

# PLL없이 동작하는 S/PDIF IC 설계에 관한 연구

## Study on the Design of S/PDIF IC which Can Operate without PLL

김경수\*, 박주성\*, 김석찬\*  
(Kyoung-Soo Kim\*, Ju-Sung Park\*, Suk-Chan Kim\*)

\*부산대학교 전자공학과

(접수일자: 2004년 8월 18일; 수정일자: 2004년 12월 14일; 채택일자: 2004년 12월 23일)

본 논문에서는 PLL (Phase Locked Loop) 없이 동작할 수 있는 S/PDIF (Sony Philips Digital Interface) 수신기의 연구에 관하여 다룬다. 현재 대부분의 오디오 장치와 오디오 프로세서에서 S/PDIF 수신기가 사용되고 있음에도 불구하고, 국내에서는 이에 관한 연구가 많지 않은 실정이다. 현재 사용되고 있는 S/PDIF 수신용 상용 DAC (Digital-to-Analog Converters) 칩들은 모두 내부에 PLL 회로를 포함하고 있다. PLL 회로는 S/PDIF 디지털 신호로부터 클럭 정보를 뽑아내고 클럭과 입력 신호간의 동기화를 맞추는 역할을 한다. 그러나, PLL 회로는 "아날로그 회로"라는 특성 때문에 VLSI (Very Large Scale Integrated Circuits) 회로의 SOC (System On Chips) 설계에 있어 많은 어려움을 야기한다. 본 논문에서는 PLL 회로 없이 순수 디지털 회로만으로 구현된 S/PDIF 수신기를 제안하였다. 제안된 수신기의 핵심 아이디어는 16 MHz의 기본 클럭과 S/PDIF 신호의 속도비를 이용한다는 것이다. 본 논문에서는 수십만 개의 S/PDIF 입력 신호에 대한 디코딩 확인 후, PLL같은 아날로그 회로 없이 순수 디지털 회로만으로 S/PDIF 수신기를 설계할 수 있음을 확인하였다. 제안된 S/PDIF 수신기는 SOC 설계용 IP로서 활용될 수 있을 것으로 본다.

**핵심용어:** 소니/필립스 디지털 인터페이스, IEC 958, 복호화기, 디지털 아날로그 변환기

**투고분야:** 음향 신호처리 분야 (1.2)

In this paper, we deal with the research about a S/PDIF (Sony Philips Digital Interface) receiver which can operate without PLL (Phase Locked Loop) circuits. Although a S/PDIF receiver is used in most audio devices and audio processors in these days, yet there are only few domestic researches about S/PDIF. Currently used commercial DACs (Digital-to-Analog Converters) which can decode S/PDIF signals, have a PLL circuit inside them. The PLL makes it possible to extract clock information from S/PDIF digital signal and to synchronize a clock signal with input signals. But the PLL circuit makes many difficulties in designing the SOC (System On Chips) of VLSIs (Very Large Scale Integrated Circuits) because it is an "analog circuit". We proposed a S/PDIF receiver which doesn't have PLL circuits and only has pure digital circuits. The key idea of the proposed S/PDIF receiver, is to use the ratio between a 16 MHz basic input clock and S/PDIF signals. After having decoded hundreds thousands S/PDIF inputs, it went to prove that a S/PDIF receiver can be designed with pure digital circuits and without any analog circuits such as PLL circuits. We have confidence that the proposed S/PDIF receiver can be used as an IP (Intellectual Property) for the SOC design of the digital circuits.

**Keywords:** S/PDIF, IEC958, decoder, DAC

**ASK subject classification:** Acoustic Signal Processing (1.2)

## I. 서론

1980년대 초반 CD 플레이어 (Compact Disc Player)의 등장으로 디지털 오디오에 대한 관심이 대두되었다.

책임저자: 김경수 (kyskim@pusan.ac.kr)  
부산광역시 금정구 장전동 산30번지 부산대학교 제도관 6309호  
반도체설계교육센터  
(전화: 051-510-2829 ; 팩스: 051-510-2827)

하지만 초기에는 오디오 기기 내부에서는 디지털 신호가 생성되지만 이를 출력할 때는 아날로그 신호로 변환해서 사용하였다. 현대에는 신호의 품질을 손상하지 않기 위하여 가능한 디지털 형태로 유지하는 경향이 있으며, 이를 위해 오디오 기기간의 디지털 통신을 위해 몇 가지 인터페이스 방식이 제안되었으며, 그 중의 하나가 EBU (European Broadcasting Union)로부터 파생된 IEC

표 1. S/PDIF의 데이터 포맷  
Table 1. The data format of S/PDIF.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Preamble				auxiliary data				Audio sample				Audio sample																V U C P			

958 1989-03 (Consumer part)이다. 이 방식은 S/PDIF (Sony Philips Digital Interface)라고도 불린다. 대부분의 DAC (Digital-to-Analog Converter)는 음악데이터를 CD 등으로부터 받아들이기 위하여 S/PDIF 수신기를 내장하고 있으며 Yamaha의 YSS922, YSS932[1,2] Philips의 UDA1350, UDA1351H[3,4]와 Crystal semiconductor사의 CS8411/CS8412, CS8413/CS8414[5,6] 등이 있다. 이들 상용 S/PDIF 수신기는 PLL (Phase Locked Loop) 회로를 이용하여 클럭을 추출한 후 S/PDIF 프로토콜로 정보를 받아들인다. 하지만 디지털 부의 노이즈로 인하여 PLL과 같은 아날로그회로를 디지털 회로와 동시에 하나의 칩에 집적하는 데는 어려움이 따른다. 따라서 디지털방식으로만 S/PDIF 프로토콜을 구현할 수 있다면 오디오 SoC (System On a Chip) 개발에 큰 기여를 할 수 있을 것이다. 본 연구에서는 오디오 신호 송신장치로부터의 디지털 신호가 비교적 깨끗한 경우에 PLL 없이 S/PDIF를 구현 할 수 있는 방식을 제안한다. 지금까지의 구현 방식은 디지털 신호가 비교적

깨끗한 경우에도 PLL을 사용하기 때문에 칩의 면적이 크게 되는 불합리한 측면이 있다. 본 연구에서는 16 MHz의 기준 클럭을 이용하여 32 KHz, 44.1 KHz, 48 KHz 등의 다양한 오디오 샘플링 레이트를 지원할 수 있는 순수한 디지털 방식의 S/PDIF 수신기의 구현 원리, 설계 과정 및 그 검증 결과에 관해 상세하게 소개한다. 제2장에서는 S/PDIF의 방식에 대하여 소개하고, 제3장과 제4장에서는 제안된 S/PDIF 수신기의 설계 및 검증 과정을 설명하고 마지막으로 결론을 내린다.

## II. S/PDIF의 방식

CD-ROM, DAT, DCC는 S/PDIF 프로토콜을 가지는 디지털 출력을 내보내며, 오디오 데이터는 각각 44.1 KHz, 48 KHz, 32 KHz의 샘플링 레이트를 가진다. 그림 1.은 S/PDIF 프로토콜의 시스템 구성의 예를 보여준

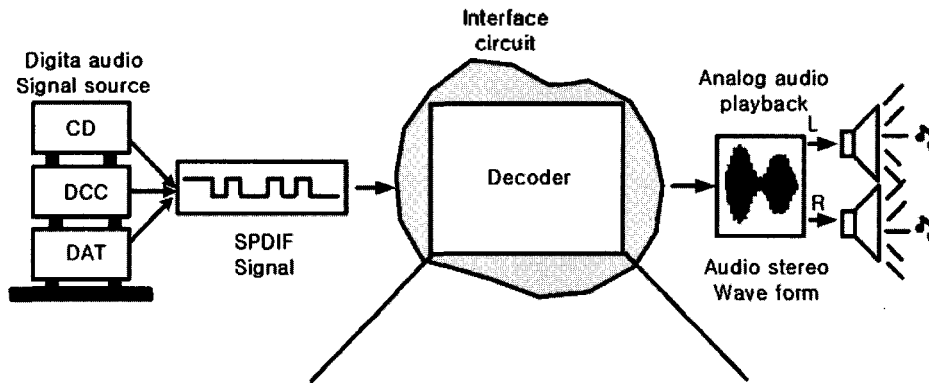


그림 1. S/PDIF 수신기의 시스템 구성도  
Fig. 1. The system configuration of a S/PDIF receiver.

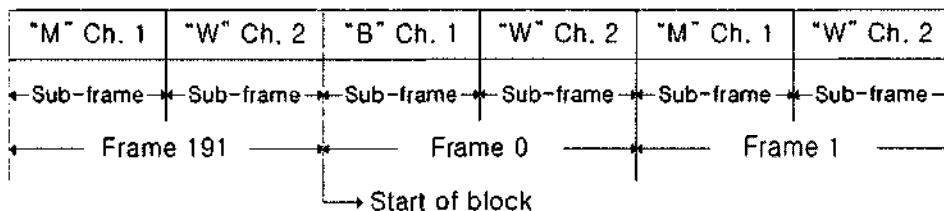


그림 2. S/PDIF 데이터 한 블록의 포맷  
Fig. 2. One block format of S/PDIF data.

표 2. 프리앰블의 인코딩 값

Table 2. The encoded values of preambles.

preamble	last cell '0'	last cell '1'
B	11101000	00010111
M	11100010	00011101
W	11100100	00011011

다. 데이터의 샘플링 레이트와 비례하여 외부로 전송될 때 CD 플레이어는 5.6MHz, DAT는 6.2MHz, DCC는 4.0 MHz의 데이터 전송 속도를 가진다.

S/PDIF 데이터는 32비트의 워드 단위 (subframe)로 전송되는데 표 1.은 S/PDIF의 한 샘플 즉 서브프레임의 비트 구성을 나타내고 있다. 비트0~비트 3은 헤드정보이고 B 프리앰블 (Left channel, 블럭 시작), M 프리앰블 (Left channel), W 프리앰블 (Right channel)의 3가지 형태가 존재한다. 각 프리앰블은 하나의 서브프레임

에 대해서 동기를 맞추기 위해서 사용된다. 표 1.에서 보는 바와 같이 S/PDIF는 하나의 서브프레임에 대해 최대 24비트 사이즈의 정밀도를 제공한다[7]. 그림 2.에 S/PDIF의 한 블록의 포맷을 나타내었다. 하나의 블록은 총 192개의 프레임으로 구성되고, 각 프레임은 채널 수에 따라 n개의 서브프레임으로 구성된다. S/PDIF의 세 가지 모드 중 CD 플레이어의 경우를 예로 들어보자. CD 플레이어의 경우는 일반적으로 16비트의 데이터만 사용하므로 표 1.에서 비트12 (LSB)~비트 27 (MSB)만 사용하고 나머지 오디오 샘플 데이터는 "0" 으로 채워 보낸다. 또한 CD 플레이어는 스테레오 (2 channel)를 사용하므로 하나의 프레임은 2개의 서브프레임으로 구성된다. 그림 2.에서 보는 바와 같이 블록의 시작시만 채널 1번에 대한 헤드로 B 프리앰블이 사용되고, 그 외의 채널 1번에 대한 헤드는 M 프리앰블로 사용된다. S/PDIF 프로토콜을 사용하는 기기의 데이터 전송 속도는 식 1.로써 계산될 수 있다. CD 플레이어의 경우는 샘플링 레이트

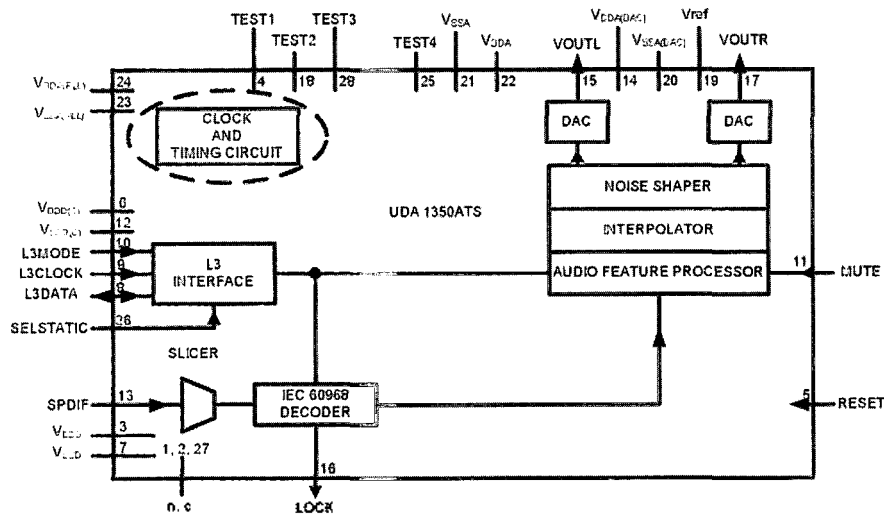


그림 3. PLL을 내장하고 있는 상용 칩의 블록도(3)  
Fig. 3. The top block of a commercial chip which includes PLL circuits(3).

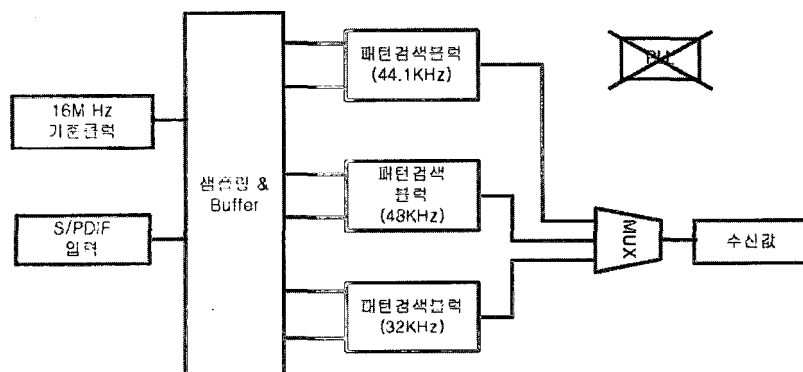


그림 4. 제안된 S/PDIF 수신기의 간략화된 구성도  
Fig. 4. The simplified configuration of a proposed S/PDIF receiver.

44.1KHz, 채널 수는 2개이므로 식 1.에 따라 계산해 보면 데이터 전송 속도는 5.6MHz가 된다.

$$\text{데이터 전송 속도(Hz)} = \text{샘플링 레이트} \times \text{샘플 전송 단위}(32) \times \text{채널 수} \times 2 \quad (\text{식 } 1)$$

※ 샘플링 레이트 : CD = 44,1KHz, DAT : 48KHz, DCC : 32KHz

프리앰블을 제외한 모든 데이터는 "bi-phase mark code" (BMC)를 사용한다. 즉, 데이터 "0"은 "01" (직전 셀 값이 1인 경우) 또는 "10" (직전 셀 값이 0인 경우)으로 코딩되고, 데이터 "1"은 "00" (직전 셀 값이 1인 경우) 또는 "11" (직전 셀 값이 0인 경우)로 코딩 된다[7]. 그러나, 프리앰블은 예외적으로 BMC (bi-phase mark code, 이하 BMC)규칙을 따르지 않으며 표 2.와 같이 코딩 된다[7].

### III. S/PDIF 수신기의 설계

#### 3.1. 제안한 S/PDIF 수신기의 동작 원리

일반적으로 S/PDIF 프로토콜을 지원하는 상용 수신기 칩은 32 KHz, 44.1 KHz, 48 KHz 샘플링 주파수의 오디오 데이터를 수신하기 위하여 PLL 회로를 사용하여 샘플링 주파수에 해당하는 클럭을 만들어 낸다. 그림 3.은 PLL 회로가 포함되어 있는 상용칩 (UDA1350ATS)[3]

의 한 예를 보여주고 있는데, 블록도 내의 "CLOCK & TIMMING CIRCUIT"가 PLL에 해당된다. 그림 3.의 PLL 회로에서는 외부 시스템 클럭 (L3CLOCK)을 이용하여 칩 내부에 사용될 여러 종류의 시스템 클럭을 생성한다. "TC 60968 DECODER" 블록에서는 PLL에서 생성된 시스템 클럭 중 S/PDIF 입력 데이터 속도에 상응하는 시스템 클럭을 이용해서 프리앰블 및 데이터를 추출해 낸다. 그림 3.의 "L3 인터페이스 블록"은 UDA1350ATS와 외부 프로세서 간의 제어 정보가 교환되며 우측 상단에 있는 블록들은 추출된 디지털 데이터를 외부 아날로그 신호로 내 보내기 위한 전처리 작업 (디엠프시스, 인터플레이션, 노이즈 셰이핑)을 한다. 그림 4.는 본 논문에서 제안하는 PLL없이 동작하는 S/PDIF 수신기의 간략화된 구성도이다. 그림 4.의 제안된 수신기와 그림 3.과 같은 상용 DAC와의 차이는 PLL 역할을 제안하는 디지털회로가 대신한다는 것이다. 본 논문에서 제안된 수신기를 S/PDIF 입력이 노이즈가 적은 비교적 깨끗한 디지털 입력이라는 전제하에 설계하였으므로 잡음 등으로 왜곡된 S/PDIF 입력인 경우에는 문제를 일으킬 수는 있지만 그렇지 않은 경우에는 오디오 프로세서에 내장되어 CD, DCC, DAT 등 S/PDIF 출력을 지원하는 장치의 출력 데이터를 문제없이 받아들일 수 있다[8]. PLL없이 클럭 정보를 뽑아낼 수 있는 기본 개념은 고정 클럭 (16 MHz)와 S/PDIF 디지털 신호가 가지는 세가지 전송 속도 (5.6 MHz, 6.2 MHz, 4.0 MHz)와의 비를 이용하는 것이다. 미리 계산되어진 비를 이용해서 샘플링된 S/PDIF 디지털 신호에서 프레임 정보를 뽑아내고, 데이

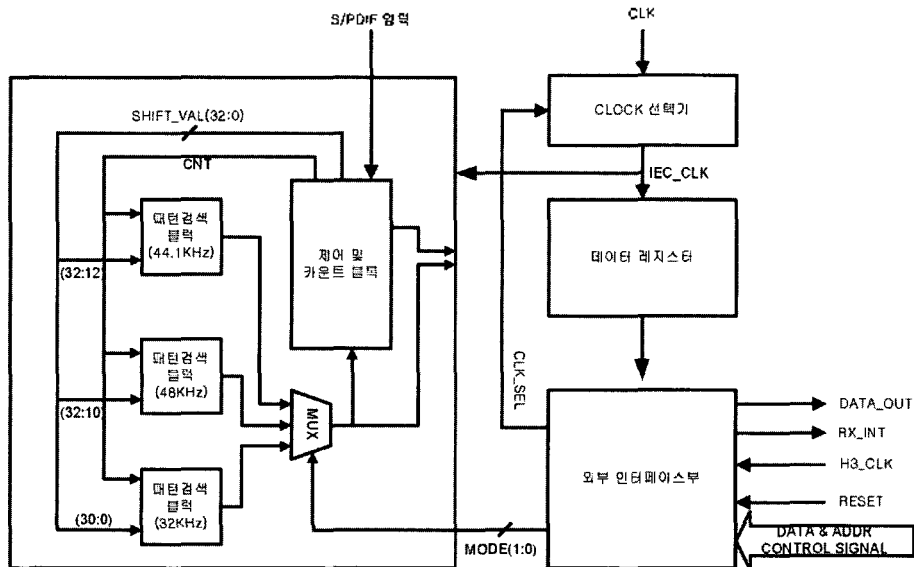


그림 5. S/PDIF 수신기의 블록 구성도  
Fig. 5. The block diagram of a designed S/PDIF receiver.

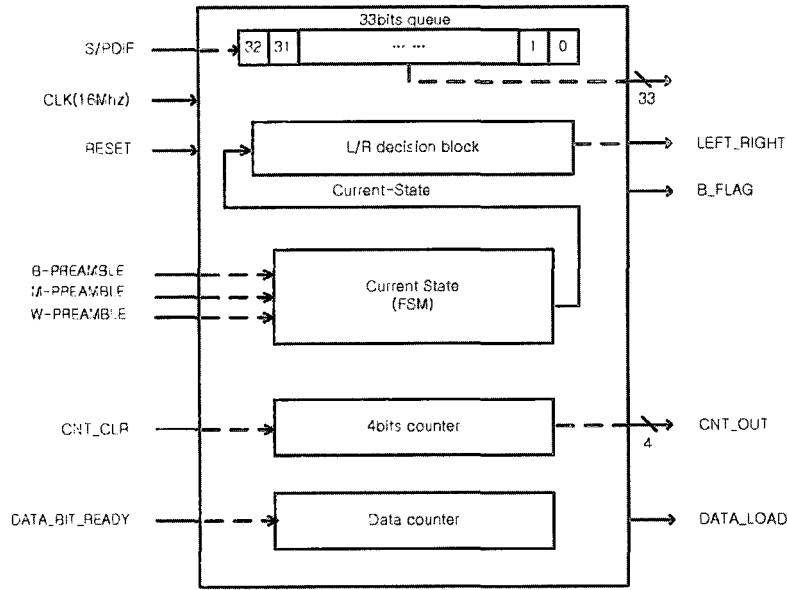


그림 6. 제어 및 카운트 블록도  
Fig. 6. The configuration of control and counter block.

터를 추출할 수 있게 하는 것이다. 16 MHz의 기준 클럭과 S/PDIF의 동기 문제는 현재 수신하고 있는 데이터의 종류가 표 1.의 포맷 중 프리엠블인 경우와 오디오 데이터인 경우로 나누어 처리하였다. 프리엠블을 수신하고 있는 경우에는 샘플링된 데이터의 선력을 적절하게 하고 샘플 데이터를 연속적으로 두 번 해석하는 과정을 거치며, 오디오 데이터를 수신하고 있는 경우에는 한 쌍의 데이터 필드를 추출한 후 카운터를 매번 클리어하는 과정을 통해서 S/PDIF 수신기가 16 MHz 클럭 신호와 S/PDIF 디지털 입력 신호 사이의 동기 어긋남의 정도에 관계없이 동작한다. S/PDIF 디지털 입력을 받은 후 3개의 독립적인 패턴 검색 회로에서 3가지의 데이터 전송 속도를 고려하여 신호를 해석하여 하나의 해석 결과를 선택한다.

3.2. 제안한 S/PDIF 수신기의 구성

제안하는 S/PDIF 수신기는 그림 5.와 같이 구성되며, 크게 S/PDIF 데이터를 샘플링하고 다양한 카운트를 하는 제어 및 카운트 블록과 기준 클럭 (16MHz) 를 이용해서 16MHz로 샘플링된 데이터에서 각 입력 패턴-44.1KHz, 48KHz, 32KHz-에 맞는 실제 데이터 정보를 뽑아내는 3가지의 패턴 검색 블록과 해석된 24비트의 데이터를 저장하는 데이터 레지스터와 임베디드 될 코어와의 인터페이스를 담당하는 외부 인터페이스부와 16 MHz 클럭 소스를 외부에서 받을지 임베디드 코어에서 받을지를 선택하는 클럭 선택기로 이루어진다. 이 중 S/PDIF 수신기

의 핵심이 되는 제어 및 카운트 블록과 3가지의 패턴 검색 블록에 대하여 구현 원리와 설계 과정을 자세히 설명한다.

3.3. 기능 블록의 설계 및 동작

3.3.1. 제어 및 카운트 블록의 설계

제어 및 카운트 블록에서는 주 클럭 신호 (16 MHz)를 이용해서 S/PDIF 입력 신호를 매 클럭 상승 에지에서 샘플링한다. 그림 6.은 제어 및 카운트 블록의 구성을 나타내고 있으며, 그 동작을 살펴보면, 먼저 주 클럭 신호 (16 MHz)를 이용하여 S/PDIF 신호를 샘플링한 후 샘플링된 값을 33비트의 큐의 MSB부터 차곡차곡 채운다. 또한 패턴 검색 블록에서 샘플링된 값에서 실제 디지털 값을 추출하기 위해 사용되는 기준 카운트 값을 생성한다. 프리엠블의 수신을 완료한 후 실제 데이터 값들을 받기 시작하는데, 이때 현재 몇 비트 까지의 데이터를 받

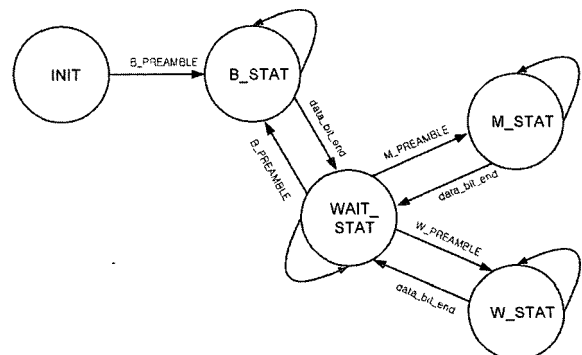


그림 7. Current state 블록 상태도  
Fig. 7. The state machine of current state block.

표 3. 패턴 검색 블록의 사용 정보

Table 3. The information for each patten search blocks.

	44.1 KHz 패턴 검색 블록	48 KHz 패턴 검색 블록	32KHz 패턴 검색 블록
S/PDIF 데이터 레이트 (샘플링 레이트)	5.6 MHz (44.1 KHz)	6.2 MHz (44.1 KHz)	4 MHz (32 KHz)
기준 클럭(16MHz)과의 속도비	2.857	2.581	4
프리앰블 해석을 위해 필요한 샘플의 비트수	23	21	33

고 있는가를 알려주는 데이터 카운트 값 또한 이 블록에서 생성된다. 현재 받고 있는 데이터의 채널 번호는 L/R 결정 블록 (L/R decision block)에서 결정된다. 마지막으로 현재의 수신 상태 (초기 상태, 대기 상태, B-state, M-state, W-state)를 결정하는 FSM (Current State 블록)을 포함하고 있다. 현재의 수신 상태를 나타내는 FSM (Current state 블록)은 그림 7.과 같이 이루어져 있다. 이 FSM 상태도에서 B\_PREAMBLE, M\_PREAMBLE, W\_PREAMBLE 신호는 패턴 검색 블록에서 만들어지는 신호로서 각 신호는 프리앰블이 수신 완료되었음을 나타내고 계속해서 들어오는 데이터 값 수신을 시작하라는 알람신호이다. 그림 7.의 상태도에서 각 스테이트 (B-state, M-state, W-state)는 각 프리앰블의 데이터를 수신 중이라는 것을 의미한다. 데이터 수신 완료 신호 (data\_bit\_end)는 표 1.의 디지털 입력 포맷 중 프리앰블을 제외한 나머지 데이터 28비트가 모두 받아들였다 것을 의미한다. 28비트의 데이터를 모두 수신 완료한 후에는 대기 상태 (wait\_stat)로 전환되는데 또다시 새로운 프리앰블이 모두 수신될 때까지 이 상태에서 머물게 된다.

### 3.3.2. 패턴 검색 블록의 설계

그림 5.에서 패턴 검색 블록은 S/PDIF 디지털 입력 신호의 샘플링 레이트에 따라 프리앰블과 데이터를 추출을 하는 역할을 한다. 패턴 검색 블록은 제어 및 카운트 블록으로부터 33비트의 큐의 값 중 상위 21비트~33비트만을 받아들이는데, 이는 16 MHz의 주 클럭과 해당 S/PDIF의 입력의 속도비에 따라 샘플 중 일부만으로 데이터를 추출할 수 있기 때문이다. 표 3.에 각 패턴 검색 블록에서 프리앰블을 추출하기 위해 제어 및 카운트 블록으로부터 받아 들이는 데이터의 비트수를 나타내었다.

#### 가. 프리앰블 찾기

II장에서 설명된 바와 같이 표 2.의 데이터 포맷 중 MSB에 해당되는 4비트의 프리앰블은 예외적으로 BMC 규칙을 따르지 않는다. 본 논문에서는 16 MHz의 입력 클럭과 S/PDIF의 입력 신호의 속도비를 이용해서 B, M, W preamble을 찾았다. S/PDIF 입력 신호와 기준 클럭 신호 (16 MHz)와의 속도비를 표 3.에 나타내었다. 표 3.에서 S/PDIF 데이터 레이트는 입력 S/PDIF 신호 자체의 속도를 나타내며, 샘플링 레이트는 S/PDIF 신호가

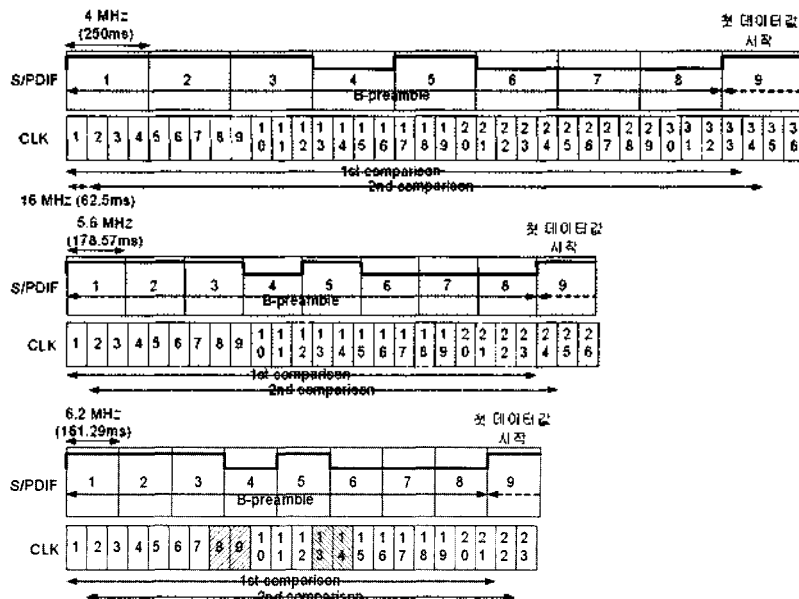


그림 8. S/PDIF(4 MHz, 5.6 MHz, 6.2 MHz) vs. 기준 클럭(16 MHz)의 시간축 비교

Fig. 8. The time axis comparison between S/PDIF(4 MHz, 5.6 MHz, 6.2 MHz) vs. basic clock(16 MHz).

인코딩될 때 사용된 샘플링 레이트를 나타낸다. 표 3.에서 기준 클럭 (16 MHz)과의 속도비는 S/PDIF 신호 자체의 속도와 기준 클럭 (16 MHz)의 속도비를 의미하며, 프리앰블 해석을 위해 필요한 샘플의 비트수는 그림 8.의 분석을 통해서 얻어진 값이다. 그림 8.은 3가지 경우의 S/PDIF 입력 신호와 기준 클럭에 대한 시간축 비교를 나타내었는데, 이의 분석을 통해서 프리앰블을 해석하기 위해 필요한 샘플의 비트수와 기준클럭으로 샘플링된 데이터에서 취해야 할 값을 찾을 수 있다. 그림 8.에서 프리앰블을 해석하기 위해 샘플링된 데이터에서 값을 취함에 있어 기본 방향은 프리앰블의 시작과 끝 부분에 위치한 샘플들은 가능한 많이 취해서 프리앰블 추출에 있어 신뢰성을 높이고, 프리앰블의 중간 부분은 최소한의 샘플을 취한다는 것이다. 그림 8.의 시간축에서 회색으로 채워져 있는 샘플은 프리앰블 판별을 위해 선택된 값들을 의미한다. 샘플의 선택은 위에서 제시한 샘플 선택의 기본 방향과 시행착오를 통해서 이루어졌으며, 그림 8.과 같이 샘플이 선택될 경우 기준 클럭과 S/PDIF 입력 신호의 동기가 어긋났을 경우 (기준 클럭의 시간축이 좌우로 쉬프트된 경우)라도 두 번의 판별 과정-2nd comparison-내에 프리앰블을 정확히 판별할 수 있으므로, 입력 신호와 기준클럭의 동기의 불일치는 문제가 되지 않는다.

아래 그림 9.는 S/PDIF 입력 신호의 속도가 5.6 MHz인 경우 그림 8.에서 결정된 샘플 선택 위치에 따라 프리앰블을 판별하는 예를 나타내고 있다. 첫 번째 그림의 B-프리앰블의 경우, 선택된 샘플의 값이 "11111010000" 일때 B-프리앰블로 판별된다. 그림 9.의 경우는 직전 셀 값이 '0' 인 경우에 해당하며 직전 셀 값이 '1' 인 경우는 그림 9.를 반대로 고려하면 된다.

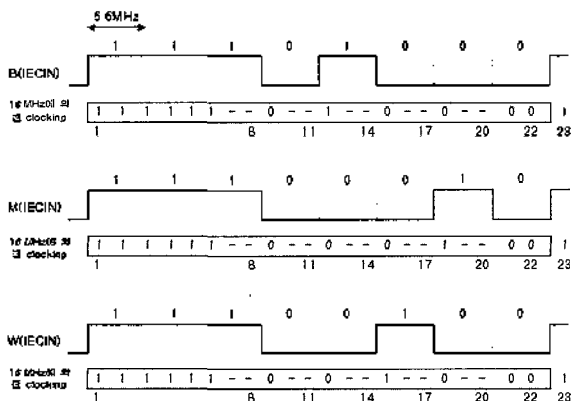


그림 9 . 5.6 MHz S/PDIF 입력의 프리앰블 추출  
Fig. 9. The preamble extraction of 5.6 MHz S/PDIF input.

나. 데이터 추출

프리앰블을 찾은 후에는 다음과 같은 방법으로 데이터 값들을 추출할 수 있다. S/PDIF 데이터는 BMC 코드 규칙을 따르므로 데이터 비트가 '0'이면 그림 10.의 a와 b는 서로 같은 level이 되며, 그렇지 않고 '1' 이면 a와 b는 서로 다른 level로 인코딩된다. 아래 그림 10.의 카운트 값 (cnt)은 제어 및 카운터 블록에서 생성되는 값으로, 프리앰블 수신이 완료되고 실제 데이터 값의 첫 입력이 시작될 때 카운트가 시작된다. 또한 카운트 값은 16 MHz의 클럭 신호에 따라 매 샘플링 타임마다 증가한다. 실제 데이터가 추출되는 과정을 S/PDIF 입력 신호 속도가 5.6 MHz인 그림 10.을 통해서 살펴보자. 제어 및 카운터 블록에서 샘플링이 이루어질 때마다 그림 10.의 23rd 위치에 현재 샘플링된 값이 채워지며, 직전에 샘플링 된 값은 1비트 쉬프트된다. 추출되는 데이터 값은 카운트 값(cnt)이 '2' 인 경우에 그림 10.에서와 같이 23rd 값과 21st 값의 XOR에 의해서 결정된다. 기준 클럭과 S/PDIF의 신호비가 2배에서 3배 사이 (2.857)의 값을 가지므로 2번의 샘플 타임 경과 후의 값(b)과 비교하는 것이 바람직하다. 한 쌍의 인코딩된 BMC 데이터 (a, b)를 수신한 후 다음 한쌍의 BMC 데이터 (c, d)의 수신을 시작하게 되는데, 여기서 BMC 코드의 특성을 이용한다. 인코딩된 데이터값 b와 c는 항상 다른 값을 가진다. 이는 bi-phase 코드의 성질이다. 카운트 값 (Cnt)이 '3' 보다 크거나 같은 경우에는 현재 샘플링 값 (23rd 값)과 직전 샘플링 값 (22nd 값)을 서로 비교하게 된다. 이때 두 값이 서로 다르면 새로운 한쌍의 데이터 필드 (c, d)가 시작되었음을 의미하고, 그렇지 않은 경우는 아직 새로운 데이터 필드가 시작되지 않았다는 것을 나타낸다. 그림 10.에서 보듯이 새로운 데이터 필드 (c, d)가 시작되면

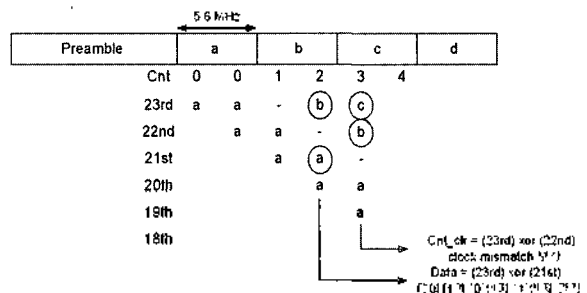


그림 10 . 5.6 MHz S/PDIF 입력의 데이터 값 추출  
Fig. 10. The data value extraction of 5.6 MHz S/PDIF input.

표 4. 데이터 값 추출 및 클럭 불일치 보상

Table 4. The extraction of data values and the compensation of clock mismatch.

	5.6 MHz S/PDIF	6.2 MHz S/PDIF	4 MHz S/PDIF	비고
데이터값 결정 (카운터 값)	23rd xor 21st (2)	21st xor 19th (2)	33rd xor 30th (3)	데이터 값이 '0'인지 '1'인지 결정
카운터 클리어 신호 (카운터 값)	23rd xor 22nd (3)	21st xor 20th (3)	33rd xor 32nd (5)	clock mismatch 보상

카운트 클리어 신호가 생성되어 새롭게 카운트가 시작된다. 새로운 데이터 필드 (c,d)의 추출은 데이터 필드 (a,b)와 동일한 방법으로 행해진다. 이러한 방법으로 56 비트의 인코딩된 S/PDIF 데이터 부분을 추출하게 된다. 한 쌍의 데이터 필드의 추출 완료 후 매번 카운트를 클리어하는 과정을 통해서 16 MHz 클럭 신호와 S/PDIF 디지털 입력 신호 사이의 동기 불일치에 따른 문제를 해결할 수 있다. S/PDIF 입력 신호 속도가 6.2 MHz, 4MHz인 경우에도 동일한 방법으로 데이터를 추출한다. S/PDIF 입력 신호의 속도에 따른 데이터 결정 시기 및 방법을 표 4.에 요약하였다.

### IV. 검증 결과

설계된 회로를 검증하기 위해서 그림 11.과 같이 인코딩된 S/PDIF 디지털 입력을 작성하여 텍스트 파일로 저장한 후 S/PDIF의 3가지 모드-5.6 MHz, 6.2 MHz, 4 MHz-로 입력을 받아 최종 추출 결과를 BMC로 인코딩되기 전의 원래 데이터 값 (original data value)와 비교하였다. II장 표 1.에서 나타난 32비트의 데이터는 그림 11.과 같이 BMC로 표현될 때 총 64비트로 인코딩 된다. 64비트 중 처음 8비트는 BMC 규칙을 따르지 않으며 표 2.에 정의된 형태로 인코딩되는 프리엠블 데이터이며, 그 외 56비트는 BMC로 인코딩된 28비트의 데이터에 해

당된다. S/PDIF 입력 신호의 3가지 형태 중 5.6 MHz의 속도 (44.1 KHz 샘플링)를 갖는 S/PDIF 신호의 시뮬레이션 결과를 그림 12.~그림 14.에서 나타내었다. 그림 12.는 44.1 KHz의 샘플링 레이트를 가지는 S/PDIF의 데이터에 대한 프리엠블 추출 과정 및 결과이다. 프리엠블 추출 완료 신호 (b\_preamble\_44k)는 b 프리엠블과 다음 데이터의 첫 번째 비트 값이 들어 왔음을 알리고, FSM (current state 블록)의 현재 상태는 b 스테이트로 변하며, 실제 S/PDIF 데이터 추출 과정이 시작된다. 그림 13.에서는 프리엠블 추출 후에 그림 11.의 연속된 BMC 입력을 받아서 28비트의 데이터 추출이 완료된 모습을 보여주고 있는데, 이를 알려주는 신호가 데이터 로드 (data\_load) 신호이다. 추출된 결과는 내부 데이터 레지스터에 저장되는데 그림 13.에서 최초로 수신된 28 비트 데이터 결과값 (data)이 "231F731h"임을 보여주고 있다. 이는 그림 11.의 입력 중 두 번째 줄의 데이터 (1st data)의 추출 결과이다. 그림 14.는 연속된 데이터 추출 결과를 중심으로 나타내었는데, 추출된 결과값 (data)는 그림 11.의 입력값과 동일함을 알 수 있다. 48 KHz, 32 KHz로 샘플링된 S/PDIF 신호의 검증도 동일한 방법으로 시뮬레이션을 통하여 검증하였다.

시뮬레이션을 통하여 설계개념을 확인한 후 설계의 신

S/PDIF 8, 7, 6, 5, 4, 3, 2, 1 (32비트 → Encoded 64비트)		Original data value(28 bits)
비트 번호	S/PDIF 8 이하 용도(56비트), S/C 코드	원래 데이터 값
00001110	00011100001110000111000011100001110000111000011100001100	B = 231F731h (1st)
11101000	10110010	W = 231F731h (2nd)
11100101	101010100110010010011001001001100100100110010011001001100	M = 2CE99CEh (3rd)
11100101	101010100110010010011001001001100100100110010011001001100	B = 231F731h (4th)
00010110	100110010010010010011001001001100100100110010010011001001100	W = 231F731h (5th)
00010110	100110010010010010011001001001100100100110010010011001001100	W = 231F731h (6th)
00011010	100110010010010010011001001001100100100110010010011001001100	M = 2CE99CEh (7th)
00011010	100110010010010010011001001001100100100110010010011001001100	W = 231F731h (8th)
11101001	100110010010010010011001001001100100100110010010011001001100	B = 0000000h (9th)
11101001	100110010010010010011001001001100100100110010010011001001100	W = 231F731h (10th)
11101001	100110010010010010011001001001100100100110010010011001001100	B = 231F731h (11th)
11101001	100110010010010010011001001001100100100110010010011001001100	M = 2CE99CEh (12th)
11101001	100110010010010010011001001001100100100110010010011001001100	W = 231F731h (13th)
11101001	100110010010010010011001001001100100100110010010011001001100	B = 231F731h (14th)
11101001	100110010010010010011001001001100100100110010010011001001100	W = 231F731h (15th)

그림 11. S/PDIF 입력 파일의 예  
Fig. 11. The example of S/PDIF stimulus files.

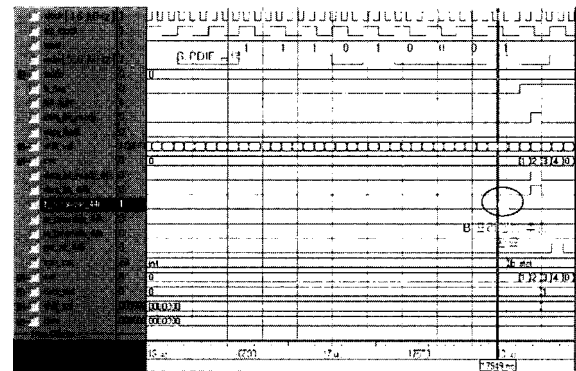


그림 12. 44.1 KHz S/PDIF 입력에 대한 프리엠블 추출 결과  
Fig. 12. The result of a preamble extraction for 44.1 KHz S/PDIF input.



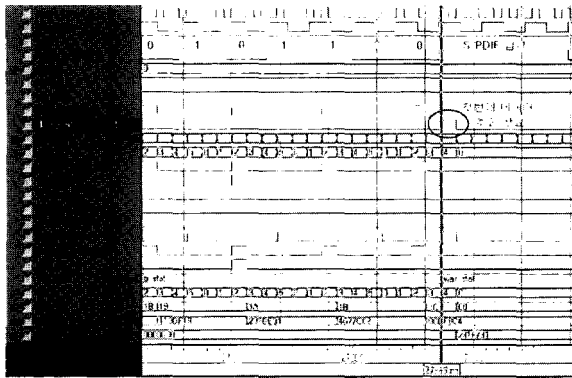


그림 13 . 44.1 KHz S/PDIF 입력에 대한 최초 데이터 추출  
 Fig. 13. The extraction of the first data for 44.1 KHz S/PDIF input.

회상을 검증하기 위하여 FPGA를 이용하여 80C51XA 호환 16비트 마이크로 컨트롤러에[9] 임베디드 IP 형태로 적용하였다. 대상 컨트롤러는 16 MHz의 동작 주파수를 가지며 32개의 명령어를 가지는 16비트 CISC 컨트롤러이다. S/PDIF 수신기의 제어 레지스터를 타겟 컨트롤러의 SFR (Special Function Register) 영역의 일부로 할당하는 방법으로 제안된 수신기를 제어하였다. 그림 15. 는 제안된 수신기를 16비트 마이크로 컨트롤러에 적용한 후의 시스템 구성을 나타낸다. 시스템의 구성은 크게 2개의 FPGA (XC1V1000-560C)로 이루어져 있으며, 우측 상단의 FPGA는 16비트 마이크로 컨트롤러와 S/PDIF 수신기를 포함하고 있으며, 좌측 하단의 FPGA는 PC 및 코덱 관련 인터페이스 회로가 포함하고 있다. FPGA로 구현 후 총 67,152의 게이트 수를 보였으며, 제안된 수신기가 차지하는 게이트 수는 5,230 게이트이다. FPGA 수준에서 S/PDIF 동작의 검증은 그림 15.와 같이 각기 다른 전송 속도를 가지는 오디오 장치 (CD, DCC, DAT)의 S/PDIF 출력을 구현된 보드 입력으로 받아 스피커를

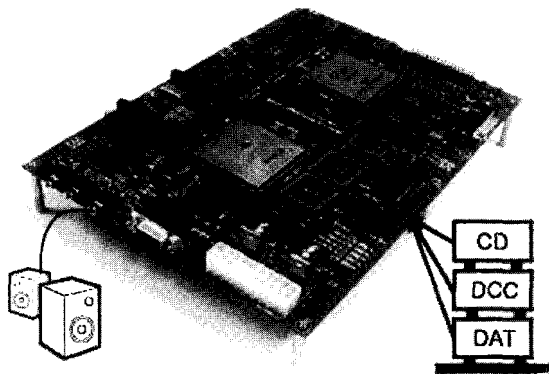


그림 15. 16비트 마이크로 컨트롤러에 임베디드된 S/PDIF 수신기의 시스템 구성  
 Fig. 15. The system configuration after the S/PDIF receiver was embedded to a 16 bits micro controller.

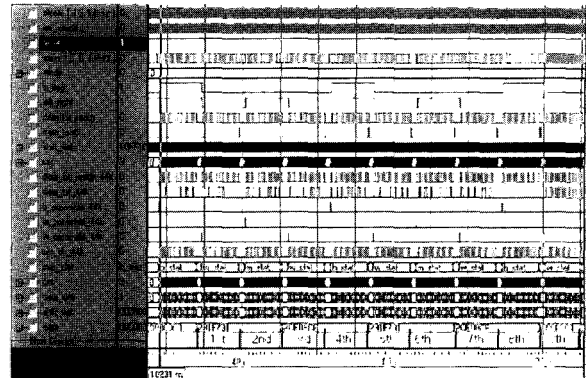


그림 14 . 44.1 KHz S/PDIF 입력에 대한 데이터 추출 결과  
 Fig. 14. The result of extraction for 44.1 KHz S/PDIF input.

통해서 들어보는 청음 테스트를 실시하였다. 다양한 장르의 곡에 대한 장시간의 오디오 청취를 통해서 본 논문에서 제안한 PLL이 없는 수신기가 실제 SOC 환경에서 올바르게 동작할 수 있음을 확인하였다.

### V. 결론

본 논문에서는 오디오 신호가 비교적 깨끗한 경우에 PLL 없이 S/PDIF 프로토콜을 구현할 수 있는 방법을 제안하고 다양한 종류의 오디오 입력에 대해 추출 결과값을 검증하였다. 제안하는 수신기는 PLL 회로 없이도 다양한 샘플링 레이트의 오디오 신호를 추출할 수 있다. 제안된 S/PDIF 수신기는 S/PDIF 디지털 신호에 포함되어 있는 클럭 추출과 입력 신호와의 동기를 맞추는 역할을 하는 PLL 회로의 기능을 16 MHz의 기준 클럭과 입력 신호의 데이터 레이트 비만을 이용하여 구현하였다. S/PDIF 데이터의 3가지 샘플링 레이트와 -32 KHz, 44.1 KHz, 48 KHz- 16 MHz 의 기준 클럭의 비를 미리 계산해 두고 기준 클럭을 이용해서 입력 데이터를 샘플링 한 후 계산된 비를 참조해서 각 모드에 대해 적절하게 샘플링 데이터를 선택하는 방법으로 데이터를 추출하였으며, 입력데이터와 기준 클럭이 정수배가 되지 않기 때문에 발생하는 동기 불일치 문제는 샘플 데이터를 연속적으로 두 번 해석하거나 (프리앰블인 경우) 한 쌍의 데이터 필드를 추출한 후 카운터를 매번 클리어하는 방법 (데이터인 경우)을 통하여 해결하였다. 제안하는 수신기의 동작을 시뮬레이션을 통하여 확인한 후 FPGA를 이용하여 16비트 마이크로컨트롤러에 IP형태로 적용하였다. 3가지의 각기 다른 전송 속도를 가지는 오디오 장

치 (CD, DCC, DAT)의 S/PDIF 출력을 구현된 보드 입력으로 받아 장시간 동안 음악을 정상적으로 재생하는 것을 확인함으로써 제안한 PLL이 없는 S/PDIF 수신기가 IP 형태로 삽입된 후 실제 하드웨어 환경에서도 올바르게 동작할 수 있음 확인하였다. 본 논문에서의 제안한 수신기는 오디오 신호 송신장치로부터의 디지털 데이터의 입력에 노이즈가 없다는 가정에서 출발하여 입력 데이터 자체에 노이즈가 있을 경우의 성능은 PLL을 내장한 칩보다는 성능이 좋지는 않지만, 입력 데이터의 노이즈가 적은 경우나 디지털 프로세서 칩의 IP로서 내장되어 사용될 때는 PLL을 대체하여 사용될 수 있을 것이다. 본 논문에서 제안하는 방식을 이용하면 아날로그 부분을 제거할 수 있어 칩 면적을 줄일 수 있을 뿐만 아니라, 아날로그 회로와 디지털회로가 하나의 칩에 집적하는 경우에 발생하는 노이즈 문제를 해결할 수 있어 레이아웃에 소요되는 시간을 단축할 수 있으며 칩의 성공률을 높일 수 있다.

### 감사의 글

본 연구는 산업자원부의 지역혁신인력양성사업, 정보통신부의 IT SoC 핵심설계인력양성 사업, 산업자원부의 반도체설계인력양성사업의 지원으로 수행되었습니다.

### 참고 문헌

1. YAMAHA CORPORATION, YSS922 CATALOG LSI-4SS932A1, 2000.2
2. YAMAHA CORPORATION, YSS932 CATALOG LSI-4SS932A0, 2001.5
3. Philips Electronics, UDA1350 IEC60958 audio DAC, 2001 Mar 27, <http://www.semiconductors.philips.com>
4. Philips Electronics, UDA1351H 96kHz IEC958 audio DAC, 2000 Jul 27, <http://www.semiconductors.philips.com>
5. Crystal semiconductor Corporation, CS8411/CS8412 Digital Audio Interface Receiver Data Sheet, Oct 1998
6. Crystal semiconductor Corporation, CS8413/CS8414 96 kHz Digital Audio Receiver Data Sheet, Oct 1998
7. Crystal semiconductor product division, Cirrus Logic Inc, AN22REV2, pp.1-3
8. 우종식 외 6, "음악 속도변환이 가능한 SOC형 오디오 프로세서 개발", 정보통신산업기술개발사업, 보이스반도체, 2003. 2
9. 김경수, 박주성, "Design of CISC Micro controller and Study on Verification Step", 대한전자공학회, 제41권 SD편 제6호, pp.7180, 2004. 6

### 저자 이력

• 김 경 수 (Kyoung-Soo Kim)



1998년 2월 부산대학교 전자공학과 졸업 (공학사)  
 2000년 2월 부산대학교 전자공학과 졸업 (공학석사)  
 2000년 3월 현재 부산대학교 전자공학과 박사과정  
 ※주관심분야: 반도체, DSP 설계, ASIC 설계, 검증 자동화

• 박 주 성 (Ju-Sung Park)



1976년 2월 부산대학교 전자공학과 졸업 (공학사)  
 1978년 6월 KAIST 졸업 (공학석사)  
 1978년 1985년 ETRI 실장  
 1989년 6월 Univ. of Florida 전자공학과 박사 취득  
 1991년 현재 부산대학교 전자공학과 교수  
 1998년 현재 부산대 IDEC 센터장  
 ※주관심분야: DSP 설계, ASIC 설계, 반도체 소자 모델링, 음성/사운드 신호 처리 및 구현, SOC 설계

• 김 석 찬 (Suk-Chan Kim)



1993년 2월 부산대학교 전자공학과 공학사  
 1995년 2월 한국과학기술원 전자공학과 공학석사  
 2000년 2월 한국과학기술원 전자공학과 공학박사  
 2001년 8월 2002년 2월 Post Doctoral Researcher, Dept. of EE, Lehigh University  
 2002년 3 현재 부산대학교 전자공학과 조교수  
 ※주관심분야: 무선 휴대 통신, 신호처리, 체적 코딩