

논문 2010-47TC-10-7

# Pipeline CORDIC을 이용한 저전력 주파수 옵셋 동기화기 설계 및 구현

(Low-Power Frequency Offset Synchronization Block Design and  
Implementation using Pipeline CORDIC)

하 준 형\*, 정 요 성\*\*, 조 용 훈\*\*\*, 장 영 범\*\*\*\*

(Jun Hyung Ha, Yo Sung Jung, Yong Hoon Cho, and Young Beom Jang)

## 요 약

이 논문에서는 pipeline CORDIC(COordinate Rotation DIgital Computer)을 이용한 저전력 주파수 옵셋 동기화기 구조를 제안하였다. 주파수 옵셋 동기화기의 핵심 블록은 주파수 옵셋 추정부와 보상부이다. 제안된 주파수 옵셋 추정부에서는 sequential CORDIC을 사용하여 구현면적을 감소시켰으며 한번에 2 단계씩 CORDIC을 수행하는 방식을 사용하여 연산 속도를 높였다. 또한 제안된 주파수 옵셋 보상부에서는 pipeline CORDIC을 사용하여 구현면적을 줄임과 동시에 계산 속도를 향상시킬 수 있었다. MatLab을 사용하여 제안 구조가 주파수 옵셋을 추정 및 보상하는 function을 검증하였다. 제안 구조에 대하여 Verilog-HDL로 코딩하고 Synopsys tool을 사용하여 합성하여 구현면적을 실험하였다.

## Abstract

In this paper, a low-power frequency offset synchronization structure using CORDIC algorithm is proposed. Main blocks of frequency offset synchronization are estimation and compensation block. In the proposed frequency offset estimation block, implementation area is reduced by using sequential CORDIC, and throughput is accelerated by using 2 step CORDIC. In the proposed frequency offset compensation block, pipeline CORDIC is utilized for area reduction and high speed processing. Through MatLab simulation, function for proposed structure is verified. Proposed frequency offset synchronization structure is implemented by Verilog-HDL coding and implementation area is estimated by Synopsys logic synthesis tool.

**Keywords :** OFDM, CORDIC, Pipeline, WLAN, Frequency offset synchronization

## I. 서 론

직교 주파수 분할 다중화(Orthogonal Frequency

Division Multiplexing, OFDM) 시스템은 기존 단일 반송파 통신 시스템과 비교하여 많은 장점을 가지고 있어서 다양한 유무선 데이터 전송 시스템 표준에 채택되어 사용되고 있다. 이와 같은 직교 주파수 분할 다중 방식은 다수의 부반송파를 사용하여 데이터를 블록(심볼)단위로 처리하므로 주파수 옵셋(Frequency offset)에 매우 민감하다. 주파수 옵셋은 송신기와 수신기의 발진기(oscillator) 동작 불일치 또는 도플러(Doppler) 효과 등에 의해서 발생하게 되며, 이는 각 부반송파간의 직교성에 영향을 주게 되어 OFDM 시스템 전체의 성능에 큰 영향을 미친다. 주파수 옵셋의 영향은 신호의 크기 감소와 위상을 회전시켜, 인접 반송파간의 간섭(Inter

\* 학생회원, 상명대학교 컴퓨터정보통신공학과  
(Graduate School, Sangmyung University)

\*\* 정회원, (주)다물멀티미디어  
(TAMUL Inc.)

\*\*\* 정회원, (주)코메스타  
(COMESTA Inc.)

\*\*\*\* 정회원, 상명대학교 공과대학 정보통신공학과  
(College of Engineering, Sangmyung University)

※ 본 연구는 교육과학기술부와 한국산업기술진흥원의 지역혁신인력양성사업으로 수행된 연구결과임.

접수일자: 2010년1월11일, 수정완료일: 2010년10월15일

Channel Interference)을 일으킨다. 그러므로 OFDM 통신 방식에서는 시스템 성능에 대한 신뢰성을 확보하기 위해서 주파수 오프셋에 대한 동기화가 반드시 필요하다. 따라서 OFDM 시스템의 수신단에서는 FFT를 수행하기 이전에 송수신기의 주파수 오프셋 동기화가 선행되어야만 하며 반도체 구현을 위하여 동기화기에 대한 고속처리 및 저 전력 구조 연구가 필요하다. 이를 위하여 CORDIC(COordinate Rotation Digital Computer) 알고리즘을 사용한 주파수 오프셋 동기화기가 제안되었다.<sup>[1~2]</sup> OFDM의 주파수 오프셋 동기화 기술은 주파수 오프셋 추정과 보상을 필요로 한다. 일반적으로 주파수 오프셋 동기화의 추정 블록은 벡터모드 CORDIC을 사용하고, 보상 블록은 회전모드 CORDIC을 사용하여 구현된다.<sup>[3]</sup>

OFDM의 주파수 오프셋 동기화기에서 추정부는 한번만 동작하고 보상부는 추정부에서 계산된 각도를 사용하여 반복적으로 동작하게 된다. 따라서 이 논문에서는 추정부의 연산속도를 높이고 구현면적을 줄이기 위해 2stage Sequential CORDIC 구조를 제안하고 보상부에서는 연산속도를 높이고 구현면적을 줄이기 위하여 삼각함수부와 complex multiplier를 통합한 pipeline CORDIC 구조를 제안한다.

II장에서는 프리앰블을 이용한 IEEE 802.11a WLAN 주파수 오프셋 동기화기에 대해 살펴보고 III장에서는 제안된 CORDIC 구조를 이용한 주파수 오프셋 동기화기의 구조 및 설계에 대해서 기술한다. IV장에서는 제안된 구조에 대한 시뮬레이션 및 합성결과를 통해 성능을 입증하고 V장에서 결론을 맺는다.

## II. OFDM용 주파수 오프셋 동기화기

### 1. 프리앰블의 구조

IEEE 802.11a OFDM 무선통신방식의 송신단에서 만들어진 패킷은 프리앰블(Preamble), 헤더(Header), 그리고 실제 전송하고자 하는 데이터인 페이로드(Payload)로 구성된다. IEEE 802.11a에서 사용되는 동기화를 위한 프리앰블의 구조는 그림 1과 같이 160개의 Short training symbol과 160개의 Long training symbol로 구성된다.<sup>[4]</sup> Short training symbol은  $s_1$ 부터  $s_{10}$ 까지 10개의 같은 심볼로 구성되며 각 심볼은 16개의 샘플로 구성된다. 수신단에서  $s_1$ 부터  $s_7$ 까지의 심볼은 신호 검출에 사용되며,  $s_8$ 부터  $s_{10}$ 까지는 대략

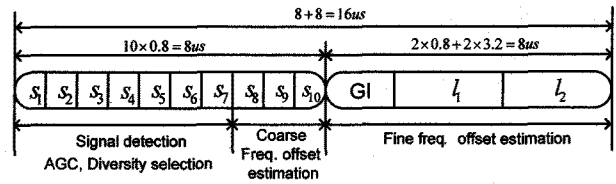


그림 1. 프리앰블의 구조  
Fig. 1. Structure of preamble.

(Coarse) 주파수 오프셋 동기화를 하는데 사용된다. Long training symbol은  $l_1$ 과  $l_2$ 의 두 개의 같은 심볼로 구성되며 수신단에서 채널 추정과 미세(Fine) 주파수 오프셋 동기화를 위해 사용된다.

### 2. 프리앰블을 이용한 주파수 오프셋 동기화

프리앰블을 이용한 수신단에서의 동기화는 신호 검출, 대략(Coarse) 주파수 오프셋 동기화, 미세(Fine) 주파수 오프셋 동기화의 순서로 이루어진다. 먼저 신호 검출은 이미 알고 있는 Short training symbol과 수신단의  $s_1$ 부터  $s_7$ 의 Cross-correlation 방법을 이용하여 peakdetection이 7개가 되면 start signal을 출력하게 된다. 그리고  $s_8$ 부터  $s_{10}$ 의 Short training symbol을 이용하여 오프셋 각도를 계산하고(추정) 128개의 Long training symbol에 오프셋 각도를 보상함으로써 대략(Coarse) 주파수 오프셋 동기화를 마치게 된다. 이러한 주파수 오프셋 동기화기의 블록도는 그림 2와 같다.

그림 2에서 먼저 Preamble중  $s_8$ 부터  $s_9$ 까지의 Short training symbol의 I와 Q가 Auto-Correlator로 입력되어 다음 식의 분자와 분모가 계산된다.

$$R(n) = \frac{\sum_{k=0}^{L-1} \text{Im}(r_{n,k} \times r_{n-1,k}^*)}{\sum_{k=0}^{L-1} \text{Re}(r_{n,k} \times r_{n-1,k}^*)} = \frac{y}{x} \quad (1)$$

식 (1)에서  $L$ 은 샘플의 수이다. 대략추정의 경우에는 16개의 샘플을 이용하고 미세추정의 경우에는 64개의 샘플을 이용한다.  $r_{n,k}$ 과  $r_{n-1,k}$ 은 각각  $n$ 번째 심볼과

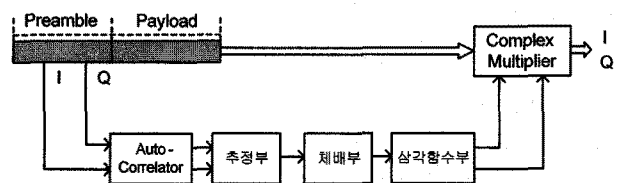


그림 2. 주파수 오프셋 동기화기의 블록도  
Fig. 2. Block diagram of frequency offset synchronizer.

그 이전 심볼  $n-1$ 번째의  $k$ 번째 샘플을 나타낸다.

그림 2의 추정부에서는 식 (1)에서 구한 Auto-Correlator 결과 값을 이용하여 다음 식과 같은 주파수 오프셋 각도  $\theta$ 를 추정한다.

$$\theta = \frac{1}{2\pi} \tan^{-1} \left( \frac{y}{x} \right) \quad (2)$$

Preamble의  $s_9$ 부터  $s_{10}$ 까지의 샘플을 사용해 식 (1)과 (2)의 연산을 반복하여 주파수 오프셋 각도  $\theta$ 를 다시 추정한다. 이와 같이 2번의 추정 연산의 결과를 평균하여 최종 오프셋 추정 값을 결정한다.

추정부에 구해진  $\theta$ 값은 삼각함수부에서  $\theta$ 값을 이용하여  $\cos\theta$ 와  $\sin\theta$ 값을 구하게 된다. 이  $\cos\theta$ 와  $\sin\theta$ 값이 최종적으로 추정된 Frequency offset 값이며 마지막으로 Complex Multiplier 블록에서 Payload의 I와 Q 값과 곱해져 1 샘플의 Long training symbol에 대한 주파수 오프셋이 보상된다. 나머지 127 샘플의 Long training symbol에 대한 보상은 체배부에서 체배된  $\theta$ 값을 사용하여 반복된다.<sup>[5]</sup>

### III. 제안된 주파수 오프셋 동기화기 구조

#### 1. 제안된 동기화기 블록도

지난 절에서 기술한 IEEE 802.11a 주파수 오프셋 동기화 알고리즘을 저전력으로 구현하기 위하여 CORDIC 알고리즘을 사용하는 것이 효과적이다.<sup>[6-8]</sup> CORDIC을 사용한 제안된 주파수 오프셋 동기화기의 전체 블록도는 그림 3과 같다.

그림 3의 Auto-Correlator 블록에서는 식 (1)의 벡터의  $x$  값과  $y$  값을 계산한다. CORDIC 추정부에서는 CORDIC을 사용하여 벡터의 각도  $\theta$ 를 계산한다. 벡터 모드 CORDIC을 사용하면 쉽게 벡터의 각도를 계산할 수 있다. 이 논문에서는 추정 성능에 영향을 미치는

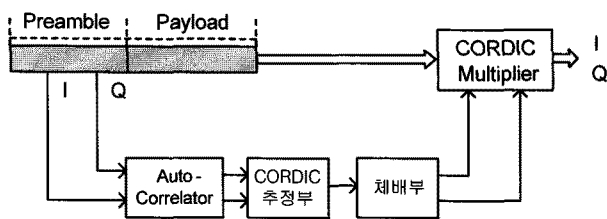


그림 3. 주파수 오프셋 동기화기의 블록도

Fig. 3. Block diagram of proposed frequency offset synchronizer using CORDIC.

CORDIC 연산 반복 횟수는 16회로 정하였다. 제안된 CORDIC 추정부는 구현면적을 줄이기 위해 Sequential 방식으로 CORDIC 벡터 모드를 구현하되 연산 속도를 높이기 위하여 한번에 2 stage씩 연산하는 2 stage sequential 방식을 제안한다. 즉 총 8번 반복 연산하여  $\theta$ 값을 계산하도록 설계한다. 그림 3의 체배부에서는 계산된  $\theta$ 를 체배한다. 예를 들면 대략 주파수 오프셋 추정 및 보상에서는 Long traing symbol 128개 샘플에 대하여 보상하므로  $\theta$ 부터  $128\theta$ 까지 체배하도록 설계한다.

그림 3의 CORDIC Multiplier에서는 체배부에서 주어진 각도  $\theta$ 를 사용하여 벡터를 회전시킨다. 회전모드 CORDIC을 사용하면 벡터 회전을 쉽게 연산할 수 있다. 대략 주파수 오프셋 보상에서는 벡터 회전 연산을 Long training symbol에 대하여 128번 반복하여야 한다. 따라서 이 연산에서는 고속처리가 관건이므로 Pipeline CORDIC 구조를 제안한다. CORDIC 정세도는 16회로 설정하였으므로 첫 번째 샘플에 대한 보상만 16 clock이 필요하고 두 번째 샘플에 대한 보상은 바로 다음 clock에 출력된다. 따라서 총 128개의 샘플에 대한 보상은 143 clock이 필요하도록 설계한다. 이 논문의 핵심 제안 블록인 CORDIC 추정부와 CORDIC Multiplier에 대한 자세한 내용은 다음 절에 기술한다.

#### 2. CORDIC 추정부 세부 설계

벡터모드 CORDIC 추정부의 설계는 식 (3)의 update 과정을 16번 반복 수행하게 된다.

$$\begin{aligned} x^{(i+1)} &= x^{(i)} \pm y^{(i)} 2^{-i} \\ y^{(i+1)} &= y^{(i)} \mp x^{(i)} 2^{-i} \\ \theta^{(i+1)} &= \theta^{(i)} \pm \tan^{-1} 2^{-i} \end{aligned} \quad (3)$$

식 (3)에서  $x^{(i)}, y^{(i)}$  값은  $i$  번째 반복에서의 복소수의 실수부 값, 허수부 값을 나타낸다. 벡터모드 CORDIC은 주어진 벡터를 0도로 회전시켜서 그때까지의 회전각을 계산한다. 따라서 식 (3)의  $y$  값이 양수인지 음수인지를 판단하여 벡터의 회전방향이 결정된다. 이와 같은 16번 연산의 CORDIC은 1 clock에 전체 연산을 끝낼 수 있다. 그러나 1 clock에 최종  $\theta$ 값을 구해야 하기 때문에 cell delay 지연으로 인해 critical timing path가 발생할 수 있다. 이와 같은 문제를 해결하기 위해 1 clock에 벡터를 한번만 회전시키는 sequential CORDIC 구조가 사용되고 있다. 이 논문에서는 critical timing path의 문

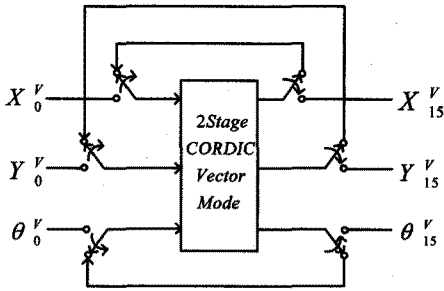


그림 4. 제안된 CORDIC 추정부 구조  
Fig. 4. Proposed CORDIC estimation block structure.

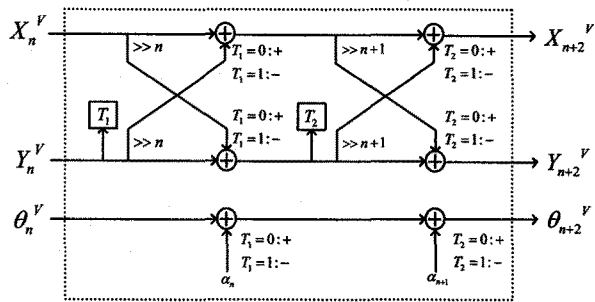


그림 5. 제안된 2 Stage CORDIC Vector Mode 구조  
Fig. 5. The 2 Stage CORDIC Vector Mode structure.

제를 해소하기 위해 sequential CORDIC 방식을 사용하며, 최종  $\theta$  값을 구해내는데 소요되는 clock을 줄이기 위해서 1 clock에 CORDIC 벡터모드 연산을 두 번 수행하는 그림 4와 같은 구조를 제안한다. 제안된 CORDIC 추정부는 최종  $\theta$  값을 구해내는데 총 8 clock이 소요된다. 그림 4의 제안 구조에서 입력 값  $X_0^V$ 와  $Y_0^V$ 은 각각 Auto-correlator 출력인 벡터의 실수 값과 허수 값이며  $\theta_0^V$ 는 0으로 초기화한다.

첫 번째 clock에서는  $X_0^V$ 와  $Y_0^V$ 의 입력값을 사용하여 벡터를 2번 회전시킨다. 이 값들은 피드백되어 다시 벡터모드의 입력으로 사용된다. 이와 같이 8 clock이 지나면 CORDIC 추정부의 연산이 끝나게 되어 그림 4의 스위치가 출력 쪽으로 스위칭 된다. 이때  $Y_{15}^V$ 는 0이 출력되며  $\theta_{15}^V$ 는 입력 벡터의 각도가 출력된다. 그림 4의 2 Stage CORDIC Vector Mode의 세부 구조는 그림 5와 같이 설계한다.

제안된 구조는 그림 5에서 보듯이 3개의 입력이 들어오도록 설계하였다. 입력 중에서  $X_n^V$ 와  $Y_n^V$ 는 현재의 벡터 값이며  $\theta_n^V$ 는 현재의 위상각이다. V 첨자를 붙인 것은 벡터 모드를 나타낸다.  $Y_n^V$ 의 MSB를 테스트하여 벡터의 회전방향을 제어하도록 설계하였다. 즉, MSB를

테스트한  $T_1$  제어신호가 0(양수)이면 벡터를 시계방향으로 회전하도록 제어한다. 즉 식 4의 X와  $\theta$ 는 +로 update되고 Y는 -로 update 된다. 이와 같이 1 stage가 끝나면 두 번째 stage도 같은 원리로 동작하도록 설계하였다.  $X_{n+2}^V$ 와  $Y_{n+2}^V$ 는 2번 회전된 벡터 값이며  $\theta_{n+2}^V$ 는 2번 회전된 후의 위상각이다.

3. CORDIC Multiplier 세부 설계

CORDIC Multiplier 블록에서는  $i, q$ 의 벡터 값과  $\theta$ 를 입력받아 다음의 벡터회전을 수행한다.

$$\begin{bmatrix} I \\ Q \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} i \\ q \end{bmatrix} \quad (4)$$

회전모드 CORDIC을 사용하면 식 (4)의 연산을 쉽게 수행할 수 있다. 대략 주파수 옵셋 보상에서는 식 (4)의 연산을 Long training symbol에 대하여 128번 반복하여야 한다. 따라서 이 연산에서는 고속처리가 관건이므로 그림 6과 같은 Pipeline CORDIC 구조를 제안한다.

그림 6에서 초기값  $X_R^0$ 와  $Y_R^0$ 는 각각 보상받아야 할  $i$ 와  $q$ 로 세팅하며  $\theta_R^0$ 은 0으로 세팅한다. 그림 6에서 보듯이 제안구조의 stage와 stage 사이에 3개의 레지스터를 사용하여 clock으로 stage를 구분하도록 설계하였다. CORDIC 정세도는 16회로 설정하였으므로 첫 번째

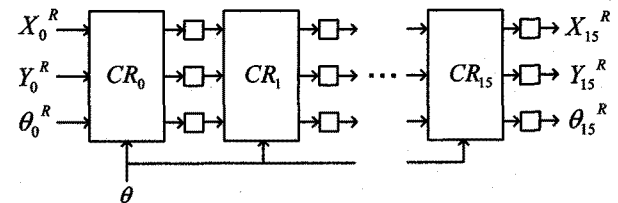


그림 6. 제안된 CORDIC Multiplier 구조  
Fig. 6. Proposed CORDIC Multiplier structure.

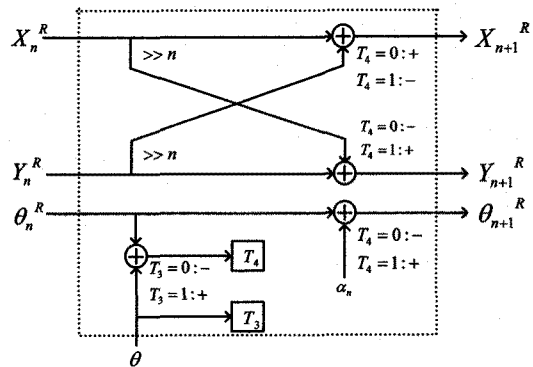


그림 7. 제안된 CORDIC Multiplier의 세부 구조  
Fig. 7. Detail structure of proposed CORDIC Multiplier.

샘플에 대한 보상만 16 clock이 필요하고 두 번째 샘플에 대한 보상은 바로 다음 clock에 출력된다. 따라서 대략 주파수 읍셋 보상의 총 128개의 샘플에 대한 보상은 143 clock이 걸리는 구조가 된다. 그림 6의 각각의 스테이지에 대한 세부 구조는 그림 7과 같이 설계하였다.

그림 7의 세부 구조에서도 식 (3)의 update 과정을 수행한다. 제안된 세부 구조는 그림 7에서 보듯이 4개의 입력이 들어오도록 설계하였다.  $X_n^R$ 과  $Y_n^R$ 은 현재의 벡터 값이며  $\theta_n^R$ 은 현재의 위상각이며 초기 값은  $0^\circ$ 로 한다. R 첨자는 회전 모드를 나타낸다.

벡터를 반시계방향으로 회전시키는  $\theta$ 가 양수인 경우를 제안 구조의 제어신호를 설명하기로 한다. 먼저 제어신호  $T_3$ 는 다음과 같이 설계한다. 즉  $\theta$ 의 MSB를 테스트한  $T_3$  제어신호가 양수(0)이므로  $\theta$ 에 연결된 모든 stage의 16개 감/가산기는 모두 감산기(subtractor)로 동작하도록 제어한다. 제어신호  $T_4$ 는 stage 마다 다르게 동작한다. 즉 현재의 위상각인  $\theta_n^R$ 에서  $\theta$ 값을 빼주어 MSB를 테스트한 제어신호  $T_4$ 를 만든다. 제어신호  $T_4$ 가 음수(1)이면 벡터를 반시계방향으로 회전하도록 하며 양수(0)이면 시계방향으로 회전하도록 제어한다.

#### IV. 구현 및 고찰

##### 1. Matlab Simulation

제안 구조를 검증하기 위하여 먼저 Matlab으로 Short training symbol과 Long training symbol의 Preamble 신호를 만들어 실험을 진행하였고 Verilog-HDL로 코딩하여 출력 값을 테스트 벡터와 비교하였다. 완성된 RTL 코드는 Synopsys Design Compiler로 합성하여 제안 구조에 대한 구현 면적을 실험하였다. 먼저 Matlab으로 Short training symbol 160개와 Long training symbol 160개의 Preamble 신호를 만들고, 80개의 Header, QPSK로 매핑된 320개의 Data인 Payload 신호를 만들었다. 이 신호가 실험에 사용된 송신 신호가 되며 그림 8과 같다.

그림 8의 첫 번째 그림은 송신 신호의 실수 값이며, 두 번째 그림은 송신 신호의 허수 값이다. 그림 8의 송신 신호에 주파수 읍셋을 인가하여 실험의 수신 신호로 사용하였다. 수신 신호 Short training symbol의  $s_8$ 부터  $s_{10}$ 까지의 신호를 Auto-correlator에 입력시켜 벡터를 생성하였다. 이와 같이 생성된 Auto-correlator의 출

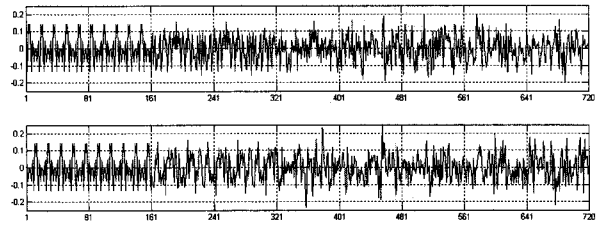


그림 8. 실험에 사용된 Preamble 및 Payload (송신신호)  
Fig. 8. Preamble and Payload used in simulation. (transmit signal)

표 1. CORDIC 추정부의 출력값  
(2 stage CORDIC Vector 모드)

Table 1. Output values of CORDIC estimation block. (2 stage CORDIC Vector mode)

|         | $x$       | $y$       | $\theta$ |
|---------|-----------|-----------|----------|
| Initial | 0.20062   | 0.031776  | 0        |
| 1 clock | 0.3168241 | -0.052648 | 18.4349  |
| 2 clock | 0.3333059 | -0.014690 | 11.5237  |
| 3 clock | 0.3344160 | -0.004303 | 9.7372   |
| 4 clock | 0.3344904 | -0.001691 | 9.2897   |
| 5 clock | 0.3344978 | 0.000268  | 8.9540   |
| 6 clock | 0.3344981 | 0.0001050 | 8.9819   |
| 7 clock | 0.3344981 | -0.00001  | 9.0029   |
| 8 clock | 0.3344981 | -0.000007 | 9.0012   |

력은 각각 0.20062와 0.031776이다. 이 벡터를 제안된 CORDIC 추정부의 입력으로 사용하였다. 제안 구조의 2 stage CORDIC Vector 모드를 실험하여  $\theta$ 값을 추정 한 결과는 표 1과 같다.

표 1에서 보듯이 2 stage CORDIC Vector 모드의 최종 벡터는 0.3344981773과 -0.00000719가 출력되었으며 추정 각은  $9.0012^\circ$ 가 출력되었다. 계산된  $\theta$  값은 Auto-correlator부에서 16 sample에 대한 누적 값이므로 결과 값을 16으로 나누어 주면 최종  $\theta$ 값을 구할 수 있으며 그 값은  $0.5625^\circ$ 이다.

이 최종  $\theta$ 값은 체배하여 CORDIC Multiplier의 입력으로 들어가 CORDIC 회전 모드를 수행하여 주파수 읍셋을 보상하게 된다. CORDIC Multiplier는 대략 주파수 읍셋 보상을 위하여 Long training symbol 128 샘플의 읍셋을 보상한다. 제안된 CORDIC Multiplier 구조를 MatLab으로 구현하여 읍셋을 보상하였다. 표 2는 128개 중에서 32 샘플의 Long training symbol의 보상 결과 값을 보여준다. 이 보상된 값들은 다음 절에서 수행한 Verilog-HDL 코딩의 테스트 벡터로 사용한다.

표 2. CORDIC Multiplier의 출력값

Table 2. Output values of CORDIC Multiplier.

|    | I        |          |          | Q        |          |          |
|----|----------|----------|----------|----------|----------|----------|
|    | 송신       | 수신       | 보상       | 송신       | 수신       | 보상       |
| 1  | 0.15625  | -0.48284 | 0.15625  | 0        | 0.14860  | -0.00004 |
| 2  | -0.00512 | 0.11570  | -0.00515 | -0.12033 | 0.03344  | -0.12032 |
| 3  | 0.03975  | 0.09200  | 0.03972  | -0.11116 | 0.07400  | -0.11118 |
| 4  | 0.09683  | -0.11058 | 0.09685  | 0.08280  | 0.06328  | 0.08277  |
| 5  | 0.02111  | -0.03347 | 0.02112  | 0.02789  | 0.01016  | 0.02788  |
| 6  | 0.05982  | 0.06073  | 0.05980  | -0.08770 | 0.08708  | -0.08772 |
| 7  | -0.11513 | 0.09335  | -0.11514 | -0.05518 | -0.08710 | -0.05515 |
| 8  | -0.03831 | 0.11278  | -0.03835 | -0.10617 | 0.00412  | -0.10616 |
| 9  | 0.09754  | -0.01341 | 0.09753  | -0.02589 | 0.10002  | -0.02591 |
| 10 | 0.05334  | -0.02464 | 0.05334  | 0.00407  | 0.04748  | 0.00406  |
| 11 | 0.00099  | 0.10497  | 0.00096  | -0.11500 | 0.04700  | -0.11500 |
| 12 | -0.13680 | 0.09927  | -0.13682 | -0.04738 | -0.10538 | -0.04735 |
| 13 | 0.02447  | 0.04291  | 0.02446  | -0.05853 | 0.04673  | -0.05854 |
| 14 | 0.05867  | -0.01158 | 0.05866  | -0.01493 | 0.05942  | -0.01495 |
| 15 | -0.02248 | -0.13474 | -0.02244 | 0.16066  | -0.09034 | 0.16066  |
| 16 | 0.11924  | -0.04942 | 0.11923  | -0.00409 | 0.10859  | -0.00413 |
| 17 | 0.06250  | 0.02731  | 0.06248  | -0.06250 | 0.08406  | -0.06252 |
| 18 | 0.03692  | -0.10427 | 0.03694  | 0.09834  | -0.01278 | 0.09833  |
| 19 | -0.05721 | -0.00769 | -0.05720 | 0.03930  | -0.06900 | 0.03931  |
| 20 | -0.13126 | 0.00579  | -0.13124 | 0.06522  | -0.14646 | 0.06527  |
| 21 | 0.08222  | -0.12075 | 0.08224  | 0.09236  | 0.02660  | 0.09233  |
| 22 | 0.06956  | -0.04683 | 0.06956  | 0.01412  | 0.05333  | 0.01410  |
| 24 | -0.06031 | 0.04773  | -0.06029 | 0.08129  | -0.09313 | 0.08130  |
| 25 | -0.03504 | 0.14696  | 0.03508  | -0.02180 | -0.03721 | -0.02179 |
| 26 | -0.12189 | 0.07874  | -0.12891 | -0.01657 | 0.04890  | -0.01653 |
| 27 | -0.12732 | 0.08591  | -0.12733 | -0.02050 | -0.09618 | -0.02047 |
| 28 | 0.07507  | 0.02087  | 0.07505  | -0.07404 | 0.10336  | -0.07406 |
| 29 | -0.00280 | -0.04315 | -0.00279 | 0.05377  | -0.03221 | 0.05378  |
| 30 | -0.09189 | -0.04330 | -0.09185 | 0.11513  | -0.14080 | 0.11516  |
| 31 | 0.09171  | -0.13930 | 0.09175  | 0.10587  | 0.01471  | 0.10584  |
| 32 | 0.01228  | -0.08664 | 0.01231  | 0.09760  | -0.04658 | 0.09760  |

Matlab을 사용한 실험에서 보상된 신호와 송신 신호와의 실수부와 허수부의 MSE는 각각 0.000001536과 0.00000125이다.

2. Verilog-HDL Function Simulation 및 합성

이 절에서는 제안 구조에 대하여 Verilog-HDL로 simulation을 수행하여 출력 값을 MatLab에서 만든 테스트 벡터와 비교하였다. 또한 완성된 RTL 코드는 Synopsys Design Compiler 합성 툴을 사용하여 제안

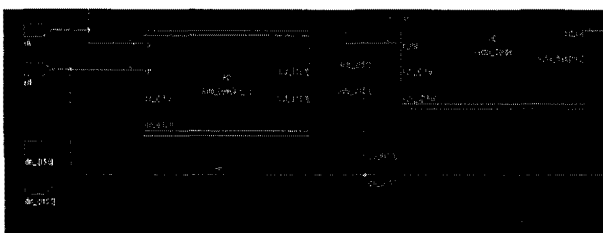


그림 9. 제안 구조의 전체 Schematic  
Fig. 9. Schematic of the entire proposed structure.

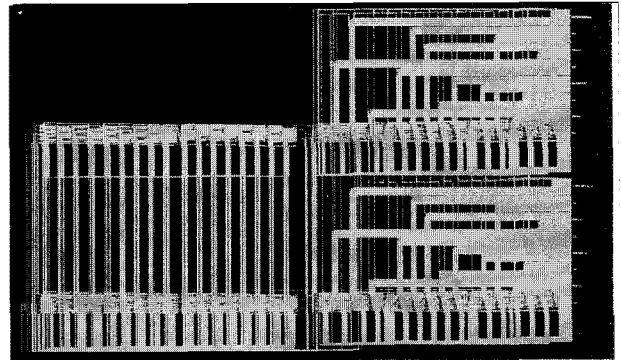


그림 10. Auto-correlator부의 합성 후 Schematic  
Fig. 10. Schematic of the Auto-correlator block.

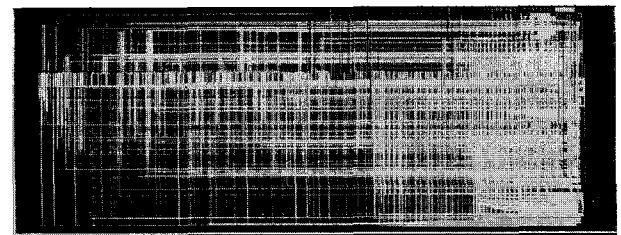


그림 11. CORDIC 추정부의 합성 후 Schematic  
Fig. 11. Schematic of the CORDIC estimator block.

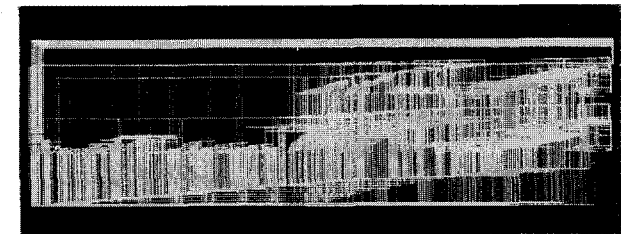


그림 12. CORDIC Multiplier부의 합성 후 Schematic  
Fig. 12. Schematic of the CORDIC Multiplier block.

표 4. 제안 구조의 면적

Table 4. The area of the proposed structure.

|                   | Combinational area | Non combinational area | Total cell area |
|-------------------|--------------------|------------------------|-----------------|
| Auto-correlator   | 115,775            | 66,740                 | 182,516         |
| CORDIC estimator  | 27,968             | 3,602                  | 31,570          |
| CORDIC Multiplier | 83,306             | 54,858                 | 138,164         |
| Total             | 227,050            | 125,202                | 352,252         |

구조의 구현 면적을 simulation하였으며 합성을 위하여 매그나칩 0.25-Micron 2.5V 공정을 사용하였다. 그림 9는 합성 툴을 사용한 전체 블록에 대한 schematic이다.

제안 구조의 합성 후 schematic은 그림 10, 11, 12와 같다.

제안구조의 면적을 계산하기 위하여 Synopsys

Design Compiler 합성 Tool을 사용한 결과는 표 4와 같다. 세부 구조인 Auto-corelator, CORDIC 추정부, CORDIC Multiplier의 구현 면적은 각각  $182,516 \mu m^2$ ,  $31,570 \mu m^2$ ,  $13,816 \mu m^2$ 로 계산되었으며 제안 구조 총 구현 면적은  $352,252 \mu m^2$ 로 계산되었다.

## V. 결 론

이 논문에서는 OFDM 통신 방식인 IEEE 802.11a WLAN의 저전력 주파수 옵셋 동기화기 구조를 제안하였다. 주파수 옵셋 동기화기의 핵심 블록인 옵셋 추정부와 옵셋 보상부의 세부 설계를 통하여 제안 구조의 효율성을 보였다. 옵셋 추정부는 구현 면적을 줄이기 위해 sequential CORDIC을 사용하되 연산 속도를 높이기 위하여 2 stage sequential 방식을 제안하였다. 옵셋 보상부는 Pipeline CORDIC을 사용하여 구현 면적을 줄임과 동시에 계산 속도를 높일 수 있었다.

## 참 고 문 헌

- [1] J. Granado, A. Torralba, J. Chavez, and V. Baena-Lecuyer, "Design of an efficient CORDIC-based architecture for synchronization in OFDM," *IEEE Trans. Consum. Electron.*, vol. 52, No. 3, pp.774-782, Aug. 2006.
- [2] A. R. S. Bahai, B. R. Saltzberg, and M. Ergen, "Multi-Carrier Digital Communications: Theory and Applications of OFDM", Springer, New York, 2004.
- [3] K. I. Lee, J. Kim, J. Lee, and Y. S. Cho "A Compact CORDIC Algorithm for Synchronization of Carrier Frequency Offset in OFDM Modems," *IEICE Trans. Commun.* vol. E89-B, no. 3, pp. 952-954, Mar. 2006.
- [4] IEEE, *Wireless MAC and PHY Specifications: High Speed Physical Layer in the 5 GHz Band*, P802.11a.D7.0, July 1999.
- [5] J. J. van de Beek, M. Sandell, and P. O. Borjesson, "ML estimation of time and frequency offset in OFDM systems," *IEEE Trans. Signal Process.*, vol. 45, no. 7, pp. 1800-1805, Jul. 1997.
- [6] Y. H. Hu, "CORDIC-based VLSI architectures for digital signal processing," *IEEE Signal Process. Mag.*, vol.9, pp. 17-35, Jul. 1992.
- [7] S. Chang and E. J. Powers, "An efficient frequency offset estimation in OFDM-based WALANS systems", *Electronics Letters*, vol 39, no. 21, pp. 1554-1555, 2003.
- [8] Z. R. Cheng, M. Gao, L. Li, D. Zhang, and Y. Song, "A design of modified CORDIC-based cosine and sine generator." *IEEE 5th International Conference on ASIC*, vol.2, pp. 765-768, Oct. 2003.

저 자 소 개



하 준 형(학생회원)  
 2009년 상명대학교 정보통신 공학과 졸업.(공학사)  
 2009년~현재 상명대학교 컴퓨터 정보통신공학과 대학원 석사과정.

<주관심분야 : 통신신호처리, SoC 설계>



장 영 범(정회원)  
 1981년 연세대학교 전기공학과 졸업.(공학사)  
 1990년 Polytechnic University 대학원 졸업.(공학석사)  
 1994년 Polytechnic University 대학원 졸업.(공학박사)

1981년~1999년 삼성전자 System LSI 사업부 수석연구원.

2002년~현재 상명대학교 정보통신공학과 교수.

<주관심분야 : 통신신호처리, 비디오신호처리, SoC 설계>



정 요 성(정회원)  
 2008년 상명대학교 정보통신 공학과 졸업.(공학사)  
 2010년 상명대학교 대학원 컴퓨터 정보통신공학과 대학원 졸업. (공학석사)  
 2010년~현재 (주)다물멀티미디어 주임연구원.

<주관심분야 : 통신신호처리, SoC 설계>



조 용 훈(정회원)  
 1986년 연세대학교 전기공학 졸업.(공학사)  
 1988년 연세대학교 대학원 전기공학과 졸업. (공학석사)  
 2001년 연세대학교 대학원 전기 전자공학과 졸업. (공학박사)

2006년~2008년 한남대 겸임교수

2002년~현재 (주)코메스타 부사장/연구소장, 전주대학교 겸임교수.

<주관심분야 : 디지털 통신, 디지털 모뎀, 이동통신 시스템, 광대역 전송 기술>