

**BỘ GIÁO DỤC VÀ ĐÀO TẠO  
ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH  
TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT  
KHOA ĐIỆN - ĐIỆN TỬ  
BỘ MÔN ĐIỆN TỬ**

# **LUẬN VĂN TỐT NGHIỆP**

**ĐỀ TÀI 2**

**ỨNG DỤNG VI MẠCH SỐ LẬP TRÌNH**

**SINH VIÊN THỰC HIỆN : TRƯƠNG PHƯỚC TOÀN  
LỚP : 95KDD  
GIÁO VIÊN HƯỚNG DẪN : TRẦN VĂN TRỌNG**

Với sự tiến bộ không ngừng của khoa học kỹ thuật, đặc biệt là ngành điện tử đã ứng dụng rất nhiều trong công nghiệp. Trong lĩnh vực điều khiển, từ khi công nghệ chế tạo loại vi mạch lập trình phát triển đã đem đến các kỹ thuật điều khiển hiện đại có nhiều ưu điểm so với việc sử dụng các mạch điều khiển được lắp ráp từ các linh kiện rời như kích thước mạch nhỏ, gọn, giá thành rẻ, độ làm việc tin cậy và công suất tiêu thụ thấp ...

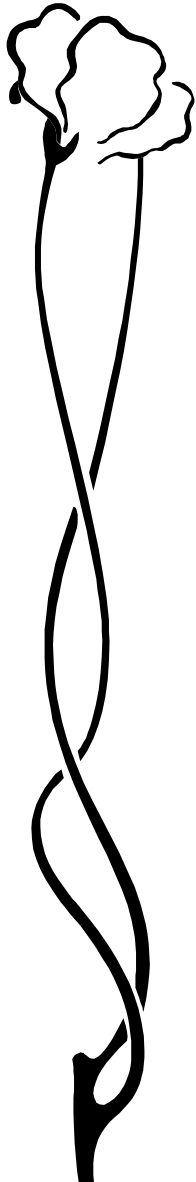
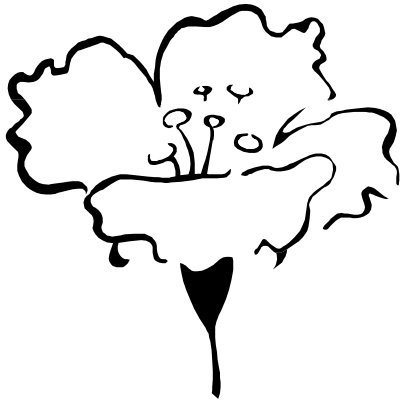
Ngày nay lĩnh vực điều khiển đã được ứng dụng rộng rãi trong các thiết bị, sản phẩm phục vụ cho nhu cầu sinh hoạt hằng ngày của con người như máy giặt, đồng hồ điện tử ... nhằm giúp chg đời sống ngày càng hiện đại và tiện lợi hơn.

Đề tài ứng dụng vi mạch số lập trình rất phong phú đa dạng, có nhiều loại hình khác nhau dựa vào công dụng và độ phức tạp. Do tài liệu tham khảo tiếng việt hạn chế, trình độ có hạn và kinh nghiệm trong thực tiễn còn non kém, nên đề tài chắc chắn còn nhiều thiếu sót.

Rất mong được nhận những ý kiến đóng góp, giúp đỡ chân tình, quý báu của quý thầy cô cùng các bạn sinh viên.

Tháng 2 năm 1999

Trương Phước Toàn



## LỜI CẢM ƠN

Con xin tỏ lòng biết ơn vô hạn đến ba mẹ và gia đình, những người thân yêu nhất, đã hết lòng dạy dỗ cho con ăn học nên người.

Con xin tỏ lòng biết ơn đến thầy hướng dẫn TRẦN VĂN TRỌNG đã tận tình chỉ dạy, hướng dẫn, nâng đỡ nhiều ý kiến quý báu và đã nhiều kiến thức cho em trong thời gian qua.

Em xin cảm ơn quý thầy cô trong khoa Niên - Niên tốt nghiệp Nữ học Sở phạm Kỹ thuật đã hết lòng dạy dỗ em trong thời gian học ở trường.

Xin cảm ơn các bạn cùng khoa đã tận tình giúp đỡ để tôi hoàn thành tốt luận văn này.

Sinh viên thực hiện

Trương Phước Toàn

# MỤC LỤC

Trang

<b>PHẦN I LÝ THUYẾT</b> .....	1
<b>CHƯƠNG I GIỚI THIỆU CÁC CÔNG LOGIC CƠ BẢN</b> .....	2
I/ CÔNG LOGIC VÀ (AND) ,HOẶC (OR) ,KHÔNG (NOT).. .....	& 2
1/ Công logic VÀ. ....	2
2/ Công logic HOẶC .....	2
3/ Công logic KHÔNG .. & .....	3
II/ CÔNG LOGIC KHÔNG-VÀ (NAND) ,KHÔNG-HOẶC (NOR).. ...	4
1/ Công NAND. ....	4
2/ Công NOR .....	4
III/ CÔNG LOGIC EXOR ,EXNOR. ....	5
1/Công EXOR .....	5
2/Công EXNOR. ....	5
IV/ BIẾN ĐỔI CÁC HÀM QUAN HỆ RA HÀM LOGIC NAND, NOR .	6
<b>CHƯƠNG II MẠCH LOGIC TỔ HỢP</b> .....	8
I/ ĐẶC ĐIỂM CƠ BẢN CỦA MẠCH TỔ HỢP .....	8
II/ PHƯƠNG PHÁP BIỂU THỊ VÀ PHÂN TÍCH CHỨC NĂNG LOGIC..8	
III/ PHƯƠNG PHÁP THIẾT KẾ LOGIC MẠCH TỔ HỢP .....	9
1/ Phân tích yêu cầu .....	9
2/ Lập bảng sự thật .....	9
3/ Tiến hành đơn giản hóa .....	11
<b>CHƯƠNG III GIỚI THIỆU VI MẠCH SỐ LẬP TRÌNH</b> .....	12
1/ LỊCH SỬ PHÁT TRIỂN CỦA VI MẠCH SỐ LẬP TRÌNH .....	12
2/ CẤU TRÚC CƠ BẢN CỦA CÁC HỘ VI MẠCH LẬP TRÌNH ...	16
3/ CÁC PHẦN MỀM HỖ TRỢ CỦA PLD .....	40
4/ GIỚI THIỆU PHẦN MỀM SYNARYO .....	44
<b>PHẦN II THI CÔNG</b> .....	48
<b>PHẦN III KẾT LUẬN</b> .....	60

# *PHẦN I*

# LÝ THUYẾT

## CHƯƠNG I : GIỚI THIỆU CÁC CÔNG LOGIC CƠ BẢN

### I/ HÀM LOGIC VÀ (AND) , HOẶC (OR) ,KHÔNG (NOT).

#### 1/ Cổng logic .

Gọi A là biến số nhị phân có mức logic là 0 hoặc 1, và Y là một biến số nhị phân tùy thuộc vào A:  $Y = f(A)$ .

Trong trường hợp này có hai khả năng xảy ra:

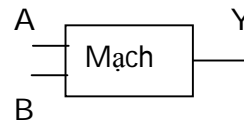
- $Y = A$ , A= 0 thì Y = 0  
hay A= 1 thì Y = 1
- $Y = \bar{A}$ , A= 0 thì Y = 1  
hay A= 1 thì Y = 0

Khi Y tùy thuộc vào hai biến số nhị phân A, B

$$\Rightarrow Y = f(A,B)$$

Vì biến số A,B chỉ có thể là 0 hay 1 nên A và B chỉ có thể tạo ra 4 tổ hợp khác nhau là:

A	B
0	0
0	1
1	0
1	1



Bảng liệt kê tất cả các tổ hợp khả dĩ của các biến số và hàm số tương ứng gọi là bảng sự thật. Khi có 3 hay nhiều biến số (A,B ,C) số lượng hàm số khả dĩ tăng nhanh.

Mạch điện tử thực hiện quan hệ logic :

$$Y = f(A) \text{ hay } Y = f(A,B).$$

gọi là mạch logic, trong đó các biến số A,B .. là các ngõ vào và hàm số Y là các ngõ ra. Một mạch logic diễn tả quan hệ giữa các ngõ vào và ngõ ra nghĩa là thực hiện được một hàm logic, do đó có bao nhiêu hàm số logic thì có bấy nhiêu mạch logic .

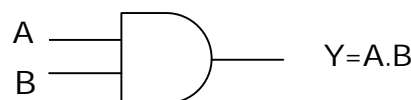
Lưu ý rằng khi biểu diễn mối quan hệ toán học ta gọi là hàm số logic còn khi biểu diễn mối quan hệ về mạch tín hiệu ta gọi là cổng logic.

#### 2/ Cổng logic VÀ (AND).

Hàm logic VÀ được định nghĩa theo bảng sự thật sau:

Bảng sự thật:

A	B	Y
0	0	0
0	1	0
1	0	0
1	1	1



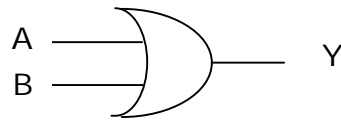
Ký hiệu toán học của hàm số VÀ.  
 $Y = A.B$

Kí hiệu cổng VÀ (AND)

**3/ Cổng logic HOẶC (OR).**

Hàm số HOẶC của hai biến số A,B được định nghĩa ở bảng sự thật sau:  
 Bảng sự thật:

A	B	Y
0	0	0
0	1	1
1	0	1
1	1	1



Kí hiệu cổng HOẶC

Ngõ ra Y là 1 khi có ít nhất một biến số là 1, do đó chỉ bằng 0 ở trường hợp khi cả hai biến số bằng 0.

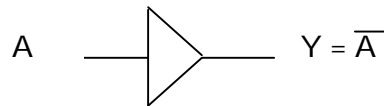
Ký hiệu toán học của cổng HOẶC là:  
 $Y = A+B$

**4/ Cổng logic KHÔNG (NOT).**

Hàm VÀ và hàm HOẶC tác động lên hai hay nhiều biến số trong khi đó hàm KHÔNG có thể xem như chỉ có thể tác động lên một biến số.

Bảng sự thật :

A	Y
0	1
1	0



Kí hiệu cổng NOT

Hàm KHÔNG có tác động phủ định hay đảo .Sở dĩ có sự đồng hóa này là vì ta đang liên hệ với số nhị phân có hai trạng thái 0 hay 1. Do đó phủ định của 0 là 1.

**III/ CÔNG LOGIC KHÔNG -VÀ (NAND) , KHÔNG-HOẶC (NOR).**

**1/ Cổng logic NAND .**

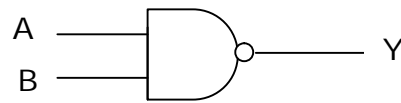
Xét trường hợp có hai biến số A,B ngõ ra ở cổng VÀ  $Y = A.B$  nên ngõ ra ở cổng KHÔNG là đảo của Y:

$$Y = A.B\overline{\quad}$$

Về hoạt động của cổng NAND thì từ các tổ hợp của A,B ta lập bảng trạng thái rồi lấy đảo để có Y đảo. Tuy nhiên có thể đi trực tiếp bằng cách lập bảng sự thật sau:

Bảng sự thật :

A	B	Y
0	0	1
0	1	1
1	0	1
1	1	0



Kí hiệu cổng NAND.

**2/ Cổng NOR.**

Xét trường hợp hai ngõ vào là A,B .Ngõ ra ở cổng NOR là :

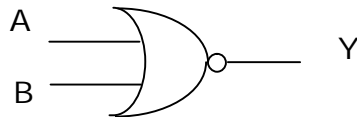
$$Y = A+B$$

nên ngõ ra ở cổng đảo sẽ là :

$$Y = \overline{A+B}$$

Bảng sự thật :

A	B	Y
0	0	1
0	1	0
1	0	0
1	1	0



Kí hiệu cổng NOR.

**III/ HÀM LOGIC EXOR VÀ EXNOR.**

**1/ Cổng logic EXOR.**

Hàm HOẶC được gọi là HOẶC bao gồm vì nó không giải quyết được bài toán cộng nhị phân. Lý do là khi cả hai biến số đều là 1 thì  $Y = 1$  thay vì là 0. Mặc dù HOẶC như vậy vẫn có ý nghĩa thực tế nên vẫn được dùng, nhưng người ta phải định nghĩa một cổng logic khác là HOẶC LOẠI TRỪ (EXOR) cổng này có ý nghĩa là loại trường hợp khi A,B đồng thời là 1 thì  $Y = 0$

Ký hiệu :  $Y = A \oplus B$

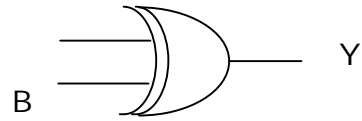
Bảng sự thật:

A	B	Y
0	0	0
0	1	1
1	0	1
1	1	0

A



0	0	0
0	1	1
1	0	1
1	1	0



Kí hiệu cổng EXOR.

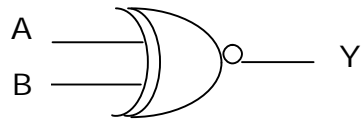
**2/ Cổng EXNOR.**

Hàm EXNOR được thực hiện bằng cách thêm cổng NOT sau cổng EXOR, do đó hoạt động logic của EXNOR là đảo so với EXOR.

Ký hiệu :  $Y = A \oplus B$

Bảng sự thật:

	B	Y
0	0	1
0	1	0
1	0	0
1	1	1



**IV/ BIẾN ĐỔI CÁC HÀM QUAN HỆ RA HÀM LOGIC NAND , NOR.**

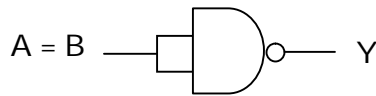
Mối liên hệ cơ bản giữa ba cổng AND, OR, NOT không những có thể thay bằng các cổng NAND mà còn có thể biến thành cổng NOR với cùng một chức năng logic, việc làm này rất thường được áp dụng khi thực hiện các mạch logic. Trong thực tế vì toàn bộ sơ đồ nếu được kết hợp cùng một loại cổng duy nhất thì sẽ giảm được số lượng vi mạch cần thiết. Quá trình biến đổi này dựa trên một nguyên tắc được trình bày như sau:

+ Cổng NOT được thay bằng cổng NAND và cổng NOR.

- Dựa vào bảng sự thật của cổng NAND suy ra trường hợp là khi cả A,B đồng thời bằng 0, thì  $Y = 1$

và  $A = 1, B = 1$  thì  $Y = 1$ .

Sơ đồ minh họa :

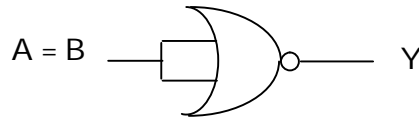


Tương tự dựa vào bảng sự thật của cổng NOR suy ra :

$A = 0, B = 0 \Rightarrow Y = 1$

và  $A = 1, B = 1 \Rightarrow Y = 0$

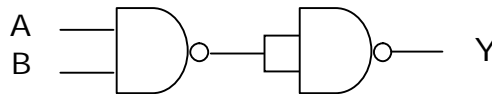
Sơ đồ minh họa :



+ Cổng AND được thay bằng cổng NAND và cổng NOR. Tương tự như các trường hợp trên, dựa vào bảng sự thật:

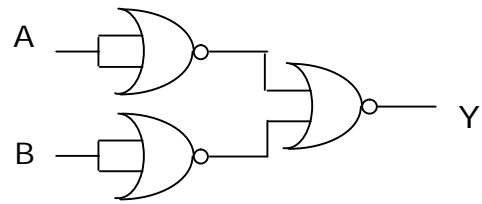
- Ngõ ra của cổng AND  $Y = A \cdot B$  còn cổng NAND  $Y' = \overline{A \cdot B} \Rightarrow$  đảo  $Y' = Y$

Sơ đồ minh họa:

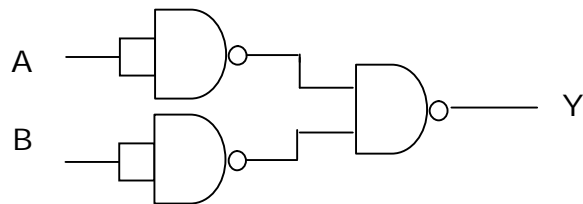


- Ngõ ra của cổng NOR  $Y = \overline{A \cdot B}$ . Ta có  $Y = A \cdot \overline{\overline{B}} = \overline{\overline{A} + B}$

Sơ đồ minh họa :

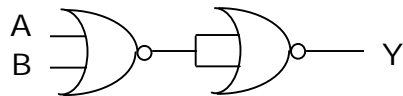


+ Cổng OR được thay bằng cổng NAND và cổng NOR. Biểu thức cổng OR  $Y = A.B$ ,  $\Rightarrow Y' = A + \overline{B} = \overline{\overline{A}.B}$   
 Sơ đồ minh họa :



- Biểu thức cổng NOR  $Y' = A.B \Rightarrow Y = \overline{\overline{A}.B} = Y$

Sơ đồ minh họa :



## CHƯƠNG II MẠCH LOGIC TỔ HỢP

### I/ ĐẶC ĐIỂM CƠ BẢN CỦA MẠCH TỔ HỢP.

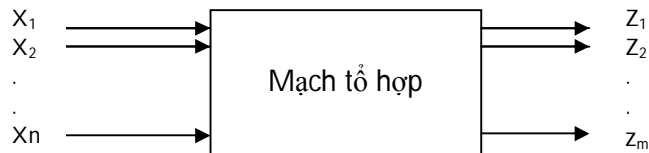
Trong mạch số, mạch tổ hợp là mạch mà trị số ổn định của tín hiệu ra ở thời điểm bất kỳ chỉ phụ thuộc vào tổ hợp các giá trị tín hiệu ngõ vào ở thời điểm đó. Trong mạch tổ hợp, trạng thái mạch điện trước thời điểm xét, tức trước khi có tín hiệu ngõ vào, không ảnh hưởng đến tín hiệu đầu ra. Đặc điểm cấu trúc mạch tổ hợp là được cấu trúc từ các cổng logic.

### II/ PHƯƠNG PHÁP BIỂU THỊ VÀ PHÂN TÍCH CHỨC NĂNG LOGIC .

#### 1/ Phương pháp biểu thị chức năng logic.

Các phương pháp thường dùng để biểu thị chức năng logic của mạch tổ hợp là hàm số logic, bảng sự thật, sơ đồ logic, bảng Karnaugh, cũng có khi biểu thị bằng đồ thị thời gian dạng sóng.

Đối với vi mạch cỡ nhỏ (SSI) thường biểu thị bằng hàm logic. Đối với cỡ vừa thường biểu thị bằng bảng sự thật, hay là bảng chức năng. Bảng chức năng dùng hình thức liệt kê, với mức logic cao (H) và mức logic thấp (L), để mô tả quan hệ logic giữa tín hiệu ngõ ra với tín hiệu ngõ vào của mạch điện đang xét. Chỉ cần thay giá trị logic cho trạng thái trong bảng chức năng, thì ta có bảng sự thật tương ứng.



Hình 2-1 : Sơ đồ khối mạch tổ hợp

Như hình 2-1 cho biết, thường có nhiều tín hiệu ngõ vào và nhiều tín hiệu ngõ ra. Một cách tổng quát, hàm logic của tín hiệu ngõ ra có thể viết dưới dạng :

$$\begin{aligned} Z_1 &= f_1(x_1, x_2, \dots, x_n) \\ Z_2 &= f_2(x_1, x_2, \dots, x_n) \\ &\dots\dots\dots \\ Z_m &= f_m(x_1, x_2, \dots, x_n) \end{aligned}$$

Cũng có thể viết dưới dạng đại lượng vector như sau:

$$Z = F(X).$$

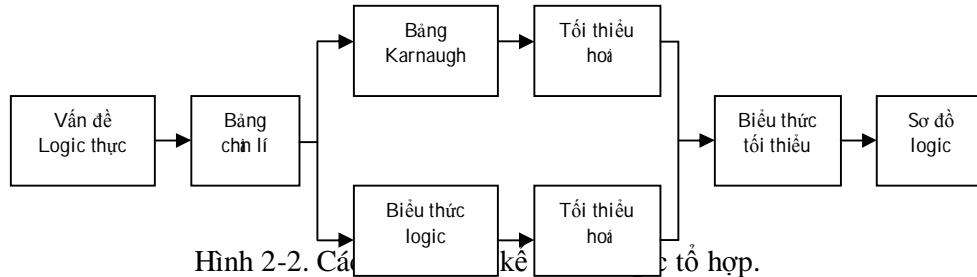
#### 2/ Phương pháp phân tích chức năng logic.

Các bước phân tích, bắt đầu từ sơ đồ mạch logic đã cho, để cuối cùng tìm ra hàm logic hoặc bảng sự thật.

- Viết biểu thức: tuân tự từ ngõ vào đến ngõ ra (hay cũng có thể ngược lại), viết ra biểu thức hàm logic của tín hiệu ngõ ra.
- Rút gọn: khi cần thiết thì rút gọn đến tối thiểu biểu thức ở trên bằng phương pháp đại số hay phương pháp hình vẽ.
- Kê bảng sự thật: khi cần thiết thì tìm ra bảng sự thật bằng cách tiến hành tính toán các giá trị hàm logic tín hiệu ngõ ra tương ứng với tổ hợp có thể của các giá trị tín hiệu ngõ vào.

**III/ PHƯƠNG PHÁP THIẾT KẾ LOGIC MẠCH TỔ HỢP.**

Phương pháp thiết kế logic là các bước cơ bản tìm ra sơ đồ mạch điện logic từ yêu cầu nhiệm vụ logic đã cho.



Hình 2-2. Các bước thiết kế mạch tổ hợp.

Hình 2-2 là quá trình thiết kế nói chung của mạch tổ hợp, trong đó bao gồm 4 bước chính :

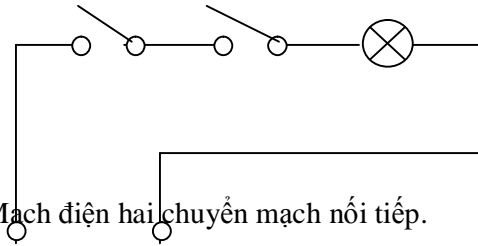
**1/ Phân tích yêu cầu:**

Yêu cầu nhiệm vụ thiết kế của vấn đề logic thực có thể là một đoạn văn, cũng có thể là bài toán logic cụ thể. Nhiệm vụ phân tích là xác định cái nào là biến số ngõ vào, cái nào là hàm số đầu ra và mối quan hệ logic giữa chúng với nhau. Muốn phân tích đúng thì phải tìm hiểu xem xét một cách sâu sắc yêu cầu thiết kế, đó là một việc khó nhưng quan trọng trong vấn đề thiết kế.

**2/ Kê bảng sự thật :**

Nói chung, đầu tiên chúng ta liệt kê thành bảng về quan hệ tương ứng nhau giữa trạng thái tín hiệu ngõ vào với trạng thái hàm số ngõ ra. Đó là bảng kê yêu cầu chức năng logic, gọi tắt là bảng chức năng. Việc này có vẻ dễ và trực quan. Tiếp theo, ta thay giá trị logic cho trạng thái, tức là dùng các số 0 và 1 biểu thị các trạng thái tương ứng của ngõ vào và ngõ ra. Kết quả ta có bảng giá trị thực logic, gọi tắt là bảng sự thật. Đây chính là hình thức đại số của yêu cầu thiết kế. Cần lưu ý rằng từ một bảng chức năng có thể được bảng sự thật khác nhau nếu thay giá trị logic khác nhau (tức là quan hệ logic giữa ngõ ra với ngõ vào cũng phụ thuộc việc thay giá trị ).

Ví dụ: Sơ đồ mạch nguyên lí hình 2-3 dùng hai chuyển mạch A,B mắc nối tiếp điều khiển bóng đèn Y.



Hình 2-3. Mạch điện hai chuyển mạch nối tiếp.  
Bảng sự thật

A	B	Z
0	0	0
0	1	0
1	0	0
1	1	1

Bảng sự thật trên có được từ xem trực tiếp các khả năng có thể của mạch định hình 2-3. Nếu thay thế giá trị logic theo 4 cách khác nhau thì từ các bảng sự thật a, b, c, d ta được các biểu thức logic khác nhau.

Bảng sự thật trong 4 tình huống thay giá trị khác nhau.

A	B	Z
0	0	0
0	1	0
1	0	0
1	1	1

a)  $Z = A.B$

A	B	Z
1	1	1
1	0	1
0	1	1
0	0	0

b)  $Z = A + B$

A	B	Z
0	0	1
0	1	1
1	0	1
1	1	0

c)  $Z = \overline{A.B}$

A	B	Z
1	1	0
1	0	0
0	1	0
0	0	1

d)  $Z = \overline{A + B}$

Từ bảng sự thật trên, ta thấy rằng chúng ta sẽ có mối quan hệ logic khác nhau nếu thay giá trị theo cách khác nhau. Chúng ta phải căn cứ vào giá trị thay thế trạng thái để xác định ý nghĩa cụ thể của 0 và 1 (tức là ý nghĩa cụ thể của bảng sự thật).

Khi liệt kê bảng chức năng hoặc bảng sự thật, có thể không liệt kê các tổ hợp trạng thái tín hiệu ngõ vào nào không thể có hay bị cấm. Những tổ hợp này cũng có thể được liệt kê, nhưng tại ngõ ra, ở trạng thái tương ứng ta ghi một dấu chéo " X ", thường sử dụng các trạng thái đánh dấu chéo để tối thiểu hoá hàm logic.

### 3/ Tiến hành tối thiểu hoá.

Nếu số biến số tương đối ít thì có thể dùng phương pháp hình vẽ. Nếu số biến số tương đối nhiều, khi đó không tiện dùng phương pháp hình vẽ, thì dùng phương pháp đại số.

## CHƯƠNG III VI MẠCH SỐ LẬP TRÌNH

### I/ LỊCH SỬ PHÁT TRIỂN CỦA VI MẠCH SỐ LẬP TRÌNH .

Trước thời kỳ vi mạch số lập trình (Programmable Logic Device) ra đời, thiết kế logic số truyền thống thì bao gồm nhiều vi mạch TTL loại MSI và SSI kết hợp lại để tạo ra các hàm logic mong muốn. Những nhà thiết kế dựa vào những sách tra cứu các vi mạch số để tìm hiểu các thông số kỹ thuật, sau đó mới quyết định sử dụng các vi mạch số cần thiết cho yêu cầu thiết kế của họ. Điều bất lợi của việc thiết kế này là trong một board sử dụng nhiều vi mạch, do đó khi sửa chữa thì gặp nhiều khó khăn.

Vào năm 1975, công ty SIGNETICS đã giới thiệu vi mạch số lập trình không có bộ nhớ đầu tiên 82S100 (hiện nay là PLS100) gọi là mảng logic lập trình trường (Field-Programmable Logic Array). Napoleon Cavlan, người được gọi là cha đẻ của mạch logic lập trình, lúc bấy giờ là nhà quản lý những ứng dụng PLA của Signetics đã thực sự hiểu rằng sử dụng PLA là phương pháp tốt hơn để thiết kế và thay đổi hệ thống số. Trong khi đó, công ty Harris đã sớm giới thiệu PROM, họ trình bày triển vọng của PROM và đã ứng dụng vào trong một số mạch logic.

Công ty National Semiconductor đã chế tạo mặt nạ lập trình cho PLA, cấu tạo của nó gồm một mảng AND lập trình kèm với mảng OR lập trình, cho phép thực hiện tổ hợp tổng các tích số của hàm logic tiêu chuẩn. Bằng cách kết hợp công nghệ PROM sử dụng nguyên tắc cầu chì với khái niệm PLA, Cavlan đã thuyết phục được các nhà quản lý công ty Signetics để đưa dự án PLAvào sản xuất.

Vi mạch PLA đầu tiên 82S100, là thành viên đầu tiên của họ vi mạch IFL (Intergrated Fuse Logic) có hình dạng 28 chân. Cấu trúc của PLA gồm một mảng AND lập trình và một mảng OR lập trình, nó cho phép thực hiện tổ hợp logic tổng của các tích số đơn giản .

Kỹ sư John Martin Birkner là một người quan tâm đến PLA, vì ông ấy hiểu rằng nhiều phương pháp thiết kế logic được học trong trường thì không áp dụng được nhiều trong công việc hiện tại. Do đó, vào năm 1975 ông ấy đã rời thung lũng Silicon để đến công ty Monolithic Memories (MMI), đây là công ty chế tạo PROM và các vi mạch logic tiêu chuẩn. Vì vậy, Birkner có điều kiện hơn trong việc tìm hiểu PLA và công nhận những ưu điểm của mạch logic lập trình nhưng đồng thời ông cũng nhận ra khuyết điểm của PLA là có hai mảng lập trình. Sau đó, Birkner đã đưa ra khái niệm mới về vi mạch số lập trình, vi mạch này cũng tương tự FLA nhưng thay vì có hai mảng lập trình thì PAL (Programmable Array Logic ) chỉ có một mảng AND lập trình và theo sau là mảng OR được giữ cố định (không lập trình). Như vậy mỗi cổng OR sẽ có một tích số cố định được nối với ngõ vào của nó, do vậy sẽ giảm được kích thước của vi mạch và cho phép tín hiệu được truyền nhanh hơn trong khi vẫn cho phép thực hiện các tổ hợp logic. PAL được đóng vỏ 20 chân. Sau một thời gian thuyết phục các nhà quản lý của công ty MMI thấy rõ những lợi điểm của PAL và đồng ý sản xuất. Vi mạch đầu tiên thuộc họ PAL được phổ biến là PAL 16L8, PAL 16R4, PAL 16R6, PAL 16R8. Các vi mạch này có thời gian truyền trì hoãn 35ns. Mỗi vi mạch có 8 ngõ ra và 16 ngõ vào, trong đó ký tự L trong ký hiệu của vi mạch biểu thị 8 tổ hợp ngõ ra tác động ở mức thấp, ký tự R cho biết có 4, 6 hay 8 thanh ghi ở ngõ ra tương ứng.

Sau một thời gian khởi đầu chậm, cuối cùng PAL đã được thiết kế trong hệ thống thực. Những công ty máy tính mini đã nhận thấy được ưu điểm của PAL là cho phép họ giảm số board cần thiết để thực hiện tốt những yêu cầu thiết kế, công ty MMI đã chọn phương pháp sản xuất PAL công đoạn mặt nạ chế tạo theo yêu cầu khách hàng. Vào lúc này MMI lại giới thiệu một họ vi mạch mới HAL (Hard Array Logic) và đề sản xuất những chi tiết này cho hãng Data General and Digital Equipment. MMI đã thay đổi cách sắp xếp công đoạn mặt nạ cầu chì và thay vào đó là lớp liên kết kim loại phù hợp yêu cầu

thiết kế của khách hàng. Những chi tiết này có nhiều lợi ích gồm mang lại những kết quả tốt và kiểm tra dễ dàng hơn. Đồng thời khách hàng cũng được lợi hơn bởi không phải quan tâm đến lập trình và kiểm tra các chi tiết. Điều này đã mang lại sự cải tiến về phương pháp chế tạo PAL, và được sự chấp nhận của thị trường. Vào năm 1978, MMI đã xuất bản sách hướng dẫn PAL đầu tiên. Đó là một bước khởi đầu để PAL mở rộng thế giới của những người thiết kế mạch logic. Ngoài ra trong sách hướng dẫn còn trình bày danh sách chương trình gốc của ngôn ngữ lập trình FORTRAN cho PALASM (PAL Assembler) đó là phần mềm dành cho việc thiết kế mạch logic PAL. PALASM có thể biên soạn, định nghĩa logic cho một khuôn thức. Ngoài ra PALASM cũng có khả năng mô phỏng sự vận hành trên phương trình mạch logic theo nguyên tắc PAL. Trong việc liên kết với những nhà thiết kế để định rõ những “vector kiểm tra”, PALASM có thể là một sự thật phù hợp. Tất cả những đặc điểm của PAL bao gồm việc khắc phục những khuyết điểm của PLA kết hợp với việc thúc đẩy sử dụng PAL đã mang đến kết quả tốt đẹp. PAL đã nhanh chóng vượt qua họ vi mạch IFL của công ty Signetics và được phổ biến trên thị trường, thuật ngữ PAL đã trở nên đồng nghĩa với PLD.

Trong lúc ấy, công ty Signetics tiếp tục phát triển họ IFL, và vào năm 1977 Signetics giới thiệu họ vi mạch FPGA (Field Programmable Gate Array) 82S103, vào năm 1979 là họ FPLS (Field Programmable Logic Sequencer). Họ FPGA có cấu tạo một mảng AND ở mức đơn với ngõ vào lập trình được và cực tính ngõ ra cũng vậy cho phép thực hiện các hàm logic cơ bản (AND, OR, NAND, NOR, INVERT), cấu trúc của họ FPLS có chức các FlipFlop để thực hiện các trạng thái của hàm tuần tự. Đồng thời Signetics cũng giới thiệu AMAZE (Automated Map and Zap Equations) là chương trình biên dịch để hỗ trợ cho những vi mạch của họ. Tương tự, những công ty chế tạo PLD khác đã lần lượt giới thiệu những phần mềm hỗ trợ của họ.

Cả 2 công ty Signetics và MMI tiếp tục giới thiệu những PLD mới để đáp ứng tính đa dạng theo các yêu cầu thiết kế. Vào giữa năm 1980, mạch logic lập trình đã được thừa nhận cùng với sự phát triển tính đa dạng của IFL và PAL đã có nhiều giá trị cho những người thiết kế. Mặc dù sự khởi đầu thành công của PLD, tuy nhiên chỉ một số ít các nhà thiết kế quen với việc dùng PLD, một số trường đại học đã đưa vi mạch logic lập trình vào những khóa học thiết kế của họ.

Tuy thế, kĩ thuật logic lập trình tiếp tục cải tiến và những vi mạch phát triển ở giai đoạn thứ hai được giới thiệu vào năm 1983. Công ty Advance Micro Devices (AMD) đã giới thiệu PAL22V10 với những đặc điểm đặc biệt là sự linh động của những cổng PLD ở 10 ngõ vào. Mỗi cổng PLD có khả năng tổ hợp hoặc với thanh ghi ở ngõ ra hoặc một ngõ vào. Cổng đệm ngõ ra ba trạng thái được điều khiển bởi một tích số riêng cho phép vận hành hai chiều. Tất cả thanh ghi đều được reset tự động trong quá trình tắt hay mở và mỗi thanh ghi có khả năng “đặt trước”, đó là đặc điểm đặc biệt cho việc kiểm tra sau này.

Với những vi mạch mới, được giới thiệu thường xuyên trên thị trường đã dẫn đến việc cần thiết phải có một phần mềm hỗ trợ trong quá trình sử dụng PLD để đạt hiệu quả cao.

Bob Osann đã nhận thấy được sự cần thiết của một chương trình biên dịch PLD vận hành dùng cho tất cả PLD của những công ty chế tạo khác nhau.

Vào tháng 9/1983, Công ty Assisted Technology đã đưa ra phiên bản 1.01a của chương trình biên dịch PLD có tên là CUPL (Universal Compiler for Programmable). Chương trình này hỗ trợ cho 29 loại vi mạch, sự ra đời của CUPL đã gây được sự chú ý của nhiều công ty chế tạo. Công ty Data I/O, nhà chế tạo các vi mạch lập trình lớn nhất trên thế giới (EPROM, PROM, PLD), đã quyết định phát triển phần mềm hỗ trợ cho riêng họ. Năm 1984, Data I/O giới thiệu ABEL (Advanced Boolean Expression Language), đó là chương trình biên dịch PLD có đặc điểm tương tự như CUPL nhưng nó được đầu tư tiếp thị nên được các nhà thiết kế chấp nhận. Vì vậy, ABEL đã sớm theo kịp CUPL trên thị trường.



Sự ra đời của chương trình biên dịch vạn năng cho PLD đã thúc đẩy nền công nghiệp thiết kế số sẵn sàng cho việc áp dụng PLD cho những thiết kế mới. Những chương trình biên dịch vạn năng này đã được cải tiến hơn so với các chương trình biên dịch PALASM và AMAZE, nó được cung cấp cho các nhà thiết kế để thực hiện các mạch logic và mô phỏng những thiết bị. Đó là những đặc điểm tiêu chuẩn của hai bộ biên dịch vạn năng CUPL và ABAL. JEDEC ( the Joint Electron Device Engineering Council) dự định sản xuất một bộ biên dịch PLD tạo ra một tiêu chuẩn để sử dụng cho tất cả các công ty chế tạo PLD hiện nay và tương lai. Vào 10/1983, the JEDEC Solid State Products Engineering Council đưa ra tiêu chuẩn JEDEC thứ 3“. Tiêu chuẩn khuôn thức chuyển đổi giữa hệ thống tạo dữ liệu và thiết bị lập trình cho PLD”. Tháng 5/1986, JEDEC tiếp tục đưa ra tiêu chuẩn 3-A, tiêu chuẩn này trở thành tiêu chuẩn chung cho công nghiệp PLD.

Tháng 7/1984, công ty Altera giới thiệu EP300. Đó là vi mạch sử dụng công nghệ CMOS của EPROM, nó có đặc tính là công suất tiêu thụ thấp, có thể xóa được (dùng tia cực tím) cùng một số đặc tính mở rộng khác. Năm 1985, một họ PLD mới được công ty Lattice Semiconductor giới thiệu là GAL (Generic Array Logic). Lattice dùng công nghệ CMOS của EEPROM, có các đặc tính kỹ thuật như công suất thấp, có thể lập trình nhiều lần ( xóa bằng điện áp với thời gian xóa khoảng vài giây). Vi mạch đầu tiên của họ GAL được kí hiệu là GAL16V8 có khả năng thay thế hoạt động của PAL (đối với vi mạch cùng loại).

Ngày càng nhiều công ty tham gia vào thị trường PLD để tạo ra những vi mạch đặc biệt và sử dụng nhiều công nghệ chế tạo khác nhau. Vào năm 1985, công ty Xilen tạo ra một họ mới là LCA (Logic Call Array). Cấu trúc của LCA có 3 đoạn: một ma trận của khối logic được bao quanh là khối vào ra và một mạng đường dữ liệu nối gián tiếp. Đặc biệt của LCA là PLD đầu tiên sử dụng tế bào RAM động cho chức năng logic. Ưu điểm của cấu trúc này là khách hàng có thể kiểm tra được chương trình của vi mạch, do bản chất dễ xóa của LCA, nên cần phải lưu trữ cấu hình của LCA ở bộ nhớ ngoài. Vì vậy, LCA không được sử dụng ở những trường hợp đòi hỏi sự hoạt động ngay lập tức khi khởi động máy. Đi kèm với LCA là chương trình soạn thảo XACT và bộ mô phỏng giúp cho việc sửa lỗi cho những thiết kế trên LCA được thuận tiện.

Năm 1985, công ty Signetics với một khái niệm mới là PML (Programmable Macro Logic). Vi mạch PML đầu tiên của Signetics PMLS 501, vi mạch này sử dụng công nghệ lưỡng cực, và được đóng vỏ 52 chân .

Vào năm 1986, công ty ExMicroelectronic giới thiệu họ ERASIC (Erasable Application Specific 7C) sử dụng công nghệ EEPROM CMOS. Vi mạch đầu tiên là XL78C00 có dạng 24 chân và điều đặc biệt là XL78C00 có thể thay thế chức năng cho PAL và EPLA cùng loại (không tính đến tốc độ), đi kèm là một phần mềm hỗ trợ ERASIC.

Vào năm 1986, công ty Signetics quyết định thay đổi họ IFL thành họ PLS (Programmable Logic From Signetics). Ví dụ như từ 82S100 thành PLS100, từ 82S157 thành PLS157. Sau đó 2 năm, công ty Actel đã cải tiến khuyết điểm họ LCA là vi mạch có thể hoạt động không nhất thiết phải có bộ nhớ ngoài. Đồng thời công ty Gazelle Microcircuit đã công bố phát minh công nghệ GaAs

(Gallium Arsenide). Đặc điểm của công nghệ này là cải tiến tốc độ , công suất của các vi mạch trên nền tảng là công nghệ silicon, cho phép vi mạch làm việc với tốc độ nhanh hơn công suất tiêu tán khi ở mức trung bình.

Ứng dụng đầu tiên của công nghệ GaAs được công ty Gazelle đưa ra là phiên bản của PAL 22V10. Ưu điểm của mạch này là cho phép vi mạch GaAs có thể tương hợp với các vi mạch TTL, do đó công nghệ GaAs đã được ứng dụng rộng rãi. Sau một thời gian cải tiến không ngừng, những PLD thế hệ sau đã được ứng dụng rộng rãi trong kỹ thuật phần cứng, nó trở thành công cụ cần thiết cho những kỹ sư thiết kế.

Sự phát triển trong công nghiệp PLD nói riêng và với công nghiệp bán dẫn nói chung đã tạo nên sự cạnh tranh của các công ty chế tạo PLD trên thế giới. Do đó, đã có nhiều xung đột xảy ra giữa các công ty trong việc cạnh tranh thị trường.

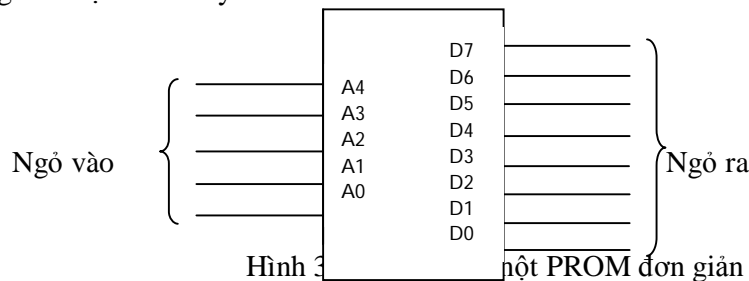
Vào năm 1986 công ty MMI đã kiện hai công ty Altera và Lattice vì đã vi phạm bản quyền PAL. Kết quả là hai công ty này đã chấp nhận thua kiện và phải mua bản quyền. Sau đó công ty MMI mua cổ phần trong công ty Xilinx và sở hữu bản quyền họ LCA. Sau đó 1 năm công ty MMI hợp với AMD trở thành một tập đoàn sản xuất các linh kiện bán dẫn hàng đầu trên thế giới. Tuy đã hợp nhất hai công ty nhưng họ vẫn tiếp tục phát triển các họ vi mạch hiện có vì những họ PLD này đã trở nên phổ biến trên thị trường. Vào năm 1987, công ty National Semiconductor đã mua lại công ty Fairchild và tiếp tục phát triển họ PAL FASTPLA trên thị trường.

**II/ CẤU TRÚC CƠ BẢN CỦA CÁC HỌ VI MẠCH LẬP TRÌNH (PLD).**

Vi mạch số lập trình trải qua thời gian dài phát triển và cải tiến đã thực sự mở ra một hướng đi mới cho những nhà thiết kế. Ưu điểm của PLD là giải quyết được vô số những vấn đề thiết kế nhờ vào nhiều họ PLD khác nhau. Những họ vi mạch này có cấu trúc và công nghệ chế tạo khác nhau, do đó chúng có những đặc điểm riêng để ứng dụng vào nhiều lĩnh vực trong công nghiệp. Mặc khác người thiết kế còn quan tâm đến các thông số kỹ thuật của vi mạch như tốc độ, công suất tiêu thụ, nguồn cung cấp và công cụ hỗ trợ để lập trình.

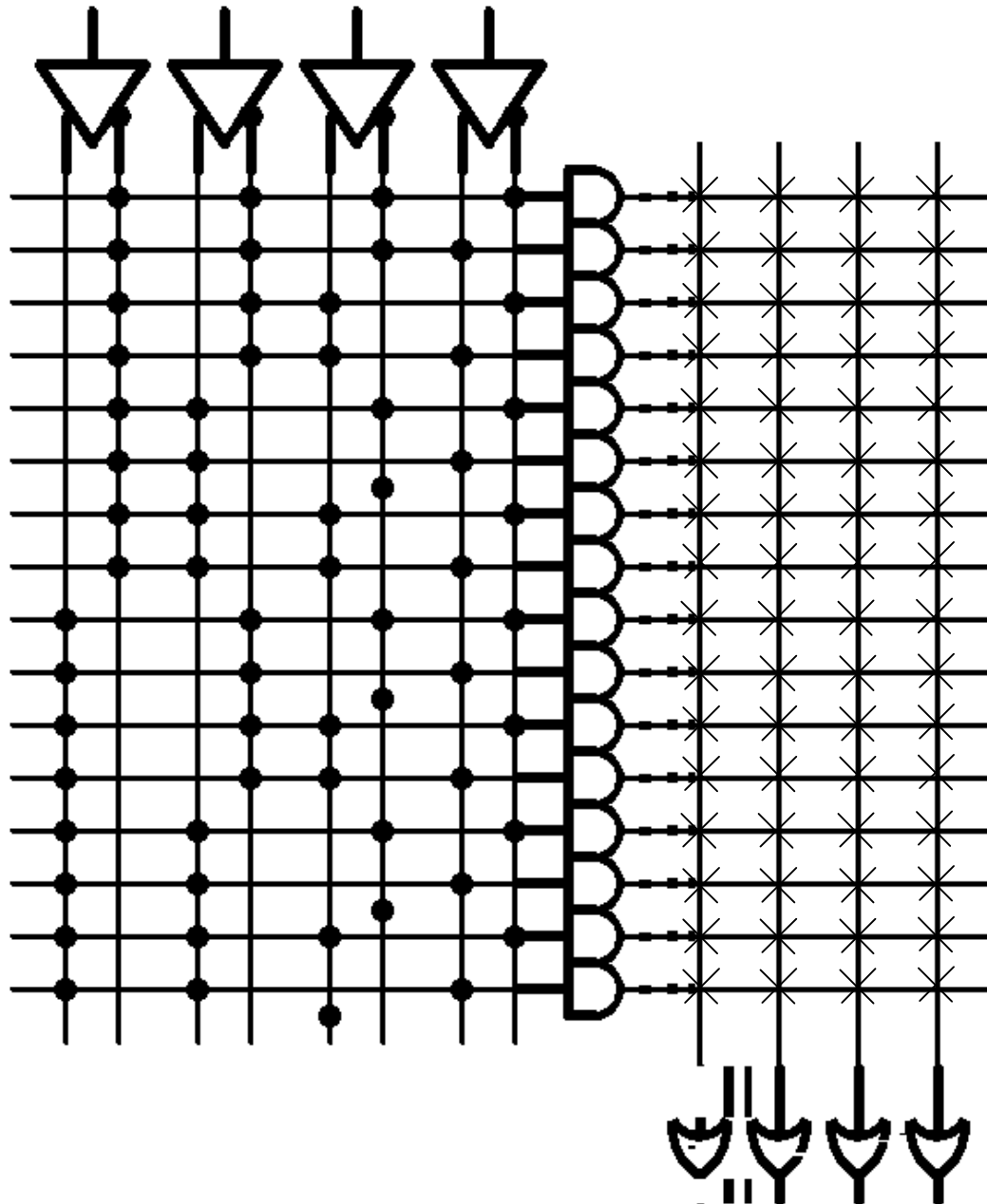
**1. Họ vi mạch PROM (Programmable Read Only Memory).**

PROM gọi là bộ nhớ chỉ đọc lập trình được. Đây là họ vi mạch đầu tiên được sử dụng như là những vi mạch số lập trình theo quan điểm của vi mạch số. Cấu trúc của PROM rất đơn giản bao gồm một mảng tế bào nhớ với những đường địa chỉ ngõ vào và những đường dữ liệu ngõ ra. Số đường địa chỉ và dữ liệu cho biết ma trận nhớ của PROM. Một PROM đơn giản được trình bày ở hình 3.1



PROM có 5 đường điều khiển ngõ vào cho phép tạo ra 32 tổ hợp logic và 8 đường dữ liệu ra tạo thành một ma trận nhớ 32x8, vì vậy có tổng cộng 256 tế bào nhớ. Cấu trúc của PROM gồm một mảng AND cố định theo sau là mảng OR lập trình, được minh họa ở hình 3.2.

Hình 3-7 Sơ đồ logic của PROM



Chú thích:

- Dấu X trong hình biểu hiện những điểm lập trình (được kết nối thông qua một cầu chì)

Hình 3.2 Sơ đồ logic của PROM

- D:

Ở mức

làng

OR. Do đó bất kỳ một liên kết nào bị loại bỏ (nghĩa là cầu chì ở đó bị đứt, thì biến đó sẽ không có mặt ở biểu thức ngõ ra).

Các hàm ở ngõ ra thay đổi tùy thuộc vào sự kết nối của các biến ở ngõ vào.

PROM thường được sử dụng để giải mã địa chỉ và ứng dụng để lưu trữ dữ liệu. Khi thiết kế các PROM, người thiết kế phải chú ý đến sự thay đổi mức logic ngõ vào (xây ra trong thời gian ngắn) khi địa chỉ ngõ vào thay đổi. Phương thức ghi của PROM là khi có một tín xung clock đồng bộ thì mạch ngõ ra chuyển sang trạng thái khác. Đặc điểm này sẽ giúp khắc phục được vấn đề trễ nhiệm ở PROM.

Khi khảo sát PROM, người ta thường quan tâm đến tốc độ truy xuất dữ liệu. Thông thường các loại PROM có thời gian truy xuất dưới 60 ns. Các loại PROM thường sử dụng công nghệ lưỡng cực là nguyên tắc cơ bản để chế tạo. Tuy nhiên, khoa học tiến bộ đã phát minh ra công nghệ CMOS cho phép rút ngắn thời gian truy xuất. Công nghệ CMOS được dùng để chế tạo EPROM, đó là một dạng PROM có thể xóa được bằng tia cực tím. Nó đã tạo ra một bước tiến đáng kể như: EPROM WS57C256F của công ty WaferScale Integration có dung lượng 32Kx8 với thời gian truy xuất là 55 ns, công ty Cypress Semiconductor giới thiệu PROM CY7C245 có dung lượng là 2048x8 với thời gian truy xuất là 25 ns.

Trên đây là một vài ví dụ cho thấy công nghệ CMOS được chấp nhận cho những ứng dụng thiết kế mạch.

## 2. Họ vi mạch FPLA ( Field Programmable Logic Array)

Họ vi mạch FPLA đầu tiên được công ty Signetics giới thiệu vào năm 1975. Cấu trúc của FPLA là một mảng AND – OR đơn giản, được trình bày ở hình 3. 3.

Mảng AND – OR có thể lập trình để thực hiện 4 hàm logic bất kỳ với hai biến ngõ vào. Mỗi biến ngõ vào được đưa qua cổng đệm để tạo hai mức logic 0 và 1. Mỗi mức logic này được nối với ngõ vào cổng AND thông qua một cầu chì lập trình. Tất cả 4 cầu chì được giữ nguyên.

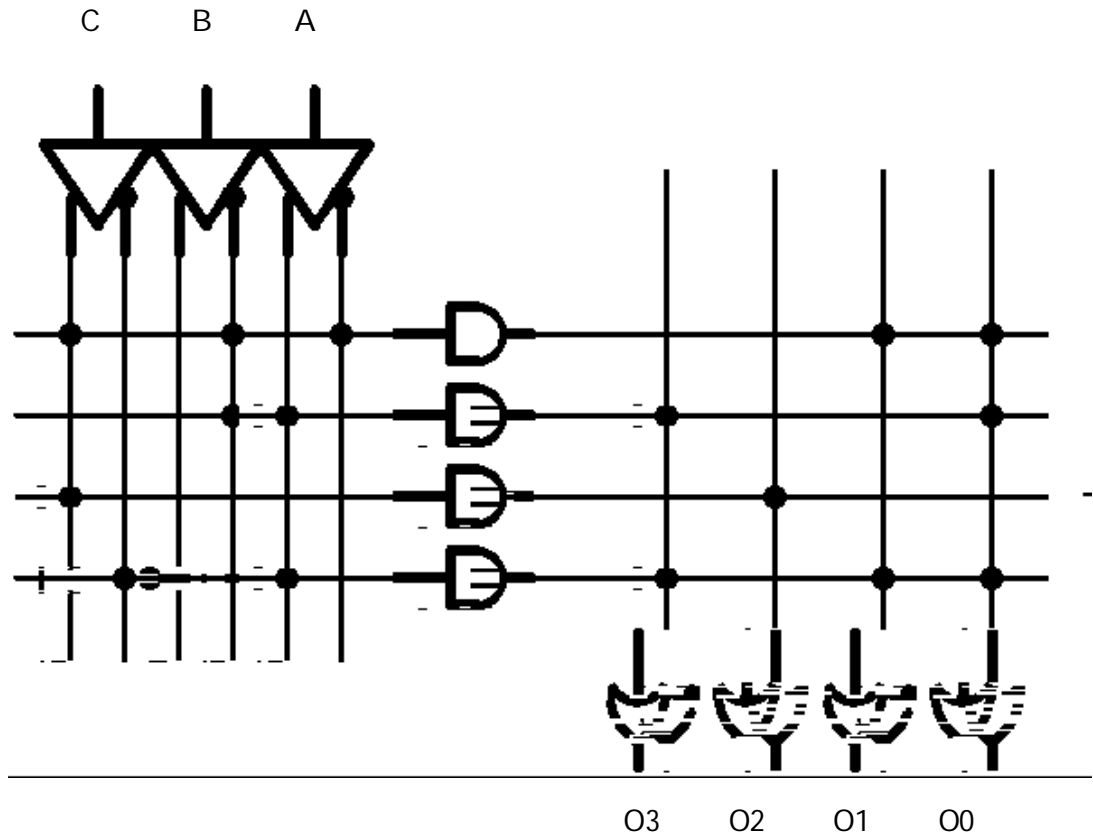
Nếu tất cả cầu chì đều thông, ví dụ như cổng ANDK thì biểu thức ngõ ra cho cổng sẽ là:

$$K = A \text{ AND } \overline{A} \text{ AND } B \text{ AND } \overline{B} = A\overline{A}B\overline{B}$$

Từ kết quả trên cho thấy ngõ ra của cổng AND luôn ở mức thấp, điều này không có lợi. Tuy nhiên nếu ta lập trình cho 4 cầu chì trên, ví dụ ta chọn A x B, lúc này giá trị của 2 biến này sẽ không có trong biểu thức.

Biểu thức ngõ ra cổng AND K là:  $K = A.B$

Nguyên tắc ở đây là lựa chọn những giá trị để lập trình, khi một cầu chì được chọn nghĩa là giá trị của nó sẽ không có mặt trong biểu thức.



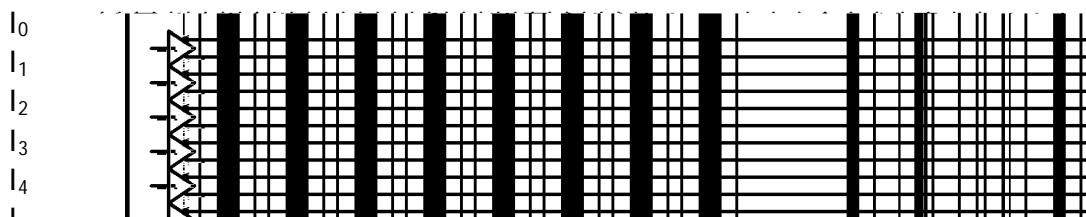
Hình 3.3 . Sơ đồ biểu thức ngõ ra của FPLA

Lưu ý mảng OR trong mạch ở hình 3.4. Mỗi ngõ ra cổng AND được nối tới 1 ngõ vào cổng OR thông qua một cầu chì và một Diode. Xét biểu thức F1 giả sử các cầu chì đều thông, ta có :

$$F1 = K + L + M + N$$

Với K,L,M,N là những tích số của AXB, F1 là tổng các tích số của hai biến A và B. Bây giờ ta sẽ lập trình bằng cách làm đứt các cầu chì thì các số hạng ứng với những cầu chì bị đứt sẽ không có mặt trong biểu thức. Bằng cách lập trình các cầu chì ở mảng AND – OR (nghĩa là loại bỏ giá trị giá trị của nó trong biểu thức) FPLA có thể tạo ra các hàm logic khác nhau theo mạch thiết kế chỉ với hai biến ngõ vào. Lưu ý những Diode trong mảng OR được dùng để bảo vệ ngăn mạch.

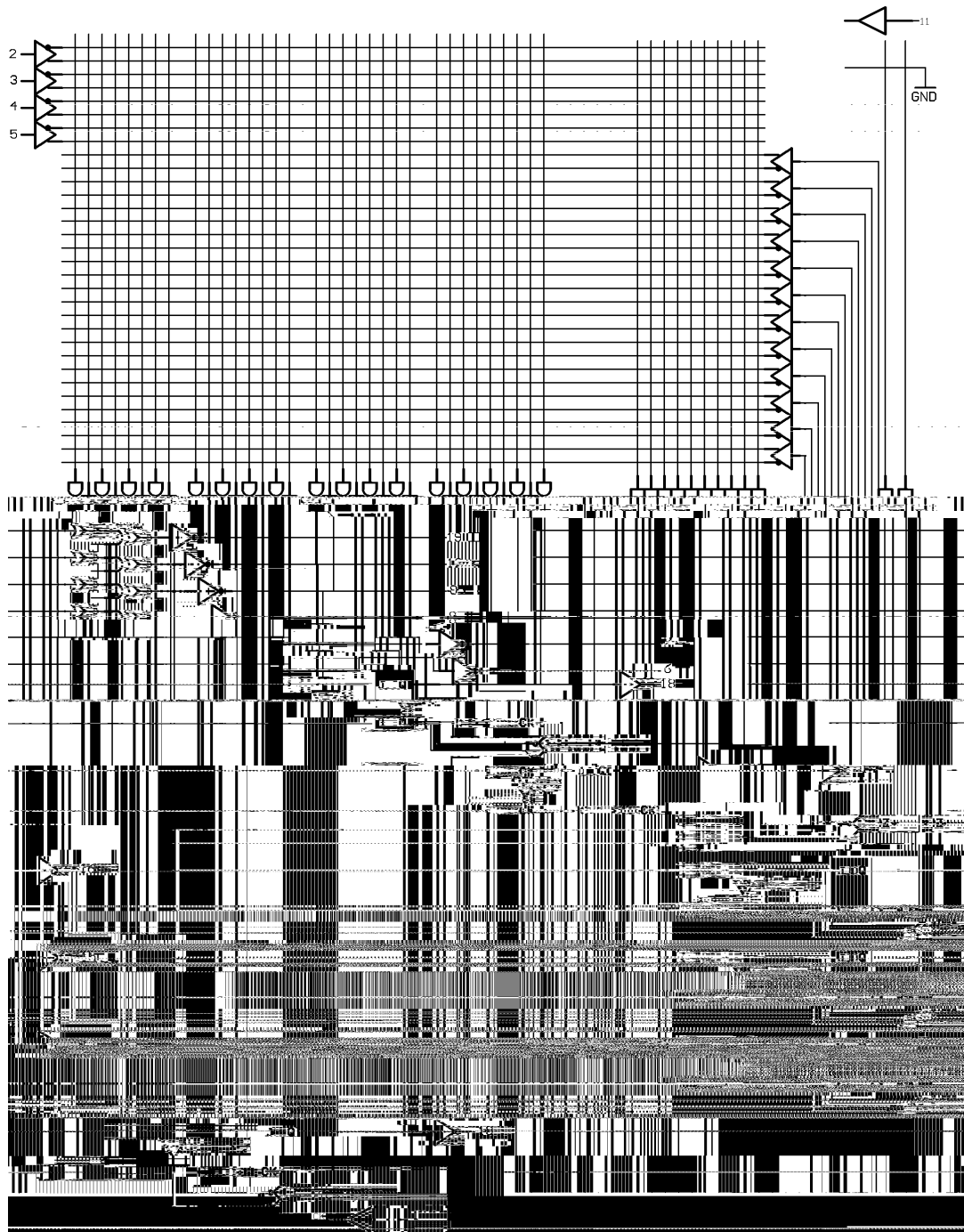
Sơ đồ mạch trong hình 3.4 là một ví dụ đơn giản của họ vi mạch mảng logic lập trình trường. Nếu vi mạch do công ty chế tạo đã được lập trình bằng công đoạn mặt nạ với công nghệ lưỡng cực thì chương trình cố định không thay đổi được. Do đó vi mạch này được gọi là PLA. Nếu vi mạch được sản xuất để người sử dụng có thể lập trình thì gọi là FPLA.



Hình 3.4. Sơ đồ logic của FPLA PLS 153

### ***3. Họ vi mạch FPLS ( Field Programable Logic Sequencer)***

Họ FPLS được giới thiệu vào năm 1979, FPLS có cấu trúc mô phỏng theo cấu trúc của FPLA nhưng được bổ sung thêm những thanh ghi cho phép “preloading” trạng thái của thiết bị. Một vài thanh ghi ở ngõ ra được đưa hồi tiếp về mảng AND lập trình và một số khác có những thanh ghi ngầm (những thanh ghi được bổ sung trên chip và không nối với chân của ngõ vào hay ngõ ra) bổ sung với thanh ghi ngõ ra, nó có thể hồi tiếp hoặc không hồi tiếp.



Hình 3.5 . Sơ đồ logic FPLS PLS157

Sơ đồ logic của vi mạch PLS157 được công ty Signetics giới thiệu được trình bày ở hình 3.5, có hình dáng bên ngoài 20 chân, có cấu trúc  $16 \times 45 \times 12$ . PLS157 có 6 thanh ghi và 6 tổ hợp ở ngõ ra. Các tổ hợp ở ngõ ra có chức năng nhất, những thanh ghi được cấu tạo bằng những công đảo  $M(M_0-M_5)$ . Cấu trúc mới của PLS157 có những đặc điểm đáng lưu ý là những thanh ghi cho phép chốt những tín hiệu ở ngõ vào và những tín hiệu này được đưa tới mảng AND. Ngoài ra vi mạch còn

được thiết kế một mảng bổ sung (mảng bù). Đây là tổng số hạng bù và được thực hiện như một ngõ vào của mảng AND, nó cho phép bổ sung thêm nhiều tổ hợp.

#### ***4. Họ vi mạch FPGA ( Field Programmable Gate Array).***

Họ FPGA được Signetics giới thiệu vào năm 1977 được sử dụng để thay thế cho những công nghệ nhiều ngõ vào tiêu chuẩn, cấu trúc của nó bao gồm một mảng AND lập trình, với lập trình cực tính ở ngõ ra. Chỉ với một công AND có thể biến đổi thành công NAND, NOR hay công OR. Mỗi công AND trong FPGA có thể biến đổi thành các công logic khác nhau.

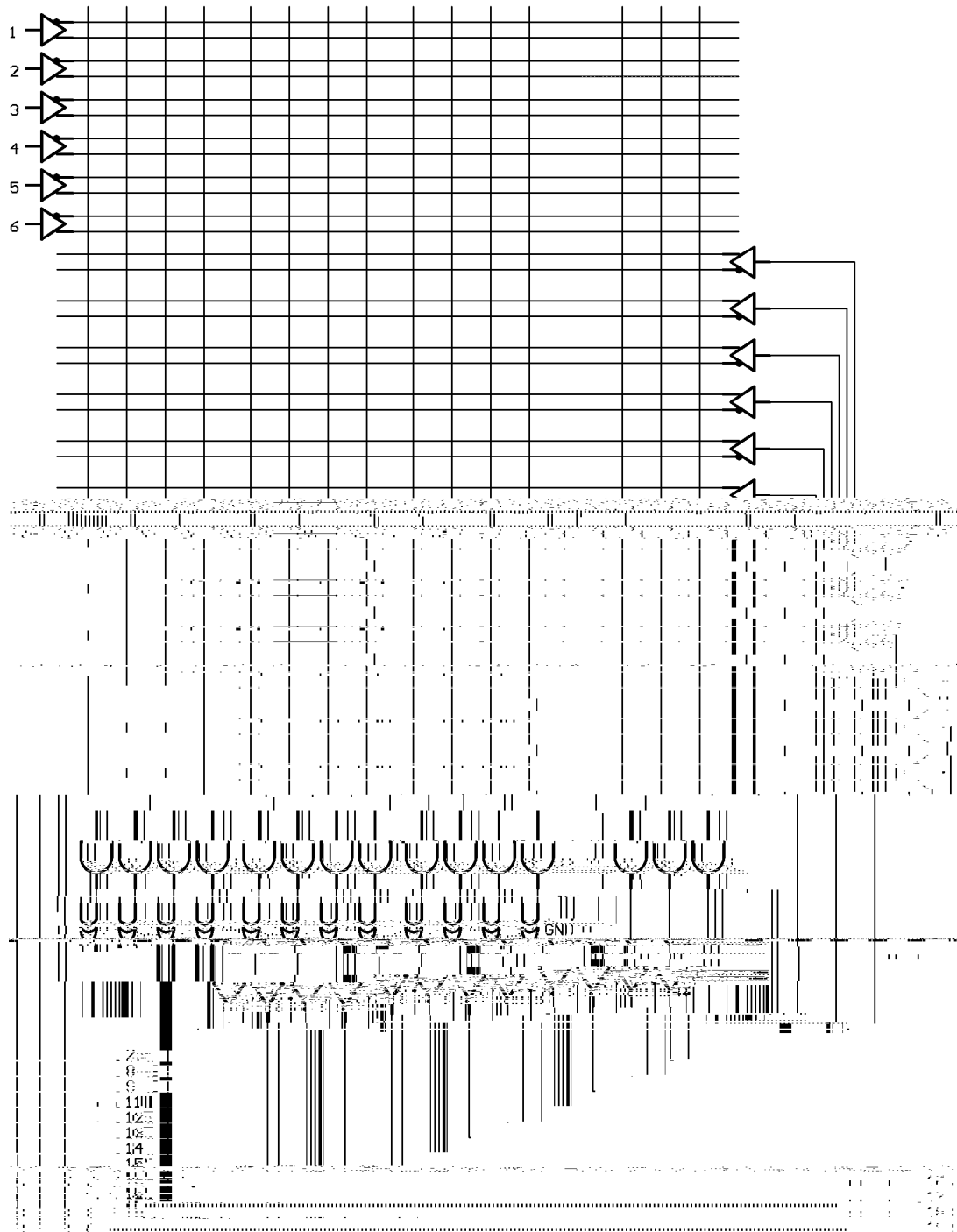
FPGA cũng được bổ sung linh động hơn những công tiêu chuẩn khác. Vi mạch đại diện cho họ FPGA là PLS151, có hình dáng 20 chân được trình bày ở hình 3.6. PLS151 có 6 ngõ vào, 12 ngõ ra và có tín hiệu hồi tiếp đưa về mảng AND được sử dụng như những ngõ vào. Có thêm 3 tích số được tạo ra bởi 3 đường điều khiển, các tín hiệu này điều khiển những công đệm ngõ ra 3 trạng thái. FPGA thích hợp trong các thiết kế để giải mã địa chỉ và được thêm vào các chức năng khác.

#### ***5. Họ vi mạch PAL ( Programmable Array Logic).***

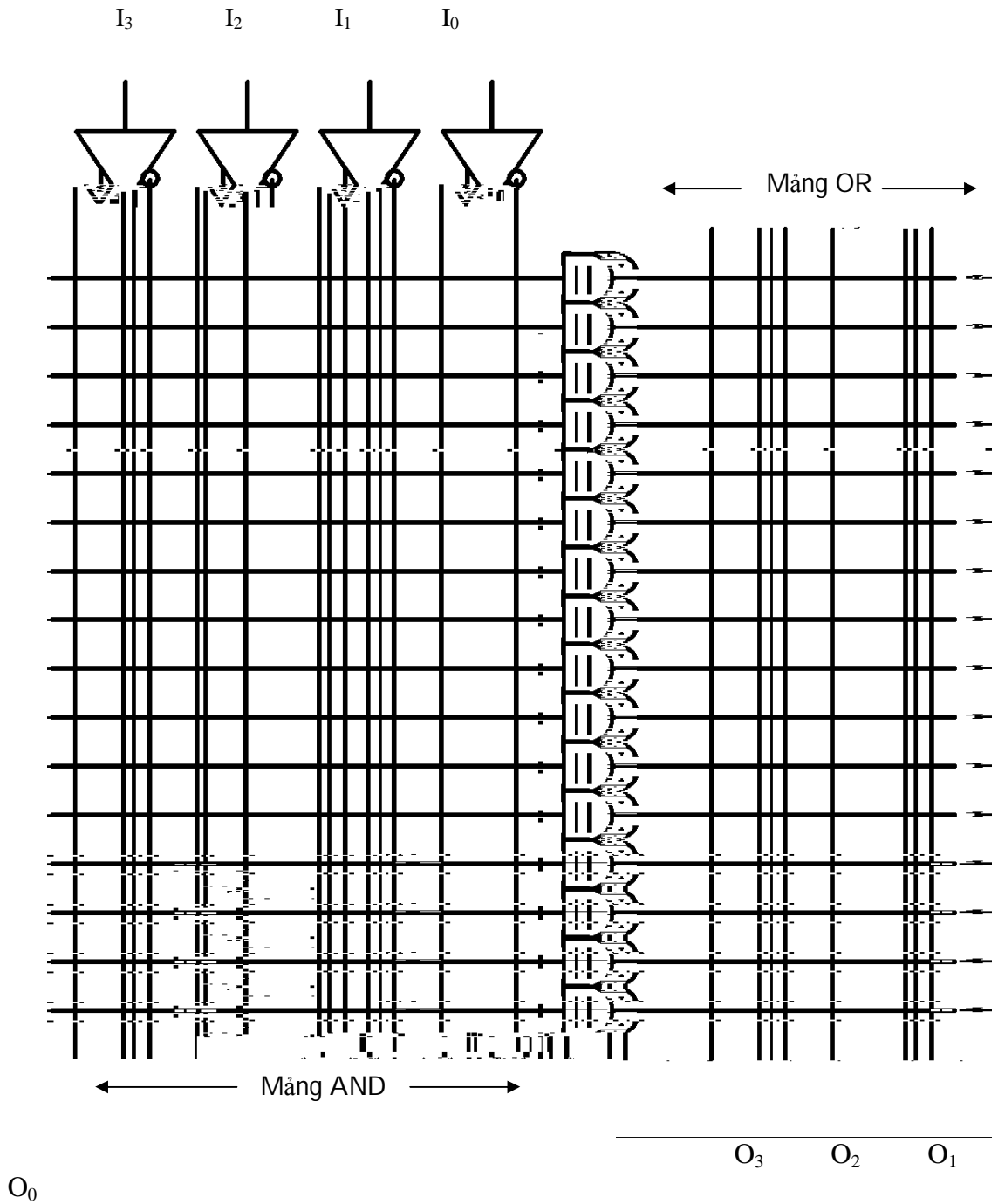
PAL là một họ phổ biến nhất trong họ PLD được MONOLITHIC MEMORIES INC giới thiệu vào năm 1978. PAL được đăng ký bản quyền về cấu trúc của công ty MMI. Cấu trúc của PAL bao gồm một mảng AND lập trình theo sau là một mảng OR cố định, cấu trúc này được cải tiến từ những khuyết điểm của họ FPLA. Hình 3.7 minh họa cho cấu trúc đơn giản của PAL. Do loại bỏ việc sử dụng cầu chì ở mảng OR, do đó số lượng tinh thể Silicon được sử dụng giảm, dẫn đến giá thành của PAL thấp hơn so với FPLA. Mặt khác thời gian trì hoãn của PAL ngắn hơn so với FPLA do giảm được sự trì hoãn khi truyền qua mảng OR.

Khảo sát PAL16L8 có hình dáng 20 chân sơ đồ logic được trình bày ở hình 3.8. Vi mạch này có 8 tổ hợp ngõ ra, mỗi ngõ ra được đảo với 7 tích số của ngõ vào, 6 trong 8 ngõ ra được hồi tiếp về mảng AND, cho phép những chân này được sử dụng với chức năng I/O. Do PAL16L8 có ngõ ra tác động ở mức thấp nên nó có thể kết hợp với các IC khác cùng một mức tác động.





Hình 3.6 . Sơ đồ logic của FPGA PLS151



Hình 3.7 . Sơ đồ logic của PAL



### Hình 3.8. Sơ đồ logic của PAL16L8

PAL16L8 được ứng dụng trong lĩnh vực giải mã địa chỉ, nó thuận tiện trong việc kết hợp với các bộ vi xử lý và thiết bị ngoại vi vì cùng một mức tác động. Với những đặc tính như tốc độ tương đối cao, giá thành thấp, thời gian truyền trì hoãn khoảng 7,5ns nên PAL16L8 rất phổ biến trong công nghiệp PLD. Ngoài ra PAL16L8 có một đặc điểm mới so với các họ trước là có cầu chì bảo vệ, nó dùng để chống sự sao chép, giúp bảo vệ nội dung bên trong. Ngoài PAL16L8 công ty MMI còn giới thiệu các loại vi mạch khác như PAL16R4, PAL16R6, PAL16R8. Các vi mạch này có cấu tạo giống như PAL16L8 nhưng ở ngõ ra sử dụng thêm các FF D để chốt tín hiệu ngõ ra.

Một thế hệ vi mạch PAL được công ty AMD giới thiệu là PAL22V10 với hình dáng 24 chân được chế tạo bằng công nghệ CMOS thay thế cho công nghệ lưỡng cực. Đặc trưng của vi mạch này là ở ngõ ra được cho qua cổng PLD.

Ngoài việc tăng số biến ngõ vào vi mạch này còn có một số đặc điểm nữa là trong hàm logic các thành phần tích số có thể thay đổi từ 8 đến 16 biến. Điều này sẽ giúp cho vi mạch thực hiện nhiều phương trình phức tạp. Nhờ vào cấu tạo ở ngõ ra các cổng PLD nên các ngõ ra hoặc vào của vi mạch có đặc tính giao tiếp 2 chiều, điều này làm tăng khả năng xử lý của vi mạch và tạo sự thuận lợi cho việc thiết kế. Do những đặc điểm đã được cải tiến nên các thế hệ vi mạch PAL được phổ biến rộng rãi (đặc biệt là nhóm vi mạch 20 chân) và PAL được xem là họ vi mạch đại diện cho họ vi mạch số lập trình.

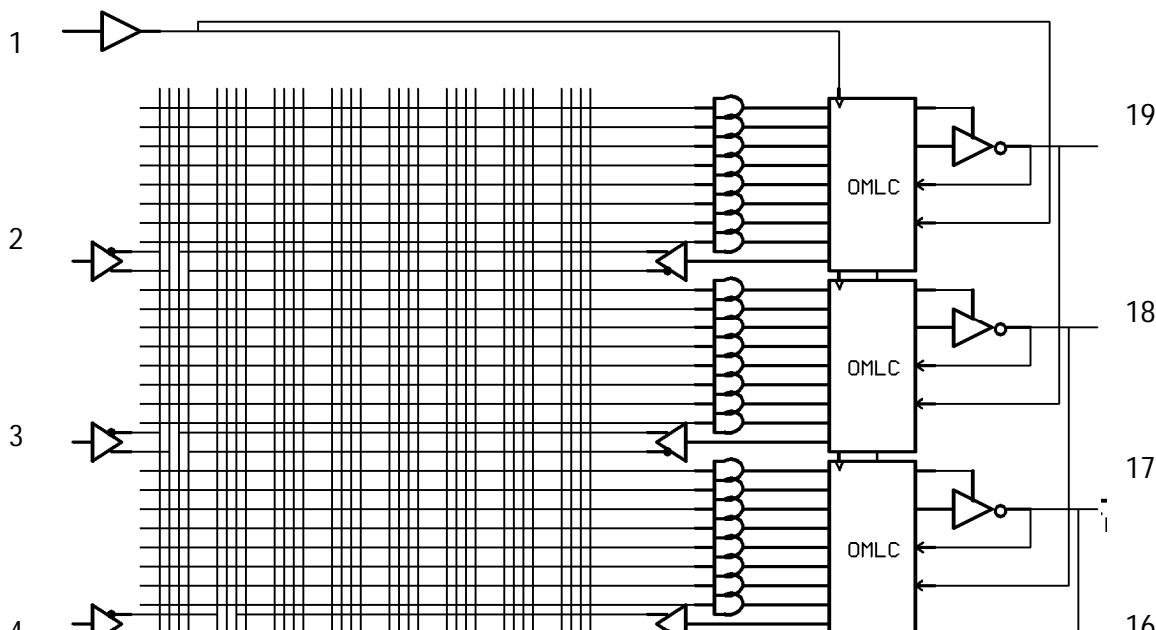
Ngoài ra các công ty chế tạo PAL có chọn lựa trong việc ký hiệu các số trên một vi mạch. Điều này cung cấp cho người sử dụng những thông tin cần thiết có liên quan đến ứng dụng của vi mạch. Các ký hiệu trong việc đánh số của họ PAL nói chung bao gồm 2 số đếm được tách rời nhau bởi 1 hay 2 ký tự. Số đầu tiên trong tên vi mạch cho biết số ngõ vào của vi mạch (đây chính là số biến ngõ vào của mảng AND). Số thứ hai biểu thị số ngõ ra của vi mạch. Ký tự nằm giữa 2 số chỉ ra ý nghĩa các thuộc tính của ngõ ra. Một số mã ký tự có ý nghĩa là:

- H tác động mức thấp.
- L tác động mức cao.
- P tác động ngõ ra có thể lập trình.
- C phân bổ sung các ngõ ra.
- S bộ tuần tự.

Các ký hiệu của vi mạch họ PAL được xem là những hướng dẫn cơ bản của vi mạch. Ngoài ra các công ty chế tạo còn cung cấp bản thông số kỹ thuật và sơ đồ logic của vi mạch kèm theo để làm tài liệu tham khảo cho các nhà thiết kế.

**6. Họ vi mạch GAL ( Generic Array Logic).**

GAL là một nhóm của công nghệ EEPLD, nó được giới thiệu và phát triển bởi công ty Lattice Semiconductor Comp. Công ty này đã đưa ra một khái niệm về cổng PLD có ký hiệu là OLMCs (Output Logic Macrocells).



### Hình 3.9 . Sơ đồ logic của GAL16V8

Vi mạch này cũng có những đặc điểm là có thể xóa bằng điện và lập trình lại bằng các phần mềm và công cụ hỗ trợ. Khảo sát cấu trúc của vi mạch GAL16V8 được trình bày ở hình 3.10, GAL16V8 có hình dạng 20 chân là một vi mạch phổ biến trong họ GAL.

Mỗi một OLMC có 8 ngõ vào tương đương với 8 tích số trong một biểu thức. Ngoài ra OLMC cũng có tín hiệu hồi tiếp đưa về để điều khiển, tín hiệu xung đồng hồ, tín hiệu hồi tiếp về mảng AND. Các vi mạch GAL đều có hỗ trợ những thanh ghi “Preload”, điều này có ích trong việc kiểm tra vi mạch. Mặt khác một thể hệ vi mạch mới được phát triển là vi mạch lập trình hệ thống ký hiệu là ispEELD (In-system Programmable).

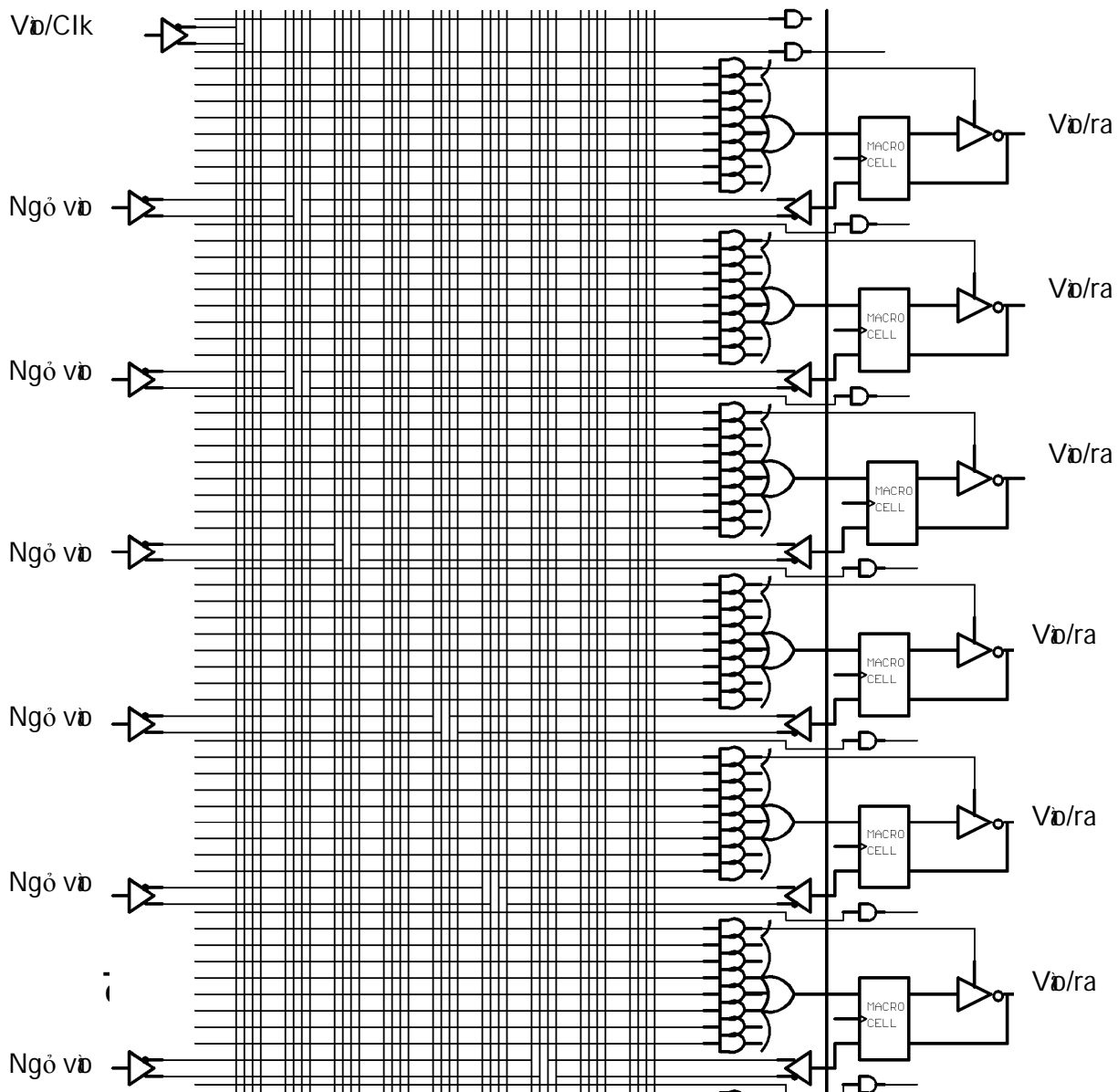
Vi mạch đầu tiên là ispGAL16Z8, cấu trúc của nó gần giống với GAL16V8 nhưng được thêm vào 4 chân để điều khiển lập trình. Trong hệ thống ispGAL16Z8 cho phép chu kỳ lập trình là 10000 lần và dữ liệu được giữ cố định trong khoảng thời gian 20 năm. Đó cũng là quy định của những vi mạch theo nguyên tắc EPROM. Cấu trúc của họ GAL là sự lặp lại cấu trúc của họ PAL và những đặc điểm của họ GAL được thiết kế để kết hợp với những vi mạch họ PAL. Điều này được thể hiện qua việc ký hiệu các vi mạch họ GAL và cấu trúc tế bào bảo vệ của nó.

### **7. Họ vi mạch PEEL (*Programmable Electrically Erasable Logic*).**

Họ PEEL được công ty International Cmos Technology INC giới thiệu. Nó được chế tạo với công nghệ EEPROM. Cấu trúc của PEEL cũng tương tự như PAL

và GAL, nó được xóa bằng điện và lập trình cũng nhờ vào phần mềm hỗ trợ. Khảo sát vi mạch PEEL18CV8 được trình bày ở hình 3.10.

Vi mạch có 20 chân với 8 ngõ ra được cấu tạo bởi cổng PLD, mỗi ngõ ra có 8 tích số trong một hàm của biểu thức và có một tích số riêng để điều khiển cổng đệm ngõ ra. Cụm tính ngõ ra cũng được lập trình các thanh ghi ở ngõ ra của vi mạch được Reset không đồng bộ, ngoài ra các thanh ghi có thể được chốt bên trong khi ngõ ra được điều khiển bởi một biểu thức của tổng các số hạng của ngõ vào. Đặc điểm này được cải tiến hơn số với các vi mạch PAL16V10 hay GAL16V8.



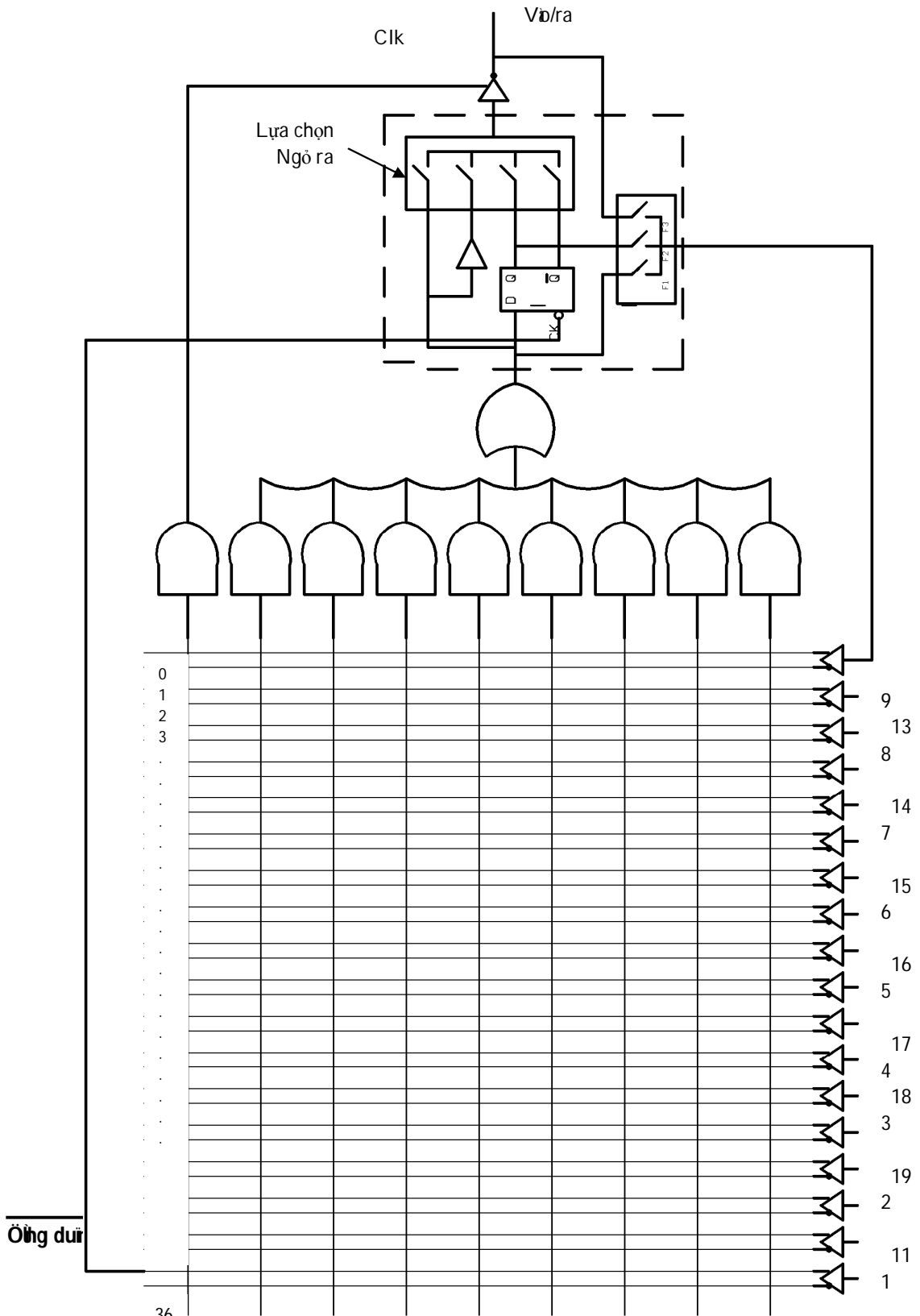
Hình 3.10 . Sơ đồ logic của PEEL18CV8

### 8. Họ vi mạch EPLD (Erasable PLD).

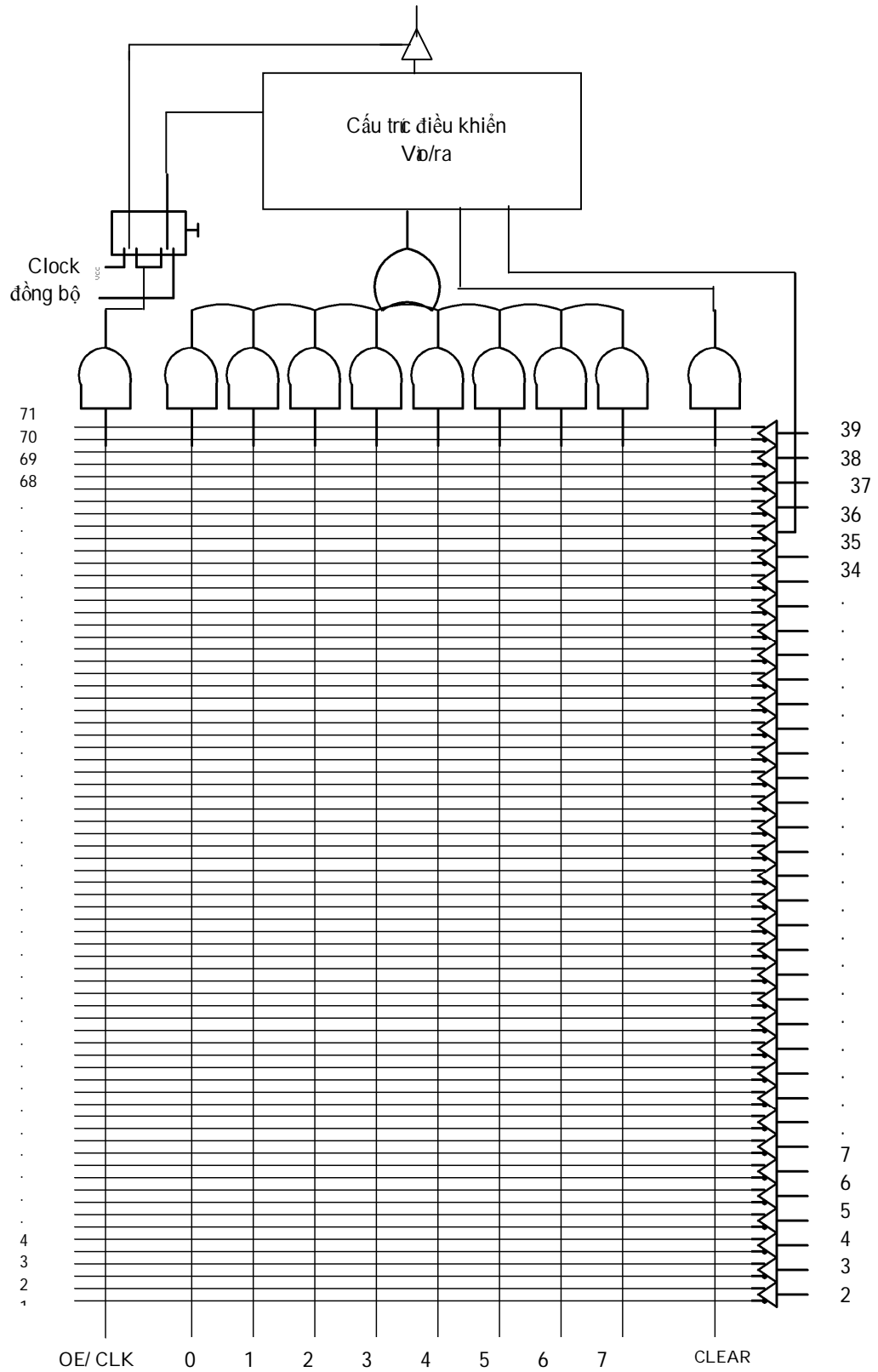
Công ty Altera lần đầu tiên giới thiệu thuật ngữ xóa các PLD bằng tia cực tím và nó đã trở thành thuật ngữ chung cho công nghệ PLD để tham khảo cho các vi mạch lập trình xóa bằng tia cực tím. Từ khi khởi đầu, công ty Altera thay đổi công nghệ chế tạo PLD từ công nghệ lưỡng cực sang công nghệ CMOS vì công nghệ CMOS đạt được hiệu suất cao về không gian (mật độ tích hợp cao hơn). Như PAL16L8 có mật độ tích hợp từ 100 lên 150 cổng, PAL22V10 có 500 đến 600 cổng và EP310 (là vi mạch đại diện cho họ EPLD) có trên 1000 cổng. Hình 3.12 trình bày sơ đồ khối của EP310 có 20 chân cấu trúc gồm 8 cổng PLD, xung xóa không đồng bộ và có thể đặt trước các tích số.

Cấu trúc của cổng PLD bao gồm cả khối điều khiển cấu trúc I/O. Cấu hình của ACB giống như cấu trúc của cổng PLD của vi mạch PAL và GAL nhưng có chức năng hoạt động đơn giản hơn. Trong đó mỗi cổng có 8 biến ngõ vào cùng với một biến để điều khiển cổng đệm ngõ ra. Nhờ vào cấu trúc ACB I/O mà EP310 có các tín hiệu tổ hợp ngõ ra tác động ở mức cao hoặc thấp hay các tín hiệu được ghi cũng tác động ở mức cao hoặc thấp. Đối với tín hiệu hồi tiếp về mảng AND được đưa về từ thanh ghi ở ngõ ra. Các cổng đệm ngõ ra được điều khiển bằng các biến riêng cho phép các chân của vi mạch có thể hoạt động hai chiều. Ngoài ra EP310 cũng có cầu chì bảo vệ chống sao chép và giờ đây cầu chì bảo vệ trở thành một tiêu chuẩn cho các thế hệ PLD mới. Một số vi mạch tiêu biểu cho họ EPLD là EP900, có cấu tạo 40 chân, bên trong có 24 khối ACB, mật độ tích hợp hơn 1000 cổng với các tổ hợp ngõ ra có lựa chọn. Nhưng trong tương lai kỹ thuật ngày càng phát triển thì mật độ tích hợp có thể lên đến hơn 10000 cổng logic trong một chip.

Ngoài vi mạch EP900 thì công ty Altera còn giới thiệu vi mạch EP1800 có 68 chân với các chức năng được mở rộng hơn so với EP900 vì số cổng logic trong IC được tăng gấp đôi và số ngõ vào cũng vậy. Vi mạch EP1800 có thể thực hiện đồng thời 4 chức năng khác nhau, có thể xem như đó là 4 vi mạch rời. Những vi mạch số lập trình đang hướng đến mật độ tích hợp trên 1000 cổng logic trong một chip thì đang gây ảnh hưởng đến các PLD có mật độ tích hợp thấp. Công ty Cypress Semiconductor đang sản xuất các sản phẩm ứng dụng công nghệ CMOS có tốc độ cao. Sản phẩm cạnh tranh của họ chủ yếu là các họ PAL thông thường và PAL 22V10 đã tạo ra thế hệ PAL có công suất thấp, tốc độ cao nên được ứng dụng rộng rãi vào các lĩnh vực công nghiệp khác nhau. Những vi mạch phổ biến của công ty Cypress Semiconductor như CY7C330, CY7C331 và CY7C332.







### 9. Họ vi mạch PML ( Programmable Macro Logic).

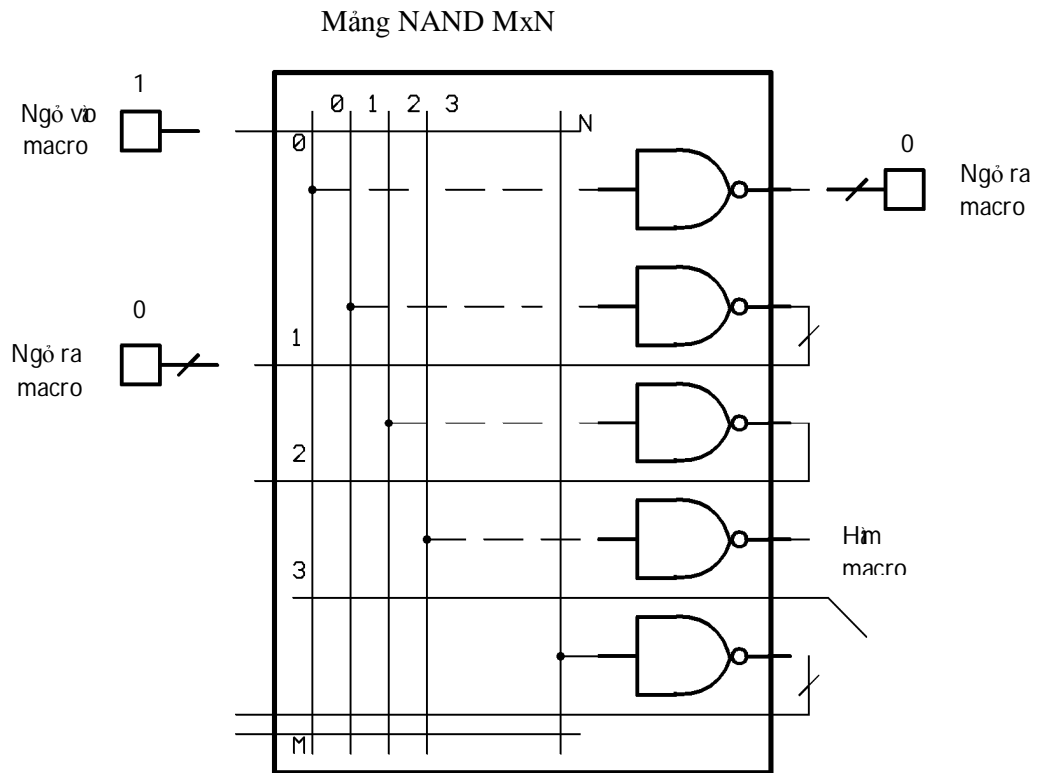
Họ vi mạch được công ty Signetics sử dụng cấu trúc mới gọi là “foldback” (gấp về). Mạch logic “foldback” sử dụng một cổng NAND đơn hay mảng NOR kết hợp với một cấu trúc liên kết lập trình trung tâm cho phép thực hiện nhiều mức logic khác nhau để liên kết với macro ngõ vào và ngõ ra. Như trong họ vi mạch PML, một mảng NAND được sử dụng vì cổng NAND có tốc độ truyền nhanh nhất trong công nghệ lưỡng cực.

Từ khóa macro để tham khảo một khối chức năng và có thể xác định một tín hiệu ngõ vào, một cổng đệm ngõ ra hay bất cứ một hàm logic nào như FF, mạch đếm hay mạch tổ hợp. Công ty Signetics phân loại các macro như sau: ngõ vào là macro ngõ vào, macro ngõ ra và những khối chức năng khác như thanh ghi hay mạch tổ hợp thì gọi là macro chức năng. Macro của họ ML mô tả ở hình 3.13. So với cấu trúc mảng AND – OR của các họ IC PAL và FPLA thì cấu trúc mảng NAND phức tạp hơn. Để đơn giản cho việc tìm hiểu, xét ví dụ sau:

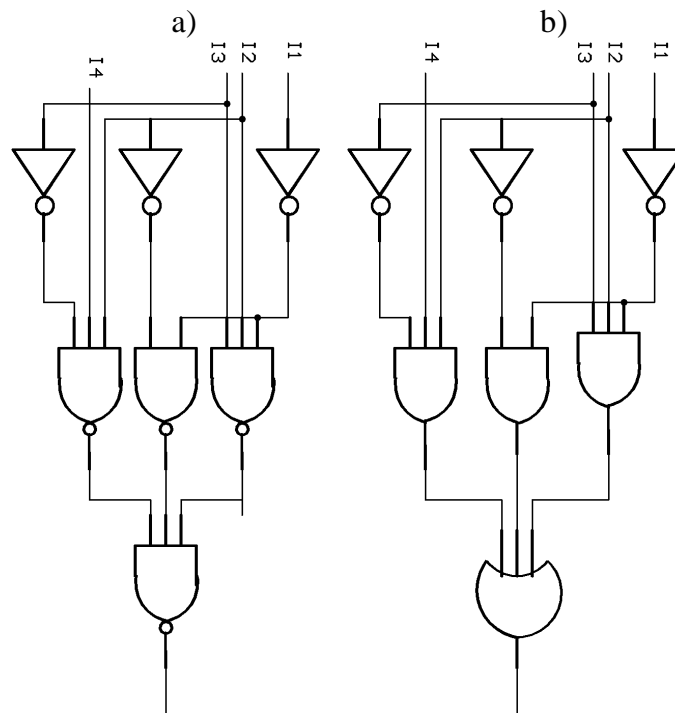
Hình 3.14 a trình bày một mạch logic đơn giản sử dụng cấu trúc AND – OR của họ vi mạch PAL và hình 3.14 b sử dụng cấu trúc NAND – NAND với chức năng tương tự nhưng có ưu điểm là không bị giới hạn với các hàm có hơn 2 cấp logic. Mặc khác, cấu trúc của PML tận dụng tối đa các cổng logic và các khả năng hoạt động của vi mạch.

Ví dụ như ở họ PAL và FPLA khi cần thêm một biến ở ngõ vào sẽ chọn đường tín hiệu ngõ ra xem như một đường tín hiệu ngõ vào, do đó sẽ làm tăng thời gian truyền và lãng phí một ngõ ra. Đối với họ PML thì tất cả các ngõ vào của cổng NAND được sử dụng như là ngõ vào của tín hiệu và các hàm logic được thực hiện với cấu trúc 3 cấp logic. Vi mạch đại diện cho họ PML là PLHS 501 có cấu tạo gồm 72 cổng NAND trong đó có 44 cổng NAND được dùng để hỗ trợ cho macro ngõ ra. Vi mạch có 24 ngõ vào, 8 cổng đệm XOR ở ngõ ra với 4 cổng tác động ở mức thấp, có 4 cổng tác động ở mức cao và có 8 đường dữ liệu 2 chiều. Vi mạch có 52 chân với kiểu chân theo dạng PLCC. Hình 3.15 trình bày cấu trúc của PLHS 501.

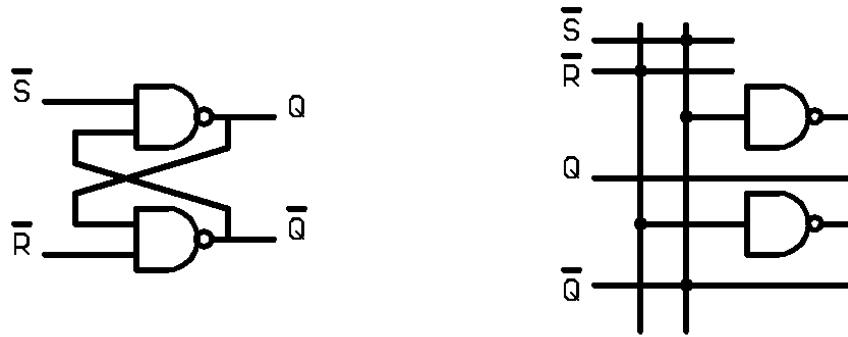
Các cổng đệm ngõ ra 3 trạng thái được điều khiển bằng từng cổng NAND riêng để tạo nên tính linh hoạt trong thiết kế.



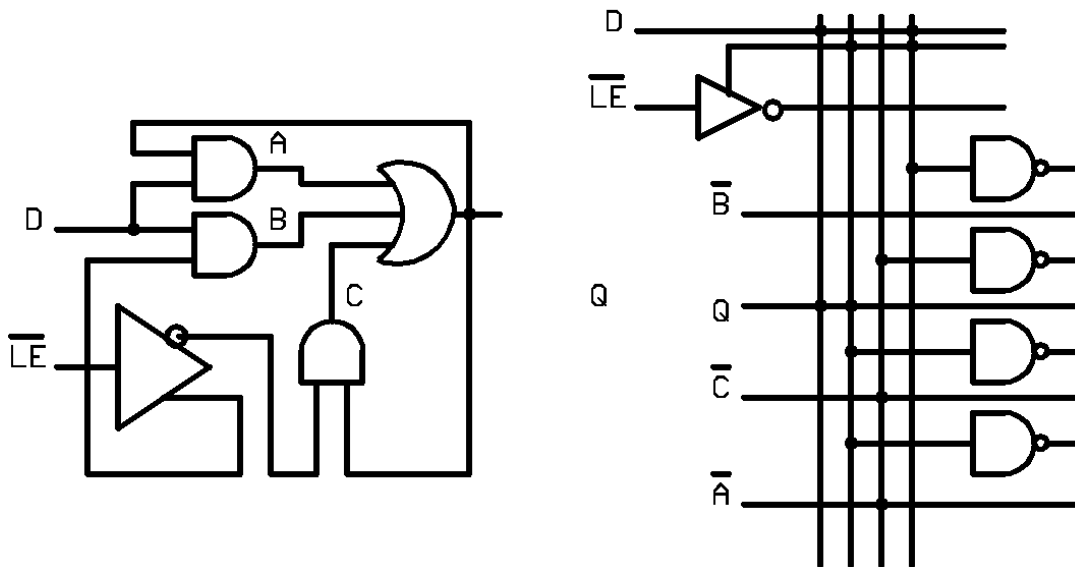
Hình 3.13 . Sơ đồ logic lập trình macro



Hình 3.14. Mạch logic sử dụng cấu trúc “flodback”

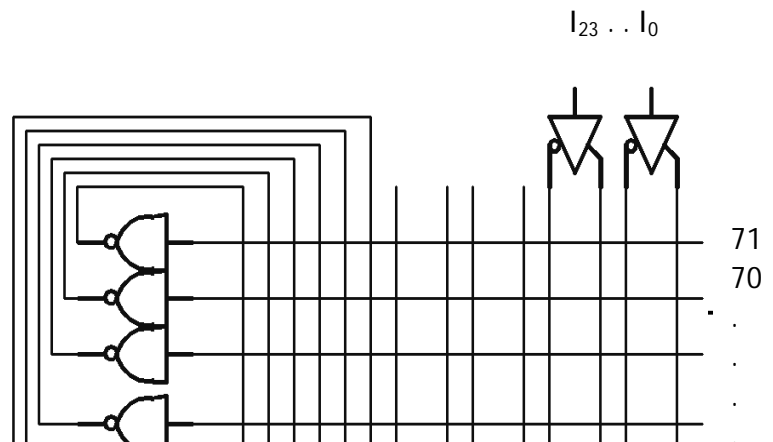


a) Mạch lật RS



b) Mạch lật D

Hình 3.15 . Mạch lật RS và D sử dụng cấu trúc PML



Hình 3.16. Sơ đồ chức năng PLS 501

#### ***10. Họ vi mạch ERASIC(Erasable Programmable Application Specific IC).***

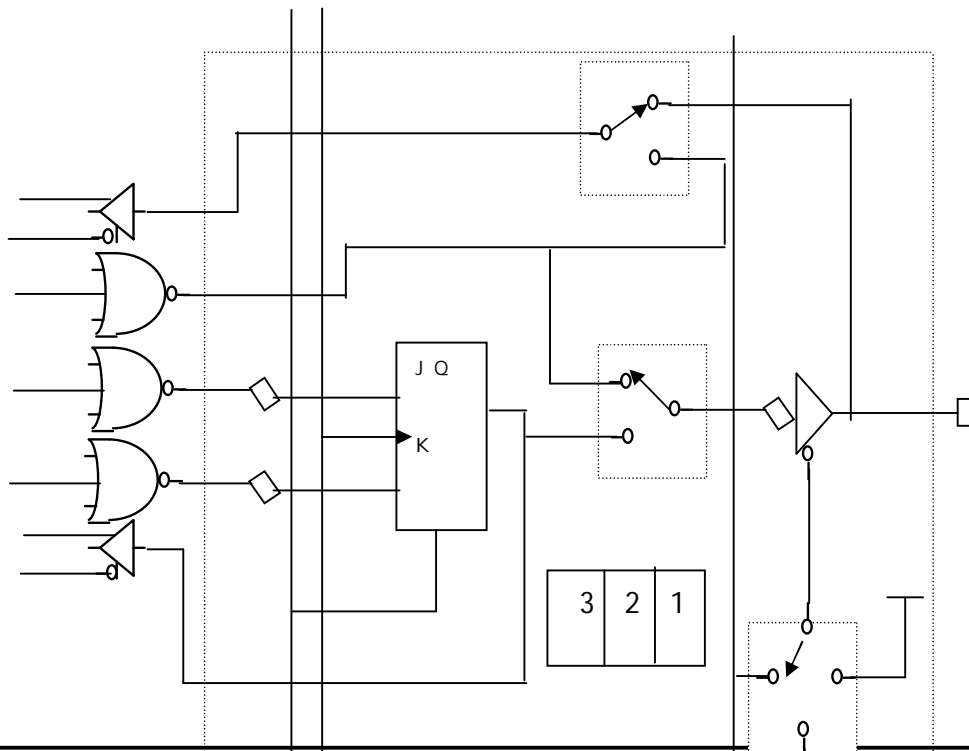
Họ vi mạch ERASIC được giới thiệu bởi công ty Exel Microelectronics có cấu trúc tương tự như họ PML nhưng được chế tạo bằng công nghệ CMOS EEPROM khác với họ PML dùng công nghệ lưỡng cực . Một đặc điểm khác biệt nữa là họ ERASIC sử dụng cấu trúc mảng NOR , vì trong công nghệ CMOS công NOR có thời gian truyền nhanh nhất. Vi mạch đầu tiên của họ này là XL 78C800 có 24 chân với mật độ thích hợp khoảng 800 cổng.

XL78C800 có 12 ngõ vào và 10 chân I/O được liên kết với các cổng lập trình. Chân số 1 là đường cung cấp tín hiệu xung clock cho FF JK, chân 13 dùng để điều khiển các cổng đệm ngõ ra cho các cổng lập trình, 8 ngõ vào được đưa vào mảng NOR thông qua các mạch lật, 2 cổng NOR được sử dụng để điều khiển mạch lật. Ngõ ra của công NOR được cấu tạo bằng các khối PCE (Polarity Control Element ) để tăng tính linh hoạt.

XL78C800 có 32 biến ở ngõ vào cổng NOR, hai biến dùng để điều khiển mạch lật và có 30 biến dùng cho cổng lập trình.

Hình 3.17 trình bày sơ đồ logic của cổng lập trình trong vi mạch XL78C800. Có ba ngõ vào của tín hiệu J, K, O được đưa vào cổng lập trình. Với hai biến J, K để điều khiển FF JK. Từ cổng lập trình cũng có 4 đường tín hiệu được đưa về mảng NOR. Chân của vi mạch kết hợp với các cổng lập trình được kết nối với ngõ vào của bộ đa hợp (IN-MUX) và nối với cổng đệm ngõ ra. Ngoài ra còn có các đường tín hiệu xung clock và xung xóa không đồng bộ cho FF JK, một đường tín hiệu OE để điều khiển cổng đệm ngõ ra. Bộ đa hợp ngõ ra sẽ lựa chọn các tín hiệu từ biến O hay từ ngõ ra của FF JK. Ở IN- MUX cho phép tín hiệu vào đi mới mảng NOR từ chân IC hay từ biến O của mảng NOR. Chân điều khiển của bộ đa hợp ngõ ra OE-MUX dùng để điều khiển cổng đệm ngõ ra hoạt động 1 chiều hay 2 chiều hay ở trạng thái tổng trở cao. Sơ đồ cổng lập trình ở hình 3.17

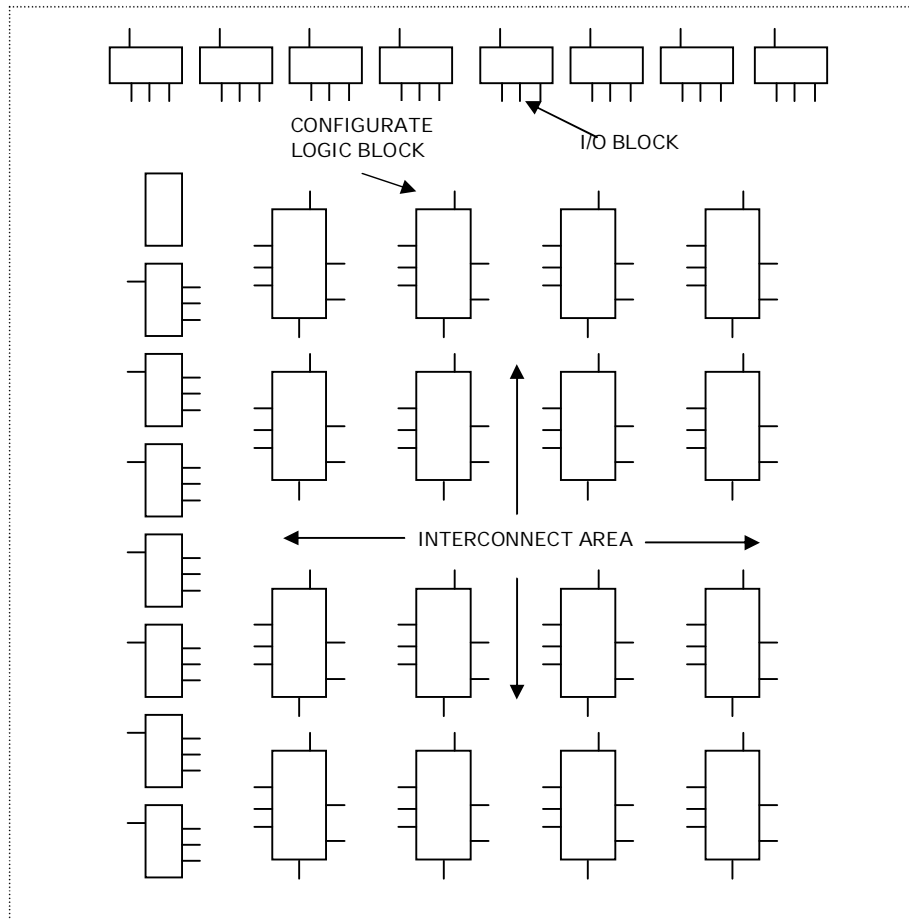
Sơ đồ cổng lập trình trên trình bày 3 chế độ làm việc của vi mạch. Thời gian truyền của vi mạch họ ERASIC là 35 ns cộng với thời gian truyền qua mảng NOR là 20 ns do đó thời gian truyền của vi mạch là 55 ns, dòng tiêu thụ 35 mA, so với dòng tiêu thụ của PLHS 501 là 250 mA. Qua sự so sánh trên cho thấy công suất tiêu thụ của họ ERASIC thấp hơn họ PML, đó là ưu điểm của công nghệ CMOS. \_



Hình 3.17 . Sơ đồ khối công lập trình ERASIC XL78C800.

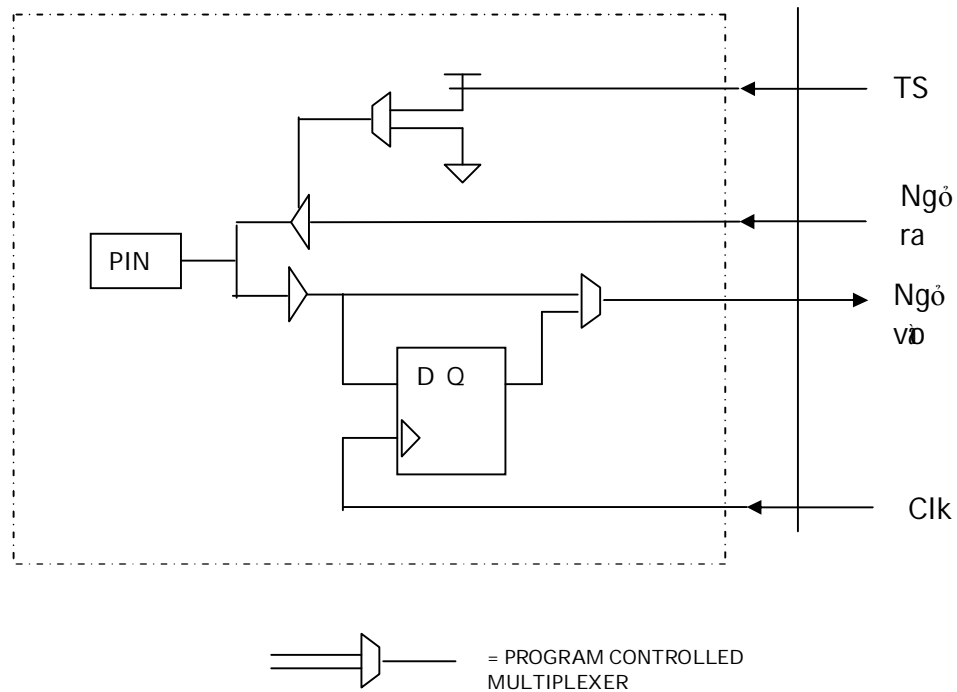
**11. Họ vi mạch LCA ( Logic Cell Array).**

Họ LCA được công ty Xilinx giới thiệu dựa theo các cấu trúc của công ty MMI, đã trình bày một cấu trúc độc đáo trong các họ của PLD. Cấu trúc truyền thống của các họ vi mạch PAL và FPLA là các mảng AND – OR. Các công lập trình có cấu trúc của họ LCA gọi là cấu trúc lập trình cho người sử dụng. Đặc biệt là trong cấu tạo của LCA, họ dùng RAM động để tạo ra các chức năng logic theo yêu cầu thiết kế. Nhược điểm của các tế bào RAM động thường không ổn định. Do đó các chức năng sẽ trở lại trạng thái ban đầu khi mất điện. Để hỗ trợ cho vấn đề này họ sử dụng thêm phương pháp lưu trữ mới có chức năng tương tự như ROM. Cấu trúc của LCA được mô tả ở hình 3.18, bao gồm một khối IOB bao quanh ma trận của khối LCB.



Hình 3.18. Cấu trúc LCA

Liên kết các đường tín hiệu dọc và ngang giữa 2 khối giúp cho việc kết nối giữa 2 khối thêm thuận tiện. Vi mạch đầu tiên của họ LCA là XC 2064, có mật độ thích hợp khá phức tạp khoảng 1200 cổng logic, 58 khối IOB cùng một ma trận 8x8 hàng và cột tạo ra 64 khối CLB. Tạo ra một vi mạch khác là XC 2018 có mật độ tích hợp khoảng 1800 cổng, có 74 khối IOB cùng một ma trận 10x10 tạo ra 100 khối CLB. Vi mạch có các đường tín hiệu xung clock, tín hiệu reset đặc biệt và mạch tạo dao động thạch anh bên trong IC dùng để kết nối với các phần tử dao động bằng thạch anh bên ngoài.



Hình 3.19. Cấu hình khối vào / ra ( IOB) của LCA

Hình 3.19 trình bày sơ đồ của khối IOB, bao gồm 1 cổng đệm ngõ vào, bộ đa hợp IN – MUX và FFD. Mức điện áp ngưỡng ở ngõ vào cổng đệm thích hợp cho cả hai họ TTL và CMOS. Ngõ ra của FFD được nối với ngõ vào của bộ đa hợp và ở ngõ ra của bộ đa hợp có thể nối 1 hay nhiều khối CLB. Ngõ ra của khối IOB gồm 1 cổng đệm 3 trạng thái được nối thẳng tới chân IC.

### III/ CÁC PHẦN MỀM HỖ TRỢ CỦA PLD.

Các phần mềm hỗ trợ cho các vi mạch lập trình được các công ty phát triển liên tục, ngày càng có nhiều tính đa dạng, có thể hỗ trợ cho nhiều loại vi mạch khác nhau nên có tính cạnh tranh mạnh mẽ trong thị trường vi mạch lập trình.

#### 1. Phần mềm PALASM 2 (PAL Assembler)



PALASM 2 của công ty MMI là phần mềm tiêu chuẩn cho các vi mạch lập trình. Đây là bộ biên dịch thế hệ thứ 2 hỗ trợ cho các vi mạch hoạt động không đồng bộ, như các vi mạch họ PAL của công ty MMI, vi mạch họ PLA và các vi mạch của công ty AMD.

## **2. Phần mềm AMAZE.**

Phần mềm AMAZE được công ty Signetics phát triển và nó được cung cấp cho các khách hàng sử dụng vi mạch lập trình của công ty. Module chính của phần mềm AMAZE là BLAST ( Boolean logic & State Transfer) dùng để biên dịch các thông tin ngõ vào chuyển đổi sang các file chương trình chuẩn của Signetics (các file có phần mở rộng là ‘ STD ’). AMAZE hỗ trợ để mô phỏng các vector kiểm tra để thiết kế theo yêu cầu của người sử dụng.

## **3. Phần mềm PLAN ( Programmable Logic Analysis).**

Phần mềm PLAN được công ty National Semiconductor giới thiệu hỗ trợ cho các vi mạch lập trình cỡ vừa và nhỏ. PLAN là một ngôn ngữ đơn giản, dùng để thực hiện các biểu thức của đại số Boolean và có khả năng giao tiếp với các công cụ lập trình để lập trình cho vi mạch.

## **4. Phần mềm HELD (Harris Enhanced Language for Programmable Logic).**

Công ty Harris phát triển phần mềm HELD để hỗ trợ cho các khách hàng sử dụng vi mạch lập trình của họ. HELD sử dụng giao diện tương tự như phần mềm PLAN nhưng cũng có những điểm khác biệt. HELD không có khả năng lựa chọn các vi mạch lập trình nhưng có khả năng kiểm tra lỗi tổng quát. Ngoài ra HELD còn yêu cầu các phương trình ngõ vào ở dạng tổng các tích ( SOP).

## **5. Phần mềm PLPL (Programmable Logic Programming Language).**

PLPL được công ty Avanced Micro Devices giới thiệu vào năm 1984. Đây là phần mềm tiên bộ nhất so với các phần mềm trước, có những đặc điểm mới và khả năng cài đặt được mở rộng hơn so với phần mềm AMAZE. Những đặc điểm mới như cho phép định nghĩa và sử dụng các chân của vi mạch cho một nhóm tín hiệu cũng như sử dụng các phương trình của đại số Boolean. PLPL cũng hỗ trợ các phương trình phức tạp có nhiều cấp logic khác nhau. Ngoài ra bộ biên dịch này cũng để ứng dụng nguyên lý Demorgan, các hàm của đại số Boolean nhưng không bắt được ở dạng tổng của các tích do đó cho phép cú pháp linh hoạt hơn.

## **6. Phần mềm APEEL (Assembler for Programmable Electrically Erasable Logic).**

Vào năm 1987, Công ty International Cmos Technology giới thiệu trình biên dịch APEEL. APEEL là một trình biên dịch đơn giản phù hợp với các yêu cầu thiết kế vừa và nhỏ và có chức năng mô phỏng. APEEL gồm một chương trình soạn thảo toàn màn hình và ở ngõ ra theo tiêu chuẩn của JEDEC. Nhưng khuyết điểm của bộ biên dịch này là không hỗ trợ để tối giản các biểu thức logic. Phần mềm APEEL cài đặt trên các máy tính cá nhân của công ty IBM và các công ty khác thích hợp với nó.

## **7. Phần mềm IPLDS II (Intel Programmable Logic Development System II).**

Phần mềm IPLDS II được công ty Intel giới thiệu để hỗ trợ cho các vi mạch họ EPLD. Điều cơ bản của phần mềm này là cho phép thiết kế theo 2 phương pháp là phương pháp dùng phương trình đại số Boolean và phương pháp liệt kê các lệnh. Để tối giản các biểu thức logic IPLDS II sử dụng thuật giải đơn giản ESPRESSO II – MV. Đó là thuật giải được phát triển bởi đại học California, nó được dùng để thực hiện việc rút gọn các tích số trong các hàm logic của các vi mạch do công ty Intel sản xuất. Tương tự như các phần mềm trước, IPLDS II cài đặt được trong các máy tính của công ty IBM và các máy tính khác có cấu hình thích hợp, được sử dụng kèm với công cụ lập trình cho vi mạch.

**8. Phần mềm CUPL ( Universal Compiler for Programmable Logic ).**

CUPL được công ty Assited Technology giới thiệu vào năm 1983. Đây là bộ biên dịch vạn năng được hỗ trợ cho 29 loại vi mạch các loại kể cả PROM và các công ty chế tạo vi mạch lập trình khác. CUPL là một ngôn ngữ mạnh hỗ trợ cho các phương trình của đại số Boolean , bảng sự thật và thiết kế sơ đồ trạng thái, CUPL được sử dụng hầu hết các máy vi tính cá nhân trên các hệ điều hành khác nhau như trên máy vi tính của công ty IBM hay CP/M, VAX/ UNIX và VAX/ VMS.

**9. Phần mềm ABEL (Advanced Boolean Expression Language).**

ABEL là phần mềm của công ty Data I/O, nó được sử dụng hầu hết các loại vi mạch lập trình khác nhau kể cả EPROM. Đây là bộ biên dịch vạn năng có nhiều chức năng hỗ trợ tương tự như CUPL.

Trên đây là giới thiệu sơ lược các phần mềm hỗ trợ cho vi mạch lập trình để soạn thảo là lập trình cho các vi mạch. Ngoài ra còn nhiều phần mềm của các công ty khác được sản xuất để hỗ trợ cho các vi mạch lập trình của họ.

Sau đây là bảng tóm tắt các ngôn ngữ thiết kế cho các vi mạch lập trình

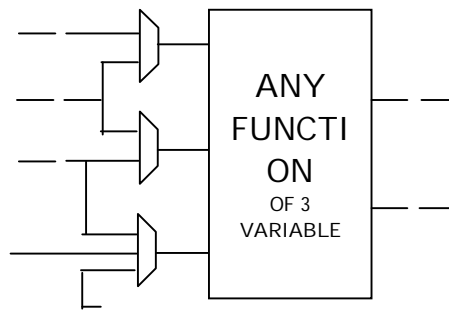
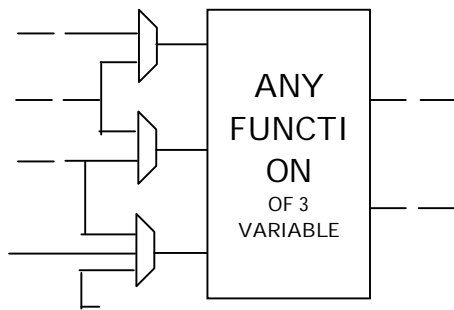
Phần mềm	Hỗ trợ cho các vi mạch	Phương trình đại số Boolean	Bảng sự thật	Sơ đồ nguyên lý	Dạng sóng	Rút gọn biểu thức logic
PALASM 2 (MMI)	X	*				
AMAZE (Signetics)	X	*		*		
PLAN (National)	X	"				
HELP (Harris)	X	*				
PLPL (AMD)	X	*				*
APEEL (ICT)	X	*				
A+PLUS (Altera)	X	*	*	*		*
iPLDS II (Intel)	X	*	*	*		*

ERASIC (Exel)	X	*	*	*		*
CUPL (Logical Dev)	X X	*	*	*		*
ABEL (Data I/O)	X X	*	*	*	*	*
ELDS (Pistohl)	X X	*				*
LOG/IC (Elan)	X X	*	*	*	*	*
PLDesigner (Minc)	XX	*	*	*		*

Giải thích:

X : Chỉ hỗ trợ cho vi mạch do chính công ty sản xuất.

XX : Hỗ trợ cho nhiều loại vi mạch lập trình.

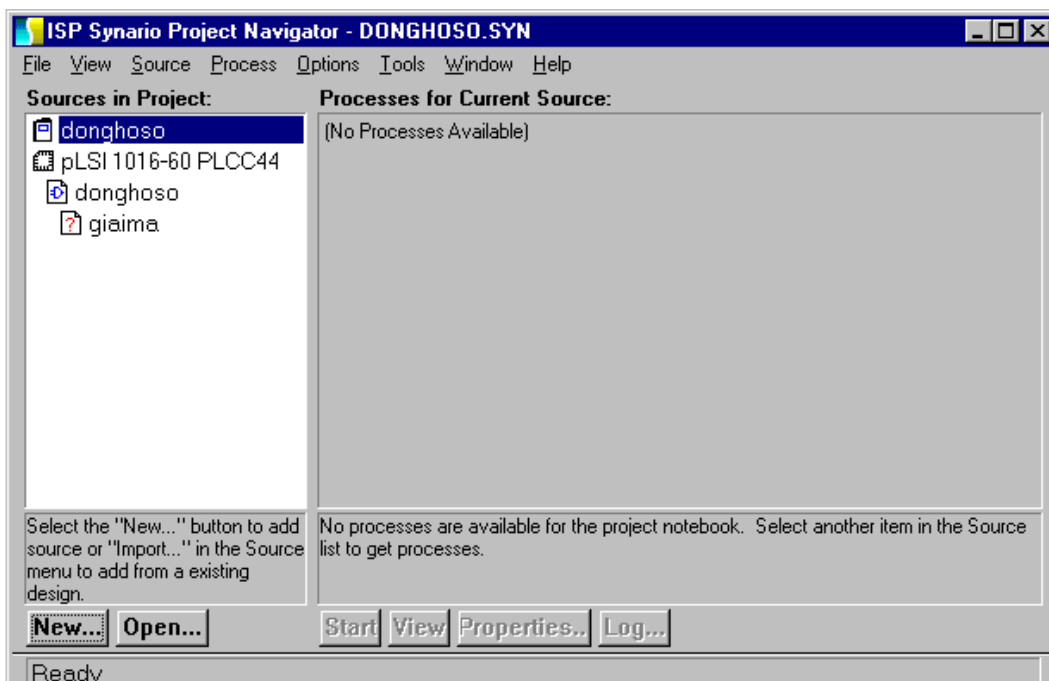


#### IV/ GIỚI THIỆU PHẦN MỀM SYNARIO.

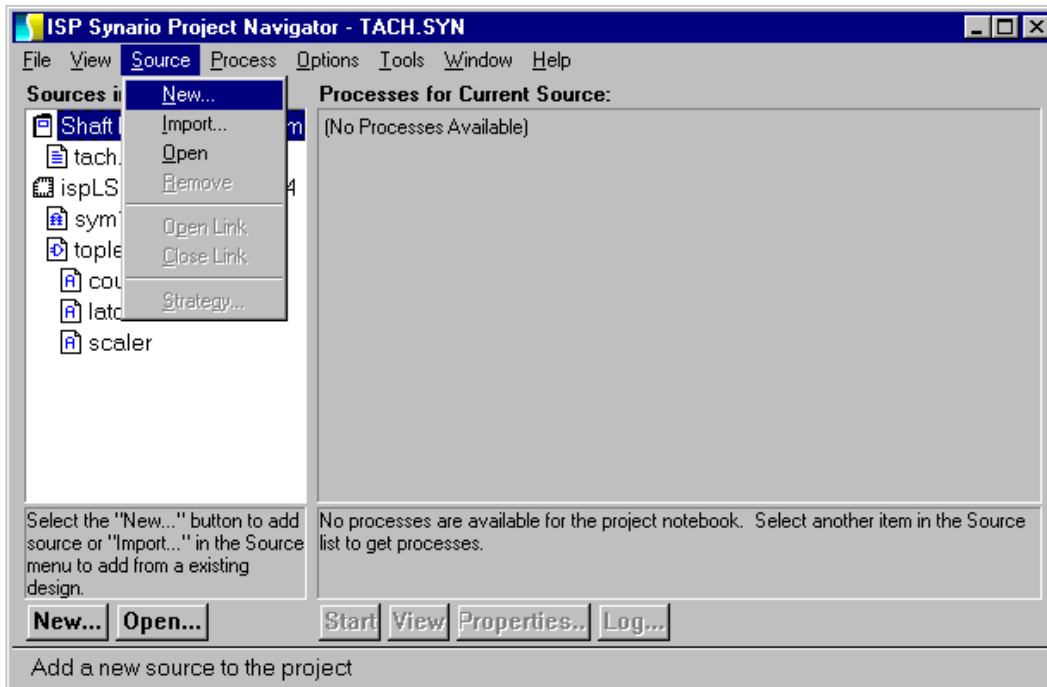
Phần mềm Synario của công ty Lattice cho phép lập trình các vi mạch PLD đến 14.000 cổng, chương trình có thể được soạn thảo dưới dạng sơ đồ mạch logic hoặc bằng ngôn ngữ AHDL.



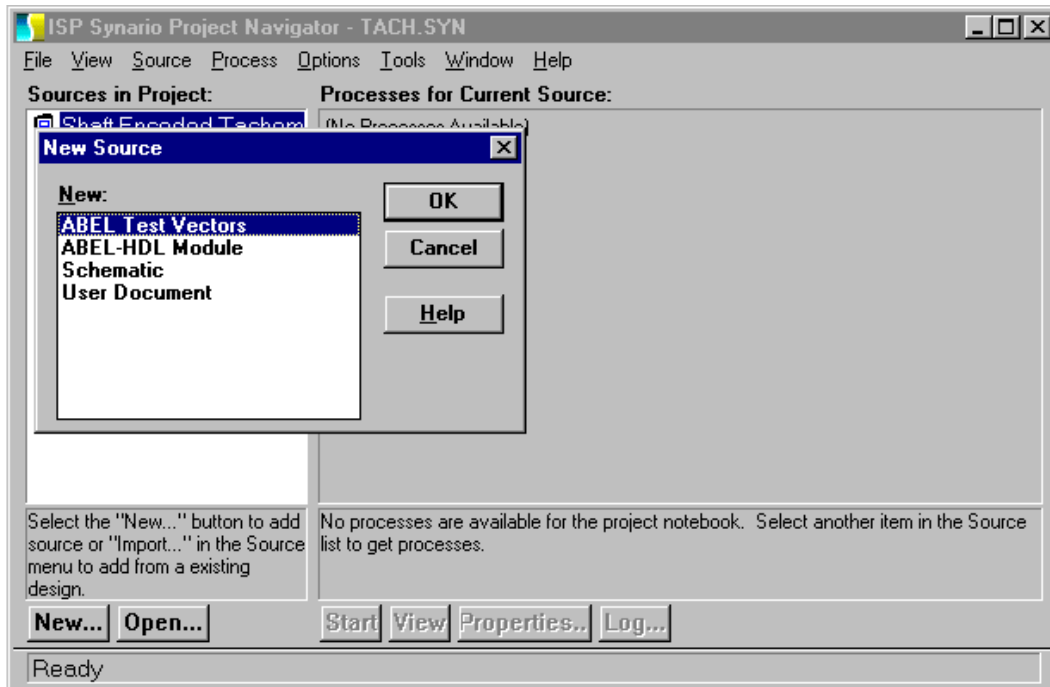
Gọi chương trình bằng cách kích đúp vào biểu tượng ISP Synario, màn hình chính của Synario xuất hiện như hình dưới đây



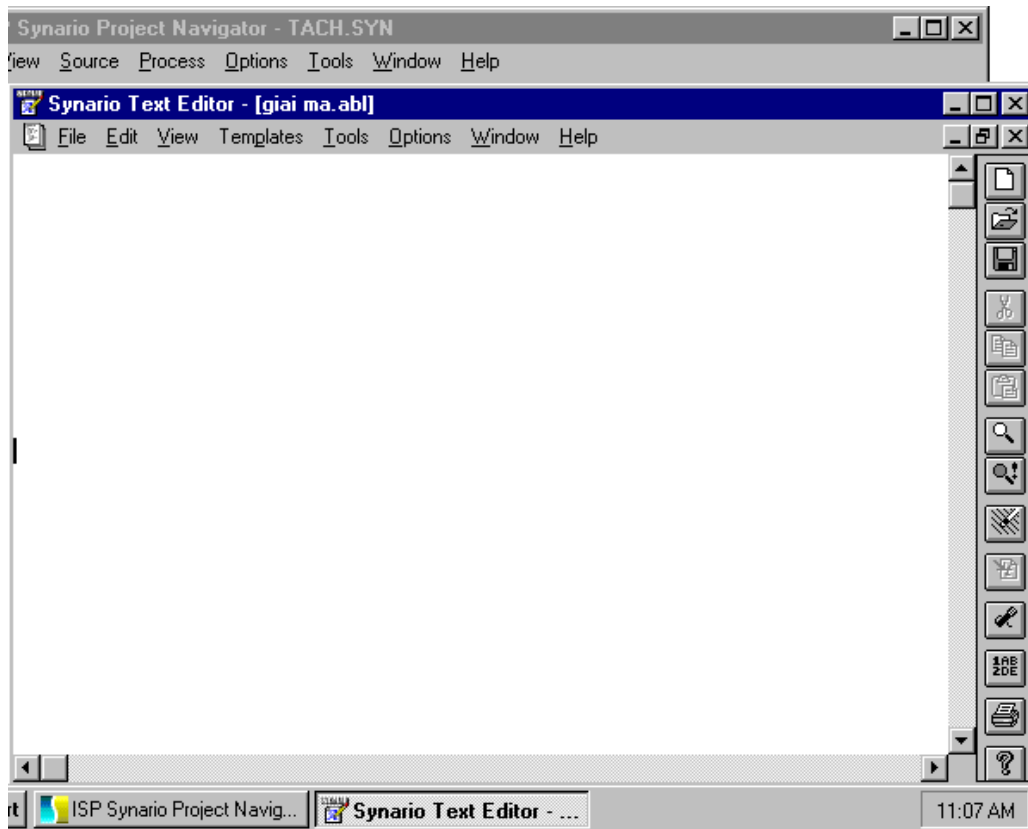
Bắt đầu soạn thảo bằng cách chọn **source - new** như màn hình dưới đây.



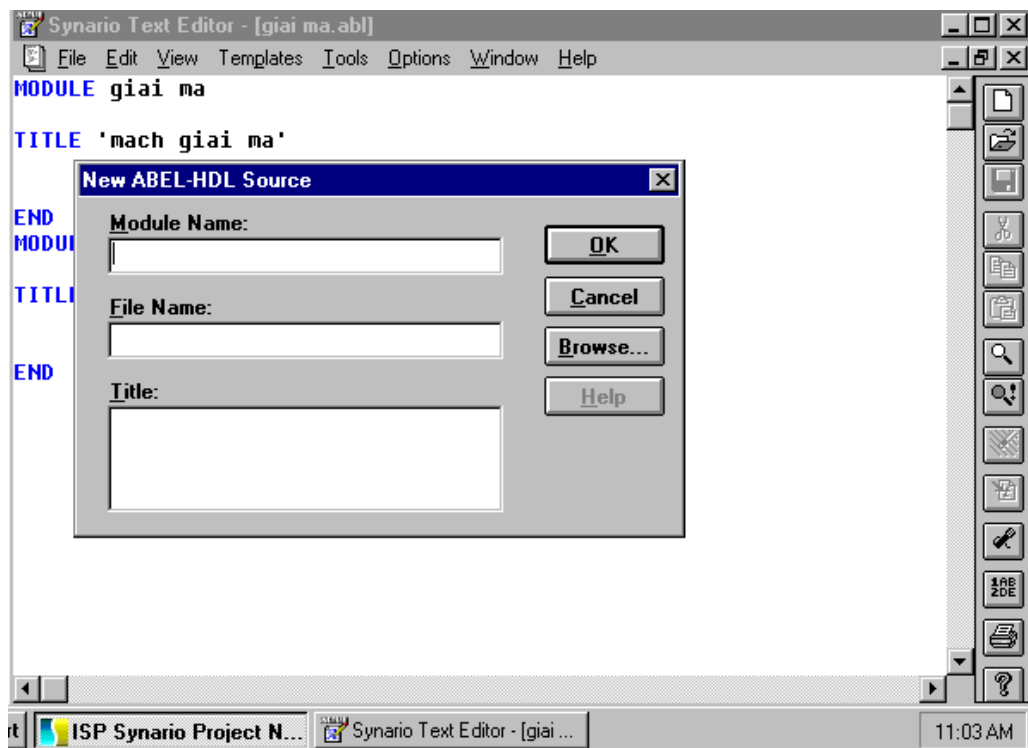
Chọn cách soạn thảo bằng sơ đồ bằng cách chọn **Schematic** hay lập trình ngôn ngữ AHDL chọn **ABEL-HDL Module**.



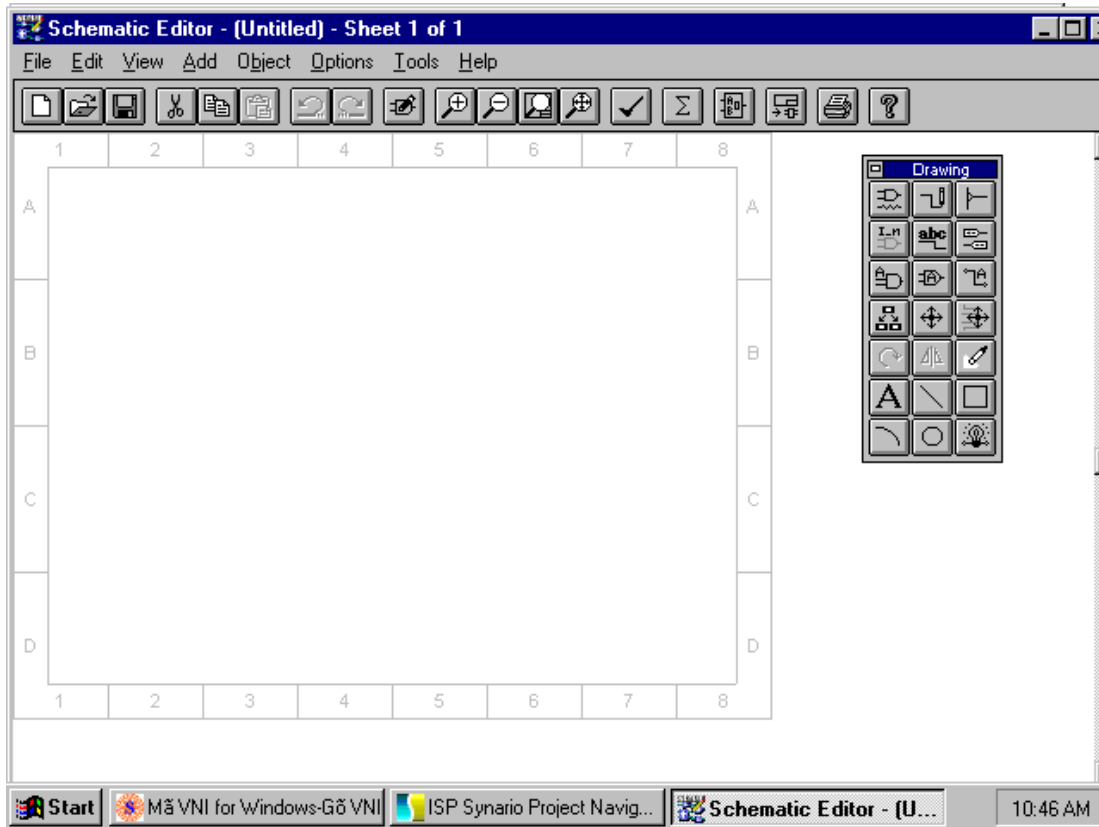
Màn hình soạn thảo ngôn ngữ AHDL.



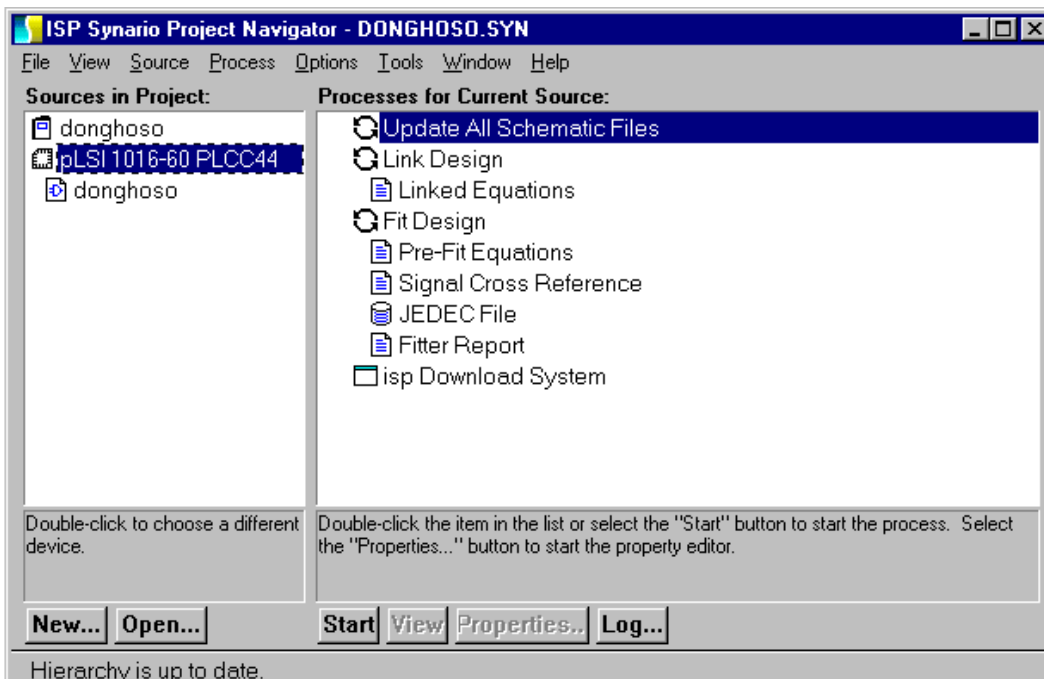
Chọn tên module, tên file, title.



Màn hình soạn thảo sơ đồ mạch logic



Trên màn hình chính có chứa sơ đồ tên "donghoso". Bên phải màn hình có 3 đề mục lệnh nhằm mục đích kiểm tra, đơn giản và thiết lập sơ đồ logic trong vi mạch.



*PHẦN II*

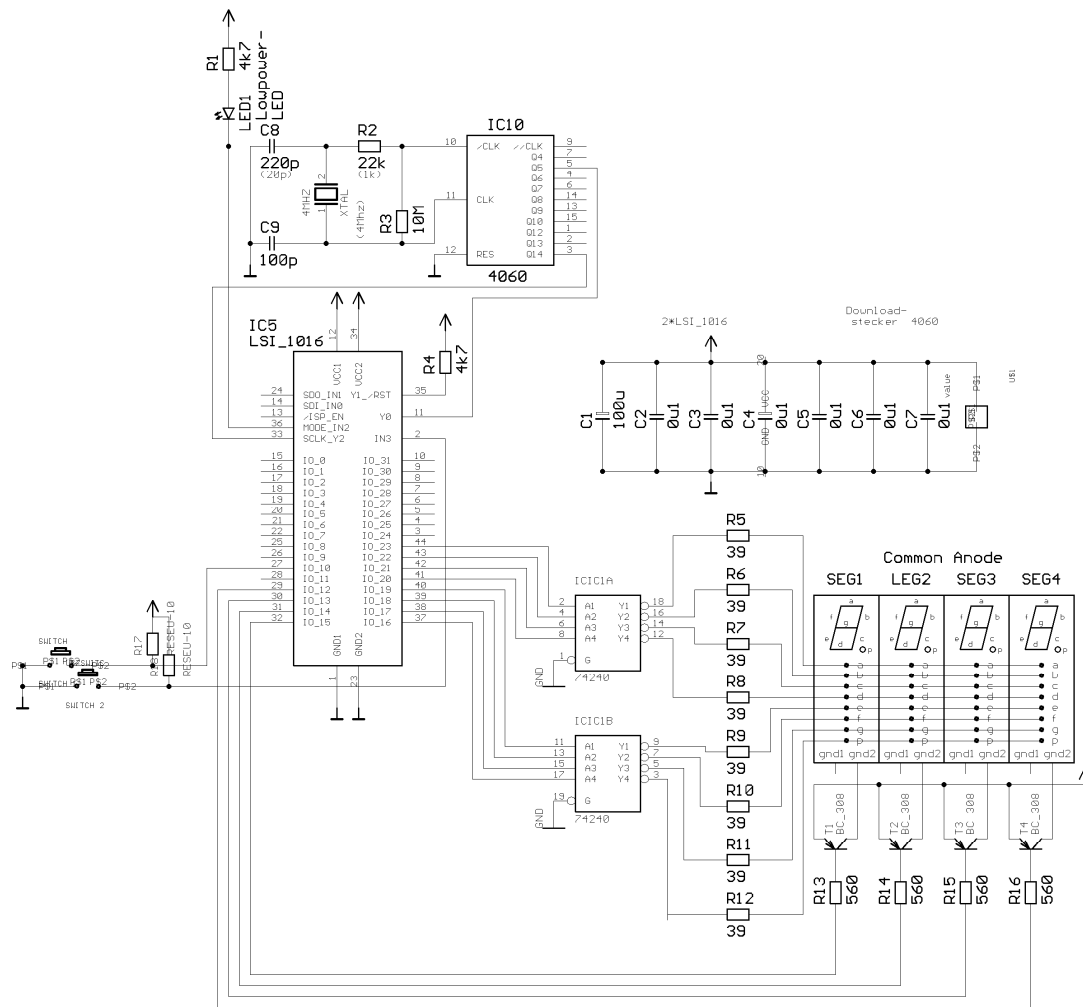
**THI CÔNG**



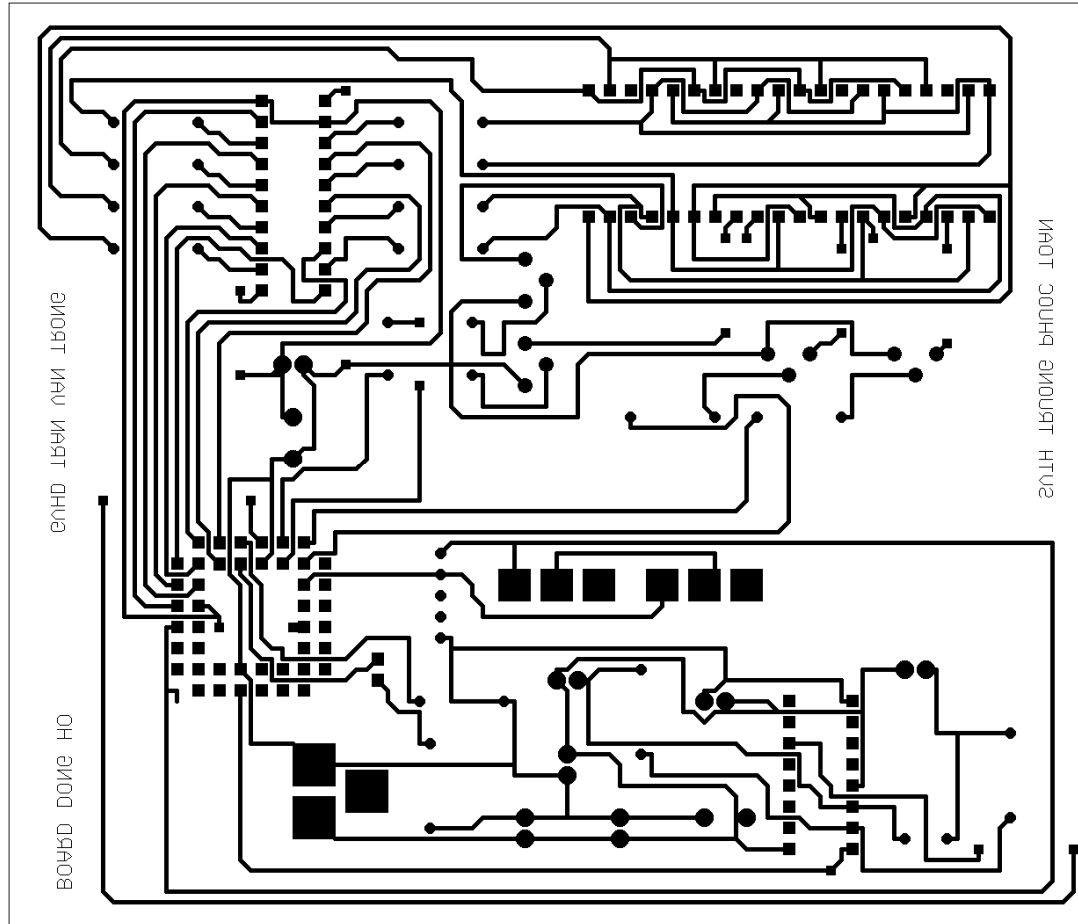
## I/ PHẦN CỨNG

Ứng dụng của vi mạch số lập trình hiện nay rất phổ biến trong các sản phẩm công nghiệp cũng như các sản phẩm trong sinh hoạt gia đình như: máy giặt, bếp điện tự động,... do trình độ còn hạn chế nên em chỉ áp dụng một ứng dụng nhỏ của vi mạch số lập trình là thi công mạch đồng hồ hiển thị số để thấy rõ những ưu điểm của PLD.

Sau đây em xin trình bày sơ đồ nguyên lý, sơ đồ mạch in và sơ đồ bố trí linh kiện của mạch đồng hồ số.

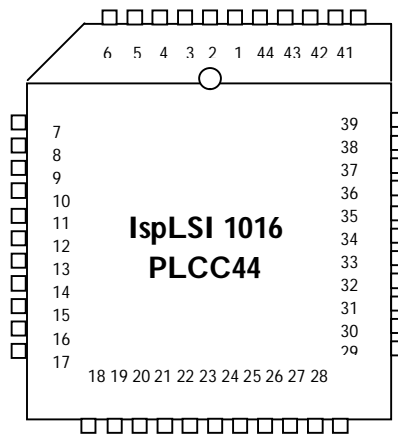


Sơ đồ nguyên lý



Sơ đồ mạch in

*Giới thiệu vi mạch lập trình của công ty lattice.*  
 Vi mạch iSP LSI 1016 là vi mạch được sử dụng để lập trình mạch đồng hồ hiển thị số:



Vi mạch iSP LSI 1016 là vi mạch có cấu tạo từ các cổng logic lập trình với mật độ tích hợp khoảng 2.000 cổng:

Thời gian trì hoãn  $T_{pd} = 7,5 \text{ ns}$

Tần số hoạt động  $f_{max} = 125\text{MHz}$

Vi mạch có 44 chân với kiểu vỏ PLCC

Vi mạch có 32 ngõ vào/ra do đó có thể được dùng với nhiều ứng dụng khác nhau trong cùng một board.

*Chức năng các chân của IC isp LSI 1016*

Số thứ tự chân	Chức năng	Số thứ tự chân	Chức năng
1	GND	23	GND
2	IN3	24	SOD/ IN 1
3	I/O 24	25	I/O 8
4	I/O 25	26	I/O 9
5	I/O 26	27	I/O 10
6	I/O 27	28	I/O 11
7	I/O 28	29	I/O 12
8	I/O 29	30	I/O 13
9	I/O 30	31	I/O 14
10	I/O 31	32	I/O 15
11	YO	33	Y2/ SCLK
12	Vcc (5V)	34	Vcc
13	$\overline{\text{IspEN/NC}}$	35	Y1/ $\overline{\text{RESET}}$
14	SDI/INO	36	IN2/ MODE
15	I/O 0	37	I/O 18
16	I/O 1	38	I/O 17
17	I/O 2	39	I/O 16
18	I/O 3	40	I/O 19
19	I/O 4	41	I/O 20
20	I/O 5	42	I/O 21
21	I/O 6	43	I/O 22
22	I/O 7	44	I/O 23

*Nguyên lý hoạt động của mạch.*

Mạch điện gồm có IC 4060 kết hợp với thạch anh 4MHz, điện trở R2,R3 và tụ C8,C9 tạo thành mạch dao động cung cấp hai tín hiệu 500ms và 1ms cho vi mạch lập trình để thực hiện các chức năng đếm, giải mã, và hiển thị led 7 đoạn bằng phương pháp quét. Mạch đồng hồ hiển số có 4 led: 2 led hiển thị giờ và 2 led hiển thị phút.

IC74240 có chức năng đệm đảo kết hợp với điện trở R5.. R12 làm mạch đệm để hiển thị. Bốn BJT có chức năng điều khiển quét.

Hai nút nhấn S1 và S2 dùng để điều chỉnh giờ và phút.

Ưu điểm của mạch điện này là:

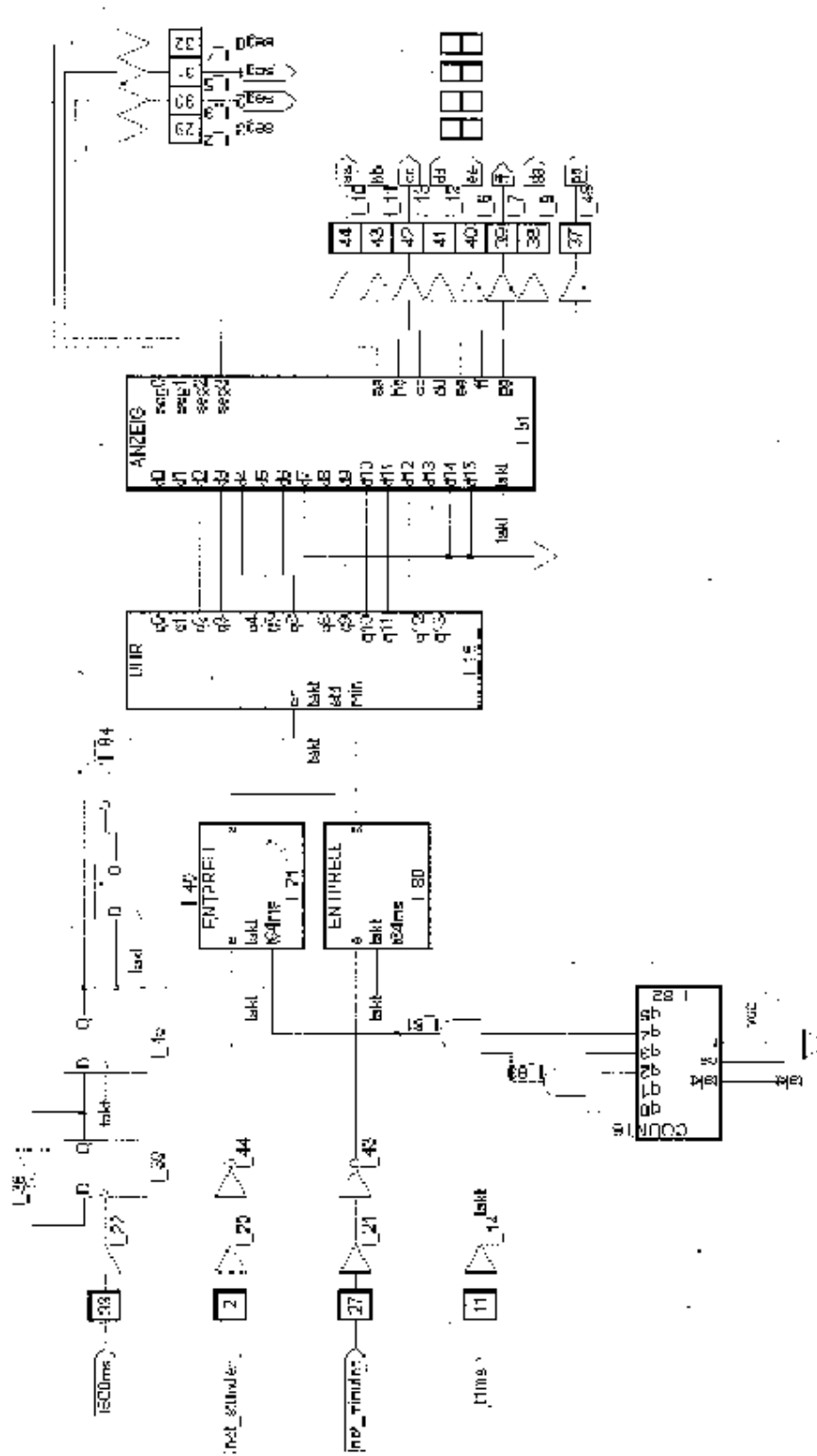
- Có ít linh kiện trong một board do đó mạch điện đơn giản thuận lợi trong việc kiểm tra khi có sự cố hư hỏng .
- Công suất tiêu thụ của mạch giảm đáng kể do dùng phương pháp quét led.

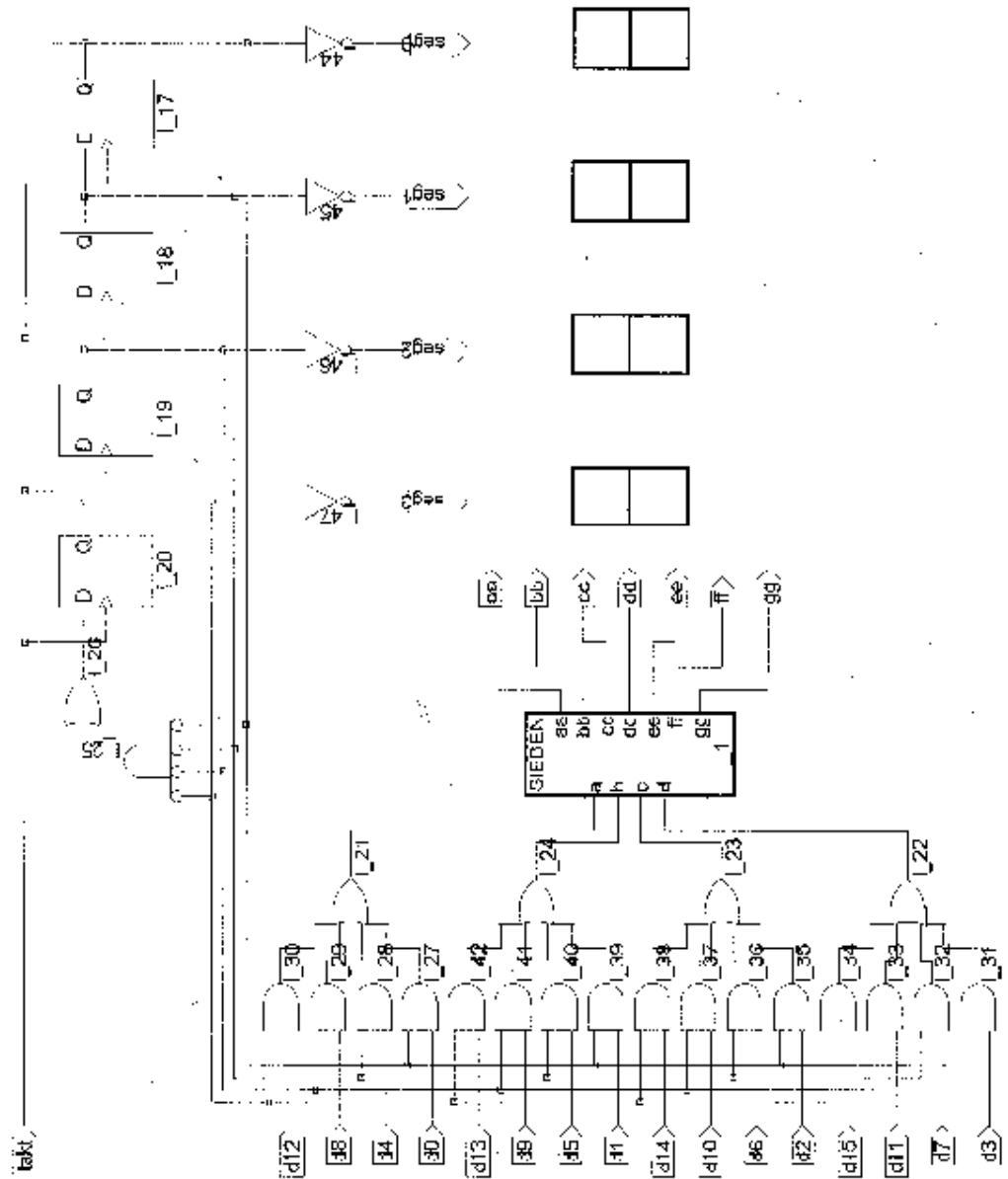
Khuyết điểm:

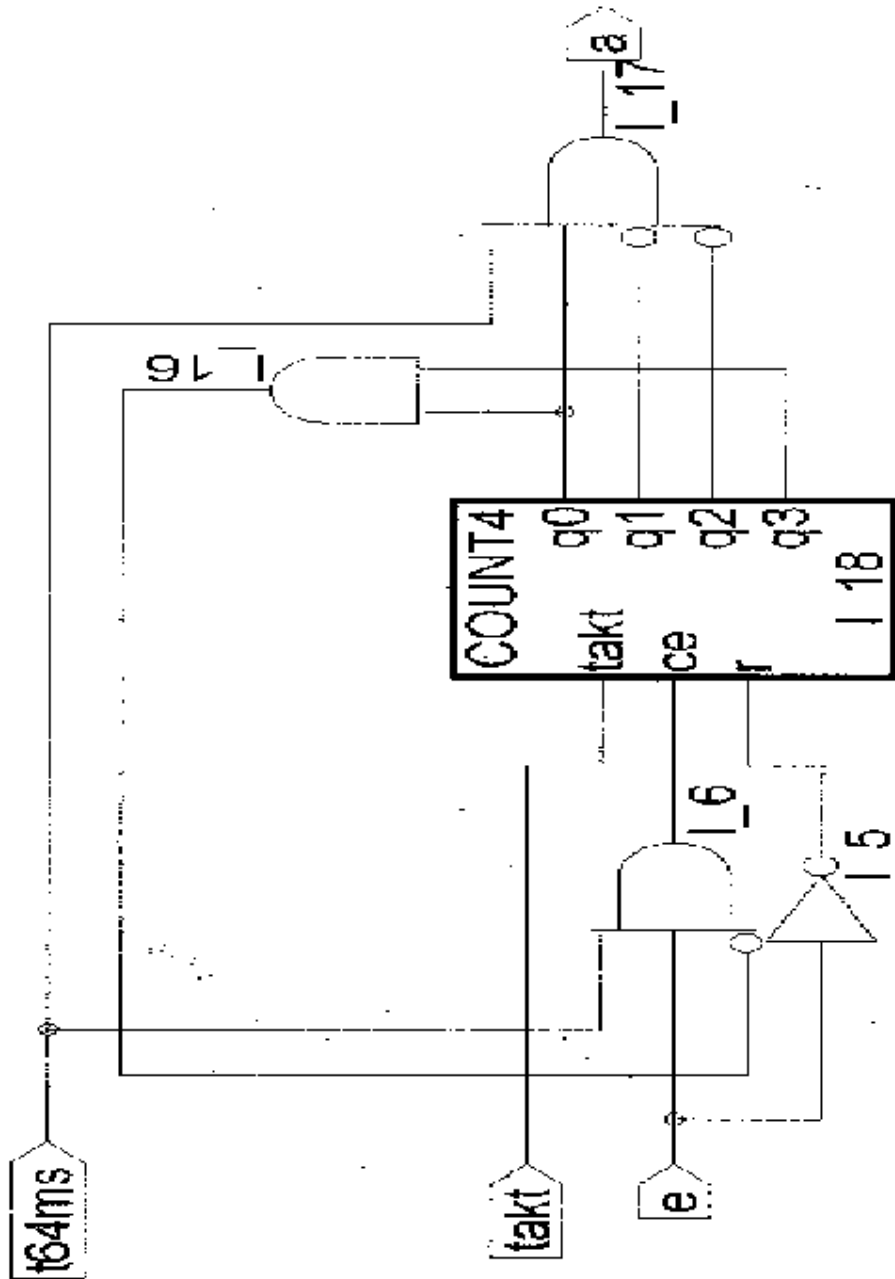
- Chưa tận dụng hết khả năng hoạt động của vi mạch do chỉ sử dụng có 13 đường vào ra.
- Do vi mạch lập trình khan hiếm trên thị trường nên giá thành cao.

**II/ PHẦN MỀM**

Sơ đồ mạch logic của mạch đồng hồ hiện số.







```

module giai ma
title 'giai ma'
''inputs
    a,b,c,d                pin 1,2,3,4;
''outputs
    aa,bb,cc,dd,ee,ff,gg  pin 5,6,7,8,9,10,11 listype'com';
equations
    aa= !d&!c&!b&!a # !d&!c&!b&!a # !d&!c&!b&!a # !d&!c&!b&!a #
!d&!c&!b&!a # !d&!c&!b& d&!c&!b&!a # !d&!c&!b&!a # !d&!c&!b&!a #
!d&!c&!b&!a # !d&!c&!b&!a # d&!c&!b&
    bb = !d&!c&!b&!a # !d&!c&!b&!a # !d&!c&!b&!a # !d&!c&!b&!a #
!d&!c&!b&!a # !d&!c&!b& d&!c&!b&!a # !d&!c&!b&!a # !d&!c&!b&!a #
!d&!c&!b&!a ;
    cc = !d&!c&!b&!a # !d&!c&!b&!a # !d&!c&!b&!a # !d&!c&!b&!a #
!d&!c&!b&!a # !d&!c&!b& d&!c&!b&!a # !d&!c&!b&!a # !d&!c&!b&!a #
!d&!c&!b&!a # !d&!c&!b&!a # d&!c&!b&
    dd = !d&!c&!b&!a # !d&!c&!b&!a # !d&!c&!b&!a # !d&!c&!b&!a #
!d&!c&!b&!a # !d&!c&!b& d&!c&!b&!a # !d&!c&!b&!a # !d&!c&!b&!a #
!d&!c&!b&!a ;
    ee = !d&!c&!b&!a # !d&!c&!b&!a # !d&!c&!b&!a # !d&!c&!b&!a #
!d&!c&!b&!a # !d&!c&!b& d&!c&!b&!a # !d&!c&!b&!a # !d&!c&!b&!a #
!d&!c&!b&!a # !d&!c&!b&!a # d&!c&!b&
    ff = !d&!c&!b&!a # !d&!c&!b&!a # !d&!c&!b&!a # !d&!c&!b&!a #
!d&!c&!b&!a # !d&!c&!b& d&!c&!b&!a # !d&!c&!b&!a # !d&!c&!b&!a #
!d&!c&!b&!a ;
    gg = !d&!c&!b&!a # !d&!c&!b&!a # !d&!c&!b&!a # !d&!c&!b&!a #
!d&!c&!b&!a # !d&!c&!b& d&!c&!b&!a # !d&!c&!b&!a # !d&!c&!b&!a #
!d&!c&!b&!a # !d&!c&!b&!a # d&!c&!b&
end
    
```



```

module count 4
title 'dem mode 4'
`constants
    c,x,p = . c . , c . , . p . ;
`inputs
    takt , up ,dn ,r , 19    pin 1, 2, 3, 4, 5 ;
`outputs
    q0 ,q1 ,q2 ,q3          pin 6, 7, 8, 9 istype 'reg' ;
equations
    (q0 . c , q1 . c , q2 . c , q3 . c ) =takt ;
`
    count up          count down          load 9    reset
q0 := ( ( ( up          # dn          )$q0 # 19 )&!r;
q1 := ( ( ( up&q0)      # (dn $! q0)    )$q1  &!19 )&!r ;
q2 := ( ( (up&q0&q1)    # (dn $!q0$q1)  )$q2  &!19 )&!r;
q3 := ( ( (up&q0&q1&q2) # (dn $!q0$q1$q2) )$q3  # !19 )&!r;
end
    
```

## *PHẦN III*

# KẾT LUẬN

### *Kết luận*

Qua 8 tuần nhận đề tài, mặc dù kiến thức và tài liệu có hạn chế nhưng em đã cố gắng hoàn thành tập luận văn đúng thời hạn đề ra. Qua thực tế kiểm nghiệm, Em có nhận xét về ưu khuyết điểm của đề tài.

Ưu điểm của mạch là:

- Mạch điện đơn giản (chỉ với 3 IC số trong mạch), dễ thực hiện, dễ kiểm tra các linh kiện khi gặp sự cố.
- Vi mạch lập trình có thể ứng dụng vào nhiều công việc khác nhau bằng cách lập trình cho vi mạch.
- Công cụ để lập trình đơn giản, dễ thực hiện.
- Làm giảm đáng kể số lượng IC trong một board.
- Hoạt động của vi mạch đảm bảo độ tin cậy, chính xác.

Khuyết điểm:

Do vi mạch lập trình trên thị trường coồn khan hiếm nên giá thành khá cao.

### *Hướng phát triển đề tài.*

Để thấy rõ những ưu điểm của vi mạch lập trình, với đề tài trên có thể thực hiện các ứng dụng thiết thực trong các yêu cầu của thực tế như thi công một mạch điện có nhiều chức năng vừa hiển thị giờ, có chuông báo giờ hẹn trước, điều khiển đóng ngắt các thiết bị từ xa...

Một lần nữa em xin cảm ơn thầy **TRẦN VĂN TRỌNG** đã tận tình hướng dẫn cho em. Xin cảm ơn quý thầy cô trong khoa đã dạy dỗ em và các bạn cùng khóa đã giúp đỡ em trong thời gian qua.

## TÀI LIỆU THAM KHẢO

- Giáo trình vi mạch số lập trình của tác giả: Thầy Trần Văn Trọng.
- Cơ sở kĩ thuật điện tử số Vũ Đức Thọ dịch.
- Tra cứu vi mạch số TTL và CMOS.
- Vi mạch số tập 1 Nguyễn Hữu Phương.
- Programmable Logic Designer' s Guide Roger C& Alford