

# OFDM 기반 WAVE 시스템의 시간동기 하드웨어 설계

정회원 현 트롱안\*, 종신회원 김진상\*, 정회원 조원경\*

## Hardware Design for Timing Synchronization of OFDM-Based WAVE Systems

Tronganh Huynh\* *Regular Member*, Jinsang Kim\* *Lifelong Member*,  
Won-Kyung Cho\* *Regular Member*

### 요 약

5.9 GHz WAVE(Wireless Access for the Vehicular Environment)는 노변-차량, 차량-차량 통신을 통하여 공공 안전과 개인통신을 지원하기 위한 중단거리 무선통신 방식이다. WAVE 물리계층의 핵심기술은 시간동기오류에 민감한 OFDM 방식이며 통신링크상의 지연을 최소화하여 고속의 차량통신 환경을 제공하는 것이 매우 중요하다. 본 논문에서는 오류에 강인하고 복잡도가 낮고 지연시간이 적은 WAVE 시스템 응용을 위한 시간동기 알고리즘과 하드웨어 구조를 제안한다. 제안된 알고리즘은 기존의 알고리즘에 비교하여 연산의 복잡도와 지연시간이 감소되며 하드웨어 구조는 파이프라인 구조와 고속 동작에 영향을 줄 수 있는 RAM이 필요하지 않다는 장점이 있다. Matlab과 FPGA를 이용한 하드웨어 구현을 통한 동기화 오차율(SER) 실험결과, 제안된 알고리즘이 고속 이동환경에 대해 강인하고 효율적이라는 확인하였다.

**Key Words** : 시간동기, OFDM, FPGA, ADSRC, 시간오프셋

### ABSTRACT

WAVE is a short-to-medium range communication standard that supports both public safety and private operations in roadside-to-vehicle and vehicle-to-vehicle communication environments. The core technology of physical layer in WAVE is orthogonal frequency division multiplexing (OFDM), which is sensitive to timing synchronization error. Besides, minimizing the latency in communication link is an essential characteristic of WAVE system. In this paper, a robust, low-complexity and small-latency timing synchronization algorithm suitable for WAVE system and its efficient hardware architecture are proposed. The comparison between proposed algorithm and other algorithms in terms of computational complexity and latency has shown the advantage of the proposed algorithm. The proposed architecture does not require RAM (Random Access Memory) which can affect the pipelining ability and high speed operation of the hardware implementation. Synchronization error rate (SER) evaluation using both Matlab and FPGA implementation shows that the proposed algorithm exhibits a good performance over the existing algorithms.

### I. 서 론

WAVE 표준은 IEEE 802.11.p의 확장 버전으로

ITS (Intelligent Transportation System) 시스템에 대부분 응용되어 발전되어 왔으며, 노변장치와 차량 또는 차량과 차량 사이의 통신을 통하여 공공의 안

\* 본 연구는 정부의 재원으로 한국과학기술진흥재단의 지원을 받아 수행된 연구임(과제번호:KRF-2006-521-D00337)

\* 경희대학교 전자전파공학과 (jskim27@khu.ac.kr)

논문번호: KICS2007-04-184, 접수일자: 2007년 4월 19일, 최종논문접수일자: 2008년 3월 29일

전 및 개인 통신을 지원하는 중단거리 무선 데이터 통신 서비스다. WAVE는 통신 링크 지연의 최소화 및 좁은 통신 구역간에 원활한 접속 및 해지가 중요할 때, 고속 데이터 전송을 지원할 수 있다는 특징을 가지고 있다 [1].

WAVE의 물리 계층은 직교 주파수분할 다중접속(OFDM) 기술에 기반을 두고 있다. OFDM은 동기화 오차에 대해 큰 민감성을 보이고 있는데[2-3], 이에 대해 WAVE 시스템에서는 차량 속도가 120 mph까지 통신이 이루어지므로 도플러 효과에 의한 영향이 매우 크며, 다중 경로로 인하여 발생하는 왜곡과 함께 WAVE 시스템의 시간 동기화의 성능을 감소시키는 결과를 가져온다.

이 논문에서는 WAVE 시스템의 시간 동기화 알고리즘과 하드웨어 구조를 제안한다. 제안된 알고리즘은 연산의 복잡성은 줄이고 오류에 대한 강인성을 갖는다는 특징이 있다. 제안된 알고리즘은 상호 상관(cross-correlation)에 기반되었으며 이를 이용하여 짧은 훈련 심볼의 시작점과, 첫 번째 긴 훈련 심볼의 보호구간을 찾는다. 제안된 알고리즘은 Xilinx Vertex II XC2V2000 FPGA으로 구현되어 졌으며, Matlab과 FGPA를 이용한 post-layout 시뮬레이션을 통해, 동기화 오차율(SER)이 측정하였다. 제안된 시간 동기화 알고리즘 및 하드웨어 구조는 고속 이동 환경 중에서도 우수한 성능을 나타내고 있음을 확인하였다.

## II. WAVE 시스템 모델

각 WAVE 프레임은 프리앰블, 신호, 데이터의 3 개 구간으로 구성된다. 프리앰블 구간은 주로 시간 동기화, 주파수 오프셋(offset) 추정, 채널 상태 추정 등을 위해 사용 되어지는 구간이다. 다음의 신호 구간은 WAVE의 데이터 전송율과 데이터 길이 정보 등을 전송하는 구간이며, 데이터 구간은 데이터를 실어서 전송하는 구간이다.

그림 1은 WAVE 모델의 물리계층의 블록 다이어그램이다. 프리앰블, 신호, 데이터 구간의 생성은 송신기에서 이루어진다. 대부분의 기능은 일반적인 OFDM 기반 시스템의 구조와 거의 비슷하다.

수신기의 시간 및 주파수 동기 블록에서 수신 신호의 타이밍 오프셋(offset)과 주파수 오프셋 조정이 이루어진다. Guard Removal(GR) 블록에서는 보호 구간의 데이터들을 제거하게 되며, FFT 블록을 통하여 OFDM 심볼은 병렬 데이터 심볼 형태로 변환하

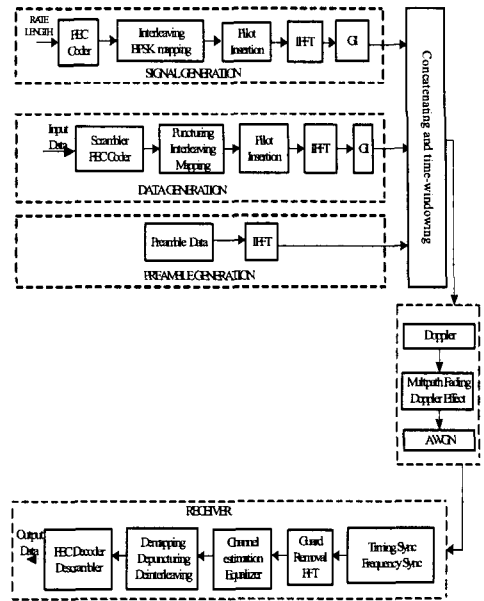


그림 1. WAVE 모델의 물리계층 블록도

게 된다. 채널추정과 등화기 블록은 두 개의 긴 훈련심볼과 파일럿 심볼의 상태 정보들을 기초로 등화과정을 수행함으로써, 시스템의 BER 성능을 향상시키게 된다. 등화 과정을 거친 데이터는 복조과정 및 천공과 인터리빙, 스크램블링의 역과정과 디코딩 과정을 거친다.

## III. WAVE 시간동기 알고리즘

OFDM 기반 시스템의 다양한 시간동기 알고리즘은 [5]-[13]에 소개되었다. [5]에서는 심볼의 타이밍을 찾기 위해 수신된 신호의 에너지와 상관 함수 사이의 비를 나타내는 타이밍 메트릭 (metric) 값이 제안되었다. [6]에서는 ML (Maximum Likelihood) 시간동기 방법이 제안되었다. 이 방법은 정확성이 SNR 추정 기법에 의존되어 있는데, 고속 이동과 다중경로에 의한 페이딩 환경에서는 SNR 추정이 종종 정확하지 않다는 문제가 있다. [7]에서는 대략적(coarse)추정과 상세(fine)추정을 포함한 새로운 타이밍 동기화 방법이 제안되었다.

본 논문에서는 WAVE 시스템의 시간동기 알고리즘을 제안한다. 제안된 알고리즘은 채널 환경에 강하면서 복잡성이 감소되도록 한다. 제안된 시간동기 알고리즘은 프리앰블의 시작점을 찾는 것으로부터 출발한다. 그림 2는 7번째 심볼의 7번째 샘플로 초기 설정하여 시작하는 예를 보여준다.

제안된 알고리즘은 다음과 같이 2 단계로 구성된다.

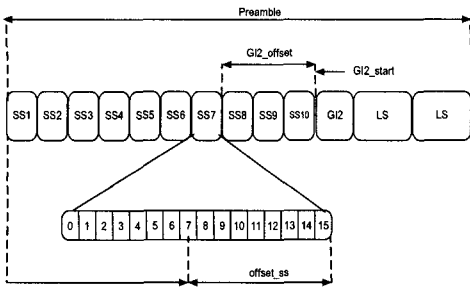


그림 2. WAVE 프레임의 프리앰블 시작점의 예.

1단계: 짧은 훈련심볼의 시작점을 탐색

수신신호  $r(n)$ 과 짧은 훈련신호(short training signal)  $rs(k)$ 의 상호상관 함수  $R_1(n)$ 은 짧은 훈련 심볼의 시작점을 찾는 데 사용된다. 여기서  $rs(k)$ 는 송신기와 수신기가 공통으로 알고 있는 값이다.

$$R_1(n) = \left| \sum_{k=0}^{15} r(n+k)r_k^*(k) \right|, 0 \leq n \leq 15 \quad (1)$$

$$\text{offset-ss} = \arg \max_{0 \leq n \leq 15} \{R_1(n)\}, \quad (2)$$

Offset-ss 값은 짧은 훈련심볼의 타이밍 오프셋(offset)을 나타낸다. 여기서  $R_1(n)$  값을 이용하여 값을 시작점의 값을 찾게 되는데, 하나의 짧은 훈련 심볼은 16개 샘플 주기를 갖고 있기 때문에 16개의 연속적인 샘플 연산을 수행하여  $R_1(n)$  값을 구하게 된다.

2단계: 긴 훈련심볼의 시작점을 탐색

$$R_2(m) = \frac{\left| \sum_{k=0}^{15} r(k+m)r_k^*(k) \right|^2}{\left( \sum_{k=0}^{15} |r(k+m)|^2 \right)}, m = 0, 16, 32, 48, \dots \quad (3)$$

$$GI2 - \text{offset} = \arg \max_m \{R_2(m) < \theta\}. \quad (4)$$

1단계의 과정을 마치면, 정확한 짧은 훈련심볼의 시작점을 찾는다. 그러나 이 과정에서는 프레임 상에 짧은 훈련심볼이 얼마나 남아있는지 모르기 때문에, 2단계에서 첫 번째 긴 훈련심볼의 보호구간(GI2) 시작점을 찾게 된다.  $R_2(m)$ 은  $r(n)$ 과  $rs(n)$ 의 상호상관(cross-correlation) 값을 계산한다. 여기서  $R_2(m)$  값은  $m$  값을 연속적으로 증가하지 않는 대신 16씩 단계적으로 증가시켜 계산하며, 짧은 심볼 내에서는 평평한 모양을 갖는다. 수신 신호가 긴 심볼의 보호구간(guard interval)에 있다면  $R_2(m)$  값은

뚜렷하게 감소할 것이다. 문턱값(threshold)  $\theta$ 은 수신신호의 보호구간의 시작점을 찾는 데 사용된다

### IV. WAVE 시간동기 하드웨어 구조

제안된 알고리즘의 하드웨어 설계 구조는 그림 3과 같다. 수신 신호의 실수값과 허수값, 각각은 4비트의 정수 비트와 12비트의 소수 비트로 구성된 16비트로 이루어져 있다. 시스템의 클럭(cik) 주파수는 10 MHz이며 detect-signal 신호는 새로운 WAVE 프레임이 탐지되었음을 표시하는데 사용된다.

FINE COMPARATOR 블록은 CROSS CORR 블록으로부터 타이밍 메트릭(metric) 값을 받아들여 16개의 연속적인 값 중에서 가장 최고점이 되는 것을 찾는다. 최고점을 찾으면 FINE COMPARATOR 블록은 ss-detect과 pointer-ss [4:0] 신호로써 CONTROL 블록에 알려주게 된다. 여기서 pointer-ss [4:0] 정보는 짧은 심볼상에서 정확한 위치를 나타낸다. CONTROL 블록은 WAVE BUFFER에서 사용되어지는 addr-buff [4:0] 신호를 생성하여 step 2의 CROSS CORR 블록을 위해 정확한 데이터를 전달해주게 된다.

CROSS CORR 블록은 1단계와 2단계의 타이밍 메트릭(metric) 값을 계산하는데 사용된다. 이렇게 하드웨어 자원을 공유하는 기술을 사용함으로써 하드웨어의 면적과 전력 소모를 줄일 수 있다.

1단계 연산이 수행되어진 다음, CONTROL 블록은 COARSE COMPARATOR 블록을 시작하기 위해 coarse-start 신호를 보내게 된다. COARSE COMPARATOR 블록은 2단계에서 보호 구간(guard interval) 2의 오프셋(offset)을 찾는 데 사용된다. FINE COMPARATOR 블록은 2단계에서 COARSE

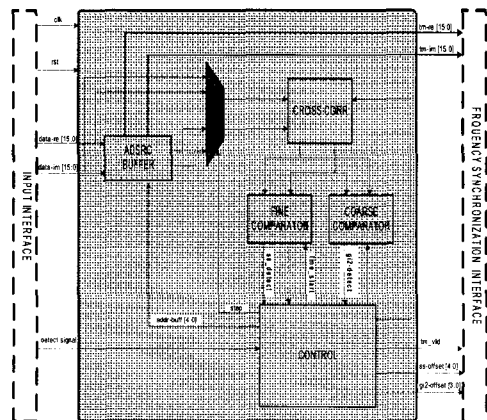


그림 3. 시간동기 하드웨어 구조

표 1. FPGA 합성결과 및 성능

|                         |                         |
|-------------------------|-------------------------|
| Number of SLICES        | 2843 out of 19200 (14%) |
| Number of external IOBs | 77 out of 404 (19%)     |
| Number of GCLKs         | 1 out of 4 (25%)        |
| Average Power           | 465 (mW)                |
| Estimated Frequency     | 128 (MHz)               |

COMPARATOR 블록은 1단계에서 클럭 게이팅(gating)을 사용하여 정지상태로 머무르게 되는데 이는 하드웨어의 전력 소모를 감소시킨다.

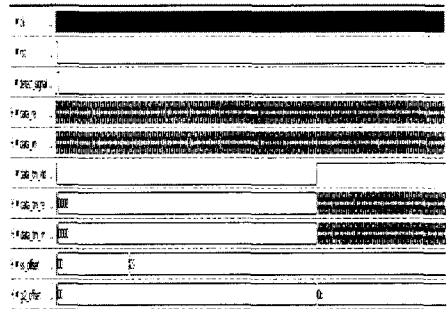
제안된 알고리즘은 VHDL로 코딩하여 SynplifyPro 8.0, 그리고 Xilinx ISE 6.0의 placed and routed 기능과 Virtex-II XC2V2000 FPGA을 사용하여 합성하였다. Active-HDL 6.2 tool은 HDL 시뮬레이션 하는데 사용되었으며 Post-Layout 시뮬레이션은 ModelSim SE 6.0 a에서 수행되었다.

표 1은 최종 post-layout 결과를 나타낸 것이다. 설계된 WAVE 시간동기 부분은 2843 슬라이스를 차지하며 전체 FPGA 면적의 14%를 차지한다. Modelsim으로부터 VCD (Value Change Dump)파일을 생성하여 그 정보를 이용하여 Xilinx의 XPower 도구를 이용하여 전력소비를 추정하였는데, 시간동기에 대한 평균전력 소모는 465 mW를 소비하는 것으로 추정되어졌다. 실제로 장치에 요구되는 클럭 주파수값은 10MHz인데 반해, 주어진 하드웨어가 지원할 수 있는 최대 주파수를 추정한 결과 128MHz까지 지원 가능함을 알 수 있다.

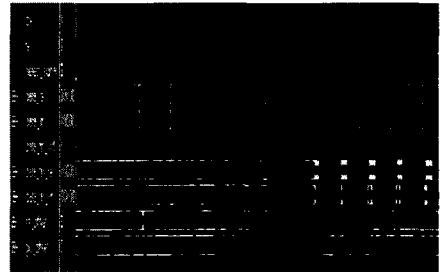
V. 시뮬레이션 결과 및 성능분석

그림 4가)와 그림 4나)는 HDL 기능검증과 FPGA 합성후 Post-layout 시뮬레이션 결과파형을 보여주며 기능이 동일함을 검증하였다.

그림 5와 그림 6은 R1(n) 값과 R2(m)값에 대한 곡선을 EbNo가 10 dB이고 이동속도가 시속 120 마일인 채널 상태에 대한 시뮬레이션 결과를 보여 준다. R1(n)의 곡선은 n=8일때 최대값을 가지며, 이것은 짧은(short) 심볼의 정확한 시점은 수신신호 상에서 8번째 샘플에 위치함을 나타낸다. R2(m)의 곡선은 m값이 1부터 8까지인 샘플상에서 거의 평평한 형태를 가지게 되는 반면에 m=9일 때 급격히 감소하게 되는 것을 알 수 있다. 이것은 GI2의 시작점을 의미한다. 이와 같이 Post-Layout 시뮬레이



가) HDL 시뮬레이션 결과



나) FPGA 합성 결과

그림 4. 시뮬레이션 결과

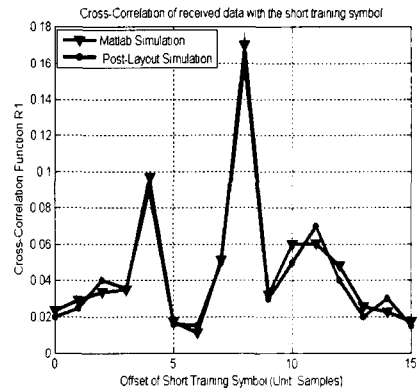


그림 5. R1(n)의 시뮬레이션 결과

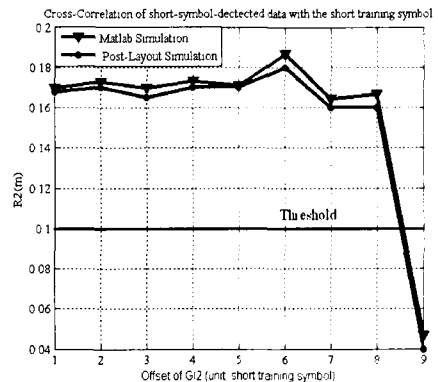


그림 6. R2(m)의 시뮬레이션 결과

선 결과는 R2(m)값과 R1(n)값이 Matlab 시뮬레이션 결과와 같다는 사실을 확인할 수 있다.

SER(Synchronization Error Rate)값을 제안된 하드웨어 구조의 성능평가에 사용하였다. 본 논문에서는 JTC(Joint Technical Committee)에 의해 제공된 실외/실내환경에 따른 채널 파라미터를 이용하여 실험하였다<sup>4)</sup>. 표 2는 이러한 채널 상태변수이다. 채널 A의 rms 지연확산은 채널 B의 지연확산보다 작으므로 채널 A의 coherence 대역폭은 채널 B 보다 크다. 그러므로 채널 A의 주파수 선택성(frequency selectivity)의 영향이 채널 B보다 덜 심각하다.

그림 7은 채널 A의 상황에서 100 km/h의 속도로 이동할 때의 SER 곡선을 보여준다. 그림 7과 같이 post-layout 시뮬레이션과 Matlab의 시뮬레이션 결과는 거의 근사함을 알 수 있다. 채널 B의 상황에서 지연 확산(delay spread)이 4us로 WAVE의 보호구간(guard interval) 1.6us 보다 크에도 불구하고, 제안된 하드웨어는 그림 8과 같이 강인함을 알 수 있다.

표 2. 채널 파라미터

| Tap | Channel A<br>(rms Delay Spread = 0.4μs) |                    | Channel B<br>(rms Delay Spread = 12μs) |                    |
|-----|---|--------------------|--|--------------------|
|     | Relative Delay (ns)                     | Average Power (db) | Relative Delay (ns)                    | Average Power (db) |
| 1   | 0                                       | -1.6               | 0                                      | -2.5               |
| 2   | 100                                     | -5.1               | 300                                    | 0                  |
| 3   | 200                                     | 0                  | 8900                                   | -12.8              |
| 4   | 500                                     | -7.6               | 12900                                  | -10                |
| 5   | 1200                                    | -6.9               | 17100                                  | -25.2              |
| 6   | 1600                                    | -27.6              | 20000                                  | -16                |

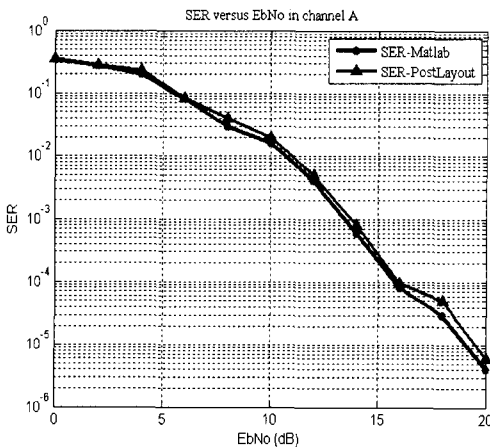


그림 7. 채널 A상의 Matlab과 Post-layout의 SER 비교

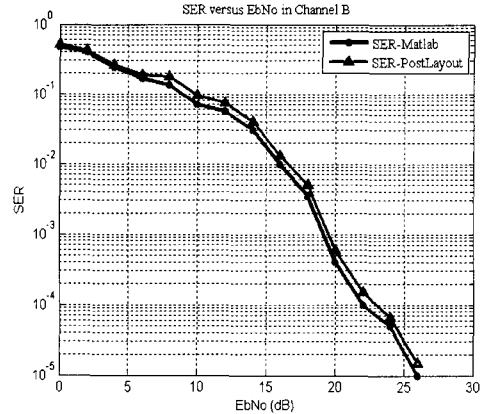


그림 8. 채널 B상의 Matlab과 Post-layout의 SER 비교

그림 9는 시속 120Km/h의 속도로 이동할 때 post-layout 시뮬레이션 상에서 channel A와 channel B를 비교한 SER 그래프를 나타낸다.

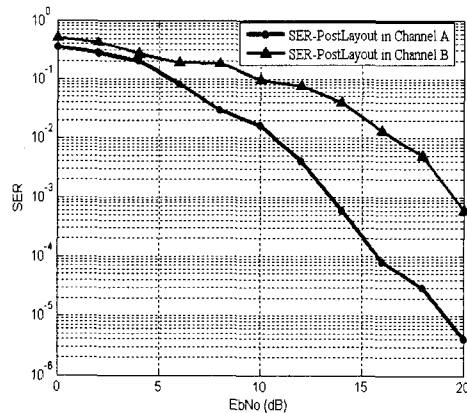


그림 9. 채널 A와 B상의 post-layout의 SER 성능비교

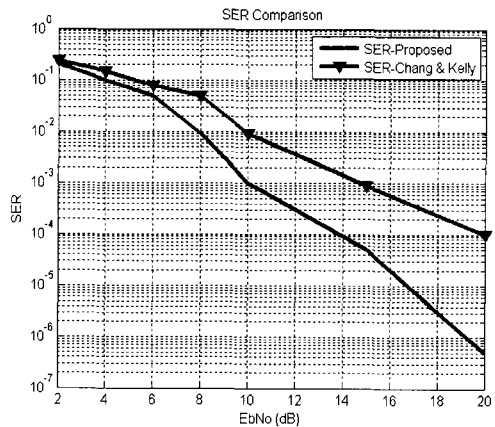


그림 10. 제안된 방법과 기존의 방법과의 비교

그림 10은 채널 A 상태에서 120Km/H로 차량이 이동할 때 제안된 방법과 기존의 방법[6]과의 SER 성능비교를 수행한 결과이다. 실험결과와 같이 제안된 알고리즘은 [6]의 방법보다 성능이 우수함을 알 수 있고 특히 높은 EbNo값에서는 성능차이가 많다. [6]의 SER값이 높은 이유는 [6]의 방법의 정확도는 수신신호 에너지 값에 따라 다르며 일반적으로 높은 이동성과 다중 페이딩 환경에서는 신호에너지 값의 변동이 심하기 때문이다.

표 3은 제안된 알고리즘과 기존의 알고리즘의 복잡도를 비교한 것이다. 표 3과 같이 제안된 알고리즘은 기존의 알고리즘 보다 연산의 복잡도가 제일 높은 곱셈연산 횟수가 아주 작음을 알 수 있다. 또한 제안된 알고리즘의 연산지연시간(latency)이 32 클럭 사이클이며, 기존의 알고리즘 [6]이 최소 48 클럭 사이클로써 제안된 알고리즘의 연산 지연시간이 적다.

표 3. 연산의 복잡도 비교

|          | 곱셈   | 덧셈   | 나눗셈 | 비퍼 길이 |
|----------|------|------|-----|-------|
| [6]      | 193  | 112  | 1   | 80    |
| [7]      | ≥192 | ≥125 | 1   | 16N   |
| 제안된 알고리즘 | 99   | 94   | 1   | 16    |

### V. 결 론

본 논문에서는 오류에 강인하고 복잡도가 낮고 지연시간이 적은 WAVE 시스템 응용을 위한 시간 동기 알고리즘과 하드웨어 구조를 제안하였다. 제안된 하드웨어 구조는 파이프라인 구조와 고속 동작에 영향을 줄 수 있는 RAM이 필요하지 않다는 장점이 있으며 기존의 알고리즘보다 연산의 복잡도가 매우 낮다는 장점을 가진다. 실험결과, 제안된 알고리즘이 고속 이동 환경에 대해 강인하고 효율적이라는 확인하였다.

### 참 고 문 헌

[1] IEEE P802.11p/D1.0, Wireless LAN Medium Access Control (MAC) and Physical PLayer (PHY) Specification - Amendment3: Wireless Access in Vehicular Environments (WAVE).  
 [2] L. Hanzo, M. Munster, B.J Choi and T. Keller, OFDM and MC-CDMA for Broad band Multi-User Communications, WLAN sand Broadcasting, Wiley, 2003.

[3] Shinsuke Hara, Ramjee Prasad, Muticarrier Technologies for 4G Mobile Communications, Artech, 2003.  
 [4] Joint Technical Committee on Wireless Access, Technical Report on RF channel characterization and System Deployment Modeling, Paper No. JTC(AIR) 1994.09. 23-065R6, Sep. 23, 1994.  
 [5] T. M. Schmidl and D. C. Cox, "Robust frequency and timing synchronization for OFDM," IEEE Transaction on Communications, , vol. 45, pp. 1613-1621, Dec. 1997.  
 [6] J.J. van de Beek, M. Sandell, "ML Estimation of Time and Frequency Offset in OFDM Systems," IEEE Transactions on signal processing, vol.45, pp.1800-1805, July 1997.  
 [7] S.Chang and B. Kelley, "Time synchronization for OFDM-based WLAN systems," IEEE Electronics Letters, vol.39, pp. 1024-1026, June 2003.

#### 현 트롱안 (Tronganh Huynh)

정회원



2004년 5월 호치민 공대 전기및 전자공학과 학사  
 2007년 2월 경희대학교 전자전파공학과 석사  
 2007년 3월 경희대학교 전자전파공학과 박사과정 재학중  
 <관심분야> 이동통신 SoC 설계

#### 김 진 상 (Jinsang Kim)

중심회원



2000년 12월 미국 콜로라도 주립대 전자공학박사  
 1990년 2월~2001년 8월 KT 연구소  
 2001년~현재 경희대학교 전자정보학부 부교수  
 <관심분야> 영상처리 및 이동통신용 SoC 설계

#### 조 원 경 (Won-Kyung Cho)

정회원



1986년 8월 한양대학교 전자공학과 공학박사  
 1980년~현재 경희대학교 전자정보학부 정교수  
 <관심분야> 컴퓨터시스템 구조, VLSI 설계