

엔진 없이 전기로 구동되는 자동차의 가상 엔진 음 구현을 위한 음원합성 IC에 관한 연구

The study of sound source synthesis IC to realize the virtual engine sound of a car powered by electricity without an engine

구재을,[†] 홍재규,¹ 송영욱,² 이기창²

(Jae-Eul Koo,[†] Jae-Gyu Hong,¹ Young-Woog Song,² and Gi-Chang Lee²)

¹아이티아이, ²Tunelt

(Received August 2, 2021; revised September 10, 2021; accepted September 23, 2021)

초 록: 본 연구는 기존의 엔진 없이 전기로 운행되는 자동차에 가상의 엔진소리를 구현해주는 반도체 에 관한 연구로, 운전자의 욕구 및 보행자의 안전을 지키기 위해 Adaptive Difference PCM(ADPCM) 방식과 Frequency modulation 방식을 조합하여 생생한 엔진음을 구현하였다. 또한 미디를 응용하는 전자음 합성 알고리즘을 제안하여 엔진음 합성방법, 엔진음 발생 시스템의 구성 모델을 제시한다. 본 연구는 운전자와 보행자를 모두 만족시키기 위해 캔 통신을 이용하여 운전자의 운전 습관에 따라 전해지는 엔진회전량, 차량 속도, 가속 페달 눌림 량, 토크 등의 정보를 받은 후, 미리 설정된 적절한 파라미터에 따라 주파수 변조를 위한 인터폴레이션, 정보량 축소를 위해 ADPCM 알고리즘, 엔진음을 쉽게 만들기 위한 미디 포맷을 이용하여 시스템과 운전자의 의도를 정확히 반영할 수 있는 인터랙션 알고리즘을 구현하였다.
핵심용어: 가상 엔진 음, 인터폴레이션, 주파수 변조, 미디, 파이프라인 구조

ABSTRACT: This study is a study on System On Chip (SOC) that implements virtual engine sound in electric vehicles without engines, and realizes vivid engine sound by combining Adaptive Difference PCM (ADPCM) method and frequency modulation method for satisfaction of driver's needs and safety of pedestrians. In addition, by proposing an electronic sound synthesis algorithm applying Musical Instrument Digital Interface (MIDI), an engine sound synthesis method and a constitutive model of an engine sound generation system are presented. In order to satisfy both drivers and pedestrians, this study uses Controller Area Network (CAN) communication to receive information such as Revolution Per Minute (RPM), vehicle speed, accelerator pedal depressed amount, torque, etc., transmitted according to the driver's driving habits, and then modulates the frequency according to the appropriate preset parameters. We implemented an interaction algorithm that accurately reflects the intention of the system and driver by using interpolation for the system, ADPCM algorithm for reducing the amount of information, and MIDI format information for making engine sound easier.

Keywords: Virtual engine sound, Interpolation, Frequency modulation, Musical Instrument Digital Interface (MIDI), Pipe-lined architecture

PACS numbers: 43.55.Ka, 43.60.Ac, 43.60.Mn

I. 서 론

4차 산업 시대는 이제 정보화 지식 사회를 지나 스마트 시대에 살고 있다. 반도체 기술의 비약적인 발

전으로 다양한 IT 플랫폼 및 초고속 기기가 속속 등장하여 실시간 멀티 커뮤니케이션 기능이 가능해져 사람과 사람, 사람과 기술 간의 인터랙션이 그 어느 때보다 중요해 졌다. 또, 단순하게 소비자 니즈를 충

[†]Corresponding author: Jae-Eul Koo (koohwass@naver.com)

ITI Electronics, STARWITH #904, 38, Heungan-daero 427ben-gil, Dongan-gu, Gyeonggi-do 14059, Republic of Korea

(Tel: 82-70-8246-7710, Fax: 82-31-471-2202)



Copyright©2021 The Acoustical Society of Korea. This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

족하는 수준을 넘어 감성적 접근과 디자인의 역할이 새롭게 조명 받고 있다. 자동차도 이러한 시대 변화와 발전에 부응하기 위하여 자동차 본체 자체가 스마트해지고 있을 뿐만 아니라 자동차에 탑재되는 모든 편의사양들도 점점 스마트하게 진화하고 있다. 본 연구는 보행자의 안전과 운전자의 즐거움을 동시에 충족할 수 있는 멋진 가상의 엔진 소리를 구현하기 위한 System On Chip(SOC) 기술에 관한 것으로 파이프라인 Harvard architecture와 Adaptive Difference PCM(ADPCM) 알고리즘을 제안하였으며 가상 엔진음 제작의 편의성을 위하여 본체와의 통신은 Controller Area Network(CAN), 소리는 미디(Musical Instrument Digital Interface, MIDI) 포맷을 응용하여 기존 기술의 음질 한계를 극복하였다. 엔진음을 만드는 기

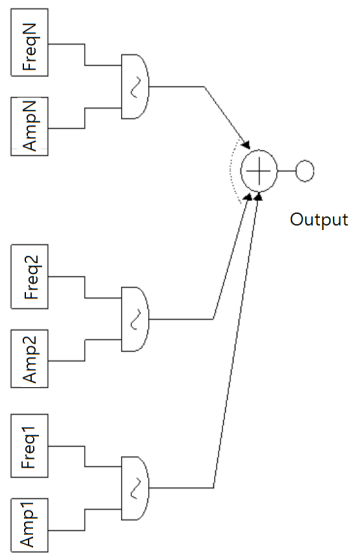


Fig. 1. Additive synthesis.

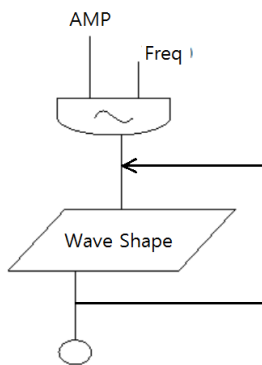


Fig. 2. Wave shaping diagram using multiple sine table.

본 기술은 가산 합성법/웨이브 셰이핑 합성법/왜곡을 이용한 합성법 등을 제안 한다. Figs. 1~3은 본 연구에서 이용한 블록도 이다.

가장 중요한 웨이브 셰이핑(Wave-shaping)은 왜곡 음원 합성 방식과 유사한 방법으로 FM과 마찬가지로 복잡한 엔진음을 구현하는데 있어서 Additive synthesis에 비해 좀 더 효율적인 방법이다. 하지만 웨이브 셰이핑에 의한 방법은 대역 제한된 스펙트럼을 발생시킬 수 있다는 점에서 Frequency Modulation(FM)과 다르다. 웨이브 셰이핑에 의해 발생된 스펙트럼은 음의 크기에 따라 달라지게 되는데, 이러한 변화는 급가속의 엔진음 스펙트럼 특성에 대응되는 것으로 전통적인 가솔린 엔진음과 닮은 엔진 음색을 효율적으로 발생시킬 수 있다. 특히 저음/저속의 엔진음을 발생시킬 때 특히 그러하다. 웨이브 셰이핑은 원래의 파형과 다른 파형을 얻기 위하여 원래 엔진음의 크기를 왜곡시키는 것이다.^[1] 대부분 왜곡은 신호에 악영향을 끼치지만 조심스럽게 제어되는 신호의 왜곡은 젊은 취향의 광범위한 엔진 음색을 만들어 낼 수 있다. FM synthesis와 마찬가지로 웨이브 셰이핑 역시 스펙트럼의 연속적인 제어가 가능하며 시간에 따라 변화하는 변화무쌍한 스펙트럼을 만들 수 있다.

기본적인 웨이브 셰이핑의 개략적인 전체적인 신호의 흐름도를 Fig. 2에 나타내었다. 그림에서 중심이 되는 요소는 웨이브셰이퍼 또는 비선형 신호 발생기라고 불리우며 신호가 이를 통과할 때 왜곡이 일어나게 된다. 이 비선형 신호발생기가 존재하므로 입력의 크기를 증가함에 따라 출력의 파형이 변화하게 된다. 이와 같이 파형이 변화하게 되므로 그의 스

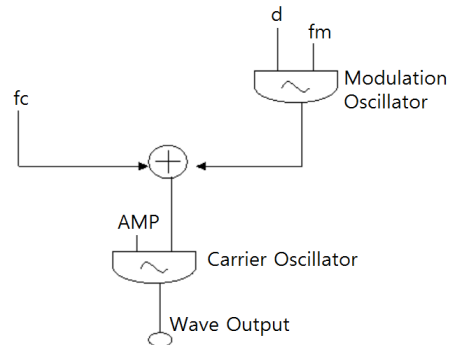


Fig. 3. Synthesis by distortion technique.

펙트럼 역시 변화하게 되고 배음들의 수와 크기가 증가하게 되어 사인파와 같은 간단한 파형으로부터 선형 인터플레이션(내삽법) 과 5차 계수를 이용한 비선형 인터플레이션을 이용하여 풍부한 스펙트럼을 갖는 파형을 얻을 수 있다.^[2]

II. 본 론

2.1 내부 H/W 구성 및 타이밍도

본 연구에서 제안한 엔진 음 합성 IC 블록도는 아래 Fig. 4와 같다. 엔진 음이 저장되어 있는 Sound MEM에는 운전자의 취향에 따라 선택할 수 있는 다양한 엔진 음을 저장할 수 있으며, 저가의 음원 모듈에서는 가장 취향에 맞는 엔진 음을 설정하여 IC 내부에 내장할 수도 있다. 본 연구에서는 다양한 활용 가능성을 두기 위하여 외부에 두는 것으로 설정하였다.

가장 중요한 부분인 Wave generator 부분을 살펴보면, 다양한 비트 레이트로 압축 인코딩 되어 저장된 데이터를 복원하기 위한 디코딩부, 주파수를 결정하는 인터플레이션 블록과 이를 반복적으로 수행하는 신호처리의 제어를 위해 어드레스 버스, 데이터 버스, 제어 버스 및 계수가 저장되어 있는 버퍼들이 있고, 엔진 음 특성상 선형 반복적인 웨이브의 재현을 위한 제어 동작 시 필요한 24 MHz 파형과 SIO에 필요한 1 MHz 펄스파형 등을 발생시키는 clock 발생부, 이 시스템의 reset 및 halt를 위한 로직들로 구성되어 있다. 엑셀레이터가 눌러 Revolution Per Minute

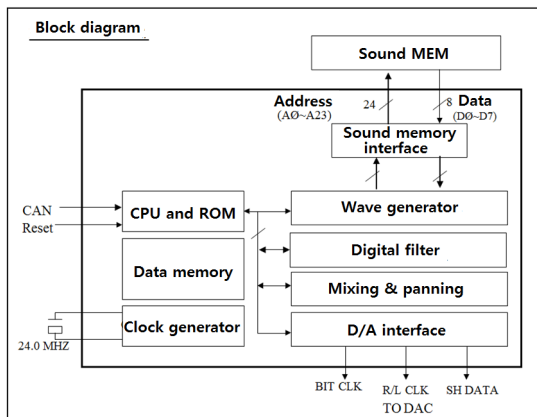


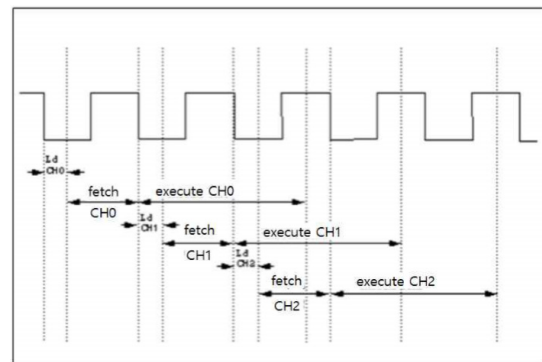
Fig. 4. IC Block diagram.

(RPM)이 올라가기 시작했을 때 디코더에서 RPM의 고속, 중속, 저속 값에 따라 계산되어서 온 정보인 테이블에 저장된 계수를 받아 프로그램 메모리에서 전달된 선형 혹은 비선형 인터플레이션 계수의 피치 값을 보정하거나 그대로 이용하여 1차의 선형 방정식과 3차의 다항식 비선형 방정식에 의해 해당 RPM에 맞는 엔진 음을 발생 시켜준다.

아래 Fig. 5는 하나의 로직으로 전체 16 채널의 엔진 음을 구현하기 위한 중첩적인 구조를 나타낸 것이다.

아래 Fig. 6은 급가속시의 엔진음을 구현하기 위한 엔벨롭의 대표적인 그래프이다.

기존의 가솔린 엔진음은 어택과 디케이 부분의 파형은 급격히 변하는 성분이 많이 포함되어 매 주기의 파형이 불규칙하고 변화가 심하다. 반면 디케이를 지나면서 써스테인 부분으로 들어서면 파형은 안정 해져서 각주기의 파형이 거의 같은 모양을 갖게 된다. 따라서 이후 뒷부분의 파형을 전부 사용하는 대신 일정한 길이만큼의 파형만을 사용해서, 그 부분을 계속 반복함으로써 데이터 양을 줄일 수 있다.



Command execution structure

Fig. 5. Pipe-lined architecture of Virtual Engine Sound System (VESS) DSP.

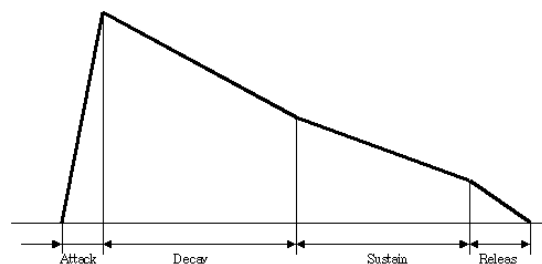


Fig. 6. Sound envelop diagram.

반복 파형을 얻기 위해 먼저 녹음된 엔진 음 파형을 정규화된 파형과 엔벨롭으로 나눈다. 정규화된 파형으로부터 어택 파형을 얻는다. 다음 Eq. (1)으로부터는 루핑용 반복 파형의 길이를 추출한다.

$$E_{\iota} = \sum_{n=0}^N (\chi(n) - \chi(n+\iota))^2, \iota = 1, \dots, L, \quad (1)$$

여기서

N ; compare length

L ; interval limit

2.2 엔진음 시뮬레이션 결과

본 연구에서 제안한 IC의 알고리즘을 시뮬레이션 한 결과는 아래 Fig. 7과 같다. 먼저 위 정규화된 엔진음의 원음 주기 별 스케일 계수를 분할 crossfading loop 방식에 의해 생성된 데이터와 곱하면 반복음의 시작과 끝에서 생기는 왜곡음을 현저히 낮출 수 있어 매우 중요하다.^[3]

아래 Fig. 7은 압축된 데이터를 디코딩 하면서 Crossfading 한 웨이브의 변화 과정을 시뮬레이션 한 결과를 보인 것이다. 데이터 복원을 위한 정확한 스케일 계수를 찾을 때는 연속적인 주기의 파워를 비교하는 방법과 단순 크기를 비교하는 방식이 있는데 파워를 이용하는 방식이 전체적인 음량 비교에 더욱 효과적이다. crossfading 할 때 가장 최적의 주기를 찾는 것이 중요하며 하나의 주기를 이용할 것인지 여러 개의 주기를 선택 이용할 것인지는 엔진음의 종류에 따라 모두 다르다.

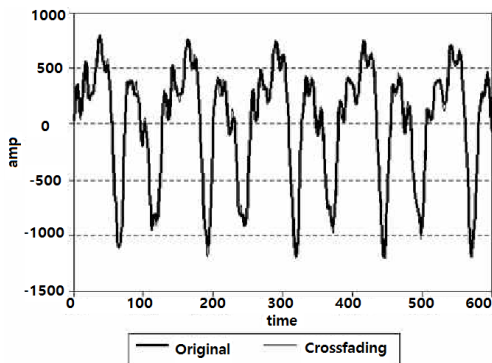


Fig. 7. Comparison original with decoding.

Fig. 7은 원 엔진 음 데이터와 인접 정보의 차를 ADPCM으로 압축한 데이터를 디코딩 하여 재생하였다.

Fig. 8은 현재 출력과 다음 번 Lookup 테이블의 출력을 합산하여 다음 출력을 만드는 디코더이다. 비트 레이트를 4 bit 에서 8 bit 까지 가변 할 수 있다.

엔진음의 변화 과정 전체를 녹음 및 샘플링 하여 메모리에 저장하는 것은 불가능하고 운전자의 습관과 RPM 변화를 인지하여 원음이 가지고 있던 주파수와 크기의 변화를 아래 Eqs. (2)와 (3)의 간단한 수식에 의하여 피치 시프팅을 이용한 방법으로 원음을 변조시켜 음색의 변화를 다양하게 만들 수 있다. 고정된 계수에 의한 피치 시프팅한 결과를 위 Fig. 9에 나타내었다. 인터폴레이션 알고리즘으로는 간단한 선형 방정식으로 표시되는 $y = ax$ 함수를 대부분 사용하였고, 고주파가 많이 포함된 엔진음은 데이터를 더 줄이는 서브샘플링 개념을 적용하였다.

$$y = ax + b. \quad (2)$$

$$y = cx^3 + dx + e. \quad (3)$$

Eq. (2)는 계수 a, b에 의해 간단한 선형방정식에 의해

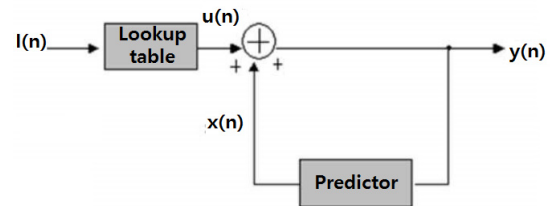


Fig. 8. Decoder for data compression.

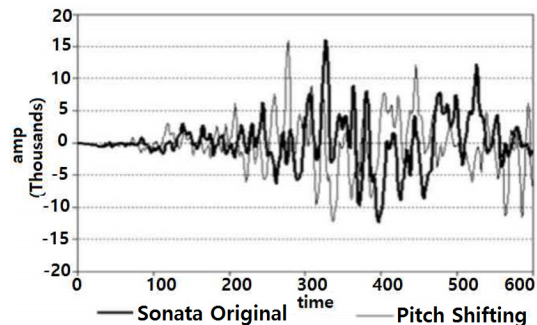


Fig. 9. Original and pitch shifting sound.

두 지점간의 사이 값을 구하는 것이고 Eq. (3)은 좀 더 정확한 사이 값을 위해서 계수를 3개 사용하는 수식이다.

본 연구에서는 비선형적으로 표시되는 3차원 함수를 사용하였다. 데이터의 서브샘플링과 비선형 인터플레이션을 적절히 이용하면 최대 1/24까지 데이터를 축소할 수 있고, 엔진 음 편집과정에서 RPM 별 엔진 음 조절시 Pitch 변화를 적절한 구간에서 응용하여 다양한 성분의 주파수 생성과 좀 더 실감나는 자연의 엔진 음을 구현 할 수 있다. 계수들의 전체 비트수는 16 Bit으로 충분하며 소수점은 Q14 형식을 적용하였다.^[4]

아래 Fig. 10은 채널 한 개를 On 하기 위한 각종 계수 파라미터를 모아 놓은 테이블이다. 메인 CPU로부터 버퍼 메모리로 넘어온 Layer-id + 차량 엑셀 세기 정보의 16 bit 정보를 받아서 Layer ON시에는 채널을 할당해 주고 미리 저장된 인터플레이션 계수 및 DMAC의 타이밍 제어를 통해 소리를 재생해 주며 레이어 OFF시에는 채널을 회수한다. 소리가 저장된 메모리와 테이블이 저장된 메모리, 프로그램 메모리의 세 개 요소가 있다. 소리저장 메모리에는 ADPCM 방식 소리 정보가 들어 있으며 오직 DMAC만이 액세스 가능하다. 테이블 메모리에는 프로그램 수행 과정에서 필요한 각 엔진음 정보 및 초기 시작 어드레스 등이 들어 있다. 이 메모리는 최종 사운드 롬에 내장시켰다. 각 채널 할당은 스택 오퍼레이션과 유사

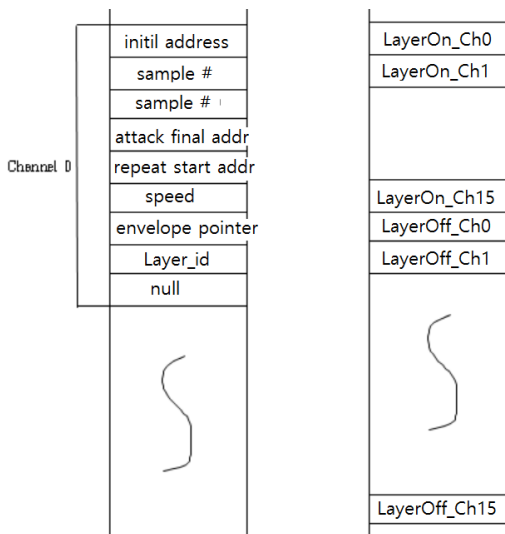


Fig. 10. Sequencing of coeff memory.

하게 동작한다. 즉 16 채널 중 어느 한 채널을 사용하고자 할 때는 스택 포인터가 가리키는 내용을 읽어 내면서 일률적으로 증가 시키고, 사용했던 채널을 회수할 때는 스택 포인터를 미리 감소 시키면서 다시 저장한다. 16 채널이 모두 사용되어진 상태에서 새로운 채널이 요구될 때는 맨 처음 시작하여 사용한 채널로 재사용한다.

2.3 알고리즘 수행 하드웨어 구성

본 연구의 시스템 하드웨어는 Fig. 11로 표현된다. 미디는 전자음악에서 사용하는 악기간의 통신 프로토콜로 본 연구에서는 개발자가 사용하기 편리하게 미디를 자동차의 종류로 대응시켜 본 포맷을 엔진 음 구현에 그대로 사용 하였다.^[5]

Fig. 11은 전체 엔진음 발생 시스템은 외부에서 미디나 웨이브 데이터를 받을 수 있는 미디 버퍼와 웨이브 버퍼가 각각 있다. 외부명령해석회로는 외부 호스트에서 써주는 데이터가 언제 시작되는지, 언제 끝나는지, 버퍼 메모리를 미디 버퍼(Pre-Setting 엔진 음)와 웨이브 버퍼(Down-Loading 엔진 음)로 꼭 나누어야 하는지, 버퍼를 나누지 않고 한가지로만 즉, 축약된 미디 데이터만 받을 것인지 웨이브 데이터만 받아도 되는지를 즉, 미디나 웨이브 중 한가지로만 사용해도 되는지를 결정하고, 수신된 데이터가 특정 시간 동안 없으니 슬립모드로 가도 되는지, 슬립모드에서 깨어나려면 어떤 조건을 주어야 하는지, 웨이브 데이터 이면 자동구간반복 기능이 있는지, 정

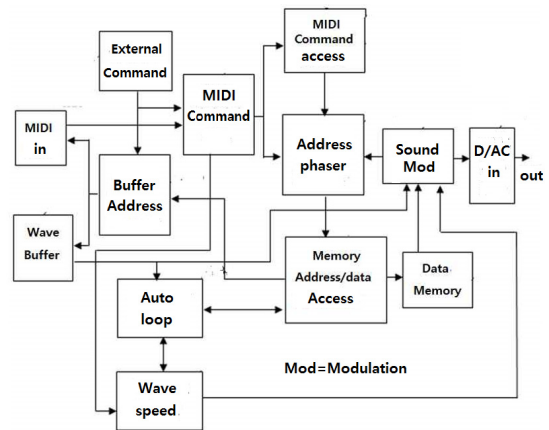


Fig. 11. Overall operation block diagram.

상적인 웨이브 이면 속도변환이 필요한지 필요 없는 지 등을 결정하는 중개자 역할을 한다.⁶⁾ 미디명령해석회로는 외부명령해석회로(씨퀵서)를 통하여 명령의 시작과 끝을 판독하여 시작을 알리는 명령이면 외부 버퍼를 읽는다. 이때 버퍼의 내용에 따라 엔진 음의 시작과 끝을 판단하고 시작일 경우에는 내부 DSP 기능부분에 적당한 채널을 할당하여 소리의 시작을 알리고 소리가 발생하는 도중에 소리의 혼 변조가 필요할 때는 미디에서 지시하는 명령에 의하여 해당 채널의 변조회로에 정보를 전달해 준다. 미디명령전달회로는 미디명령해석회로가 미디명령을 해석해서 적당한 데이터를 소리발생부의 하나인 어드레스 페이지 변조회로에 전달할 때 해석된 명령에 따라 정해진 순서대로 필요한 정보를 써넣는 회로로서 크기, 주파수, 할당채널번호, 소리의 길이 등의 정보를 전달해 준다. 이때 하나의 명령의 전달이 아직 수행이 안 됐을 때 다음명령의 통제를 할 수 있는 통제회로가 그 기능을 해준다. 웨이브속도 변환회로는 원하는 속도로 변환된 웨이브 신호를 최종 데이터 변조회로에 전달해주어 각종 데이터메모리를 통하여 들어온 데이터와 최종 혼합 변조를 할 수 있도록 해준다. 데이터 메모리는 엔진 음색 데이터 메모리, 음 크기 조절 데이터 메모리, 음 변조 데이터 메모리 등을 포함한다. 만일 세 가지의 데이터(엔진 음색, 크기 조절, 음 변조) 중 두개 이상이 동일한 싸이클에 동시에 읽으려고 할 때는 엔진 음색 데이터 메모리에 최우선권을 주고 다음 두가지(음 크기, 음 변조)의 우선권은 임의로 한다. 다중 화음제어회로는 정상적인 내부 데이터 메모리를 사용하는 미디음과 자동구간반복메모리를 사용하는 미디 음, 속도변화가 발생된 웨이브를 통합적으로 동시에 소리를 만들 수 있도록 음을 혼합하는 동시발생회로이다.

Fig. 12는 본 연구의 최종 결과인 SoC의 레이아웃 구조도이다.

1. Controller part : MIDI Rx / Tx, I2S / I2C
2. Controller memory part
3. Sound memory1 : Engine sound parameter 1
4. Sound memory address controller
5. Sound memory2 : Engine sound parameter 2

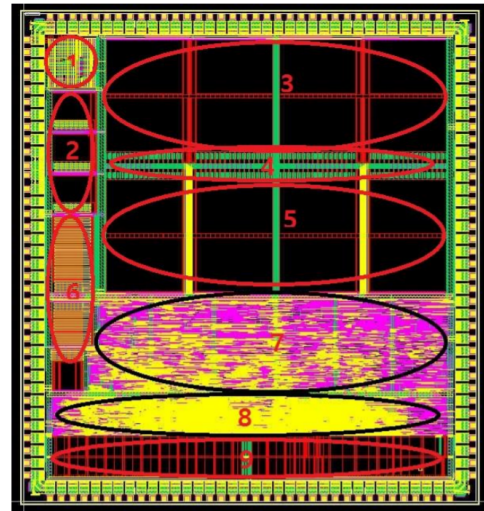


Fig. 12. (Color available online) Layout diagram of SoC.

6. Sub memory direct access part
7. DSP engine 1 : High definition filter / Env cal block
8. DSP engine 2 : Modulation / Time varying filter
9. DSP Parameter storage RAM : 16 / 32 / 8 Bit

III. 결론 및 향후 과제

본 논문은 엔진이 없는 전기차의 가상 엔진 음을 만드는 음원 IC에 관한 연구로 최종 16개의 엔진음을 동시에 발생시킬 수 있는 구조로 되어 있고, 소리의 생성을 위하여 합성음이 섞이지 않은 일반 가솔린 자동차에서 발생하는 음을 전자적인 방식으로 재생하는 기능과 일반 웨이브 혹은 미디 신호에 특정 효과음(sound modification / editing)정보를 오디오 정보와 섞어서 처리해 줄 수 있는 기능을 갖고 있다. 또한 전자 엔진 음 합성을 간단히 하기 위해 외부 중앙처리 CPU와의 데이터 전달을 단순화하여 순차적으로 정보를 주고받을 수 있는 구조로 0.18마이크론 공정의 반도체 제조 공정을 이용하여 시제품을 설계 및 제작하였다. 본 IC의 기본은 샘플링 방식에 의한 엔진 음 합성 형식이라 할 수 있는데 이는 디지털 시그널 프로세싱 이론을 이용하여 자동차 엔진 음을 전자적으로 완벽히 재생할 수 있는 가능성을 보여주는 결과라고 할 수 있다.

감사의 글

본 논문은 중소기업부의 2020' 중소기업 기술혁신 시장확장형 지원과제로 아이티아이일렉트로닉스의 연구과제인 “자율주행 및 전기자동차의 가상 엔진음 합성 IC 개발”의 연구 결과 중 일부이다.

References

1. V. Verfaillie, U. Zolzer, and D. Arfib, “Adaptive digital audio effects (A-DAFx): A new class of sound transformations,” *IEEE Trans. on Audio, Speech, and Lang. Process.* **14**, 1817-1831 (2006).
2. N. Juillerat, S. Schubiger-Banz, and S.M. Arisona, “Low latency audio pitch shifting in the time domain,” *Proc. ICALIP.* 29-35 (2008).
3. J. Chowning, “The synthesis of complex Audio spectra by means of frequency modulation,” *J. Audio Eng. Soc.* **24**, 525-534 (1973).
4. J. A. Moor, “The synthesis of complex audio spectra by means of discrete summation formulas,” *J. Audio Eng. Soc.* **2**, 717-727 (1976).
5. J. E. Koo, H. C. Bang, H. S. Yun, and W. H. Kim, “A study on the digital reverberation system for processing sound effect” (in Korean), *J. Acoust. Soc. Kr.* **11**, 5-14 (1992).
6. Y. Song, K. Lee, H. Bae, J. Ahn, J. Koo, and C. Park, “Development of hands-free tailgate using ultrasonic sensor detection laser image irradiation method” (in Korean), *Proc. Hyundai. Kia Motors Group's conf.* 25-31 (2015).

저자 약력

▶ 구 재 을 (Jae-Eul Koo)



1983년 2월 : 한국항공대 통신공학 학사
 1988년 2월 : 연세대 전자공학 석사
 1996년 2월 : 한국항공대 전자공학 박사
 1988년 ~ 1990년 : KAIST
 1990년 ~ 1994년 : 미국 AdMOS(Silicon Valley-LG Semicon)
 1996년 ~ 1998년 : 숭실대 전자공학과 겸임 교수(조교수)
 1994년 ~ 2019년 : 화음소 CEO(전 KOSDAQ)
 2019년~현재 : ITI Elec CTO

▶ 홍 재 규 (Jae-Gyu Hong)



1987년 2월 : 한양대학교 전기공학과 학사
 1989년 2월 : 한양대학교 전자공학과 석사
 1990년 2월 : 대우전자 연구소
 2015년 2월 : emdt 연구소 연구소장
 2015년 2월 ~ 현재 : ITI Elec 연구위원

▶ 송 영 옥 (Young-Woog Song)



2004년 2월 : 고려대학교 기계공학과 학사
 2006년 2월 : 서울대학교 기계공학과 석사
 2006년 ~ 2019년 1월 : 현대자동차수석연구원
 2019년 ~ 현재 : Tunelt(현대차계열) 대표

▶ 이 기 창 (Gi-Chang Lee)



2007년 2월 : 부산대학교 기계공학과 학사
 2006년 2월 : 부산대학교 기계공학과 석사
 2007년 ~ 2019년 1월 : 현대자동차 선임연구원
 2019년 ~ 현재 : Tunelt(현대차계열) CTO