

# CMOS PLA 자동 Layout Program 개발에 관한연구 (The Research on Automation Layout Program of CMOS PLA)

朴 魯 京,\* 全 興 雨,\* 文 大 哲,\*\* 車 均 鉉\*

(Nho-Kyung Park, Heung-Woo Jun, Dai-Tchul Moon, and Kyun-Hyon Tchah)

### 要 約

본 논문에서는 CMOS PLA 회로의 레이아웃을 사용자가 원하는 기능에 따라 입력 데이터를 줌으로써 자동설계 할 수 있는 CMOS PLA Generator를 개발하였다. 본 프로그램의 구성은 PLA의 기본셀을 생성시켜 주는 procedure, 입력 데이터값을 stick diagram으로 그릴 수 있는 procedure, 임의의 설계규칙 (design rules)을 자동 변환하는 procedure, CIF 입력 데이터로 IBM PC-AT에서 physical 레이아웃을 플로팅하는 procedure, 다이나믹 CMOS PLA의 레이아웃 생성 procedure 등으로 구성되어 있다. 자동으로 생성, 변환된 레이아웃 정보는 CIF 화일로 저장된다. 본 연구에서 사용한 언어는 VLSI설계 TOOL를 만들기 위한 레이아웃용 언어인 CHISEL 언어이며, VAX 11/750 (UNIX)에서 수행했다.

### Abstract

This paper deals with the CMOS PLA Generator using CHISEL language. The program which plots a CMOS PLA alphanumeric layout automatically according to desired input functions and output functions has been developed. The program consists of procedures. These procedure are drawing a stick diagram with input data, converting any design rule, plotting a physical layout at IBM PC-AT with CIF input data. Physical layout information of dynamic CMOS PLA is stored CIF form. The CMOS PLA Generator is written in CHISEL which is layout language VLSI design tools and run on a VAX 11/750 running UNIX.

### I. 서 론

PLD(programmable logic device)는 각종 custom LSI화 방법중 가장 손쉬운 방법이나, 반면에 칩 이용율과 소비전력면에 결점을 가지고 있다. 그러나, PLD는 칩 이용율이 낮다는 결점은 folding 기법등을 이용해서 칩 이용율을 높이고, 소비전력이 크다는 결점은 CMOS process 등을 적용하여 PLD 성능을 향상시키고 있다.

앞으로 게이트 어레이(gate array)가 주요 functional 요소로 사용되는 PLD와 게이트 어레이가 가미된 새로운 형태의 PLD가 제품화 될 것이다. 이에 본 연구에서는 사용자가 필요로 하는 디바이스를 구성하고자 할때는 사용자가 function에 대한 입·출력을 정해진 format(chap VI 참조)에 따라 입력시키면 원하는 디바이스가 자동으로 레이아웃 되도록 하는 CMOS PLA generator를 국내에서 처음으로 레이아웃용 CHISEL 언어<sup>1)</sup>를 이용하여 개발하였다. 본 연구에서 사용한 CHISEL 언어는 첫째, 호환성이 좋고, 범용 언어로서 다른 자동 레이아웃 프로그램을 지원해 줄수 있으며, 둘째, 다른 설계장비와도 같이 사용할 수 있으나, 다른 레이아웃 장비의 도움없이도 칩 설계를 할

\*正會員, 高麗大學校 電子·電算工學科  
(Dept. of Elec. & Comp. Eng., Korea Univ.)

\*\*正會員, 湖西大學校 通信工學科  
(Dept. of Telcom. Eng., Hoseo College)

接受日字 : 1986年 9月 15日

수 있는 장점을 가진 언어로서 값비싼 CAD장비 없이도 쉽게 설계자가 직접 범용터미날 상에서 설계한 IC 레이아웃을 볼 수 있어 디버깅이 용이함으로 교육용 레이아웃 언어로도 널리 이용될 것이다. 또한, 본 연구에서 사용한 레이아웃 방식은 알파벳 문자로 심볼을 나타내는 alphanumeric 레이아웃 방식이며, 레이아웃 설계규칙은 NCR 설계규칙을 적용하였고,<sup>6,7)</sup> CMOS PLA generator를 회로에 직접 적용하여 그 성능을 입증하였다.

그리고 출력으로 저장된 CIF 파일을 이용하여 플롯터로 그린 레이아웃과 프린터로 작성한 alphanumeric 레이아웃의 타당성을 확인하여 보았다.<sup>8,4)</sup>

전체적인 프로그램의 길이는 약 1,800 라인정도이다. 여기서 사용한 CHISEL 언어는 칩설계, 그래픽 에디터, 시스템의 placement와 routing, 실리콘 컴파일 등에도 직접 이용할 수 있는 장점을 가지고 있다.

II. CMOS PLA 설계 개요

1. CMOS PLA 구성

NAND 구조는 NOR 구조와 비교해서 속도면에서 NOR 구조보다  $N \times N$ 배 정도 느리나, 면적 면에서는 NOR 구조보다 약 30%정도 셀 면적을 작게 할 수 있는 잇점을 가지고 있어 본 연구에서는 집적도를 고려하여 NAND 구조를 이용했다.<sup>11)</sup> 논리합수를 규칙적인 구조로서 실현할 수 있는 PLA 구조는 일반적으로 그림 1(a)와 같이 구성된다.<sup>12)</sup> 본 연구에서는 주변의 다른 소자들과의 상호 연결을 원활하게 하고, 배선의 길이를 단축시킬 수 있도록 입력과 출력을 상·하에서 각각 입·출력할 수 있는 효율적인 구조를 갖는 CMOS PLA를 구성하였다. 그리고 변형된 PLA 구조는그 위상 클럭으로 동작하도록 하였다.<sup>11)</sup> 설계된 PLA의 블록도는 그림1(b)와 같다.

2. CMOS PLA 설계구성 및 레이아웃 프로시유어

1) 설계 구성도

본 연구에서는 CMOS PLA generator를 그림 2와 같이 구성하였다. 그림 2에서 PLA 기본구성 요소들은 입·출력, AND 평면, OR 평면을 구성하는데 사용되는 것으로서 모든 프로시유어에서 이용할 수 있도록 한다.<sup>14)</sup> 본 PLA 구조는 AND 평면과 OR 평면을 NAND 구조로 구현했다.

2) 프로시유어 블록

그림 2의 설계블록에 대한 PLA 구성 요소들과 완전한 PLA를 생성시키는 중요한 프로시유어는 그림 3과 같다.

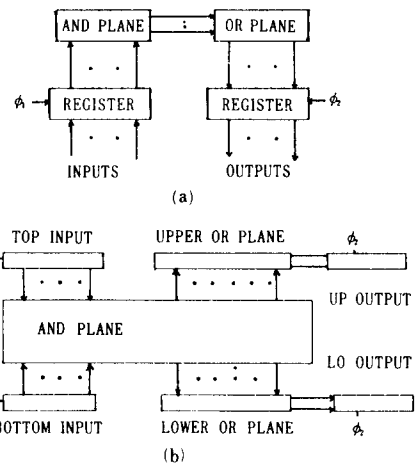


그림 1. PLA 블록도

- (a) 일반적인 PLA블럭도
- (b) 변형된 PLA블럭도

Fig. 1. PLA block Diagram.

- (a) Conventional PLA Block Diagram.
- (b) Modified PLA Block Diagram.

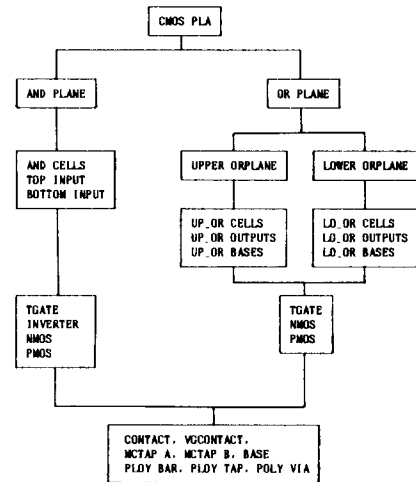


그림 2. CMOS PLA GENERATOR 구성도

Fig. 2. Block Diagram of CMOS PLA Generator.

III. CMOS PLA 기본 구성요소 설계

CMOS PLA Generator는 PLA 기본 구성요소에 대한 화일과 이를 사용하여 PLA를 레이아웃하기 위한 화일로 구성되는데 PLA 기본 구성요소들은 AND 평면, OR 평면, 입·출력단에서 모두 이용될 수 있도록 헤드 화일로 만들었다.

1) 기본 구성요소 설계

본 연구에서는 CMOS PLA를 구성하는 기본적인 구

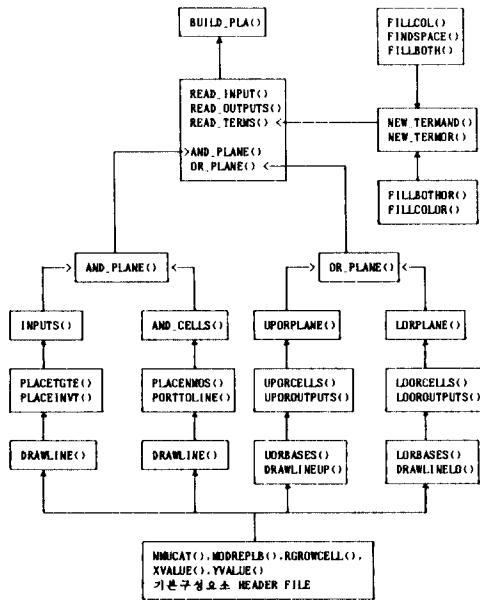


그림 3. CMOS PLA 프로시듀어 블록도  
Fig. 3. Blocnk Diagram of CMOS PLA Procedure.

성요소를 11개 (CONTACT, VGCONTACT, POLY BAR, POLY TAP, POLY VIA, MCTAP A, MCTAP B, NMOS, PMOS BASE, INVERTER, TGATE)로 선정하였다. 선정된 기본 구성요소는 NCR 설계규칙을 적용하여 설계하였다. 설계된 각 구성요소의 기능은 다음과 같다.

- (1) CONTACT : 인버터에서 P-WELL 및 기판을 VDD 및 GND에 연결하는데 사용된다.
- (2) VGCONTACT : TGATE에서 P-WELL 및 기판을 VDD 선 및 GND선에 연결하는데 사용된다.
- (3) POLY BAR : AND 평면에서 NMOS가 배치되지 않은 부분에 배열되어 배치된 NMOS간의 게이트를 연결하는데 사용된다.
- (4) POLY TAP : 인버터나 TGATE에서 입·출력단자를 연결하는데 사용된다.
- (5) POLY VIA : AND 평면의 출력과 OR 평면의 입력 연결 및 클럭들의 연결에 사용된다.
- (6) MCTAP A : 인버터에서 VDD 선이나 GND선을 PMOS의 소오스에 연결할 때 사용된다.
- (7) MCTAP B : 인버터나 TGATE에서 PMOS의 드레인 단자들을 연결하는데 사용된다.
- (8) PMOS : P-PLUS, 실리콘 및 POLY가 만나서 형성된 트랜지스터로서 AND 평면과 OR 평면에서 사용된다.

- (9) BASE : UP-OR BASES와 LO-OR BASES를 만들기 위한 구성요소로서 BASES 사이에 PMOS가 배치된다.
- (10) INVERTER : 입력신호의 위상을 반전시키는 소자로서 입력단에서 사용된다.
- (11) TGATE : Transmission gate의 약자로서 위상이 반전된 두개의 클럭으로 입력신호를 제어하는 소자이다.

선정된 11개의 기본 구성요소들은 CMOS PLA를 구성하는 기본 요소들로서 프로그램화 할 때 하나의 블록으로 호출된다. 각 기본 구성요소에 대한 프로그램을 수행한 결과로 나온 기본 구성요소들의 레이아웃 그림의 일부분은 그림 4와 같으며, 이해를 돕기 위하여 설계도형도 첨부하였다. 이때 각각의 크기는 ZOOM을 하였으므로 크기에 있어 차이가 있다. ZOOM 1에 대하여 열 방향은 1 $\mu$ m이고, 행 방향은 0.5 $\mu$ m 단위로 alphanumeric 이 표기된다.

```

    I          ppp
    ++++++++ppp+++++++
    +SSSSSSStttSSSSSSS+
    +SSSSSSStttSSSSSSS+
    +SScccSstttScccSSS+
    +SSSSSSStttSSSSSSS+
    +SSSSSSStttSSSSSSS+
    ++++++++ppp+++++++
  
```

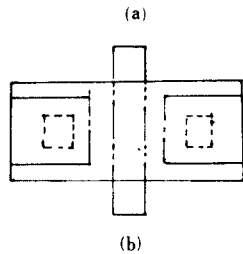


그림 4. 기본구성요소 레이아웃  
(a) Alphanumeric 레이아웃 (nmos zoom 2)  
(b) Physical 레이아웃  
Fig. 4. Layout of Basic Component.  
(a) Alphanumeric Layout (nmos zoom 2).  
(b) Physical Layout.

IV. 기본 CMOS PLA 셀 설계 및 프로그램

CMOS PLA는 다음과 같이 top input, bottom input, and cell로 구성된 AND 평면과 upper orplane, lower orplane으로 구성된 OR평면으로 이루어져 있다. 그리고 입·출력단은 매우 간결하게 inverter와 tgate로 구성한다.

1. AND 평면 설계

AND 평면은 top input, bottom input, and cells 로 구성한다. Top input은 그림 5 와 같이 구성되도록 설계하며, bottom input은 top input을 X축에 대칭시킨 형태로 설계하였다. AND 셀의 NAND 구조는 그림 6 과 같은 다이내믹 CMOS NAND로 구성한다.

다이내믹 CMOS NAND 회로는 CMOS NAND 회로 보다 칩면적이 감소되는 장점을 가지고 있다. 이러한 특징을 갖는 다이내믹 CMOS NAND 회로로 구성된 AND셀에 대한 설계도형은 그림 7 과 같은 형태를 갖는다.

2. OR 평면 설계

OR 평면은 크게 upper or 평면과 lower or 평면으로 나누어 설계한다. upper or 평면은 up or bases, up or cells, up or outputs으로 구성되고, lower or 평면은 lo or bases, lo or cells, lo or outputs로 구성된다.

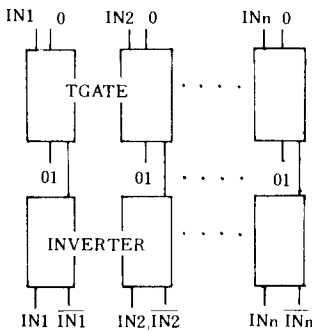


그림 5. 상측 입력  
Fig. 5. Top Input.

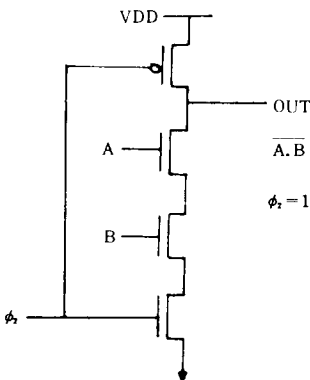


그림 6. 다이내믹 CMOS NAND  
Fig. 6. Dynamic CMOS NAND.

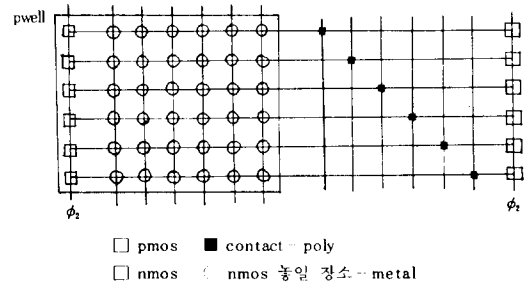


그림 7. AND 셀  
Fig. 7. AND Cells.

up or outputs은 그림 8 과 같이 구성되도록 설계하였으며, lo or outputs은 up or outputs을 X축에 대응시킨 형태로 설계한다.

up or cells와 lo or cells은 그림 9 과 같은 CMOS 변형 형태를 갖는 NAND 구조로 설계한다. 일반 CMOS 셀은 PMOS와 NMOS의 쌍으로 구성되지만, 그림 9 의 NAND 회로에서 NMOS 회로는 단 하나의 MOSFET로 구성된다. 그러므로 칩 면적이 감소되는 장점이 있다.

변형된 CMOS NAND 회로로 구성된 up or cells 에

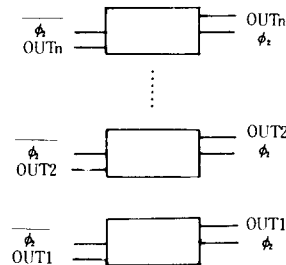


그림 8. Up-or Outputs 구조  
Fig. 8. Up-or Outputs Structure.

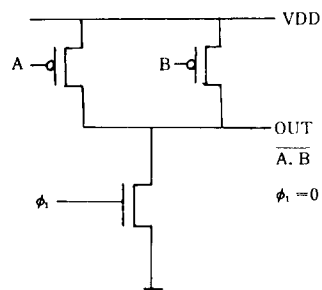


그림 9. 변형된 CMOS NAND.  
Fig. 9. Modified CMOS NAND.

대한 설계도형은 그림10(b) 같은 형태를 갖는다. 그림 10(a)의 bases는 PMOS 및 NMOS 가 배치되는 기본 틀을 형성하는데 사용된다. 그리고 lo or cells은 up or cells을 X축에 대칭시킨 형태로 설계한다.

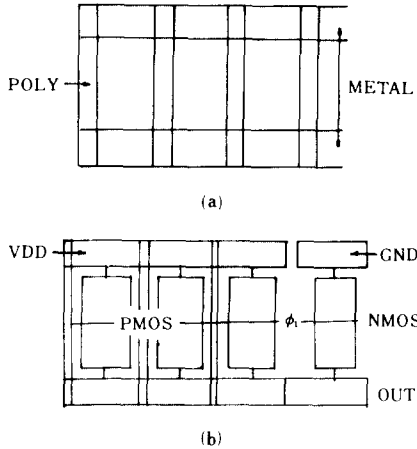


그림10. Base와 Up-or Cell 구조  
(a) BASE 구조  
(b) Up-or Cell 구조

Fig. 10. Structure of Base & Up-or Cell.  
(a) BASE Structure.  
(b) Up-or Cell Structure.

3. 기본 PLA셀 프로그램

1.2 절에서 설계한 기본 PLA셀들(AND 평면과 OR 평면)을 만들기 위한 각각의 프로시듀어들을 만든다. 이들은 III장에서 설계한 기본 구성요소들에 대한 HEADER 파일에 있는 셀 이름이 아규먼트로 사용되며 기본 PLA셀들을 만든다.

1) AND 평면

And 평면을 만들기 위한 프로시듀어는 and plane()이며, and plane() 프로시듀어는 inputs() 프로시듀어와 and cells() 프로시듀어로 되어있다. 이들에 대한 각각의 기능은 다음과 같다.

input() : 그림5의 top input과 그의 X축 대칭인 bottom input의 생성과 위치 선정을 하는 프로시듀어이다.

and cells() : 그림7의 AND cells을 생성시키는 프로시듀어이다.

2) OR 평면

OR 평면을 만들기 위한 프로시듀어는 or-plane()이며, or-plane()은 uorplane()과 lorplane()로 되어있다. 그리고 uorplane()은 uporcell()와 uporoutputs()로 구성되어 있으며, lorplane()은 loa-

cells()와 looroutputs()로 구성되어 있다. 이들에 대한 기능은 다음과 같다.

uporcells() : 그림10(b)의 uporcells을 생성시키는 프로시듀어이다.

uporoutputs() : 그림7의 uporoutputs을 생성시키는 프로시듀어이다.

loorcells() : 그림10(b)의 X축 대칭인 loorcells을 생성시키는 프로시듀어이다.

looroutputs() : 그림7의 X축 대칭인 looroutputs을 생성시키는 프로시듀어이다.

V. CMOS PLA 설계 및 프로그램

III장과 IV장에서 설계한 기본 CMOS PLA 구성요소들과 기본 CMOS PLA 셀들을 종합하여 변형된 CMOS PLA 구조를 갖도록 전체적인 CMOS PLA를 설계하고 프로그램을 개발하였다.

1. CMOS PLA 설계

CMOS PLA 구성은 II장에서 설명한 바와 같이 AND 평면을 and cells을 중심으로 위로는 top input이, 아래로는 bottom input이 위치하도록 설계한다. AND 평면의 오른쪽에 위치한 OR 평면은 위쪽에 upper orplane을 놓고, 아래쪽에는 lower orplane이 위치하도록 전체적인 CMOS PLA를 구성하였다. 설계구성은 그림1(b)와 같다.

2. CMOS PLA 프로그램

그림1(b) 같이 구성되는 CMOS PLA를 레이아웃하기 위한 프로그램을 만들었으며, 전체적인 CMOS PLA는 AND 평면과 OR 평면으로 구성되므로 AND 평면과 OR 평면을 결합시켜 완전한 CMOS PLA를 만드는 프로시듀어를 만들었다. 이러한 기능을 갖는 프로시듀어는 build pla()이다.

build-pla()의 기능은 다음과 같다.

build-pla() : build pla() 프로시듀어는 and-plane()과 or-plane()로 만들어진 기본 셀들을 이용하여 CMOS pla의 입·출력 정보에 대한 입력에 따라서 완전한 CMOS pla를 구성하기 위한 셀들의 위치 선정과 배치를 하며 clock선, VDD선, GND선 등을 연결시키는 기능을 한다.

CMOS PLA의 자동레이아웃에 있어서 중요한 것은 입력을 읽고, 그에 따라서 AND셀의 terms로서 NMOS를 배치하는 기능이다. 이에 대한 연산 알고리즘은 그림11과 같다.

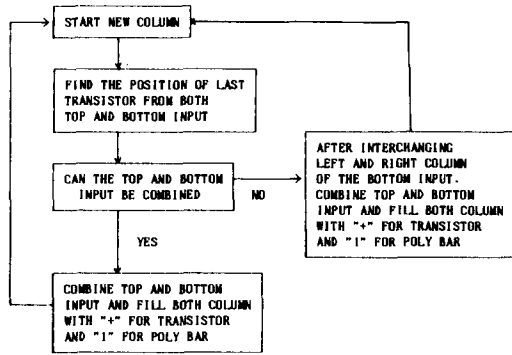


그림11. AND Terms 배치연산  
Fig. 11. AND Terms Placement Algorithm.

예를 들면, 표 1 과 표 2 에서와 같이 top input과 bottom input이 2 개의 열로 분리된다. 이때 칩면적은 줄이기 위하여, "1"은 좌측열로 가고, "0"은 우측열로 가서 하나의 열로 결합하도록 한다. "1"과 "0"은 '+' 로 표시되어 NMOS가 배치되고, 다른 문자(P)는 P loy bar로 배치되어 AND term들이 연결된다. 이 때 top과 bottom 입력에 대한 term들이 상호연결 되지 않도록, 즉 설계규칙(design rule)에 적합하도록 적어도 하나의 공간이 존재하여야 한다. 공간이 존재하지 않는 경우는 표 2 와 같이 bottom input의 좌우열을 교환한 후 결합된다. 이와같이 AND 셀의 term을 배치시키는 기능을 부가함으로써 설계규칙을 점검할 필요없이 배치가 자동적으로 된다.

이와 같은 기능을 하는 프로시듀어는 new-term and()이며, 그와 관련된 프로시듀어는 fill-col(), fillboth()와 findspace()가 있다. 그리고 orplane()의 team으로서 PMOS를 배치하는 기능을 갖는 프로시듀어는 new-termor()이며, 그와 관련된 프로시듀어로서는 fillbothor(), fillcolor()가 있다. 그외에 입·출력단을 만들고, PMOS와 NMOS를 term에 배치시킨다.

표 1. 입력배치 연산예 1  
Table 1. Example 1 of Input Placement Algorithm.

TOP INPUT	BOTTOM INPUT	INPUT SEPARATION		COMBINED INPUT	
		LEFT	RIGHT	LEFT	RIGHT
1	.	1	.	+	P
0	.	.	0	P	+
1	.	1	.	+	
.	.	.	.		
.	0	.	0		+
.	1	1	.	+	P

표 2. 입력배치 연산예 2  
Table 1. Example 2 of Input Placement Algorithm.

TOP INPUT	BOTTOM INPUT	INPUT SEPERATION		COMBINED INPUT		AFTER INTER-CHANGING LEFT & RIGHT COLUMN OF THE BOTTOM INPUT	
		LEFT	RIGGT	LEFT	RIGHT		
1	.	1	.	+	P	+	P
1	.	1	.	+	P	+	P
0	.	.	0	P	+	P	+
1	.	1	.	+			
.	1	1	.	+			+
.	0	.	0	P	+	+	P

배선등을 하는데 필요한 프로시듀어와 기능은 다음과 같다.

- placetgate(), placeinv(), placenmos(): AND 평면에서 원하는 위치에 각각 tagte, inverter, NMOS를 생성시키는 프로시듀어이다.
- drawline(): 두 점사이를 선으로 연결시켜주는 프로시듀어이다.
- porttoline(): 포트와 포트사이를 연결시켜주는 프로시듀어이다.
- numcat(s, n): N(십진수표현)에 따라서 S(STRING))을 돌려보내는 프로시듀어이다.
- xvalue(), yavlue(): 셀안에 있는 포트에서 각각 X값, Y값을 생성시키는 프로시듀어이다.

VI. 입·출력방식 및 프로그램 실험

1. 입력 데이터 작성방법

CMOS, PLA Generator는 원하는 입·출력에 따라서 CMOS PLA가 자동적으로 레이아웃이 되도록 하는 프로그램이다. 따라서 원하는 PLA 회로에 대한 레이아웃을 CMOS PLA Generator을 이용하여 그리고자 할때 입력 데이터를 CMOS PLA Generator에 입력시키는 방법에 대해서 서술한다.

예를 들어 원하는 CMOS PLA 회로가 그림12와 같을 때, 먼저 CMOS, PLA 회로에 대한 출력함수의 값을 구하면 다음과 같다.

$$\begin{aligned} \text{upper 1} & \quad \bar{T}_2 \cdot T_3 + T_3 + \bar{B}_2 \cdot \bar{B}_3 \\ \text{lower 1} & \quad T_1 \cdot \bar{T}_3 + B_1 \cdot B_3 \\ \text{upper 2} & \quad \bar{T}_1 \cdot T_2 + B_1 \cdot B_3 \\ \text{lower 2} & \quad \bar{T}_2 \cdot T_3 + \bar{B}_2 \cdot B_3 \end{aligned}$$

구한 출력함수값을 CMOS, PLA Generator에 입력시키기 위한 입력 데이터를 만들기 위해서 출력 함수값에 대한 입·출력 배열을 표 3 과 같이 작성한다.

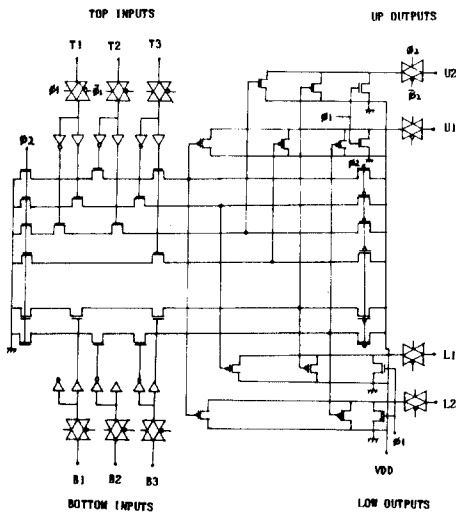


그림12. CMOS PLA 회로의 예  
Fig.12. An Example of CMOS PLA Circuit.

표 3. 예에 대한 입·출력 배열  
Table 3. Input, Output Array About Example.

입력	T1	B1	T2	B2	T3	B3	출력	U1	L1	U2	L2
	.	.	0	.	1	.		1	.	.	1
	1	.	.	.	0	.		.	1	.	.
	0	.	1	.	.	.		.	.	1	.
	.	.	.	.	1	.		1	.	.	.
	.	1	.	.	.	1		.	1	1	.
	.	.	.	0	.	0		1	.	.	1

T = TOP, B = BOTTOM U = UPPER, L = LOWER

표 3에 입력란 중에서 '0'은 입력 product항 중에서 input bar를 나타내며, '1'은 입력 product항 중에서 input를 나타낸다. 출력란 중에서 '1'은 AND 평면의 행과 연결을 나타내고, '.'은 AND 평면의 어느 행과도 연결되지 않음을 나타낸다. 표 3과 같이 작성한 입·출력 배열을 CMOS PLA Generator에 입력시키기 위해서는 표 4와 같은 입력데이터 화일로 작성한다.

즉, PLA는 AND 평면과 OR 평면 형태로 되어있어 AND 평면에는 주어진 함수의 입력값에 따라서 NMOS를 심어주고, OR 평면은 주어진 함수의 출력값에 따라서 AND 평면과 연결되는 행에 PMOS를 연결시키는 원하는 함수의 값을 가진 PLA 레이아웃을 터미널의 스크린상에서 볼 수가 있다.

2. 프로그램 실험

본 연구에서 개발한 CMOS PLA Generator를 실험하기 위해서 반가산기(half-adder), 전 가산기(full-

표 4. 예에 대한 입력화일  
Table 4. Input File About Example.

```

PLA      "EXAMPLE-T"
"TOP 1"  T C
"BOTTOM 1" B C
"TOP 2"  T C
"BOTTOM 2" B C
"TOP 3"  T C
"BOTTOM 3" B C
" "
"UPPER 1"  UP C
"LOWER 1"  LO C
"UPPER 2"  UP C
"LOWER 2"  LO C
" "
..0.1.1..1
1...0..1..
0.1.....1.
...1.1...
.1...1.11.
...0.01..1
" "
END
    
```

adder), 7-세그먼트(seven-segment)에 적용하여 본 프로그램의 타당성을 확인하였고, 여기서는 7-세그먼트에 대해서만 논하겠다. CMOS PLA Generator의 동작과정은 그림13과 같다.

7 세그먼트를 CMOS PLA Generator를 이용하여 설계하기 위해서 논리함수를 구하면 다음과 같다.

$$\begin{aligned}
 a &= x_2 \cdot x_4 + x_3 \cdot x_4 + x_1 \cdot \bar{x}_2 + \bar{x}_1 \cdot \bar{x}_2 \cdot \bar{x}_4 \\
 b &= x_1 + \bar{x}_1 \cdot \bar{x}_2 + x_3 \cdot x_4 + \bar{x}_3 \cdot \bar{x}_4 \\
 c &= x_1 + \bar{x}_3 + x_3 \cdot x_4 \\
 d &= x_1 + x_3 \cdot \bar{x}_4 + x_2 \cdot \bar{x}_3 \cdot x_4 + \bar{x}_1 \cdot \bar{x}_2 \cdot x_3 + \bar{x}_2 \cdot \bar{x}_3 \cdot \bar{x}_4 \\
 e &= x_3 \cdot \bar{x}_4 + \bar{x}_2 \cdot \bar{x}_3 \cdot \bar{x}_4 \\
 f &= x_1 + \bar{x}_3 \cdot \bar{x}_4 + x_2 \cdot \bar{x}_3 \cdot \bar{x}_4 + x_2 \cdot x_3 \cdot \bar{x}_4 \\
 g &= x_1 + x_2 \cdot \bar{x}_3 + \bar{x}_2 \cdot x_3
 \end{aligned}$$

이 된다.

논리함수 출력 a, b, c, d, e, f, g를 CMOS PLA Generator의 입력화일로 바꾸면,

$$\begin{aligned}
 U1 &= T2 \cdot T4 + T3 \cdot T4 + T1 \cdot \bar{T}2 + \bar{T}2 + T1 \cdot \bar{T}2 \cdot \bar{T}4 \\
 U2 &= T1 + \bar{T}1 \cdot \bar{T}2 + T3 \cdot \bar{T}4 + T3 \cdot T4 \\
 U3 &= T1 + \bar{T}3 + T3 \cdot T4 \\
 U4 &= T1 + T3 \cdot \bar{T}4 + T2 \cdot \bar{T}3 \cdot T4 + \bar{T}1 \cdot \bar{T}3 \cdot T3 \\
 &\quad + \bar{T}2 \cdot \bar{T}3 \cdot \bar{T}4 \\
 L1 &= T3 \cdot \bar{T}4 + \bar{B}2 \cdot \bar{B}3 \cdot \bar{B}4 \\
 L2 &= T1 + \bar{T}3 \cdot T4 + B2 \cdot \bar{B}3 \cdot B4 + B2 \cdot B3 \cdot \bar{B}4 \\
 L3 &= T1 + B2 \cdot \bar{B}3 + \bar{B}2 \cdot B3
 \end{aligned}$$

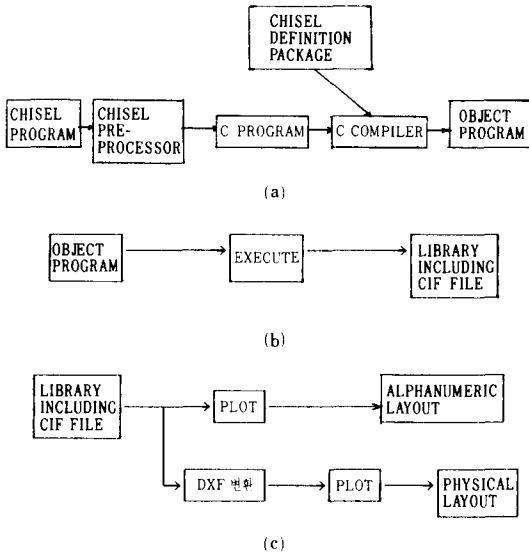


그림13. CMOS PLA Generator의 동작

- (a) CHISEL 프로그램 컴파일
- (b) CHISEL 라이브러리 생성
- (c) CHISEL 프로그램 플로팅

Fig. 13. Operation of CMOS PLA Generator.

- (a) Compilation of CHISEL Program.
- (b) Generation of CHISEL Library.
- (c) Plotting of CHISEL Program.

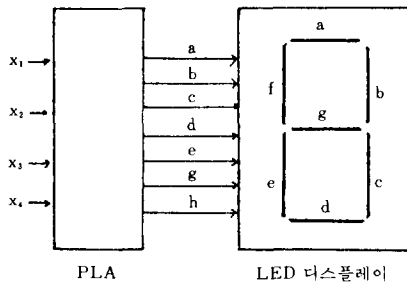


그림14. PLA와 LED 디스플레이 인터페이스

Fig. 14. The Interface Between PLA and LED Display.

U1, U2, U3, U4, L1, L2, L3에 대한 식을 입력화일로 작성하면 표5와 같다.

입력화일을 CMOS PLA Generator에 입력시켜 프로그램을 수행하면 그림15와 같이 7 세그먼트 레이아웃이 생성된다. 그림15는 32배로 축소한 것이다. 이블 수행하는 computing time은 시스템 시간으로 1.7초가 소요되었으며, 그의 반가산기와 전가산기는 각각 1.3초와 1.1초가 걸렸다.

Ⅶ. 결 론

본 연구에서는 CMOS 소자로 구성된 CMOS PLA

표 5. 7 세그먼트에 대한 입력화일  
Table 5. Input File About Seven Segment.

```

    PLA "seven-segment.t"
    "top 1" TC
    "bottom 1" BC
    "top 2" TC
    "bottom 2" BC
    "top 3" TC
    "bottom 3" BC
    "top 4" TC
    "bottom 4" BC
    "
    "upper 1" UPC
    "lower 1" LOC
    "upper 2" UPC
    "lower 2" LOC
    "upper 3" UPC
    "lower 3" LOC
    "upper 4" UPC
    "
    .. 1. . . 1. 1. . . .
    . . . 1. 1. 1. . . 1.
    1. 0. . . . 1. . . .
    0. 0. . . . 0. 1. . . .
    1. . . . . . 11111
    . . . 0. . . . . 1.
    0. 0. . . . . 1. . . .
    . . . 0. 1. . . 11. . .
    . . . 1. 0. . . 11. . . 1
    . . . 1. 0. 1 . . . 1. 1
    . 0. 0. 1. . . . . . 1
    . . . 0. 0. 0. 1. . . . 1
    . . . 1. 1. 0 . . . 1. . .
    . . . 1. 0. . . . . 1.
    . . . 0. 1. . . . . 1.
    "
    END
    
```

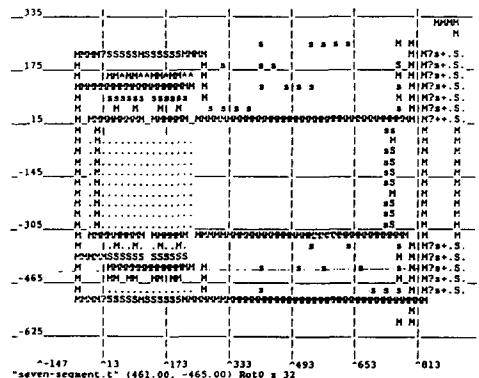


그림15. 7 세그먼트 레이아웃  
Fig. 15. Layout of Seven Segment.



를 사용자가 원하는 입·출력 정보에 따라서 자동적으로 레이아웃하는 CMOS PLA Generator를 CHISEL 언어를 이용하여 개발하였으며, 그 성능을 실험하였다.

7-세그먼트에 대하여 성능을 실험하여 그 출력이 Alphanumeric 형태로 스크린상에 정상적으로 레이아웃 되었으며, CIF 화일이 생성되었다. 간단한 회로에 대하여는 출력된 CIF 화일을 Physical 레이아웃 하여 Alphanumeric 레이아웃과 같게 나타남을 확인하였다. 본 논문에서 개발한 CMOS PLA Generator는 사용자가 원하는 입·출력에 대해서 CMOS PLA를 자동적으로 레이아웃 할 수 있고, 입·출력단을 각각 2방향으로 분리하여 뽑아낼 수 있어 주변회로와의 배선이 쉽도록 하였다. CMOS PLA Generator를 실행시키면 출력을 CIF 화일로 저장할 수 있어 다른 CAD 시스템과의 호환성이 있다. 그리고 CMOS PLAGenerator는 CHISEL 언어를 이용하여 개발되어 국내에 널리 보급되어 있는 범용 컴퓨터에서 동작이 가능하다. 또한 CHISEL 언어는 C언어를 확장한 것이기 때문에 쉽게 습득할 수 있어 레이아웃 프로그램 개발시간을 단축시킬 수 있으며, 레이아웃을 특정 CAD 시스템없이 일반 화면에서도 볼 수가 있어 값비싼 CAD 장비의 도입없어도 설계자가 IC칩을 설계할 수 있으므로 그 파급 효과가 상당히 크리라 사료된다. 본 연구는 과학기술처에서 시행한 특정연구 개발사업의 연구결과임을 밝혀둔다.

#### 參 考 文 獻

[1] Kyunhyon Tchah; Kevin Karplus "Design of CMOS PLA using Clanguage", *KIEE*, vol. 21, no. 5, pp. 61-66, 1984.

- [2] Kevin karplus, "CHISEL An Extension to the programming language C for VLSI layout", Stanford University, 1983.
- [3] C. Mead & L. conway, "Introduction to VLSI System", chap 3-4, Addison-Welsey, 1980.
- [4] Ullman, "Computational Aspects of VLSI", chap 7-8, Computer science press, 1984.
- [5] Neil Weste, Kamran Eshraghian, "Principles of CMOS VLSI Design A System Perspective", Addison-Welsey, 1985.
- [6] John Newkirk, Robert Mathews, "The VLSI Designer's Library", Addison-Welsey, 1983.
- [7] KIET, "A Study on MOS Technology Development", Chap 2, chap 5-6 1984-1985.
- [8] "Spice 2G User's Guide", Department of Electrical Enineering and Computer Science, University of California Berkeley, 1981.
- [9] J.R Egan & C.L. Liu, "Bipartite Folding and Partitioning of PLA", *IEEE Trans. On Computer-Aided Design*, vol. CAD-3, no. 3, July 1984.
- [10] 과학기술처, "다목적 공동설계 개발에 관한 연구 최종 연구보고서" 1986.
- [11] Saburo Muroga, "VLSI System Design", chap4. chap7, John Wiley & Sons Newyork 1982.
- [12] J. Mavor, M.A. Jack and P.B. Denyer, "INTRODUCTION TO MOS LSI DESIGN", Addison-welsey, 1983.