

명세서

발명의 명칭: 저면적 구조를 갖는 심볼 타이밍 옵셋 동기화기 및 그 설계방법

기술분야

- [1] 본 발명은 저면적 구조를 갖는 심볼 타이밍 옵셋 동기화기 및 그 설계방법에 관한 것으로서, 보다 상세하게는 OFDM 통신 방식인 IEEE 802.11a WLAN의 심볼 타이밍 옵셋 동기화기에 있어서 저면적 구조를 갖는 심볼 타이밍 옵셋 동기화기 및 그 설계방법에 관한 것이다.

[2]

배경기술

- [3] OFDM(Orthogonal Frequency Division Multiplexing: 직교주파수 분할) 방식의 기본원리는 고속의 전송률을 갖는 데이터열을 낮은 전송률을 갖는 많은 수의 데이터열로 나누고, 이들을 다수의 부반송파를 사용하여 동시에 전송하는 것이다. 즉, OFDM은 데이터열을 여러 개의 부채널(Sub-channel)로 동시에 나란히 전송하는 다중 반송파 전송방식의 특별한 형태로 본다. 따라서 OFDM 기법은, 1개 채널의 고속의 원천 데이터열을 다중의 채널로 동시에 전송한다는 측면에서는 "다중화 기술"이며, 다중의 반송파에 분할하여 실어 전송한다는 측면에서는 일종의 "변조 기술"이다. 각 부반송파의 파형은 시간축상으로는 직교(Orthogonal)하나, 주파수축상에서는 겹치게(Overlap)된다.
- [4] 최근, 이와 같은 OFDM 변조 방식이 다양한 유무선 데이터 전송 시스템에서 널리 사용되고 있다. OFDM 시스템은 기존의 단일 캐리어 전송방식에 비해서 주파수 이용 효율이 높고, 직교 관계를 갖는 다수의 부 반송파(Subcarriers)를 이용하여 심볼 간의 간섭(Inter Symbol Interference) 문제를 해소하였으며 보호구간(Guard Interval)을 사용하여 멀티패스페이딩 환경에서도 우수한 성능을 얻을 수 있다.
- [5] 그러나, OFDM 변조 방식은 동기 알고리즘에 취약한 단점을 지니고 있다. 즉 송신기와 수신기의 반송파 주파수 불일치로 인한 반송파 주파수 옵셋은 반송파간 간섭(Inter-Carrier Interference)을 발생시키고, OFDM 심볼의 동기를 정확히 획득하지 못하게 하여 심볼간의 간섭을 일으켜 시스템의 성능을 저하시킨다.
- [6] 따라서 OFDM 방식의 신호를 정확히 복조하기 위해서는 신호의 심볼 타이밍 동기가 매우 중요하다. 고속 데이터 전송은 실내 무선 환경에서 clock 주기의 수십 배가 넘는 다중 경로 지연을 발생시키는데, 심볼 타이밍 옵셋은 이러한 다중 경로 지연으로 인한 심볼 도착시간의 불확실성과 송, 수신단 샘플링 주파수 차이 등으로 발생하며 부 반송파의 신호의 위상을 회전시키고 FFT(Fast Fourier Transform) 윈도우 위치의 동기 오류를 일으켜 인접 심볼 간 간섭의 원인이 된다.

이와 같이 심볼 동기 오차는 시스템의 성능을 크게 저하시키는 이유가 되므로 OFDM 시스템의 수신 단에서는 주파수 오프셋 동기화 블록을 수행하기 이전에 송수신단의 심볼 타이밍 오프셋 동기가 선행되어야만 한다.

[7]

발명의 상세한 설명

기술적 과제

- [8] 본 발명의 실시예는 상기와 같은 필요성에 부응하기 위하여 창안된 것으로서, OFDM 통신 방식인 IEEE 802.11a WLAN의 심볼 타이밍 오프셋 동기화기에 있어서 저면적 구조를 갖는 심볼 타이밍 오프셋 동기화기 및 그 설계방법을 제공하는 것을 목적으로 한다.

[9]

과제 해결 수단

- [10] 전술한 목적을 달성하기 위한 본 발명의 실시예에 따른 심볼 타이밍 오프셋 동기화기는, 심볼 타이밍 오프셋 동기화기에 있어서, 직접형 필터 구조를 전치 직접형 필터 구조로 변환한 후, 상기 전치 직접형 필터 구조의 필터 계수를 CSD(Canonic Signed Digit)형으로 변환하며, CSD형 계수에 기초하여 필터 구조가 재설계된 것을 특징으로 한다.
- [11] 바람직하게는, 상기 전치 직접형 필터 구조의 훈련 심볼들을 2의 보수형으로 나타낸 후, 이를 상기 CSD형으로 변환한다.
- [12] 여기서, 상기 전치 직접형 필터 구조는 $r_1(n)$, $r_2(n)$, $i_1(n)$ 및 $i_2(n)$ 의 4개의 필터(여기서, r 은 실수(real), i 는 허수(image)를 의미)로 구성될 수 있다.
- [13] 이 경우, $r_1(n)$ 과 $i_1(n)$ 은 같은 필터이며, $r_2(n)$ 과 $i_2(n)$ 은 같은 필터일 수 있다.
- [14] 전술한 목적을 달성하기 위한 본 발명의 실시예에 따른 심볼 타이밍 오프셋 동기화기의 설계방법은, 심볼 타이밍 오프셋 동기화기의 설계방법에 있어서, 직접형 필터 구조를 전치 직접형 필터 구조로 변환하는 단계; 상기 전치 직접형 필터 구조의 필터 계수를 CSD형으로 변환하는 단계; 및 CSD형 계수에 기초하여 필터 구조를 재설계하는 단계를 포함하는 것을 특징으로 한다.
- [15] 바람직하게는, 상기의 심볼 타이밍 오프셋 동기화기의 설계방법은, 상기 전치 직접형 필터 구조의 훈련 심볼들을 2의 보수형으로 표현하는 단계를 더 포함하며, 상기 2의 보수형으로 표현된 심볼들에 기초하여 상기 CSD형으로 변환할 수 있다.
- [16] 여기서, 상기 전치 직접형 필터 구조는 $r_1(n)$, $r_2(n)$, $i_1(n)$ 및 $i_2(n)$ 의 4개의 필터(여기서, r 은 실수(real), i 는 허수(image)를 의미)로 구성될 수 있다.
- [17] 이 경우, $r_1(n)$ 과 $i_1(n)$ 은 같은 필터이며, $r_2(n)$ 과 $i_2(n)$ 은 같은 필터일 수 있다.
- [18]

발명의 효과

- [19] 본 발명의 실시예에 따르면, 심볼 타이밍 오프셋 동기화기의 곱셈기를 필터

개념으로 설계를 하여 1단계로 전치직접형 구조로 변환한 후에 2단계로 필터 계수를 CSD(Canonic Signed Digit)형 필터로 변환함으로써 저면적 심볼 타이밍 옵셋 동기화기 구조를 얻을 수 있게 된다.

[20]

도면의 간단한 설명

[21] 도 1은 IEEE 802.11a에서 사용되는 프리앰블 구조를 나타낸 도면이다.

[22] 도 2는 WLAN용 심볼 타이밍 옵셋과 주파수 옵셋 동기화기 블록도를 나타낸 도면이다.

[23] 도 3은 심볼 타이밍 옵셋 동기화기의 구조 예를 나타낸 도면이다.

[24] 도 4는 본 발명의 실시예에 따른 심볼 타이밍 옵셋 동기화기의 전치 직접형 전체 구조를 나타낸 도면이다.

[25] 도 5는 본 발명의 실시예에 따라 재설계된 저면적 $r_1(n)$ 필터 구조의 예를 나타낸 도면이다.

[26] 도 6은 본 발명의 실시예에 따라 재설계된 저면적 $r_2(n)$ 필터 구조의 예를 나타낸 도면이다.

[27]

발명의 실시를 위한 최선의 형태

[28] 이하, 첨부된 도면을 참조하여 본 발명의 실시예를 상세하게 설명한다. 이하의 설명에 있어서, 당업자에게 주지 저명한 기술에 대해서는 그 상세한 설명을 생략할 수 있다.

[29] 또한, 본 발명의 구성 요소를 설명하는 데 있어서, 동일한 명칭의 구성 요소에 대하여 도면에 따라 다른 참조부호를 부여할 수도 있으며, 서로 다른 도면임에도 불구하고 동일한 참조부호를 부여할 수도 있다. 그러나, 이와 같은 경우라 하더라도 해당 구성 요소가 실시예에 따라 서로 다른 기능을 갖는다는 것을 의미하거나, 서로 다른 실시예에서 동일한 기능을 갖는다는 것을 의미하는 것은 아니며, 각각의 구성 요소의 기능은 해당 실시예에서의 각각의 구성요소에 대한 설명에 기초하여 판단하여야 할 것이다.

[30] 또한, 본 발명의 실시예를 설명함에 있어, 관련된 공지 구성 또는 기능에 대한 구체적인 설명이 본 발명의 요지를 흐릴 수 있다고 판단되는 경우에는 그 상세한 설명은 생략할 수 있다.

[31] 또한, 본 발명의 구성 요소를 설명하는 데 있어서, 제 1, 제 2, A, B, (a), (b) 등의 용어를 사용할 수 있다. 이러한 용어는 그 구성 요소를 다른 구성 요소와 구별하기 위한 것일 뿐, 그 용어에 의해 해당 구성 요소의 본질이나 차례 또는 순서 등이 한정되지 않는다. 어떤 구성 요소가 다른 구성요소에 "연결", "결합" 또는 "접속"된다고 기재된 경우, 그 구성 요소는 그 다른 구성요소에 직접적으로 연결되거나 접속될 수 있지만, 각 구성 요소 사이에 또 다른 구성 요소가 "연결", "결합" 또는 "접속"될 수도 있다고 이해되어야 할 것이다.

- [32] WLAN(IEEE 802.11a) OFDM 무선통신 방식의 송신단에서 만들어지는 패킷은 프리앰블(Preamble), 헤더(Header), 그리고 실제 전송하고자하는 데이터인 페이로드(Payload)로 구성된다. IEEE 802.11a에서 사용되는 프리앰블의 구조는 도 1에 도시한 바와 같이 Short training symbol 160 샘플과 Long training symbol 160 샘플로 구성되어 있다.
- [33] 여기서, Short training symbol은 신호 검출, AGC, Diversity selection, Timing synchronization, Coarse frequency offset estimation을 하는데 이용된다. 또한, Long training symbol은 Channel estimation과 Fine frequency offset estimation을 하는데 사용된다.
- [34] Short training sequences는 s_1 부터 s_{10} 까지 10개의 같은 심볼로 구성되며 각각의 심볼은 16개의 실수부와 16개의 허수부로 구성되어 있다. 또한, Long training sequences는 l_1 과 l_2 의 두 개의 심볼로 구성되며 각각의 심볼은 64개의 실수부와 64개의 허수부로 구성되어 있다.
- [35] 프리앰블을 이용한 심볼 타이밍 옵셋 동기화는 주기적인 특성을 갖는 훈련 심볼을 이용하여 타이밍을 동기화시켜 심볼의 시작점을 정확히 찾는 작업이다. 이 시작점을 주파수 옵셋 동기화기 블록에 정확히 알려주어 FFT/IFFT 블록에 입력되는 심볼의 타이밍을 일치시킨다. 프리앰블을 이용한 WLAN용 심볼 타이밍 옵셋 동기화기와 주파수 옵셋 동기화기의 블록도는 도 2에 도시한 바와 같다.
- [36] 도 2에 나타난 바와 같이, 먼저 Preamble 중 s_1 부터 s_7 까지의 Short training symbol의 I와 Q가 심볼 타이밍 옵셋 동기화기로 입력되어 타이밍 동기를 획득하여야 한다. 타이밍 옵셋을 찾기 위하여 다음 식을 계산한다.
- [37] [수학식 1]
- [38]
- $$R_n = \left| \sum_{n=0}^{15} Y_n X_{15-n}^* \right|$$
- [39] 수학식 1에서 R_n 은 상호상관 방법으로 계산된 상관 값이며 X_{15-n} 은 수신기가 미리 알고 있는 Short training 심볼의 15-n번째 값이고 *은 conjugate를 의미하고 은 수신된 신호이다. 수신신호의 상호상관 값 은 수신 받은 신호와 Short training 심볼의 16개의 샘플을 복소 곱셈하여 얻은 값들이다. 따라서 수학식 1을 수신 신호에 대하여 연속으로 수행하면 16개의 반복적인 실수와 허수 샘플들이 끝나는 시점에서 상관 값이 크게 증가하여 피크를 나타내므로 심볼 타이밍 동기를 정확히 얻을 수 있다. 수학식 1에서 미리 알고 있는 Short training symbol과 입력신호와의 연산은 다음 식으로 표현할 수 있다.
- [40] [수학식 2]
- [41]

$$\begin{aligned}
R_n &= |YX^*| \\
&= \left| \sum (y_r + jy_i)(x_r - jx_i) \right| \\
&= \left| \sum (y_r x_r - jy_r x_i + jy_i x_r + y_i x_i) \right| \\
&= \left| \sum [(y_r x_r + y_i x_i) + j(y_i x_r - y_r x_i)] \right| \\
&= |r(n) + ji(n)| = \sqrt{r^2(n) + i^2(n)}
\end{aligned}$$

- [42] 수학식 2에서 복소 신호 X는 수신기에서 미리 알고 있는 Short training symbol이며 x_r 은 실수 값, x_i 는 허수 값을 나타낸다. 복소 신호 Y는 수신된 신호이며 y_r 은 실수 값, y_i 는 허수 값을 나타낸다. 이 식을 수행하면 상판 값이 구해지는데 여기서 2개의 실수부 곱셈 연산 블록과 2개의 허수부 곱셈 연산 블록이 필요하므로 총 4개의 곱셈 연산 블록이 필요하다. 이와 같은 4개의 곱셈 연산 블록을 포함한 심볼 타이밍 옵셋 동기화기 구조의 일 예는 도 3과 같다.
- [43] 도 3의 심볼타이밍 옵셋 동기화기 구조에서 y_r 은 수신신호의 실수, y_i 는 수신신호의 허수를 나타내고 x_{r0} 은 알고 있는 훈련 심볼의 0번째 실수샘플 값이고 15번까지 총 16개가 있고, x_{i0} 는 훈련 심볼의 0번째 허수샘플 값이며 15번까지 총 16개가 있다. 프리앰블 입력 값은 복소수로 각각의 곱셈기 파트별로 실수인 y_r , 허수인 y_i 가 입력된다. 입력 신호들은 쉬프트 레지스터인 매 clock마다 D0로 입력되며 그 옆의 쉬프트 레지스터로 이동된다. 쉬프트 레지스터는 D0부터 D14까지 총 15개가 있다. 각 쉬프트 레지스터에 입력된 값들은 매 클럭(clock)마다 각각의 곱셈기 파트에서 x_r 과 x_i 를 이용해 곱셈연산을 수행한다.
- [44] 도 3의 실수 곱셈 연산 블록은 $\sum y_r x_r$ 의 값과 $\sum y_i x_i$ 값의 합을 연산하고, 허수 곱셈 연산 블록은 $\sum y_i x_r$ 의 값과 $\sum y_r x_i$ 값의 합을 연산한다. 도 3에 도시된 심볼 타이밍 동기화기는 64개의 곱셈기를 사용하고 있으므로 구현 면적이 커지게 된다. 따라서 도 3과 같은 심볼 타이밍 옵셋 동기화기 구조에서 곱셈기 파트를 효율적으로 설계하여 저면적으로 구현할 수 있는 구조가 제안될 필요가 있다.
- [45] 본 발명의 실시예에 따른 심볼 타이밍 옵셋 동기화기의 구조는 심볼 타이밍 옵셋 동기화기를 하나의 필터 개념으로 접근하여 설계한다. 필터 개념으로 접근하면 프리앰블의 Short training symbol의 16개 필터계수의 필터로 볼 수 있다. 예를 들어, 도 3의 구조에 대한 저면적 구조는 전치직접형 구조설계 단계(1단계), CSD 계수 설계 단계(2단계)의 2 단계로 이루어질 수 있다.
- [46] 15개의 곱셈기를 사용하는 심볼 타이밍 옵셋 동기화기는 매 클럭(clock)마다 복소수의 입력이 들어오며, 이는 수신기에서 미리 알고 있는 Short training

symbol과 곱셈연산을 수행한다. 곱셈기 파트는 실수 파트 2개와 허수 파트 2개로 총 4개의 파트로 구성이 되며 하나의 곱셈기 파트는 15개의 곱셈기와 15개의 쉬프트 레지스터로 이루어진다. 따라서 총 곱셈기와 쉬프트 레지스터는 각각 60개가 필요하며 매 clock 마다 60번의 연산이 수행되어 구현 면적이 커지게 되는 단점이 있다. 이와 같은 단점은 필터 계수를 CSD형으로 변환하여 partial product를 줄인 후, 덧셈기와 쉬프트만을 사용하여 구현한다. 이 방법을 이용하기 위하여 1단계로 전치 직접형 필터 구조를 사용한다. 예를 들어, 도 3과 같은 심볼 타이밍 옵셋 동기화기 구조에서는 직접형(Direct form) 필터 구조를 사용하고 있는데, 이를 도 4와 같은 전치 직접형(Transposed direct form) 필터 구조로 변환한다.

- [47] 도 4에 도시된 바와 같은 심볼 타이밍 옵셋 동기화기의 구조는 4개의 필터로 구성되어 있다. 즉 $r_1(n)$, $r_2(n)$, $i_1(n)$, $i_2(n)$ 의 4개의 필터로 구성된다. 여기에서 $r_1(n)$ 필터와 $i_1(n)$ 필터는 계수가 같으므로 같은 필터이며, $r_2(n)$ 필터와 $i_2(n)$ 필터도 계수가 같으므로 같은 필터이다. 이와 같은 필터 계수를 CSD(Canonic Signed Digit)형으로 변환하여 $r_1(n)$ 필터와 $r_2(n)$ 필터에 대한 저면적을 구현할 수 있다.
- [48] 먼저, $r_1(n)$ 필터에 대한 저면적 필터 구조의 설계에 대하여 살펴본다.
- [49] 도 4에 도시한 바와 같은 필터 구조에 대하여 덧셈기를 사용하는 저면적 구조를 설계하기 위하여, 사용되는 필터 계수를 CSD형으로 변환할 수 있다. CSD형으로 변환하려면 훈련 심볼들을 2의 보수형으로 나타낸 후 이를 CSD형으로 변환한다. CSD형으로 구현하면 사용되는 덧셈기의 수가 적어지기 때문이다. 표 1은 필터에서 사용되는 계수의 실수 값을 16비트의 2의 보수형으로 나타낸 표이다.
- [50] [표 1] Short training symbol(REAL)의 2의 보수형($r_1(n)$ 블록)
- [51]

훈련심볼 (REAL)	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0.0460	0	0	0	0	0	1	0	1	1	1	1	0	0	0	1	1
-0.1324	1	1	1	0	1	1	1	1	0	0	0	0	1	1	1	0
-0.0135	1	1	1	1	1	1	1	0	0	1	0	0	0	1	1	0
0.1428	0	0	0	1	0	0	1	0	0	1	0	0	0	1	1	1
0.0920	0	0	0	0	1	0	1	1	1	1	0	0	0	1	1	1
0.1428	0	0	0	1	0	0	1	0	0	1	0	0	0	1	1	1
-0.0135	1	1	1	1	1	1	1	0	0	1	0	0	0	1	1	0
-0.1324	1	1	1	0	1	1	1	1	0	0	0	0	1	1	1	0
0.0460	0	0	0	0	0	1	0	1	1	1	1	0	0	0	1	1
0.0023	0	0	0	0	0	0	0	0	0	1	0	0	1	0	1	1
-0.0785	1	1	1	1	0	1	0	1	1	1	1	1	0	1	0	0
-0.0127	1	1	1	1	1	1	1	0	0	1	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
-0.0127	1	1	1	1	1	1	1	0	0	1	1	0	0	0	0	0
-0.0785	1	1	1	1	0	1	0	1	1	1	1	1	0	1	0	0
0.0023	0	0	0	0	0	0	0	0	0	1	0	0	1	0	1	1

- [52] 표 1은 2의 보수형 계수와 필요한 덧셈기의 개수를 나타내고 있다. 표 1에서의 1의 수가 덧셈기의 수와 비례한다. 즉 1의 수가 구현 면적과 비례하게 된다. 즉 2의 보수형 계수를 사용하면 상대적으로 1의 수가 많으므로 필요한 덧셈기의 수가 많아지기 때문에 구현 면적이 커진다.
- [53] 따라서 덧셈기의 수를 줄이기 위하여 CSD 형의 계수로 변환한다. 즉 필터계수로 사용되는 Short training symbol의 실수 값과 허수 값을 CSD 형으로 바꿔준다. 표 2는 표 1의 2의 보수형 계수를 16비트의 CSD 형으로 나타내었다.
- [54] [표 2] Short training symbol(REAL)의 CSD형($r_1(n)$ 블록)
- [55]

훈련심볼 (REAL)	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0.0460	0	0	0	0	1	0	N	0	0	0	N	0	0	1	0	N
-0.1324	0	0	0	N	0	0	N	0	0	0	1	0	0	0	N	0
-0.0135	0	0	0	0	0	0	N	0	0	1	0	0	1	0	N	0
0.1428	0	0	0	1	0	0	1	0	0	1	0	0	1	0	0	N
0.0920	0	0	0	1	0	N	0	0	0	N	0	0	1	0	0	N
0.1428	0	0	0	1	0	0	1	0	0	1	0	0	1	0	0	N
-0.0135	0	0	0	0	0	0	N	0	0	1	0	0	1	0	N	0
-0.1324	0	0	0	N	0	0	0	N	0	0	0	1	0	0	0	N
0.0460	0	0	0	0	1	0	N	0	0	0	N	0	0	1	0	N
0.0023	0	0	0	0	0	0	0	0	0	1	0	1	0	0	N	0
-0.0785	0	0	0	0	N	0	N	0	0	0	0	N	0	1	0	0
-0.0127	0	0	0	0	0	0	N	0	1	0	N	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
-0.0127	0	0	0	0	0	0	N	0	1	0	N	0	0	0	0	0
-0.0785	0	0	0	0	N	0	N	0	0	0	0	N	0	1	0	0
0.0023	0	0	0	0	0	0	0	0	0	1	0	1	0	0	N	0

- [56] $r_1(n)$ 펄터의 구현에서 CSD형의 펄터계수를 사용하면 2의 보수형 구조보다 덧셈기의 수를 감소시킬 수 있다. 표 2의 CSD형 계수를 순서대로 x_{n0} 부터 x_{n15} 까지로 정의하여 표현한다. 아래 첨자 r_n 은 실수를 의미한다. 따라서 표 2를 식으로 나타내면 수학식 3과 같다.

[57] [수학식 3]

[58]

$$x_{r0} = a \gg 4 - a \gg 6 - a \gg 10 + a \gg 13 - a \gg 15$$

$$x_{r1} = -a \gg 3 - a \gg 7 + a \gg 11 - a \gg 14$$

$$x_{r2} = -a \gg 6 + a \gg 9 + a \gg 12 - a \gg 14$$

$$x_{r3} = a \gg 3 + a \gg 6 + a \gg 9 + a \gg 12 - a \gg 15$$

$$x_{r4} = a \gg 3 - a \gg 5 - a \gg 9 + a \gg 12 - a \gg 15$$

$$x_{r5} = x_{r3}$$

$$x_{r6} = x_{r2}$$

$$x_{r7} = x_{r1}$$

$$x_{r8} = x_{r0}$$

$$x_{r9} = a \gg 9 + a \gg 11 - a \gg 13 - a \gg 15$$

$$x_{r10} = -a \gg 4 - a \gg 6 - a \gg 11 + a \gg 13$$

$$x_{r11} = -a \gg 6 + a \gg 8 - a \gg 10$$

$$x_{r13} = x_{r11}$$

$$x_{r14} = x_{r10}$$

$$x_{r15} = x_{r9}$$

- [59] 본 발명의 실시예에 따라 수학식 3을 덧셈기를 사용하여 구현한 $r_1(n)$ 필터는 도 5에 도시한 바와 같다. 도 5를 살펴보면 프리앰블에서 필터로 신호를 입력받아서 쉬프트와 덧셈, 뺄셈연산을 통해 각각의 15개의 연산 값들을 계산할 수 있다.
- [60] 다음에, $r_2(n)$ 필터에 대한 저면적 필터 구조의 설계에 대하여 살펴본다.
- [61] Short training symbol의 허수 값의 2의 보수형은 표 3과 같다. 허수 값은 실수 값의 8 sample 지연한 값과 같다.
- [62] [표 3] Short training symbol(IMAGE)의 2의 보수형($r_2(n)$ 블록)
- [63]

훈련심볼 (REAL)	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0.0460	0	0	0	0	0	1	0	1	1	1	1	0	0	0	1	1
0.0023	0	0	0	0	0	0	0	0	0	1	0	0	1	0	1	1
-0.0785	1	1	1	1	0	1	0	1	1	1	1	1	0	1	0	0
-0.0127	1	1	1	1	1	1	1	0	0	1	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
-0.0127	1	1	1	1	1	1	1	0	0	1	1	0	0	0	0	0
-0.0785	1	1	1	1	0	1	0	1	1	1	1	1	0	1	0	0
0.0023	0	0	0	0	0	0	0	0	0	1	0	0	1	0	1	1
0.0460	0	0	0	0	0	1	0	1	1	1	1	0	0	0	1	1
-0.1324	1	1	1	0	1	1	1	1	0	0	0	0	1	1	1	0
-0.0135	1	1	1	1	1	1	1	0	0	1	0	0	0	1	1	0
0.1428	0	0	0	1	0	0	1	0	0	1	0	0	0	1	1	1
0.0920	0	0	0	0	1	0	1	1	1	1	0	0	0	1	1	1
0.1428	0	0	0	1	0	0	1	0	0	1	0	0	0	1	1	1
-0.0135	1	1	1	1	1	1	1	0	0	1	0	0	0	1	1	0
-0.1324	1	1	1	0	1	1	1	1	0	0	0	0	1	1	1	0

[64] 표 4는 표 3에 나타난 Short training symbol의 허수 값의 2의 보수형을 CSD 형태로 표현한 것이다.

[65] [표 4] Short training symbol(IMAGE)의 CSD형($r_2(n)$ 블록)

[66]

훈련심볼 (REAL)	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0.0460	0	0	0	0	1	0	N	0	0	0	N	0	0	1	0	N
0.0023	0	0	0	0	0	0	0	0	0	1	0	1	0	N	0	N
-0.0785	0	0	0	0	N	0	N	0	0	0	0	N	0	1	0	0
-0.0127	0	0	0	0	0	0	N	0	1	0	N	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
-0.0127	0	0	0	0	0	0	N	0	1	0	N	0	0	0	0	0
-0.0785	0	0	0	0	N	0	N	0	0	0	0	N	0	1	0	0
0.0023	0	0	0	0	0	0	0	0	0	1	0	1	0	N	0	N
0.0460	0	0	0	0	1	0	N	0	0	0	N	0	0	1	0	N
-0.1324	0	0	0	N	0	0	0	N	0	0	0	1	0	0	N	0
-0.0135	0	0	0	0	0	0	N	0	0	1	0	0	1	0	N	0
0.1428	0	0	0	1	0	0	1	0	0	1	0	0	1	0	0	N
0.0920	0	0	0	1	0	N	0	0	0	N	0	0	1	0	0	N
0.1428	0	0	0	1	0	0	1	0	0	1	0	0	1	0	0	N
-0.0135	0	0	0	0	0	0	N	0	0	1	0	0	1	0	N	0
-0.1324	0	0	0	N	0	0	0	N	0	0	0	1	0	0	N	0

[67] 표 4를 식으로 표현하면 수학식 4와 같다.

[68] [수학식 4]

[69]

$$x_{r0} = a \gg 4 - a \gg 6 - a \gg 10 + a \gg 13 - a \gg 15$$

$$x_{r1} = a \gg 9 + a \gg 11 - a \gg 13 - a \gg 15$$

$$x_{r2} = -a \gg 4 - a \gg 6 - a \gg 11 + a \gg 13$$

$$x_{r3} = -a \gg 6 + a \gg 8 - a \gg 10$$

$$x_{r5} = x_{r3}$$

$$x_{r6} = x_{r2}$$

$$x_{r7} = x_{r1}$$

$$x_{r8} = x_{r0}$$

$$x_{r9} = -a \gg 3 - a \gg 7 + a \gg 11 - a \gg 14$$

$$x_{r10} = -a \gg 6 + a \gg 9 + a \gg 12 - a \gg 14$$

$$x_{r11} = a \gg 3 + a \gg 6 + a \gg 9 + a \gg 12 - a \gg 15$$

$$x_{r12} = a \gg 3 - a \gg 5 - a \gg 9 + a \gg 12 - a \gg 15$$

$$x_{r13} = x_{r11}$$

$$x_{r14} = x_{r10}$$

$$x_{r15} = x_{r9}$$

[70]

[71] 본 발명의 실시예에 따라 수학식 4를 이용해서 $r2(n)$ 필터를 구현하면 도 6과 같이 나타낼 수 있다.

[72] 본 발명의 실시예는 OFDM 통신 방식인 IEEE 802.11a WLAN의 심볼 타이밍 옵셋 동기화기 블록에 대한 저면적 구조를 제안하였다. 심볼 타이밍 옵셋 동기화기의 곱셈기를 필터 개념으로 설계를 하여 1단계로 전치직접형 구조로 변환한 후에 2단계로 필터 계수를 CSD형 필터로 변환함으로써 저면적 심볼 타이밍 옵셋 동기화기 구조를 얻을 수 있었다.

[73] 이상에서, 본 발명의 실시예를 구성하는 모든 구성 요소들이 하나로 결합하거나 결합하여 동작하는 것으로 기재되어 있다고 해서, 본 발명이 반드시 이러한 실시예에 한정되는 것은 아니다. 즉, 본 발명의 목적 범위 안에서라면, 그 모든 구성 요소들이 하나 이상으로 선택적으로 결합하여 동작할 수도 있다. 또한, 그 모든 구성 요소들이 각각 하나의 독립적인 하드웨어로 구현될 수 있지만, 각 구성 요소들의 그 일부 또는 전부가 선택적으로 조합되어 하나 또는 복수 개의 하드웨어에서 조합된 일부 또는 전부의 기능을 수행하는 프로그램 모듈을 갖는 컴퓨터 프로그램으로서 구현될 수도 있다. 또한, 이와 같은 컴퓨터 프로그램은 USB 메모리, CD 디스크, 플래쉬 메모리 등과 같은 컴퓨터가 읽을 수 있는 저장매체(Computer Readable Media)에 저장되어 컴퓨터에 의하여 읽혀지고

실행됨으로써, 본 발명의 실시예를 구현할 수 있다. 컴퓨터 프로그램의 저장매체로서는 자기 기록매체, 광 기록매체, 캐리어 웨이브 매체 등이 포함될 수 있다.

[74] 또한, 기술적이거나 과학적인 용어를 포함한 모든 용어들은, 상세한 설명에서 다르게 정의되지 않는 한, 본 발명이 속하는 기술 분야에서 통상의 지식을 가진 자에 의해 일반적으로 이해되는 것과 동일한 의미를 갖는다. 사전에 정의된 용어와 같이 일반적으로 사용되는 용어들은 관련 기술의 문맥상의 의미와 일치하는 것으로 해석되어야 하며, 본 발명에서 명백하게 정의하지 않는 한, 이상적이거나 과도하게 형식적인 의미로 해석되지 않는다.

[75] 이상의 설명은 본 발명의 기술 사상을 예시적으로 설명한 것에 불과한 것으로서, 본 발명이 속하는 기술 분야에서 통상의 지식을 가진 자라면 본 발명의 본질적인 특성에서 벗어나지 않는 범위에서 다양한 수정 및 변형이 가능할 것이다. 또한, 본 발명에 개시된 실시예들은 본 발명의 기술 사상을 한정하기 위한 것이 아니라 설명하기 위한 것이며, 이러한 실시예에 의하여 본 발명의 기술 사상의 범위가 한정되는 것은 아니다. 따라서, 본 발명의 보호 범위는 청구범위에 의하여 해석되어야 하며, 그와 균등한 범위 내에 있는 모든 기술 사상은 본 발명의 권리범위에 포함되는 것으로 해석되어야 할 것이다.

[76]

청구범위

- [청구항 1] 심볼 타이밍 옵셋 동기화에 있어서,
직접형 필터 구조를 전치 직접형 필터 구조로 변환한 후, 상기 전치 직접형 필터 구조의 필터 계수를 CSD(Canonic Signed Digit)형으로 변환하며, CSD형 계수에 기초하여 필터 구조가 재설계된 것을 특징으로 하는 심볼 타이밍 옵셋 동기화기.
- [청구항 2] 제 1항에 있어서,
상기 전치 직접형 필터 구조의 훈련 심볼들을 2의 보수형으로 나타낸 후, 이를 상기 CSD형으로 변환한 것을 특징으로 하는 심볼 타이밍 옵셋 동기화기.
- [청구항 3] 제 1항에 있어서,
상기 전치 직접형 필터 구조는 $r_1(n)$, $r_2(n)$, $i_1(n)$ 및 $i_2(n)$ 의 4개의 필터(여기서, r 은 실수(real), i 는 허수(image)를 의미)로 구성되는 것을 특징으로 하는 심볼 타이밍 옵셋 동기화기.
- [청구항 4] 제 3항에 있어서,
 $r_1(n)$ 과 $i_1(n)$ 은 같은 필터이며, $r_2(n)$ 과 $i_2(n)$ 은 같은 필터인 것을 특징으로 하는 심볼 타이밍 옵셋 동기화기.
- [청구항 5] 심볼 타이밍 옵셋 동기화기의 설계방법에 있어서,
직접형 필터 구조를 전치 직접형 필터 구조로 변환하는 단계;
상기 전치 직접형 필터 구조의 필터 계수를 CSD형으로 변환하는 단계; 및
CSD형 계수에 기초하여 필터 구조를 재설계하는 단계
를 포함하는 것을 특징으로 하는 심볼 타이밍 옵셋 동기화기의 설계방법.
- [청구항 6] 제 5항에 있어서,
상기 전치 직접형 필터 구조의 훈련 심볼들을 2의 보수형으로 표현하는 단계
를 더 포함하며,
상기 2의 보수형으로 표현된 심볼들에 기초하여 상기 CSD형으로 변환하는 것을 특징으로 하는 심볼 타이밍 옵셋 동기화기의 설계방법.
- [청구항 7] 제 5항에 있어서,
상기 전치 직접형 필터 구조는 $r_1(n)$, $r_2(n)$, $i_1(n)$ 및 $i_2(n)$ 의 4개의 필터(여기서, r 은 실수(real), i 는 허수(image)를 의미)로 구성되는 것을 특징으로 하는 심볼 타이밍 옵셋 동기화기의 설계방법.
- [청구항 8] 제 7항에 있어서,
 $r_1(n)$ 과 $i_1(n)$ 은 같은 필터이며, $r_2(n)$ 과 $i_2(n)$ 은 같은 필터인 것을

특징으로 하는 심볼 타이밍 옵셋 동기화기의 설계방법.