

# 3GPP 표준의 터보 복호기 하드웨어 설계에 관한 연구

정회원 김 주 민\*, 정회원 정 덕 진\*

## A Study on the hardware implementation of the 3GPP standard Turbo Decoder

Ju-Min Kim\*, Duck-Jin Chung\* Regular Members

요 약

차세대 이동 통신인 IMT2000에서는 3GPP 및 3GPP2규격 모두에서 터보코드가 채널 코딩기법으로서 길쌈부호와 함께 표준으로 채택되어 있으며 특히 3GPP규격에서는 제한길이 4인 1/3 터보코드가 채택되어 있다. 본 논문에서는 상기 터보 코드를 복호하기 위한 복호기의 구조를 제시하고, 3GPP 규격의 터보 코드를 복호할 수 있는 복호기를 설계하였다. 특히 효율적인 동작을 위하여 내부 SISO 복호기로서 레지스터교환방식을 적용하고 새로운 구조의 누적 메트릭 정규화 부를 포함한 SOVA복호기를 설계하였다. 개발 터보 복호기의 성능 예측을 위하여 M ATLAB을 통하여 시뮬레이션하였으며, VHDL을 사용하여 각 모듈의 제어를 위한 제어블럭, 입력 제어 버퍼, SO VA 내부 복호기를 포함한 전체 터보 복호기를 설계하였다. 설계한 복호기는 Synopsys사의 FPGA express에서 합성하고, EPF200SRC240-3 FPGA에 이식하여 하드웨어적으로 동작을 검증하였다

ABSTRACT

Turbo codes are selected as FEC(Forward error correction) codes with convolution code in 3GPP(3rd generation partnership project) and 3GPP2 standard of IMT2000. Especially, 1/3 turbo code with K=4 is employed for 3GPP standard. In this paper, we proposed a hardware structure of a turbo decoder and developed the decoder for 3GPP standard turbo code. For its efficient operation, we design a SOVA decoder by employing a register exchange decoding block and new path metric normalization block as a SISO constituent decoder. In addition, we estimate its performance under MATLAB 6.0 and designed the turbo decoder including control block, input control buffer, SOVA constituent decoder with VHDL. Finally, we synthesized the developed turbo decoder under Synopsys FPGA Express and verified it with ALTERA EPF200SRC240-3 FPGA device.

### I. 서 론

국내 이동통신은 cdma2000 1x서비스를 시작으로 음성위주의 개인이동 통신에서 데이터를 포함한 멀티미디어 이동통신으로 변화하고 있다. cdma2000 1x서비스는 기존 2세대 이동통신보다 월등히 빠른 144kbps의 데이터 전송속도를 제공하고 있으나, 동화상 및 고품질 사운드의 멀티미디어를 지원하기에는 부족하다. 그러나 제 3세대 멀티미디어 이동통신

인인 IMT 2000 시스템과 같이 고신뢰도의 통신을 보장하는 고속 지상 통신망에서는 자동차 이동 통신에서 144kbps, 개인이동통신에서 384kbps, 사무실환경에서 2Mbps이상의 성능을 제공할 것이다. 이러한 IMT2000은 일본 및 유럽을 중심으로 하여 GSM으로부터 발전해온 W-CDMA와 북미를 중심으로 발전해온 cdma2000으로 나뉘어 지는데 이 두 가지 방식 모두에서 고 신뢰도의 데이터 통신을 제공하기 위하여 터보 코드가 표준 채널 코덱으로 채

\* 인하대학교 전자전기컴퓨터공학부 집적회로연구실(joshua1k@dreamwiz.com)

논문번호 : 020297-0708 접수일자 : 2002년 7월 8일

※ 본 연구는 2002년도 시스템 IC 2010사업의 지원으로 수행되었음.

택되었다.

특별히 W-CDMA를 위한 3GPP규격에서는 커먼 패킷채널(CPCH), 데디케이트드채널(DCH), 다운링크 스페어드채널(DSCH), 그리고 포워드억세스채널(F A -CH)에서 8 스테이트의 터보코드를 채택하였으며 최대 5114 크기의 인터리버를 포함한다. 이러한 터보 코드의 복호를 위해서는 두개의 내부 Soft-in Soft-out(SISO)복호기가 필요하며 사용되는 알고리즘은 MAP(Maximum a posteriori)방법과 SOVA (soft output viterbi algorithm)이다. 이중 SOVA는 MAP보다 비교적 낮은 복잡도를 가지기 때문에 하드웨어를 구현하는데 있어서 좀더 용이하다.<sup>[1][2]</sup>

따라서 본 논문에서는 이러한 터보코드의 복호기 구현으로 주목받고 있는 연관정 출력 비터비 알고리즘(SOVA)를 이용한 고성능 터보 복호기의 구현을 위하여, 레지스터 교환 방식의 SOVA복호기를 내부 SISO로 적용하였으며, 3GPP규격에 제시된 프라이머리터리버 및 디인터리버와, SOVA 복호기의 출력인 신뢰도 값에서 다음 복호단의 사전정보값을 추출하기위한 외부정보추출 모듈, 반복복호 제어를 위한 제어모듈로 구성된 터보 복호기의 하드웨어 구조를 제시한다. 설계된 터보복호기의 성능 예측은 MATLAB을 통하여 수행하였으며, Model Technology의 ModelSim의 VHDL을 사용하여 설계하였으며, 기능검증도 동시에 수행하였다. 설계된 터보 복호기는 ALTERA사의 EPF200SRC503-3 F PGA에 이식하여 검증하였다.

## II. 이론적 배경

### 1. 3GPP 규격의 터보 부호기

3GPP의 표준규격으로 제시된 터보부호기는 그림 1에 나타난 바와 같이 제한길이가 4인 동일한 두개의 회귀부호기를 인터리버를 사이에 두고 병렬로 동작시켜 심벌을 만드는 1/3 FEC 부호기이다. 그림 1에 도시한 터보 부호기에서  $x_k$ 는 메시지를,  $z_k$ 는 첫 번째 패리티를, 그리고  $z'_k$ 는 두 번째 패리티를 나타낸다.

따라서 위의 터보 부호는 시스티메틱 부호이므로 프레임 길이가 K인 입력 시퀀스  $x_k$ 에 따른 그 출력 비트열은  $x_1, z_1, z'_1, x_2, z_2, z'_2, \dots, x_K, z_K, z'_K$  가 되며, 모든 입력비트가 입력된 후에 두번째 내부 부호기는 동작을 정지시키고 첫번째 내부

부호기의 출력을 자신의 입력으로 사용하여 내부 메모리를 중단 시킨다. 그리고 난후 첫번째 내부 부호기의 동작을 정지시키고, 두번째 내부 부호기의 출력을 그 자신의 입력으로 사용하여 중단시킨다. 따라서 트렐리스를 중단시키기 위한 비트열은  $x_{K+1}, z_{K+1}, x_{K+2}, z_{K+2}, x_{K+3}, z_{K+3}, x_{K+1}', z_{K+1}', x_{K+2}', z_{K+2}', x_{K+3}', z_{K+3}'$  이 된다.<sup>[4]</sup>

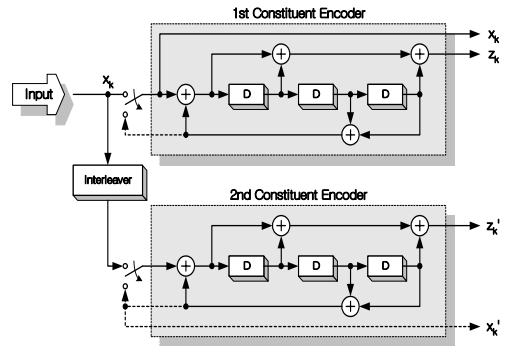


그림 1. 3GPP 규격의 터보 부호기  
Fig. 1. 3GPP standard Turbo encoder

### 2. SOVA를 이용한 터보 복호기

터보 복호기는 두개의 회귀길쌈부호기로 생성된 각각의 패리티를 이용하여 복호하므로 그림 2에 보인바와 같이 2개의 SISO(Soft-in Soft-out)복호기가 필요하다. SOVA복호 알고리즘은 최대 우호 시퀀스를 찾아 송신된 메시지를 복원한다. SOVA를 이용한 복호방법은 기존의 경관정 비터비복호의 방법을 그대로 적용하면서 복호된 비트에 대한 신뢰도 값을 계산하는 부가 첨가되어 그 출력으로서 경관정과 함께 신뢰도를 나타내는 연관정을 포함하게 된다. 따라서 그림 2에 보인바와 같이 송신된 코드는 잡음환경에서 변질되어 수신되며,  $y_{1k}, y_{2k}, y_{3k}$ 로 수신단에 입력된다. 첫번째 SOVA 복호기는  $y_{1k}, y_{2k}$ 를 입력으로 초기 복호를 수행하여 경관정 비트열과 신뢰도 정보열을 생성시킨다. 생성된 신뢰도 정보열을 기반으로 수신 메시지  $y_{1k}$ 와 첫번째 복호기의 사전정보확율과의 연산을 통하여 외부 정보열이 추출되어 인터리빙된 후 두번째 SOVA복호기의 입력의 사전정보로 사용된다. 두번째 SOVA 복호기는 인터리빙된  $y_{1k}$ 와  $y_{3k}$  그리고 첫번째 복호기로부터 생성된 외부정보를 입력으로 복호하게 된다. 두번째 복호기로부터 추출된 외부정보는 디인터리빙되어 다시 첫번째 복호기의 사전정보로 사용되므로

써 반복복호를 수행할 수 있게 된다.

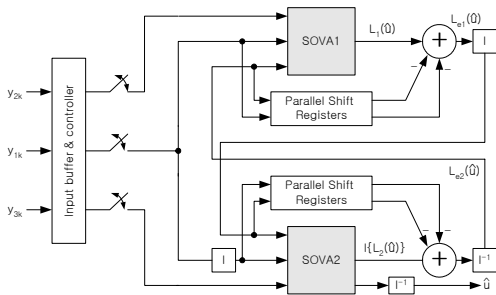


그림 2. 설계 터보 복호기의 간략도  
Fig. 2. A simplified block diagram of the developed turbo decoder

### III. 실험 및 결과

#### 1. 터보 복호기의 하드웨어 구성

그림 3에 보인바와 같이 수신된 입력 메시지(msg)와 첫번째 패리티(parity1), 및 두번째 패리티(parity2)가 입력 버퍼의 역할과 전체 터보 복호기의 제어를 담당하는 Tb\_ctl블록으로 입력된다. 입력이 완료되면 Tb\_ctl 블록은 첫번째 복호블록을 활성화 시키며 첫번째 SOVA복호기(SOVA1)으로 메시지와 첫번째 패리티가 입력되면서 복호가 시작된다. 첫번째 SOVA에서 출력된 복호 비트에 대한 우호도(Likelihood)는 Le\_extractor1의 입력으로 인가되어 지연된 입력 메시지와 사전정보 확률을 감하여 외부정보열을 생성하며, 이는 프레임 인터리버에 의해 인터리빙된 후 두번째 SOVA 복호기(SOVA2)의 사전정보로 입력된다. 또한 메시지와 두번째 패리티가 동시에 SOVA2의 입력으로 인가 되도록 delay chain를 사용하였으며, 상기 지연된 메시지는 프레임인터리버에서 인터리빙 되어 사전정보의 입력과 동시에 두 번째 SOVA복호기의 입력으로 들어가게 된다.

상술한 바와 같이 첫번째 복호가 완료되고, 두번째 복호가 개시될때는 Tb\_ctl 블록에서 비활성화되었던 두번째 복호블록을 활성화 시킨다. 첫번째 복호 과정과 동일한 동작을 통하여 SOVA2의 복호가 완료되었으나 초기 정해진 수만큼의 반복복호가 완료되지 않았다면 Le\_extractor2와 디인터리버를 통하여 다시 SOVA1의 사전정보로 입력되며 이와 동시에 메시지와 패리티가 입력되도록 버퍼출력이

활성화되어 다시 처음 복호와 같은 과정으로 복호한다. 상기와 같이 반복복호중, 정해진 반복 복호횟수만큼 반복 복호 완료한 경우 Tb\_ctl블록에서는 최종 출력 생성블록인 Out\_gen을 활성화 시켜서 최종 복호출력을 생성시킨다. 상술한 바와 같이 설계된 터보 복호기는 16회 까지 반복복호 횟수를 정하여 반복복호 할 수 있다.

입력 N\_iteration으로 정해진 반복복호 횟수 만큼 복호가 진행되었으면 두번째 SOVA복호기의 경관정 출력 값이 디 인터리빙되어 f\_hard\_out으로 최종 출력되게 된다. 여기서 프레임길이를 f\_ln이라고 하고, 반복복호 횟수를 n이라고 한다면  $(f\_ln + 4) + n(2 * f\_ln + 56)$  클럭의 지연을 갖는다.

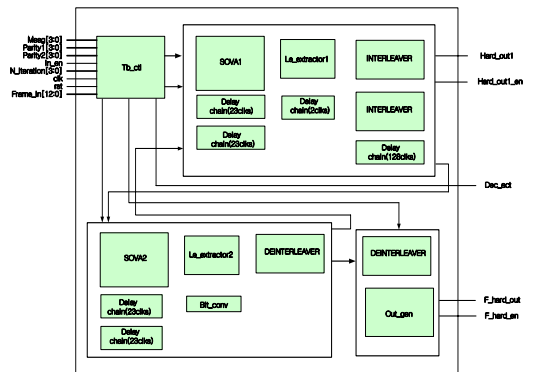
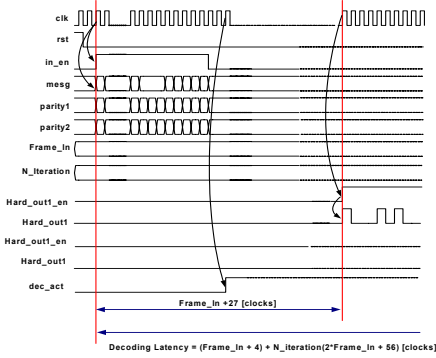
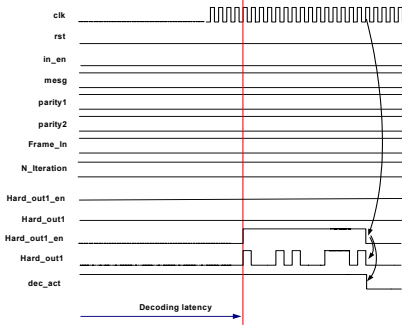


그림 3. 터보 복호기의 내부모듈 구성도  
Fig. 3. A block diagram of the implemented turbo decoder

그림 4에 터보 복호기의 주요 포트에 대한 타이밍도를 도시하였다. 먼저 rst 신호는 복호기 콘트롤 모듈을 초기화 시킴으로써 복호기 동작을 준비한다. frame\_ln은 프레임길이를 설정하는 것으로 응용분야에 맞게 초기부터 유지시킨다. in\_en신호를 입력 신호인 mesg, parity1, parity2가 입력되는 동안 high로 유지시킨다. n\_iteration 신호는 반복 복호 횟수를 결정하는 것으로 복호기 동작전에 정의한다. 입력 후 (프레임길이 + 27) 클럭의 지연후에 첫번째 복호기의 출력을 Hard\_out1에서 출력하며, 출력 중에는 Hard\_out1\_en이 high로 유지 된다. 입력후 (프레임길이+ 4) + 반복복호횟수(2 \*프레임길이 + 56)후 최종 출력이 F\_hard\_out포트에서 출력되고, 출력 중에는 F\_hard\_en이 high로 유지된다. dec\_act는 내부 입력버퍼가 다 채워진 후 복호 시작시 high로 되어 반복복호가 모두 종료되었을 때 low상태로 된다.



(a)



(b)

그림 4. 개발된 터보 복호기의 동작 타이밍도 (a) 벡터입력 (b) 복호 출력  
Fig. 4. Timing diagram of the developed turbo decoder (a) vector input (b) decoded output

2. 터보복호기 세부 블록

1) Tb\_ctl 블록

Tb\_ctl 은 터보복호기 주 제어블록으로서, 그림 5에 도시한 바와 같이 입력 버퍼(Buffer)와 복호기 컨트롤러(SOVA\_ctl)로 구성되어 있다. 입력 버퍼는 수신데이터를 내부 메모리에 차례로 저장하며, 버퍼가 다 채워지면 복호기 컨트롤러를 동작시켜 터보복호기 내부 모듈을 기동시킴으로써 반복복호를 수행하게 한다. 복호기 컨트롤러는 상기 버퍼로부터의 트리거 신호를 기점으로 버퍼내 저장 데이터를 복호블록으로 전송되게 하고, 첫번째 복호블록을 활성화시켜서 복호를 시작한다. 또한 첫번째 복호가 완료되면 적절한 시점에서 두번째 복호블록을 활성화 시키고 첫번째 복호블록을 비활성화 시켜 두번째 복호가 이루어 질 수 있도록 터보 복호기 내부 블록을 각각 제어 한다. 또한 인터리빙과 디인터리빙 시에는 첫번째 복호블록 과 두번째 복호블록 모

두가 활성화 되며, 지정된 반복복호 횟수(n\_iteratio n)만큼 모든 반복복호가 완료 되면 출력 모듈을 활성화 시켜, 최종 복호 결과를 생성시키도록 한다.

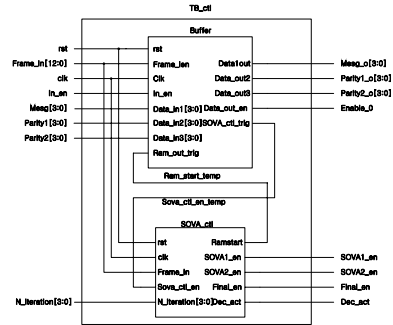


그림 5. 터보 복호기 내부 모듈 컨트롤 블록의 간략도  
Fig. 5. A block diagram of the control module of the turbo decoder

2) SOVA 복호기 블록

SOVA 복호기는 터보 복호기의 성능을 결정하는 가장 중요한 블록이므로, 이의 효율적 구조설계는 고성능의 터보복호기를 설계하기 위하여 필수적이다. 설계된 SOVA 복호기는 가지메트릭연산장치(branch calculation unit), 가산비교선택장치(add compare selection unit), 생존경로 판단 기능의 경관정 생존 메모리 장치(hard decision SMU)와 H ageneuer의 신뢰도 갱신이 이식된 연관정 생존 메모리 장치(Soft decision SMU)와 컨트롤러로 구성되었다.<sup>[3][7]</sup> 특히 본 논문에서는 낮은 복호지연과 고속동작에 유리한 터보 복호기 설계를 목적으로 그림 6(a)와 같은 누적메트릭 정규화 장치를 제안 하여 이식하였다.

기존의 비터비 복호기에서는 누적 메트릭의 오버플로우를 방지하기 위하여 오버플로우 발생시 각 스테이스의 누적메트릭에서 정해진 임계치를 각각 감산하거나, 정규화시점에서 각 스테이트의 메트릭중 가장 작은값으로 모든 누적메트릭을 감산하는 방식으로 정규화를 수행한다. 그러나 이러한 방식은 정규화 지연을 야기시켜 전체 복호기의 동작속도의 증가에 방해가 되며, 또한 하드웨어의 복잡성을 증가시킨다. 따라서 본 논문에서는 그림6(a)에 도시한 바와 같이 누적 메트릭을 신뢰도에 영향을 미치지 않는 범위로 비트를 2분하여, 정규화 요구시에 각 메트릭의 상위 수 비트만을 1비트 쉬프트 연산을 하여 정규화 한다. 이와 같이 기존의 메트릭 전체 비트수의 감산을 통한 메트릭 정규화를 일부 비트

의 1비트 쉬프트로 대치함으로써, 간단한 하드웨어 구조로 전체 가산비교선택장치의 동작속도를 높일수 있다.

또한 설계된 SOVA복호기는 그림 6(b)와 같이 레지스터 교환 방식을 사용하므로써 23 클록의 짧은 복호 지연을 가지고 있으며, 20단의 레지스터 교환 스테이지로 설계 하였다.<sup>[7]</sup> 더불어 신뢰도 갱신 방식은 Hagenauer가 제안한 방식을 적용하였다.<sup>[5]</sup> <sup>[7]</sup>이는 개발된 터보 복호기의 내부 SISO(Soft in Soft out)복호기로 사용되며 경관정 복호값을 출력할 뿐 아니라 그에 해당하는 신뢰도 값을 동시에 출력한다. 또한 1/3 터보코드의 복호를 위해서 두개의 내부 SOVA 복호기가 이식되었다.

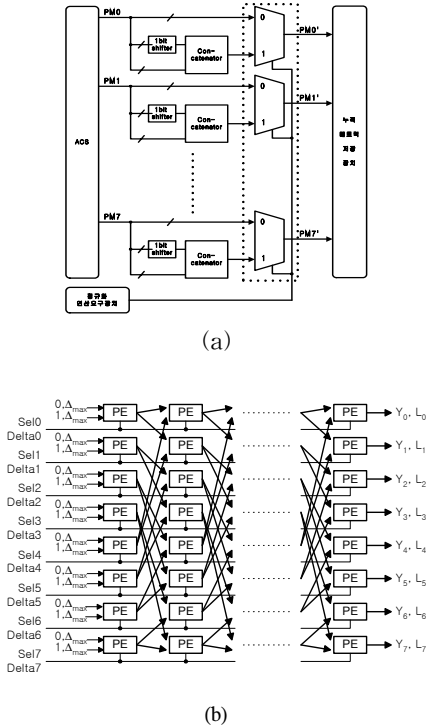


그림 6. (a) 이식된 누적메트릭 정규화 모듈 블록도 (b) 신뢰도 갱신을 포함한 레지스터교환 방식의 SMU

Fig. 6. (a) The implemented path metric normalization module (b) The register exchange SMU with reliability update

### 3) Le\_extractor 블록

SOVA복호기의 출력은 우호도(likelihood)

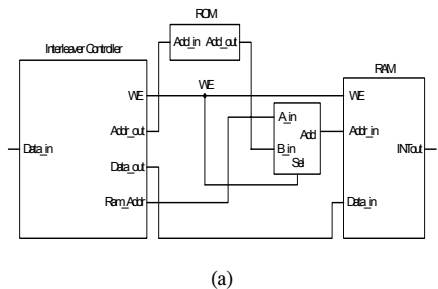
$L(\hat{u}_k)$  이므로 이로부터 외부정보 확률을 추출하기 위해서는

$$L(\hat{u}_k)_E = L(\hat{u}_k) - (L(u_k)_A + L_{c,y_{1k}}) \quad (1)$$

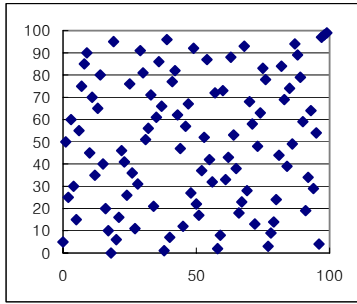
를 적용하여야 한다. <sup>[1][2][5][6]</sup> 여기서 따라서 외부 정보 확률을 추출하기 위해서는 신뢰도 값  $L(\hat{u}_k)$ 에서 메시지 정보  $L_{c,y_{1k}}$ 와 그 SOVA복호기의 사전 정보확률  $L(u_k)_A$ 값을 빼 주어야 하며, 이렇게 하여 추출된 외부정보확률  $L(\hat{u}_k)_E$ 은 그 다음 SOVA 복호기의 사전정보 확률로 사용된다. 따라서 Le\_extractor는 식(1)에 보인 바와 같은 과정을 통하여 외부 정보확률을 추출한다.

### 4) Interleaver/Deinterleaver 블록

프라임 인터리버<sup>[4][9]</sup>는 소수를 이용하여 인터리빙 패턴을 결정하며, GF 인터리버<sup>[10]</sup>와 마찬가지로 가지치기(Pruning)를 통하여 프레임 길이와 인터리빙 패턴을 일치시키는 방법으로 설계된다. 설계한 터보 복호기에서는 검증의 편의를 위하여 100 bit 크기의 인터리버를 설계 하여 이식하였으며, 인터리빙 패턴을 프라임 인터리버의 인터리빙 패턴 생성 방식에 따라 MATLAB에서 코딩하여 생성시켰다. 그림 7(a)에 설계한 인터리버의 블록도를 도시하였으며, 내부 ROM은 그림 7(b)에 보인바와 같은 인터리빙 패턴을 저장하고 있다. 그림 7(a) 도시한 바와 같이 Interleaver controller의 입력으로 들어온 데이터는 ROM에 저장되어 있는 인터리빙 패턴에 따른 RAM의 위치로 순서가 바뀌어 저장되며, 저장 후에는 RAM의 순차적인 주소지정을 통하여 데이터를 출력하므로, ROM에 저장된 인터리빙 패턴에 따라 데이터의 순서가 바뀌어 출력된다. 한편, 디인터리버 역시 인터리버와 동일한 블록으로 구성되며 단지 입력데이터가 입력 순서대로 메모리에 저장되고, 출력시에 인터리빙 패턴 순서에 따라 출력하므로써 원래의 순서로 복귀되도록 설계 하였다.



(a)



(b)

그림 7. 이식된 프라임 인터리버의 (a) 블록도 (b) 100bit 인터리빙 패턴

Fig. 7. (a) The block diagram (b) 100bit interleaving pattern of the implemented prime interleaver

5) 최종 출력 블록

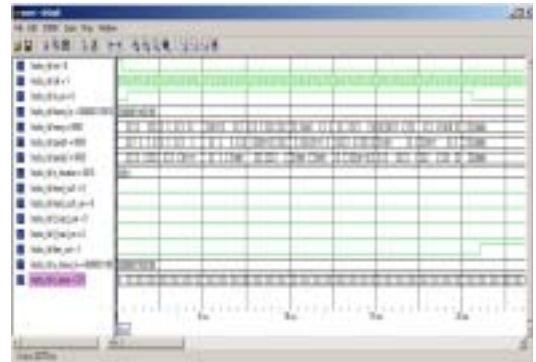
최종 복호 출력은 N\_iteration 에서 정해진 횟수만큼의 모든 반복복호가 완료된 후에만 출력된다. 반복복호가 완료되면 상기 Tb\_ctl블록에서 최종 출력 블록을 활성화 시켜 두번째 복호블록에서 최종 복호된 비트열을 디인터리빙하여 출력하게 된다.

3. 테스트 결과

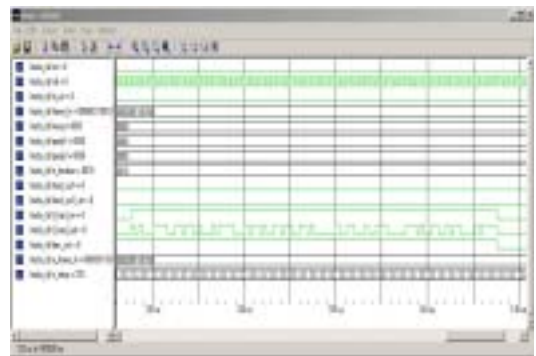
설계된 터보 복호기의 설계와 기능검증을 위하여 Model Technology사의 ModelSim EE 5.2를 사용하였다. 검증을 위한 테스트 벡터는 랜덤으로 generation된 100bit frame을 Matlab으로 코딩한 터보 부호화기를 통과시켜 메시지 비트열과 패리티 비트열을 얻고 이를 각각 6dB Eb/No의 AWGN 채널모델을 통하여 노이즈를 추가하여 채널신뢰도를 고려한 수신 데이터를 추출하였다. 이 수신 데이터를 4bit 연관정 정보로 하여 텍스트 파일로 저장하였다. Simulation을 위해서 테스트벤치에서 이 텍스트 파일을 입력 벡터로 사용하여 동작적 검증을 수행하였다. 이를 FPGA에 이식하기 위하여 설계한 코드는 Synopsys사의 FPGA Express를 이용하여 ALTERA EPF10k200SRC240-3 디바이스를 타겟으로 합성하였으며, ALTERA 사의 MAXPLUS2에서 P&R을 수행하였다. 타겟 FPGA로의 P&R이 완료된후 생성한 넷리스트를 이용하여 다시 Modelsim EE 5.2과 MAXPLUS2에서 포스트레이아웃 시뮬레이션을 동작을 검증하였고, MAXPLUS2에서의 타이밍시뮬레이션으로부터 최대 동작 클럭속도가 14.1MHz로 측정되었다. 또한 내부 테스트 회로를 추가로 구현하

여 모든 설계 회로를 EPF10K200SRC240-3 디바이스에 이식하여 HP1663A 로직분석기를 사용하여 하드웨어 동작을 최종 검증하였다.

그림 8에 설계된 터보 복호기의 시뮬레이션 결과를 도시하였으며 그림 8에 실험에 사용된 6dB Eb/No의 AWGN채널 환경에서 오류가 발생한 Lena 이미지와 설계된 터보 복호기에서 오류 정정된 Lena 이미지를 도시하였다. 그림 8(a)에 보인바와 같이 오류가 존재하는 수신데이터는 /turbo\_tb/mesg와 /turbo\_tb/parity1과 /turbo\_tb/parity2에 입력으로 추가되며 상기 입력데이터가 입력되는 동안 /turbo\_tb/in\_en은 high상태를 유지한다. 그림 8(b)는 2회 반복복호후의 복호 출력 데이터이다. 최종 복호데이터는 그림 8(b)에서 /turbo\_tb/h\_hard\_out이며, 최종복호데이터가 출력되는 동안 /turbo\_tb/h\_hard\_en은 high를 유지한다. 그림 8(b)의 결과로 6dB Eb/No의 AWGN 채널에 의하여 발생한 오류는 모두 정정되었음을 확인할 수 있었으며, 동일한 조건을 Lena 이미지에 적용하여 복호하였을 때의 오류정정 결과를 그림 9(a)(b)에 도시하였다.



(a)



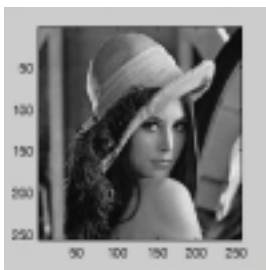
(b)

그림 8. (a) 입력 벡터가 터보 복호기로 인가될때의 시물레이션 결과 (b)최종 복호 출력때의 시물레이션 결과  
 Fig. 8. Simulation result (a) when input vectors are transferred to the turbo decoder (b) when the decoded output appears

그림 9(a)는 테스트 벡터와 동일한 6dB Eb/No의 AWGN 채널에서 오류가 발생한 Lena 이미지 이며, 그림 9(b)는 그림 9(a)의 데이터를 터보 복호기에서 복호한 Lena 이미지 이다. 그림 9(b)에 보인바와 같이 복호된 이미지에는 오류가 모두 제거되어 있음을 알 수 있다.



(a)

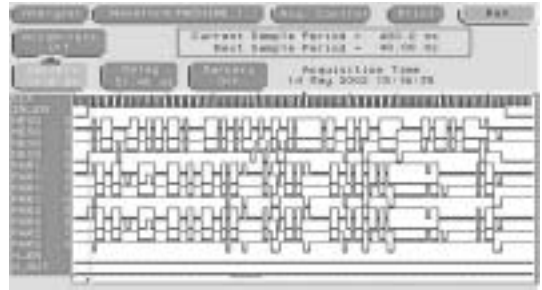


(b)

그림 9. (a) 6dB Eb/No AWGN 채널에 의해서 오류가 발생한 Lena 이미지 (b) 터보 복호기에서 2회 반복후 오류 정정된 Lena 이미지  
 Fig. 9. Lena 이미지 (a) corrupted by AWGN channel of 6dB Eb/No (b) decoded after 2 times iterative decoding

그림 10에 FPGA에 이식된 터보복호기의 측정 결과를 도시하였다. 테스트 벡터는 시물레이션을 위한 테스트 벡터와 동일하며 동작클럭은 1MHz로 하였다. 입력 타이밍도를 그림 10(a)에 도시하였으며, MSG 0~3은 메시지 입력이며, PAR1 0~3은 첫번째 패리티 입력이고, PAR2 0~3은 두번째 패리티 입력을 측정한 결과 이다. 그림 10(b)에 2회 반복복호후의 최종 복호 출력을 도시하였다. 그림 10

(b)에서 최종 복호 출력은 H\_OUT이며, 그림 8(b)의 결과와 상기 측정결과로부터 설계된 터보 복호기가 성공적으로 동작함을 알 수 있다.



(a)



(b)

그림 10. (a) 입력 벡터가 터보 복호기로 인가 될 때의 측정 결과 (b) 최종 복호 출력때의 FPGA 측정결과  
 Fig. 10. FPGA test result (a) when input vectors are transferred to the turbo decoder (b) when the decoded output appears

#### IV. 결 론

본 연구에서는 3GPP규격의 터보 복호기를 설계하였다. 설계된 터보복호기는 두개의 내부 SISO복호기로서 SOVA복호기를 사용하였으며, 전체 복호속도 결정의 가장 중요한 요소인 SOVA복호기의 동작 속도 증가를 위하여 새로운 매트릭 정규화 모듈을 이식하였으며, 복호지연의 최소화를 위하여 레지스터교환구조를 적용하였다. 또한 프라임 인터리버블록, 외부정보추출블록, 최종출력 블록을 각각 설계하여 이식하였고 상기 블록의 제어와 입력 제어를 위한 컨트롤블록을 설계하여 터보복호기를 구성하였다. 터보복호기는 Model Technology의 ModelSim의 VHDL을 사용하여 설계하였으며, Synopsys의 FPGA Express에서 ALTERA의 EPF200S

RC503-3 FPGA를 타겟으로 합성하였고, ALTERA의 MAXPLUS2에서 P&R을 수행하였다. 동작 검증 및 타이밍 분석은 ModelSim과 MAXPLUS2에서 수행하였다. 또한 설계한 터보복호기를 EPF20K10 RC503-3 FPGA에 내부 테스트 회로를 포함하여 이식하였으며, HP1663A 로직분석기에서 복호 동작을 확인 하였다. 상기와 같은 시뮬레이션 결과와 측정된 검증 결과를 비교하여 성공적인 동작을 확인 할 수 있었다. 또한 5회 반복 복호를 기준으로, ALTERA사의 MAP 기반의 터보 복호기의 복호 동작에 있어서 1샘플당 20클럭이 소요[11]되는데 비하여, 본 논문의 SOVA 기반의 터보복호기는 동일한 5회반복복호를 기준으로 1샘플당 15클럭이 소요되므로 동일한 클럭에서 ALTERA사의 터보복호기 대비 1.3배의 복호속도를 보임을 알수 있었다. 본 연구 결과는 다양한 터보 복호기의 응용을 위한 터보복호기 하드웨어 구조설계의 기반이 될 것이다.

### 참고문헌

- [1] Lang Lin, Roger S. Cheng, "On The Tail Effect of SOVA-Based Decoding For Turbo Codes," Global Telecommunications Conference, 1997. GLOBECOM'97., IEEE, vol. 2, pp. 644-648, 1997.
- [2] Lutz Papke, Patrick Robertson, "Improved Decoding with the SOVA in a Parallel Concatenated (Turbo-code) Scheme," Proceedings of the 1996 IEEE International Conference on Communications, Vol 1, pp 102-106, 1996.
- [3] O. J. Joeressen, and H. Meyr, "A 40Mb/s Soft-Output Viterbi Decoder," IEEE Journal of Solid-state Circuits, Vol. 30, No. 7, 1995.
- [4] "3G TS 25.212 v.3.2.0 Technical specification", pp. 15-16, Mar., 2000.
- [5] Joachim Hagenauer, and Peter Hoeher, "A Viterbi Algorithm with Soft-Decision Outputs and its Applications," Proceedings of IEEE Globecom Conference, Dallas TX, USA, pp. 1680-1686, 1989.
- [6] Gerhard Bauch, Volker Franz, "A comparison of Soft-In/Soft-Out Algorithm for 'Turbo-Detection'," Proceedings of the International Conference on Telecommunication, pp. 259-263, Jun. 21, 1998.
- [7] Jumin Kim, Taewhan Koh, Duckjin Chung, "VLSI implementation of SOVA decoder for 3GPP standard Turbo code with efficient ACS unit," Third International conference on Information, communication & signal processing(ICICS) Singapore, pp.326-329, 15-18 Oct., 2001.
- [8] Taewhan Koh, Jumin Kim, Duckjin Chung, "The Performance Estimation method of Turbo Internal Interleavers," Third International conference on Information, communication & signal processing(ICICS) Singapore, pp. 456-459, 15-18 Oct., 2001.
- [9] A. Shibutani, H. Suda, and Y. Yamamoto, "Performance of W-CDMA mobile radio with turbo codes using prime interleaver," Proc. IEEE VTC 2000-Spring, pp. 946-950, Tokyo, 2000.
- [10] M. Eroz, A. R. Hammons Jr., "On the Design of Prunable interleavers for Turbo Codes," Vehicular Technology Conference, 1999 IEEE 49th, vol.2, pp. 1669-1673, 1999.
- [11] ALTERA MEGA core specification, "Turbo Encoder/Decoder MegaCore Function User Guide," Version 1.0, April, 2000.



김 주 민(Ju-Min Kim)

정회원



1993년 2월 : 인하대학교  
전자재료공학과 졸업  
1995년 8월 : 인하대학교  
전자재료공학과 공학석사  
1999년 3월 ~ 현재 : 인하대  
학교 전자재료공학과 박사과정

<주관심분야> FEC 알고리즘, VLSI 구현, 신호처리

정 덕 진(Duck-Jin Chung)

정회원



1970년 2월 : 서울대학교  
전기공학과 졸업  
1984년 8월 : Utah State  
Univ. 전기공학과 공학석사  
1988년 8월 : Univ. of  
Utah 전기공학과 공학박사  
1980년 ~ 1988년: 한국

전자통신연구원 책임연구원

1989년 ~ 현재: 인하대학교 정보통신공학부  
교수

<주관심분야> FEC 알고리즘, 인공지능, Genetic Algorithm, VLSI 구현, 신호처리