

Motion JPEG용 베이스라인 기반의 디코더 설계

김경현* · 손승일* · 이민수**

*한신대학교 정보통신학과 · **엠에스웨이(주)

A Design of Baseline Based on Decoder for Motion JPEG

Kyung-hyun Kim* · Seung-il Sohn* · Min-soo Lee**

*Dept. of Information and Communication HanShin University · **MSWAY Co., Ltd.

E-mail : kkh52103@lycos.co.kr

요 약

정보화 사회가 진행되어감에 따라 카메라 센서, 디지털 카메라, 휴대폰, 영상 관련디지털 기기들이 증가하고 이로 인하여 영상정보 서비스 기술의 중요성이 크게 부각되었다. 특히 멀티미디어 응용서비스 기술에서는 영상 정보가 필수적인데, 그 영상 정보의 양이 너무 방대하여 압축 부호화를 하여 사용되고 있다. 본 논문에서는 정지영상압축 방법 중 JPEG표준에서 제시한 4가지 동작 모드 중 베이스라인을 기반으로 하는 JPEG 알고리즘을 연구하여 Motion JPEG에서 동작 가능한 디코더를 C언어를 통해 시뮬레이션하고 최적화된 결과를 VHDL로 구현하였다.

Motion JPEG의 무선전송 환경에 적용 가능한 불규칙한 스트리밍 방식의 입력데이터의 처리가 가능한 파이프라인 구조로 설계 하였다. 설계결과 Xilinx XC3S1000 FG676-4 환경에서 66.130MHz의 동작 속도를 나타내었고 최초 223클럭의 딜레이 이후 매 클럭마다 화소데이터를 얻을 수 있었다. Motion JPEG 디코더를 설계하는데 사용된 게이트는 총 54,143개이다.

1. 서 론

정보화 사회가 진행되어감에 따라 디지털 영상의 응용 분야가 급속하게 확대되고 있다. 이러한 응용분야 중 디지털 카메라의 ExIF(Exchangeable Image File Format)는 JPEG(Joint Photographic Experts Group) 압축 기술을 이용하는 업계 표준 파일 포맷이다. 이 표준의 가장 기본적인 기능들을 포함하는 기술인 JPEG 베이스라인 기술은 단 순성과 폭 넓은 지지 기반 덕분에 디지털 카메라 시스템에 있어서 핵심기술이 되었다. 또한 JPEG을 연속적으로 인코딩하는 기법을 사용한 Motion JPEG기술을 통해 보안 분야 및 교통관련 영상 등의 카메라와 관련된 동영상 캡처기술이 널리 사용되고 있다[1].

JPEG의 인코더에서는 다량의 정보를 가진 영상 신호 데이터 저장의 제약을 줄이기 위해 효율적인 영상 압축 기법을 적용 데이터양을 감소시켜 전송시킨다. 하지만 이에 따른 영상 신호의 복호화시 상대적으로 연산량 증가의 결과를 가져왔다. 이는 전력 소모 및 시간지연의 문제로써 나타나고 실시간 처리능력을 갖춘 저 전력 고속의 디코더를 위해서는 Motion JPEG 디코더 전용 하드웨어의 개발이 요구되었다[2].

이에 본 논문에서는 입력데이터의 효과적인 파이프라인 제어방식 및 고속의 연산 기법을 통해 하드웨어에 적합한 Motion JPEG용 디코더를 연

구하였다. 이를 토대로 하드웨어 설계언어인 VHDL을 이용하여 회로를 모델링하여 동작을 검증하였다.

II. JPEG 베이스라인 디코더

2-1 디코더 전체 블록도

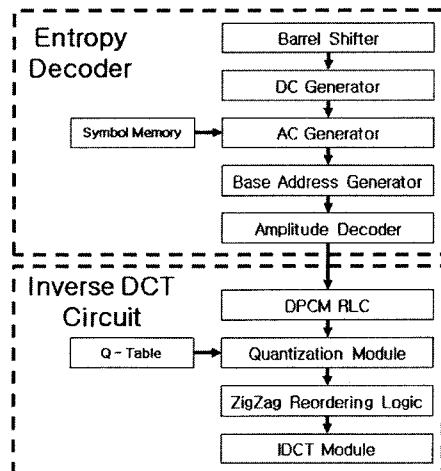


그림 1. JPEG 베이스라인 디코더 전체 블록도

그림 1은 본 논문에서 설계한 디코더 모델의 전체 블록도로서 디코더 모듈은 크게 Entropy Decoder와 Inverse Circuit의 두 가지 블록으로 구성되어 있다[3][4][9].

2-2 엔트로피 디코더

2-2-1 불규칙한 데이터 입력

디코더 연구시 가장 우선적으로 고려해야 하는 사항은 입력데이터의 저장 방식이다. 프레임메모리 방식의 디코딩 장비는 특정 타이밍에 원하는 위치의 영상 데이터를 참조하여 쓸 수 있는 장점이 있지만 영상 데이터의 저장을 위한 메모리가 이미지 사이즈만큼 추가적으로 필요하다[3].

본 논문의 연구 주제인 Motion JPEG의 영상 디코딩 장비는 무선 스트리밍 방식으로 데이터를 통신하는 경우가 대부분이며 불규칙한 데이터 입력이 이루어 질 경우를 고려하여야 한다[2]. 이에 본 논문은 메모리 사용을 최소화하여 블록사이즈 정도만 할당하여 연속적으로 데이터가 입력될 경우에도 데이터의 손실 없이 처리 가능하도록 하였다. 또한 데이터 전송 시 오류 및 손실에 의한 이미지 복원이 불가능할 때 해당 프레임의 디코딩을 중단하고 새로운 이미지정보를 바로 입력받아 처리함으로써 연속적인 이미지 처리의 오류의 누적을 막았다[2][3][10].

2-2-2 가변적인 데이터 길이 처리

엔트로피 디코딩 데이터의 입력 값은 Code word와 가변적인 유효 데이터 값이 연속적으로 연결되어 있는 형태로 입력된다. 이를 각각의 코드 정보와 고정비트의 압축 데이터로 복원하기 위해서는 한 번에 원하는 만큼 데이터를 가져올 수 있는 모듈이 필요하다[3][4]. 이에 본 논문에서는 그림2와 같이 Upperer / Lower 각각의 16비트 레지스터와 코드길이 정보에 따른 쉬프트 길이 정보를 가산기를 통해 Barrel Shifter 모듈을 제어함으로써 불필요한 지연 없이 효율적으로 데이터를 추출하는 방식을 사용하였다[2][4][11].

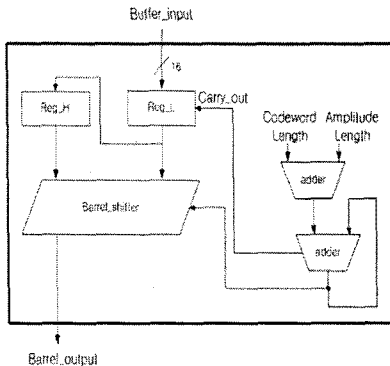


그림 2. Barrel Shifter

2-2-3 엔트로피 디코더 전체 블록도

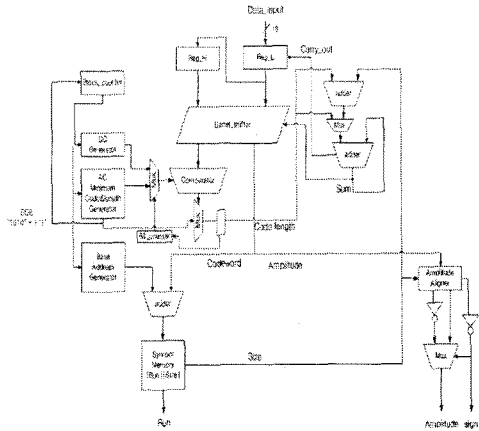


그림 3. 엔트로피 디코더 전체 블록도

그림3은 허프만테이블 기반의 Motion JPEG 디코더의 전체 블록도 이다. 최초 비트단위의 데이터들이 전송되어 처리 가능한 데이터 단위로 묶인 뒤 16비트 레지스터 Reg_L에 입력이 되고 이를 제어하는 가산기를 통해 Reg_L의 값은 16비트 단위로 Reg_H로 인가되는 형식으로 최악의 경우 Reg_H의 하위 0번째 비트와 Reg_L의 상위 15비트가 연결되어 Barrel Shifter에 입력이 될 수 있도록 한다. 이후 Block Counter 모듈의 제어에 따라서 Luma DC, Chroma DC, Luma AC, Chroma AC Generator 중 한가지의 비교기가 선택되어 동작하도록 한다. 비교기는 미리 계산된 Minimum code에 따라서 16비트 전체를 한 번에 조건 비교하여 바로 처리할 수 있도록 설계되었다. 이에 DC값의 경우 Code Length와 Category 값을 유추하여 Code word와 Amplitude 값을 각각 쉬프터에서 얻어낸다. AC값의 경우 추가적으로 Base Address Generator 모듈의 연산을 통해 Run/Size와 Code Length 값을 유추해 낸 뒤 쉬프터를 통해 유효한 데이터를 얻어낸다. 최종적으로 Run값은 별도 출력을 통해 이후 AC값의 데이터가 0인 부분을 패딩 시키며 그 외의 크기를 갖는 유효 데이터는 기존 1의 보수형태에서 2의 보수형태로 변환시켜 출력시켜 역양자화 모듈의 입력으로 사용된다[4][5][10][11].

2-3 Inverse DCT

그림 4는 역변환기의 블록도를 나타낸 것이다. 무선 전송된 입력데이터를 VLD(Variable Length Decoding), DPCM(Differential Pulse Coded Modulation), 역양자화를 통하여 얻어진 8x8블록 단위의 고정비트 데이터를 입력으로 사용한다[6].

역변환 시에도 불규칙한 데이터의 입력을 고려하여 모든 제어는 FSM(Finite State Machine)을 통하여 동작을 제어하는 구조로 설계하였다[7][8].

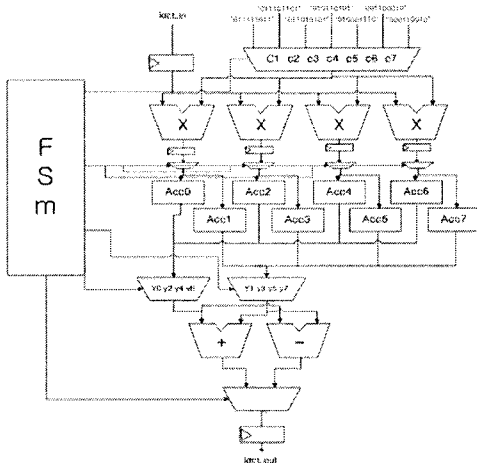


그림 4. Inverse DCT 블록도

III. 설계 결과

3-1 JPEG 디코더 타이밍 차트

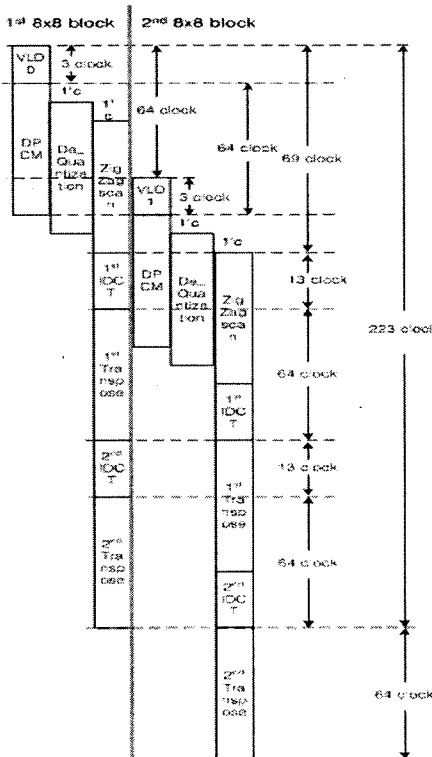


그림 5. JPEG 디코더 타이밍 차트

그림 5는 JPEG 디코더의 타이밍 차트를 나타낸 것이다. 최초 VLD 3클럭 연산 이후 8x8블록

의 가변적인 유효 데이터 값을 얻기 시작한다. 이후 DPCM모듈의 고정된 64클럭 동작을 통해 64개의 고정 블록 데이터를 생성시키며 양자화와 지그재그스캔 모듈을 연속적으로 수행한다. 이때 DPCM 모듈이 하나의 블록연산을 끝내기 3클럭 전부터 유효 화소 값을 얻는 VLD연산을 시작함으로써 VLD를 제외한 모든 모듈은 파이프라인 방식의 연속적인 동작을 통해 효율성을 증대시켰다[9][10].

결과적으로 본 논문에서는 Motion JPEG 디코딩시 무선 데이터 전송상태가 양호하여 원하는 타이밍에 VLD데이터의 입력으로 사용할 수 있다면 최초 223클럭의 딜레이 이후 매클럭 복원영상의 화소 데이터를 얻을 수 있었다.

3-2 시뮬레이션 결과

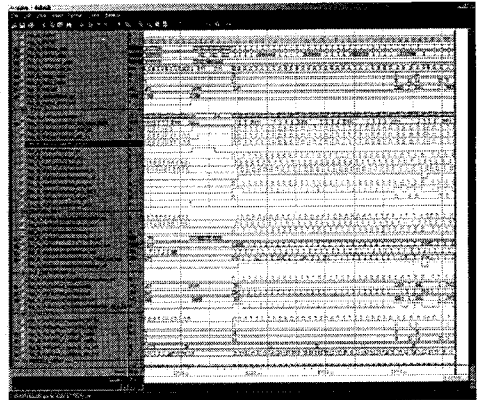


그림 6. Motion JPEG 디코더 결과파형

설계된 Motion JPEG용 베이스라인 기반의 디코더는 ModelSim-SE 6.1 시뮬레이터를 통해 그림 6과 같은 설계파형을 통해 검증하였다. 클럭주기 100ns 환경에서 (320*240) 사이즈의 영상을 인코딩하는데 총 77,023 클럭이 소요되었다.

표 1. Motion JPEG 디코더 설계 결과

분류	내용
FPGA	Xilinx xc3s1000 fg676
Tools	Xilinx ISE 7.1i
Total Equivalent gate	54,143
Maximum Frequency	66.130MHz

XilinxISE 7.1i 개발 툴을 기반으로 Xilinx XC3S1000 FG676-4 환경에서 시뮬레이션 한 결과 표1과 같이 최대 동작속도 66.130MHz를 나타내었고 총 54,143 게이트가 사용되었다.

3-3 결과영상



그림 7. 320x240 복원영상

그림 7은 설계된 Motion JPEG 디코더를 통해 복원된 (320*240) 사이즈의 영상이다. 이와 같이 연속된 시간의 복원영상들이 초당 15~30프레임 정도의 속도로 순차적으로 디코딩되어 디스플레이 장치에서 구동될 때 동영상과 같은 영상 효과를 얻게 된다.

IV. 결 론

본 논문의 입력데이터는 선행 연구되어진 JPEG 인코더의 출력 값을 통해 전달하였고, 설계된 디코더의 출력 영상 데이터는 최초 입력과 같은 raw파일의 형태로 변환하여 검증하였다.

최종적인 모듈의 실장 테스트 결과 (320*240) 사이즈의 YCbCr 4:2:2 컬러 포맷 이미지 하나를 디코딩 할 때 무선 데이터 전송 상황에 따라 최소 77,023클럭으로 처리 가능하다. 이는 66.130MHz의 동작속도를 갖는 본 논문의 성능을 고려할 때 초당 15프레임 수준으로 전송되는 Motion JPEG의 영상데이터 복호화 작업 수행이 가능하다.

향후 IC 인터페이스 모듈의 연결을 통해 카메라에서 입력받은 연속적인 영상을 영상장비의 디스플레이 장치에 실시간으로 복호화 시키는 모든 단계의 설계기술을 독자적으로 정립하고자 한다.

참고문헌

- [1] 후지와라 히로시, 정제창 역, "최신 MPEG", 교보문고 1995.
- [2] 박기현, "코덱의 세계로의 초대", 2007.
- [3] "TMS320DM6446 Digital Media System-on-Chip," <http://www.ti.com>
- [4] ITU-CCITT, "Information Technology Digital Compression and Coding of Continuous-Tone Still Images Requirements and Guideline", CCITT, 1993.
- [5] Arun N. Netravali, Barry G. Haskell, "Digital Pictures", PLENUM Press, 1994.
- [6] Vishnu Srinivasan, K. J. Ray Liu, "VLSI Design of High-Speed Time-Recursive 2-D DCT/IDCT Processor for Video Applications", IEEE, 1996.
- [7] Bret Stott, Dave Johnson, Venkatesh Akella, "Asynchronous 2-D Discrete Cosine Transform Core Processor", IEEE, 1995.
- [8] Avanindra Madisetti and Alan N. Willson, "DCT/IDCT Processor Design for HDTV Application", IEEE, 1995.
- [9] James Rosenthal, "JPEG Image Compression Using an FPGA, December 2006.
- [10] Mohammed Elbadri, Raymond Peterkin, voicu Groza, Dan Ionescu, and Abdulmoteleb El Saddik "HADWARE SURPPORT OF JPEG", IEEE, 1995.
- [11] Mario Kovac, N.Ranganathan, "A High Speed VLSI Chip for JPEG Image Compression Standard", IEEE, 1995.