

동기식 IMT-2000을 위한 터보코드의 내부 인터리버의 설계

나 병철, 정 상국, 남 명우, 노 승용
서울시립대학교 전자전기공학부
전화 : (02) 2210-2934 / 팩스 : (02) 2249-6802

Design of Internal Interleaver of Turbo Code for Sync-IMT2000

Na Byoung Chul, Jeong Sang Kook, Nam Myoung Yoo, Rho Seung Yong
School of Electric Engineering, The University of Seoul
E-mail : nbc@odyssey.uos.ac.kr

Abstract

A parallel concatenated convolutional coding scheme consists of two constituent systematic convolutional encoders linked by an interleaver. The information bits at the input of the first encoder are scrambled by the interleaver before entering the second encoder.[1]-[3] Now many different interleavers are presented for Turbo code. The topic of this paper is to design of the interleaver which is for IS-2000 Turbo code.

I. 서론

최근 IMT-2000에 관심이 집중되면서 Turbo Code를 이용한 채널부호 및 그의 응용분야가 계속 넓어지고 있다. Turbo code는 두 개의 recursive systematic convolutional encoder를 이용하며, 물론 이 두 개의 엔코더는 인터리버를 통해서 연결된다.

음성통신과는 달리 시간지연의 영향이 적은 데이터통신에서는 터보코드의 적용이 용이하다. 더 나아가 잡음 환경이 나쁜 위성 채널에서의 적용에 관한 연구가 폭 넓게 이루어지고 있으며, 현재 상당한 진전을 보이고 있다. 본 논문에서는 그러한 적용을 위한 차세대기준에 맞는 Turbo code의 internal interleaver의 설계와 문제점 보완에 그 목적을 두고 있다.

IS-2000에서는 $R = 1/2, 1/3, 1/4$ 터보코드를 정의하고 있으며, 입력 신호는 N_{turbo} 개로 하고 있다. 이 N_{turbo} 개의 입력 신호를 통해서 $(N_{turbo} + 6)/R$ 개의 coded output bits를 생성한다. 여기에 쓰이는 Turbo interleaver는 어드레스의 sequence를 산출해낸다. 또한 interleaver Parameter n 을 이용하고 있다.[4] 따라서 논문에서는 이러한 점을 감안하여 하드웨어적 기초를 바탕으로 Turbo Interleaver 설계를 하였다. 또한 일반적인 block interleaver과의 성능 비교를 통해서 IS-2000에 제시된 Turbo internal interleaver의 성능을 체크하였다.

II. 본론

서론에서 언급한 바와 같이 하드웨어 설계를 위하여 IS-2000 알고리즘에서 제안한 방법을 이용하여, data,

CRC 와 reserved input bits를 통틀어 엔코더에 입력 되는 총 비트수를 N_{turbo} 라 한다. 이와같은 입력신호를 가지는 터보 코드의 interleaving 과정은 아래와 같다.

- ① $N_{turbo} < 2^{n+5}$: 이 조건을 만족시키는 parameter n 을 구한다.
- ② counter 는 0 으로 초기화시킨다. 이 때 카운터의 길이는 $(n + 5)$ bits 이다.
- ③ counter에서 n bits의 MSB를 추출하여 1을 더해 새로운 값을 만든다. 그리고 이 새로운 값에서 n bits의 LSB를 선택한다.
- ④ counter의 LSB- 5 bits를 이용하여 Lookup Table 에서 n -bits의 output를 추출해낸다.
- ⑤ 과정 3, 4를 통해서 얻어진 값을 곱한다. 이 값의 n bits LSB를 이용하게 된다.
- ⑥ counter 의 다섯 번째 bit를 Bit-reverse 시킨다.
- ⑦ interleaver의 output address 는 과정 6과 5의 결과를 이용하여 구한다.
- ⑧ N_{turbo} 보다 같거나 크면 구해진 어드레스를 이용한다.
- ⑨ counter를 증가시키면서 과정 3에서 8을 반복한다.

[4]

위와 같은 과정을 통하여 interleaving된 결과를 읽어 낼 수 있는 output address를 생성해 낸다.

이와 같이 IS-2000에서 규정된 내용에 따라 동기식 적용을 위한 Interleaver을 설계했다. 다음은 설계된 Interleaver의 블록도이며, 본논문에 제시된 interleaver의 block length는 378bits이다. block length에 따라서 Turbo Interleaver parameter n 이 정해지며, 본논문에 서는 length 378에 따라 IS-2000에 규정되어진 $n=4$ 을 [4] 이용하여 설계를 수행하였다.

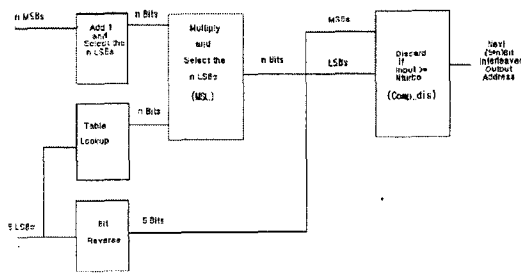


그림1. output address 생성과정 블록도

Figure 1. Turbo Interleaver Output Address Calculation

다음의 표 1은 interleaver Block size N_{turbo} 에 따른 Turbo interleaver parameter n 을 나타낸다. [4]

Turbo Interleaver Block Size N_{turbo}	Turbo Interleaver Parameter n
378	4
570	5
762	5
1,146	6
1,530	6
2,298	7
3,066	7
4,602	8
6,138	8
9,210	9
12,282	9
20,730	10

표 1. 터보 인터리버 파라미터 n

Table 1. Turbo Interleaver Parameter n

Multi media에서 필요한 4Mbps의 속도를 내기 위해 Interleaver의 속도 향상을 위하여 곱셈기는 속도가 빠른 Booth Multiplier를 사용하여 Interleaver의 속도를 향상시켰다. 또한 설계된 Interleaver의 Minimum distance는 아래의 그림과 같다. 그림에서 보이는 것과 같이 설계된 interleaver를 통한 code의 최소 거리는 블록 크기 378에서 30에 근접하는 거리를 가진다.[2]-[3]

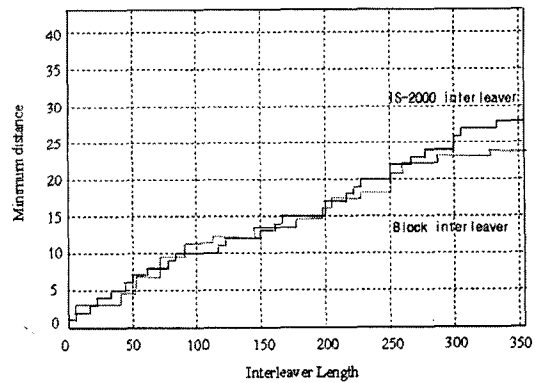


그림2. 인터리버크기에 따른 최소거리

Figure 2. Minimum Distance as a function of the Interleaver Length

이는 일반적으로 row순서로 쓰고 column순서로 읽어 내는 block interleaver에 비하여 더 나은 성능을 보이

고 있다. 그림 2에서 볼 수 있듯이 우리가 목표로 한 block size 378에서 IS-2000에서 제안된 interleaver가 일반적인 Block interleaver보다 minimum distance가 우수하다는 것을 알 수 있다.

Look-up table은 32개의 어레이구조를 가지는 ROM을 이용하여 구현하였다. 그리고, MSL 계산 부에는 속도가 빠른 Booth Multiplier를 이용하여 속도향상을 했다. 입력신호는 외부 RAM을 이용하여 저장해두었다가 생성된 output address sequence에 따라서 interleaving 된 출력신호를 만들어낸다. 생성된 회로에는 외부 RAM과의 인터페이스를 고려하였으며, 설계된 회로에는 RAM회로가 생략되어있다. 카운터는 9비트 카운터를 사용하며, 카운터 회로는 output address sequence를 완전히 생성할 때까지 1씩 증가를 시킨다. 다음은 생성된 회로의 simulation 결과이다. 시뮬레이션은 Aldec 사의 Active-hdl 시뮬레이터를 사용하였다.

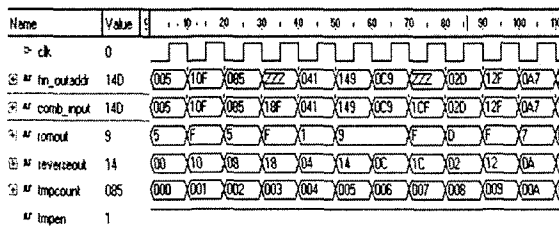


그림 3. 시뮬레이션 결과 파형
Figure 3. Waveform of Simulation

시뮬레이션 결과에서 볼 수있듯이 카운터를 통하여 생성되어지는 최종 output address sequence는 ROM에 저장되어있는 Lookup Table의 값과 MSL 결과 값을 이용한다. 여기서 한가지 중요한 점은 block length 378을 넘는 address는 생략이 된다는 것이다. 본 회로에서는 High Impedence를 사용하여, 생성되어지는 신호 열과 차별을 두었다. comb_input는 비교기에 입력되는 신호로써 378과 비교하여 최종 output address에서 High Impedence로 대체되었다. 물론 18F신호동안 전 단계의 신호를 출력할 수도 있다. 이때 counter는 9 bits all zero에서 "111111110"까지 "1"씩 증가시키면서 output address를 생성해낸다.

인터리버는 Turbo code의 성능에 막대한 영향을 끼친다. minimum distance를 볼 때, interleaver의 크기가 커질수록 좋은 성능을 보인다.. 그러나 interleaver의 크기가 커질수록 회로는 복잡해지며, 따라서 성능과 복잡도 사이에서의 trade-off는 불가피하다 할 수 있

겠다. 설계된 인터리버의 어드레스 생성회로도 는 그림 4와 같다.

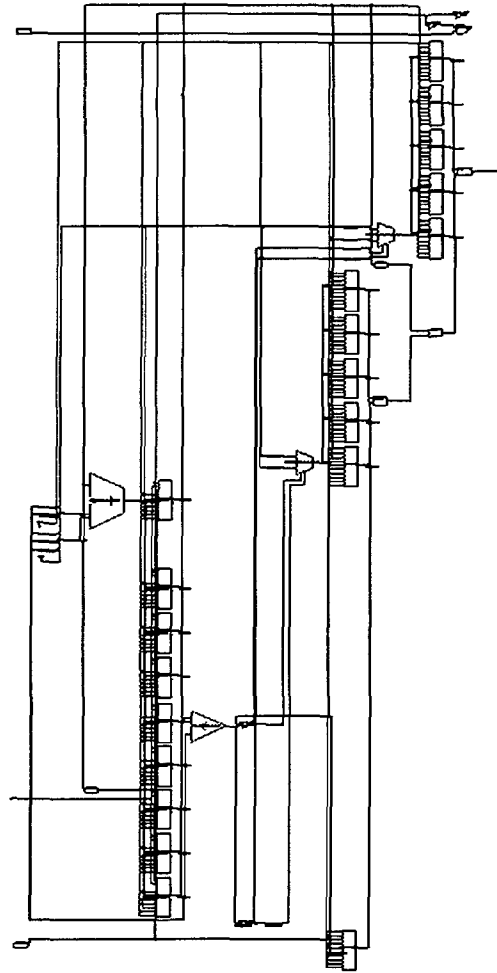


그림 4. output Address generator의 설계회로
Figure 4. Circuit of Output Address Generator

III. 결론 및 결과고찰

현재 연구되어진 인터리버를 보면 성능과 복잡도 사이의 trade-off를 감안하여 많은 연구가 이루어졌다. 본 논문의 시뮬레이션 결과에서 볼 수있듯이 block length 378보다 큰 address가 생성된다. 이는 단지 counter의 output만을 이용하여 output address sequence를 생성하게 되므로 피할 수 없는 결과물이다. 이들 신호 열은 비교기를 통하여 378보다 큰 신호는

address sequence에서 제외된다. 이는 향후 알고리즘의 개선을 통하여 이루어 질 수있을 것이다. 또한 378보다 큰 block size를 사용하게되면 MSL부의 곱셈연산에서 많은 delay가 발생하므로 이의 개선을 위해서는 제안된 Booth 곱셈기가 좋은 대안이 될 것이다.

좀더 나아가 본 논문에서 설계된 Interleaver는 Turbo code의 인터리버로써 사용 가능하며, IS-2000의 시스템 사양을 따르고 있다. 또한 시뮬레이션을 통해서 같은 길이의 Block Interleaver보다 우수한 성능을 확인했다. 또한 Booth Multiplier를 사용, Interleaver의 속도를 향상시켰다. 현재 Interleaver 에 관련된 연구는 집중적이고 폭넓게 행하여지고 있다. IS-2000에서 제안된 알고리즘도 차츰 변해가고 있는 실정이다. VHDL 코딩을 통해서 생성된 회로는 이 변화되는 알고리즘에 효과적으로 대처할 수있으리라 생각한다. 알고리즘의 변화가 혁신적인 변화를 가져오지 않는 한 제안된 알고리즘의 회로는 VHDL로 형성된 회로에서 크게 변화하지 않을 것이다.

참고문헌

- [1] Frde Daneshgaran and Marina Mondin, "Design of Interleavers for Turbo Codes:Iterative Interleaver Growth Algorithms of Polynomial Complexity", IEEE Trans. Information Theory,Vol 45, pp. 1845 -1859 , Sept 1999
- [2] Sergio Benedetto and Guido Montorsi, "Design of Parallel Concatenated Convolutional Codes", IEEE Trans.Communications,Vol 44, pp. 591-600, May 1996
- [3] Guido Masera, Gianluca Piccinini, Massimo Ruo Roch, and Maurizio Zamboni, "Vlsi Architectures for Turbo Codes", IEEE Trans. VLSI Systems, Vol.7, pp 369-378, Sept 1999
- [4] "Physical Layer Standard for cdma2000 Spread Spectrum Systems", IS-2000 표준안, pp. 2-68 ~ 2-71