig 10. System Floorplanning.  
그림 10. 시스템 플로어플래닝

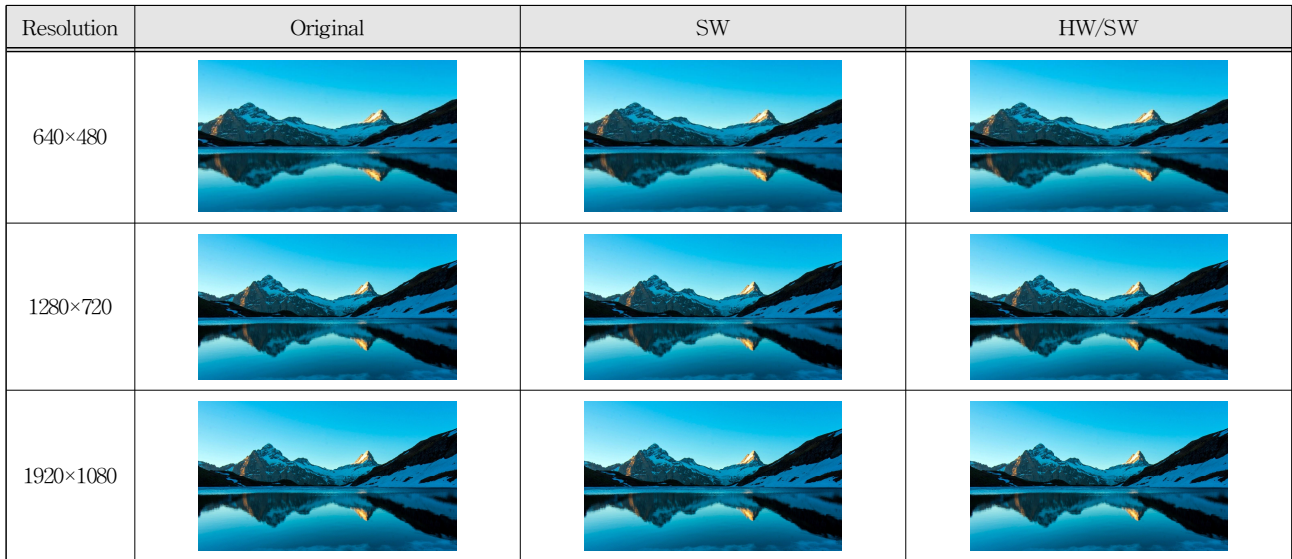


Fig. 11. Processing results of different resolutions.

그림 11. 해상도에 따른 영상 처리 결과

그림 10은 SoC 내부의 PL 영역에서 Static Logic과 하드웨어 가속기 모듈의 PRM(Partial Reconfiguration Module)을 나타낸다. 시스템이 처음 실행되면 Static Logic이 PL 영역에 다운로드 되고 PR을 통해 PRM을 부분 재구성하여 하드웨어 가속기를 작동시킨다. 필요한 연산이 끝나면 PRM에 blank bitstream을 다운로드하여 하드웨어 리소스 사용량을 줄인다.

본 논문에서 구현한 2D-IDCT 모듈은 이미지의 non-zero DCT 계수만을 연산하기 때문에 이미지의 특성에 따라 다른 연산 성능을 보여준다. 표 2는 해상도에 따른 SW design 및 HW/SW co-design

에서의 평균 2D-IDCT 연산 처리 시간을 비교한 것이며, 그림 11은 실제 처리된 이미지 결과를 나타낸다.

640×480, 1280×720, 1920×1080 해상도의 2D-IDCT 연산을 1개의 HW task를 생성하여 처리할 경우 각각 약 17.394배, 18.996배, 19.080배의 성능이 향상된 것을 확인할 수 있다. 또한 HW task를 최대 4개 생성하는 경우 각각 약 71.812배, 78.568배, 79.113배의 성능 향상이 이루어지며 이는 1개의 HW task를 생성하는 경우에 비해 약 4배의 성능 향상을 보이는 것을 확인할 수 있다. HW task를 생성하여 처리하는 경우 해상도에 따라 성능 향상이 다른 것을 확인할 수 있는데 이는 non-zero DCT 계수를 처리하는 하드웨어 가속기의 특징 때문이다. 실제 실험에 사용된 이미지의 경우 640×480 해상도에서 71.03%, 1280×720 해상도에서 65.08%, 1920×1080 해상도에서 57.43%로 해상도가 증가할수록 전체 픽셀 데이터 대비 non-zero DCT 계수의 비율이 줄어드는 것을 확인하였다. 또한 이미지의 non-zero DCT 계수의 양은 압축률에 영향을 받기 때문에 압축률에 따른 성능 향상이 다른 것을 확인할 수 있다.

표 3은 영상 압축률에 따른 SW design 및 HW/SW co-design에서의 평균 2D-IDCT 연산 처리 시간을 비교한 것이다. 1920×1080 해상도의 이미지를 사용하였고 이미지 압축률에 따른 PSNR(Peak Signal to Noise Ratio)도 표기하였다. 1개의 HW task를

Table 2. Processing Time of 2D-IDCT for Different Resolutions.  
표 2 해상도에 따른 2D-IDCT 처리 시간

Resolution	Platform	RM	Time (ms)	Speed Up (%)
640×480	SW	-	117.664	-
	HW/SW	1	6.397	1739.378
		2	3.207	3568.974
		4	1.616	7181.188
1280×720	SW	-	353.043	-
	HW/SW	1	17.656	1899.564
		2	8.837	3895.055
		4	4.437	7856.795
1920×1080	SW	-	806.173	-
	HW/SW	1	40.148	1908.003
		2	20.083	3914.206
		4	10.063	7911.259

생성하여 2D-IDCT를 수행하는 경우 압축률이 높은 순서대로 각각 약 42.918배, 26.597배, 11.001배의 성능 향상을 보여주며 PSNR이 낮을수록 높은 성능 향상을 보여준다. 영상 압축률이 높을수록 quantization 처리 이후의 non-zero DCT 계수의 개수가 줄어드는데 이는 2D-IDCT의 연산량을 감소시켜 SW에 비해 더욱 높은 성능 향상을 이룰 수 있다. 또한 최대 4개의 HW task를 생성하는 경우 각각 173.959배, 109.113배, 46.497배의 성능이 향상되어 1개의 HW task를 생성하여 처리하는 경우에 비해 약 4배의 성능 향상을 보이는 것을 확인할 수 있다.

가 높을수록 많은 성능 향상이 이루어진다. 압축률이 다를 경우 낮은 압축률에서 46.497배, 높은 압축률에서 173.959배로 압축률이 높을수록 많은 성능 향상이 이루어진다. PR을 활용하여 실시간으로 입력되는 정지 영상 데이터 특성에 따라 하드웨어 가속기를 재구성하며, 2D-IDCT 연산이 이루어지지 않을 경우 해당 하드웨어 가속기를 다시 제거할 수 있다.

### References

Table 3. Processing Time of 2D-IDCT for Images with Different Compression Ratios.

표 3. 압축률에 따른 2D-IDCT 처리 시간

Compression Ratio	PSNR (dB)	Platform	RM	Average (ms)	Speed Up (%)
109:1	29.103	SW	×	816.882	0
		HW/SW	1	18.600	4291.839
			2	9.348	8638.575
			4	4.669	17395.866
33:1	33.548	SW	×	818.471	0
		HW/SW	1	29.658	2659.697
			2	14.839	5415.475
			4	7.433	10911.314
7:1	45.708	SW	×	809.771	0
		HW/SW	1	67.473	1100.141
			2	33.744	2299.748
			4	17.049	4649.669

### III. 결론

본 논문에서는 Zynq SoC에서 재구성 가능한 하드웨어 가속기를 지원하는 멀티쓰레딩 시스템을 제안하였다.  $\mu$ C/OS-II 환경에서 재구성 가능 하드웨어 가속기를 이용하여 압축된 정지 영상의 픽셀 데이터를 복원하는 고성능 JPEG 디코더를 구현하였다. 디코딩 과정에서 가장 많은 처리 시간을 차지하는 2D-IDCT 연산을 재구성 가능한 하드웨어 가속기로 구현하고 480p, 720p, 1080p 해상도의 이미지를 사용하여 성능을 검증하였다. 실험결과 4개의 HW task를 생성하여 1080p 해상도의 이미지를 처리할 경우 최대 99 fps를 처리할 수 있고 해상도

[1] Bryan Chan Jia Ching, Ab Al-Hadi Ab Rahman, Nabihah Ahmad, "Implementation of an 8×8 Discrete Cosine Transform on Programmable System-on-hip," *Journal of Physics: Conference Series*, Vol.1049, 2018. DOI: 10.1088/1742-6596/1049/1/012084

[2] Ahmed Ben Atitallah, Patrice Kadionik, Fahmi Ghozzi, Patrice Nouel, Nouri Masmoudi, Herve Levi, "An FPGA implementation of HW/SW codesign architecture for H.263 video coding," *International Journal of Electronics and Communications*, Vol.61, NO.9, pp.605-620 2007, DOI: 10.1016/j.aeue.2006.11.001

[3] Enno Lübbers and Marco Platzner. "ReconOS: Multithreaded Programming for Reconfigurable Computers," *ACM Transactions on Embedded Computing Systems (TECS)*, Vol.9, No.1, 2009. DIO: 10.1145/1596532.1596540

[4] Xilinx, *Zynq-7000 SoC Data Sheet: Overview (DS190)*, Xilinx, 2018.

[5] Xilinx, *Vivado Design Suite User Guide Partial Reconfiguration (UG909)*, Xilinx, 2019.

[6] Xilinx, *Partial Reconfiguration User Guide (UG702)*, Xilinx, 2013.

[7] Andreas Agne, Markus Happe, Ariane Keller, Enno Lübbers, Bernhard Plattner, Marco Platzner, Christian Plessl, "ReconOS: An Operating System Approach for Reconfigurable Computing," *IEEE Micro*, Vol.34, No.1, pp.60-71, 2013. DIO: 10.1109/MM.2013.110

[8] Jooheung Lee, Narayanan Vijaykrishnan, and Mary Jane Irwin, "Inverse discrete cosine transform architecture exploiting sparseness and symmetry

properties,” *IEEE Transactions On Circuits And Systems For Video Technology*, Vol.16, No.5, pp.655-662, 2006.

DIO: 10.1109/TCSVT.2006.873155

---

## BIOGRAPHY

---

### **Hyeon-Jun Shin** (Member)



2018 : BS degree in Dept. of Electronic and Electrical Engineering, Hongik University.  
 2020 : MS degree in Dept. of Electronics and Computer Engineering, Hongik University.  
 2020 ~ : Associate Research Engineer, Chips&Media.

### **Joo-Heung Lee** (Member)



1996 : BS degree in Electronic Engineering, Inha University.  
 1998 : MS degree in Electronic Communication Engineering, Hanyang University.  
 2006 : PhD degree in Electrical Engineering, The Pennsylvania State University.

1998~2000 : Researcher, R&D Complex of LG Electronics.

2006~2011 : Assistant Professor, University of Central Florida.

2011 ~ : Professor, Electronic and Electrical Engineering, Hongik University.