MML(merged memory logic) 라이브러리 구축을 위한 반자동 아날로그 컴파일러 개발에 관한 연구

최 문석, 송 병근, 곽 계달
한양대학교 전자공학과
133-791 서울시 성동구 행당동 17 번지
mschoi@hymail.hanyang.ac.kr

A Study on the Development of Semi-automated Analog Cell Compiler for MML Library

Moon-Suk Choi, Byeong-Geun Song, Kae-Dal Kwack
Dept, of Electronic Engineering, Hanyang University,
Haengdang-Dong, 17 Sungdong-Gu, Seoul, Korea 133-791
mschoi@hymail.hanyang.ac.kr

Abstract

Today SOC(system on a chip) is a trend in VLSI design society. Especially MML(merged memory Logic) process provides designers with good chances to implement SOC which is consists of DRAM, SRAM, Logic and A/D mixed mode ciruit blocks. Designers need good circuit library which is reliable and easy to tune for specific design. For this need we present semi-automated analog compiler methodology. And we aplied this design methodology to resistor-string DAC design.

1. 서 론

미세공정(deep-submicron)과 소자 기술의 발전으로 인하여 기존의 시스템 수준 회로를 단일 칩으로(Silicon on a Chip, SOC) 구현할 수 있게 되었다. 이에 국내외 DRAM 업체를 중심으로 단일 공정에서 DRAM 과 Logic 을 동시에 구현할 수 있는 MML(Merged Memory Logic) 공정을 공급하고 있다. 이러한 공정의 발전으로 인하여, 회로 설계자는 개별 회로 설계에만 국한되지 않고, 시스템 수준의 회로 설계를 요구 받게 되었다. 시스템 수준의 설계는 설계 사양 뿐만 아니라 시스템을 구성하는 부분들(DRAM, SRAM, Digital 그리고 Analog 회로 블록 등)에 대한 지식과 함께 이에 대한 설계 능력 까지를 요구하고 있다. 그러나, 한 설계자가 이러한 여러 구성 부분들에 대한 설계능력을 갖고 있기는 매우 어려우며, 갖고 있다고 하더라도 시간과 이에 대한 많은 비용이 소모된다. 그러므로, 각 구성 부분들에 대한 라이브러리 개발이 절실히 요구된다. 그런데, 디지털 회로 블록들에 대한 설계 및 합성에 대한 자동화는 많이 진척되어 왔지만, 아날로그 회로 블록들에 대한 설계 자동화는 상대적으로 미비하여 실재 산업현장에서는 대부분이 완전 수동(full-custom) 설계 방식에 의존하고 있다. 현재까지의 아날로그 회로 설계 자동화에 대한 보고는 OP-AMP 나 필터 설계 등과 같은 전형적인 아날로그 회로들에 대하여 국한되어 왔으나, 개발 및 유지 보수에 많은 노력이 요구되고 있다. 본 연구에서는 완전 수동식과 완전 자동식에서의 아날로그 회로 설계법의 장점만을 절충한 반자동 컴파일러를 개발하여 제안한다.

2. 아날로그 설계 절차와 완전자동 컴파일러

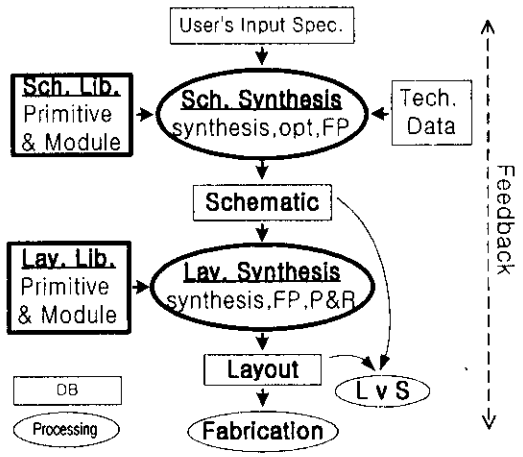

그림 1. 일반적인 아날로그회로 설계절차

아날로그 회로 설계에 대한 일반적인 절차는 그림 1 과 같이 나타낼 수 있다[1]. 그림에서 직사각형은 설계 사양, 공정자료, 레이아웃 및 회로도 등의 라이브러리나 데이터를 의미하며, 타원형은 회로 합성, 회로 최적화, 레이아웃 합성, LVS(레이아웃과 회로도의 비교검색)등의 실제 처리 절차를 의미한다. 대부분의 아

날로그 회로 설계는 주어진 공정과 설계 사양에 맞도록 먼저 **회로 설계**[1]를 한다. 이는 기본 회로 구성 요소에서부터 필요한 회로 모듈들까지를 모두 새로이 만들어 전체 회로를 구성할 수도 있고, 기존의 것들을 라이브러리화 하여 구성할 수도 있다. 이때 최종적으로 구성된 회로는 주어진 설계사양에 만족시키기 위해서 회로의 개별 트랜지스터 길이와 넓이를 최적의 값이 되도록 정한다. 이후 각 구성 부분별로 배치(floor-plan)를 한다. 회로설계를 마치면 **레이아웃 작업**[2]을 수행한다. 레이아웃 작업 역시 기본 레이아웃 구성 요소에서 모듈의 레이아웃까지 직접 그리거나 기존의 것들을 재사용하여 수행한다. 다음으로 전체 레이아웃 구성을 위한 배치 계획(floor-plan)과 배치 배선(place & routing)을 수행한다. 마지막으로 **레이아웃과 회로도의 비교 검색(LVS)**[3]을 수행한다. 이와 같은 1), 2), 3)의 과정을 반복하여 최초의 주어진 성능사양에 맞는 회로도와 레이아웃이 완성된다. 대부분의 아날로그 회로 설계는 위와 같은 설계 과정을 거치며, 경우에 따라서 부가적인 설계 절차가 추가된다. 디지털 회로 설계는 설계 자동화를 가능하게 하는 EDA (Electronic Design Automation) 소프트웨어의 지원이 충분하다. 그러나, 아날로그 회로 설계는 설계 자동화를 하기가 쉽지 않을 뿐만 아니라, 자동화 툴을 사용한 설계 회로의 실제 응용이 매우 불안하다. 지금까지 발표된 아날로그 회로 설계 자동화 툴 들은 칩 사양만 입력해 주면, 그 값에 맞는 회로도 및 레이아웃을 발생시켜 주던 방식 이였다. 그러나, 이러한 과정의 결과는 일반적으로 받아들여질 만한 수준이 아니기 때문에 매우 불안하다. 그림 1에서 굵은 선으로 표시된 부분은 아날로그 회로 설계 과정에서 가장 많은 설계시간을 요구하는 부분이다.

완전 자동화 아날로그 회로 설계 컴파일러라는 것은 이 분분을 컴파일러가 자동으로 수행하는 것을 말한다. 그러나, 이처럼 설계 과정을 완전히 자동화하더라도 이를 통해 설계된 회로는 디지털 회로처럼 그대로 채용할 수는 없다. 이는 아날로그 설계시의 모든 고려사항을 자동화 알고리즘으로 구현하는데 한계가 있기 때문이며, 이에 따라 신뢰도에 문제가 제기될 수 밖에 없다.

본 연구에서는 완전 수동 방식의 설계 방법과 완전 자동화 방식의 설계 방법의 한계를 인식하고 이를 해결하는 대안으로써 반자동 컴파일러를 개발하였다. 이는 설계 단계별로, 수동식 방식으로 수행하는 것이 유리한 단계는 수동식 방식으로 하고, 자동화 시키는 것이 유리한 단계는 자동화 방식으로 수행하도록 했다. 이렇게 함으로써 완전 수동식 방식에서의 자유로운 설계와 이로 인한 최적의 레이아웃 밀도를 얻을 수 있으며, 자동화 방식에서는 최상의 칩 설계 능력을 확보할 수 있다.

3. 반자동 아날로그 설계 방법론

앞에서도 언급했지만, 반자동 설계 방법은 전체 설계 단계에서 수동식 및 자동화를 선별적으로 시킴으로써, 두 방식의 장점만을 취하는 것이다. 이를 위해서는 먼저 설계의 어느 단계를 수동식 및 자동화로 진행시킬 것인가를 결정해야 한다.

본 논문에서는 직렬 저항(Resistor String, RS) 분배기를 이용한 DAC (Digital to Analog Converter) 설계[3]를 예로 들어 설명하겠다(그림 2.).

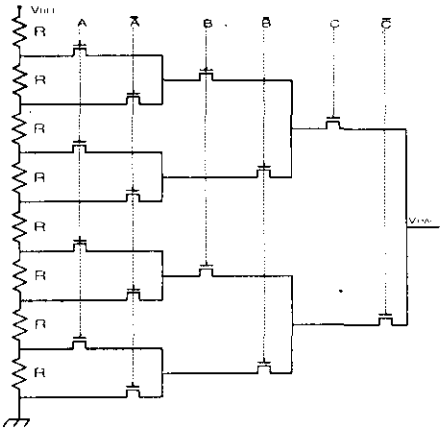

그림 2. 3b-RS DAC 의 회로도

표 1 은 3-bit RS DAC 설계를 할 때, 수동식 및 자동화의 여부를 각 설계 단계별로 나타내었다.[2]

설계 단계	풀커스텀	자동화
A. 회로설계		
a-1. primitive cell gen.		○
a-2. leaf cell gen.	○	
a-3. critical path gen.	○(생성시)	○(적용시)
a-4. floor plan	○(생성시)	○(적용시)
a-5. cir. Optimize	○	
a-6. synthesis		○
a-7. netlist gen		○
B. 레이아웃		
b-1. primitive cell gen.	○	
b-2. leaf cell gen.	○	
b-3. floor plan	○(생성시)	○(적용시)
b-4. synthesis		○
b-5 postlayout sim.	○(sim.)	○ (netlist)

표 1. 각 설계 단계별 수동, 자동화 여부

A. 회로설계

a-1) primitive cell gen : 트랜지스터와 저항 등으로 구성된 RS DAC 회로의 최소 단위를 의미한다.(GDT 환경에서는 이 primitive cell 들을 기본적으로 제공한다.)

a-2) leaf gen. : RS DAC 를 구성하는 기본 회로 블록들을 말한다. (그림 3) RS-DAC 를 구현하기 위해서는 단지 6 개의 셀(leaf cell)들만이 필요하다.

a-3) critical path gen : RS DAC 의 회로 동작 특성을 결정 짓는 회로 경로를 추출한다. 그림 4 는 3-bit RS-DAC 에서 추출된 중요 경로를 나타낸다. 이 중요 경로를 추출할 때는 실제 레이아웃에서의 배선 및 트랜지스터에 의한 기생 저항 및 커패시터값을 자동적으

로 추출하도록 하였다. 여기서 각 노드에 영향을 미치는 캐패시터값 Ck 는

$$C_k = C_u 2^k, \quad where \quad C_u \ is \ cap. \ of \ unit \ layout$$

로 구해진다.[4]

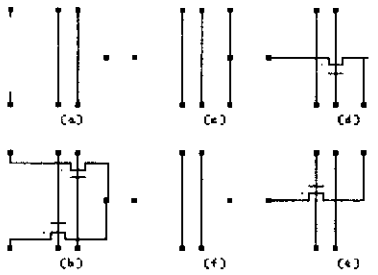

그림 3. RS-DAC 를 이루는 셀(leaf cell) 들(6 개)

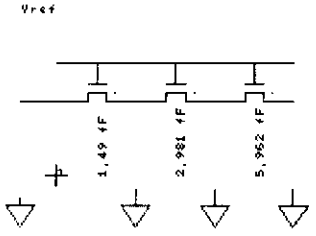

그림 4. 3-bit RS-DAC 의 중요 경로

a-4) floor plan : RS-DAC 의 전체 회로 구성 계획을 작성한다.(그림 5)

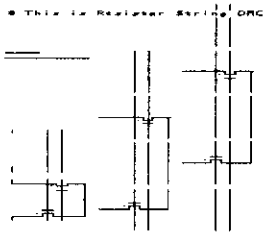

그림 5. 3-bit RS-DAC 회로의 배치 계획도

a-5) cir. Optimize : 주어진 설계 사양에 적합한 트랜지스터 사이즈를 결정한다. 이 부분은 HSPICE 를 사용하여 완전 수동식으로 수행한다.

a-6) synthesis(tiling) : 라이브러리 개발이 끝나면 사용자가 요구하는 사양에 맞춰 회로도를 생성한다. 여기서는 사용자가 선택할 수 있는 사양은 RS-DAC 의 임의의 bit 수 이다. 그림 6 은 3-bit RS-DAC 의 회로도를 생성한 것이다.

a-7) netlist gen : 설계한 회로의 시뮬레이션을 위한 HSPICE netlist 를 추출한다.

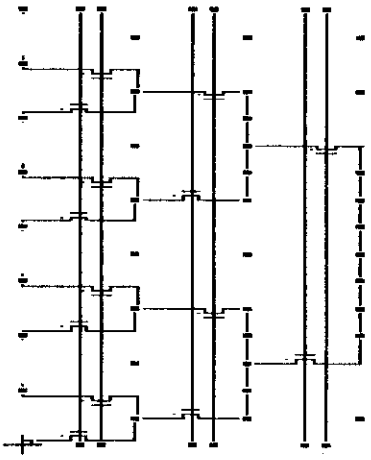

그림 6. 생성된 3-bit DAC 의 회로도

B. 레이아웃

b-1. primitive cell gen. : RS DAC 의 레이아웃을 그리기 위한 MOSFET 이나 저항 등의 최소 단위의 레이아웃을 의미한다(본 논문에서는 직접 그림).

b-2. leaf cell gen. : RS DAC 를 구성하는 기본 회로 블록들(그림 3)에 대한 레이아웃을 그림 7 에 나타내었다. 여기서도 역시 6 개의 셀(leaf cell)들만이 필요하다.

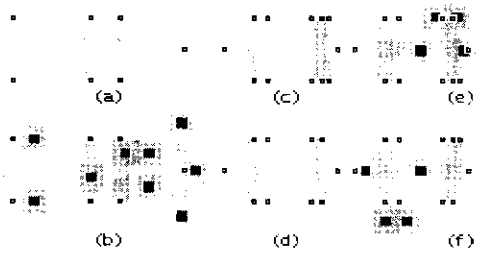

그림 7. RS-DAC 의 기본 셀(leaf cell) 들(6 개)

b-3. floor plan : 레이아웃의 배치 계획을 정한다. 그림 5 의 회로도에 상응하는 레이아웃 배치 계획도를 그림 8.에 나타내었다.

b-4. synthesis(tiling) : 라이브러리 개발이 끝나면 사용자가 요구하는 사양에 맞춰 레이아웃을 생성한다. 그림 9 는 3-bit RS DAC 의 레이아웃을 생성한 것이다. 여기에서는 폴리저항을 임의로 줄여서 나타내었다.

b-5. postlayout simulation : 최종적으로 결정된 회로도 및 레이아웃을 근거로 기생성분(R,C) 을 고려한 회로 시뮬레이션을 수행한다.

4. 결 론

full-custom 방법과 자동화 기법의 선별적 선택에 의한 반자동 설계 방법을 통한, 절충적 설계 기법은 양

쪽의 장점을 살려, 현재 개발 중인 SOC 상에서 아날로그 회로와 디지털 회로가 혼재한 상태의 MML 구현 문제를 시간적으로나, 비용적으로 크게 절감시켜주는 효과를 가져온다.

참고문헌

[1] JEF RIJMENANTS et al, "ILAC: An Automated Layout Tool for Analog CMOS Circuits", *IEEE Journal of Solid-State Circuits*, vol. 24, No. 2, pp417-425, April 1989.

[2] Mentor Graphics Co. "GDT Designer: Generator Creation and Compaction Manual", *Software Version 5.3_1*, Sep 1997.

[3] Johns Martin, "Analog Integrated Circuit Design", Wiley, pp463-465, 1997.

[4] JEN-YEN HUANG, "Resistor Termination in D/A and A/D Converters", *IEEE Journal of Solid-State Circuits*, vol. SC-15, No. 6, pp1084-1087, December 1980.

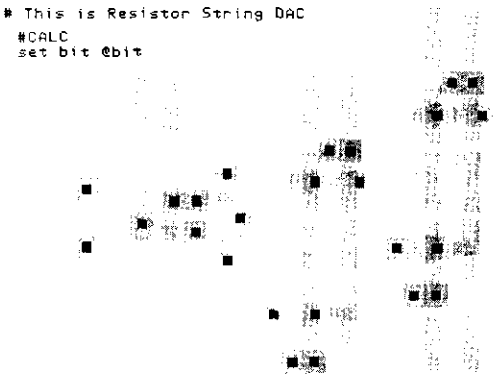

그림 8. 3-bit RS-DAC 레이아웃 배치 계획도

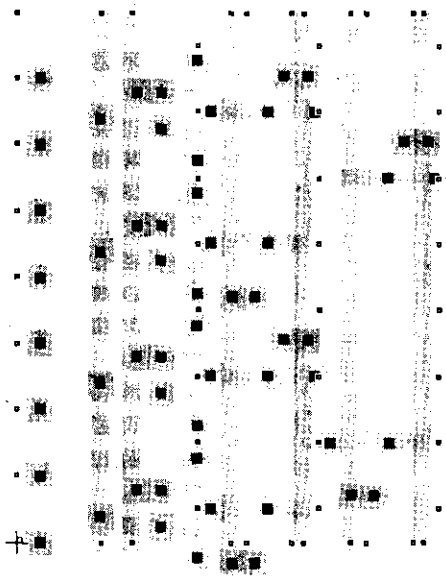

그림 9. 생성된 3-bit DAC 의 레이아웃