

TS. NGUYỄN VIỆT NGUYÊN

KỸ THUẬT SỐ

DÙNG CHO SINH VIÊN
HỆ CAO ĐẲNG NGHỀ



NHÀ XUẤT BẢN GIÁO DỤC VIỆT NAM

TS. NGUYỄN VIỆT NGUYÊN

KỸ THUẬT SỐ

(Dùng cho sinh viên hệ cao đẳng nghề)

NHÀ XUẤT BẢN GIÁO DỤC VIỆT NAM

Công ty Cổ phần Sách Đại học – Dạy nghề – Nhà xuất bản Giáo dục Việt Nam
giữ quyền công bố tác phẩm

161 – 2009/CXB/2 – 2008/GD

Mã số: 6E018Y9 – DAI

Lời nói đầu

Cuốn giáo trình *Kỹ thuật số* (mô đun 15) được biên soạn trong chương trình giảng dạy theo tín chỉ của Tổng cục Dạy nghề cho đối tượng là sinh viên các trường Cao đẳng Nghề. Nội dung cuốn sách được chuẩn hóa bao gồm 12 bài với thời lượng 45 tiết lý thuyết (3 tín chỉ) và 120 tiết thực hành.

Nội dung tổng quát được phân bố trong 12 bài như sau:

Bài 1: Cơ sở kỹ thuật số

Bài 2: Các phần tử logic cơ bản

Bài 3: Các phần tử logic thông dụng

Bài 4: Mạch mã hóa (CODE)

Bài 5: Mạch giải mã (DECODE)

Bài 6: Mạch dồn kênh (MUX)

Bài 7: Mạch phân kênh (DEMUX)

Bài 8: Mạch logic tuần tự

Bài 9: Bộ nhớ chỉ đọc (ROM)

Bài 10: Bộ nhớ ghi/đọc (RAM)

Bài 11: Mạch chuyển đổi số – tương tự (DAC)

Bài 12: Mạch chuyển đổi tương tự – số (ADC)

Mỗi bài giảng đều có câu hỏi ôn tập củng cố, bài tập áp dụng và đặc biệt có nội dung thực hành giúp học viên có thể lắp ráp mạch điện đã học, đo lường, kiểm tra các thông số của mạch.

Do nội dung lý thuyết và thực hành kỹ thuật số rất phong phú và đa dạng, trong quá trình biên soạn tác giả cố gắng chọn lọc những vấn đề cơ bản và thông dụng nhất để đưa vào giáo trình, tôn trọng tính hệ thống, tính khoa học và sự phù hợp với các đối tượng học viên để có điều kiện tiếp thu tốt nhất kiến thức cơ sở, làm nền tảng cho sự phát triển tiếp tục trong lĩnh vực kỹ thuật số và ứng dụng. Trong quá trình biên soạn, sách không tránh khỏi những thiếu

sót, tác giả và NXB Giáo dục Việt Nam rất mong được sự đóng góp của bạn đọc để lần tái bản tới được hoàn thiện hơn.

Xin chân thành cảm ơn sự quan tâm của bạn đọc gần xa.

Mọi góp ý xin gửi về Công ty CP Sách Đại học – Dạy nghề, 25 Hàn Thuyên, Hà Nội. Điện thoại 04.38264974.

Xin chân thành cảm ơn!

TÁC GIẢ

BÀI 1. CƠ SỞ KỸ THUẬT SỐ

1.1. KHÁI NIỆM TÍN HIỆU TƯƠNG TỰ VÀ TÍN HIỆU SỐ

– Các đại lượng vật lý trong tự nhiên hay trong một quá trình công nghệ mang thông tin về đối tượng và được mô tả dưới dạng các tín hiệu và là các mục tiêu cần được theo dõi, thu thập, xử lý nhờ các thiết bị điện tử. Tín hiệu có bản chất là biểu hiện vật lý của tin tức.

– Trong lý thuyết thông tin, tin tức được định nghĩa là nội dung bao hàm trong một sự kiện hay trong một tập hợp nhiều sự kiện. Bản chất của quá trình thu thập, đo lường tín hiệu chính là quá trình xử lý tin tức chứa đựng trong các tín hiệu trên. Tín hiệu được biểu diễn về toán học là một hàm số biến thiên theo thời gian $s(t)$ hay một hàm số biến thiên theo tần số $s(f)$. Từ đây, ta có các định nghĩa sau đối với một tín hiệu $s(t)$:

+ Nếu $s(t)$ xuất hiện vào lúc t_0 và biến mất vào lúc $t_0 + \tau$ với τ là một đại lượng hằng số (đo theo đơn vị thời gian) thì giá trị τ được gọi là độ dài của tín hiệu $s(t)$. Đây chính là khoảng thời gian bị mắc bận của một hệ thống thông tin đối với $s(t)$.

+ Nếu trong τ quan sát thấy $s(t)$ là một hàm biến thiên liên tục theo t thì ta gọi $s(t)$ là một tín hiệu tương tự (hay tín hiệu analog). Còn ngược lại, nếu trong τ quan sát thấy $s(t)$ là một hàm biến thiên gián đoạn theo t thì ta gọi $s(t)$ là một tín hiệu xung – số (hay tín hiệu digital).

– Một số tham số cơ bản của tín hiệu được định nghĩa như sau:

+ Năng lượng toàn phần E_s của một tín hiệu:

$$E_s = \int_{t_0}^{t_0+\tau} s^2(t)dt = \int_{-\infty}^{+\infty} s^2(t)dt \quad (1.1)$$

+ Công suất của tín hiệu P_s :

$$P_s = \frac{E_s}{\tau} \quad (1.2)$$

+ Trị hiệu dụng của $s(t)$ ký hiệu là $s_{\text{hiệu dụng}}$:

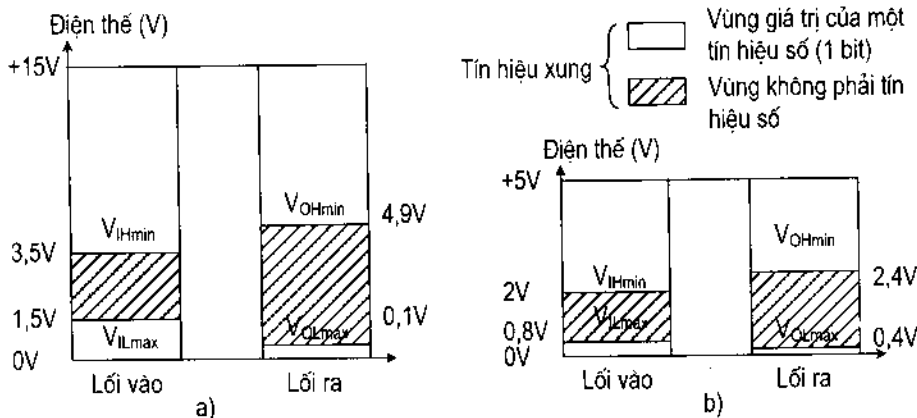
$$s_{\text{hiệu dụng}} = \sqrt{P_s} = \sqrt{\frac{E_s}{\tau}} \quad (1.3)$$

+ Trị trung bình hay thành phần không biến đổi theo thời gian (thành phần một chiều) của tín hiệu:

$$s_{\text{tb}} = \overline{s(t)} = \frac{1}{\tau} \int_{t_0}^{t_0+\tau} s(t) dt \quad (1.4)$$

– Các thông số trạng thái về điện của một phần tử hay một mạch điện tử là điện áp $u(t)$ xác định giữa hai điểm khác nhau của mạch hay dòng điện chảy $i(t)$ qua một điểm được dùng để biểu diễn tín hiệu. Khi đó, $s(t) \equiv u(t)$ hay $s(t) \equiv i(t)$. Trên thực tế, phổ biến là dùng các trạng thái điện thế (điện áp tại một điểm đang xét so với một điểm chung gọi là điện thế gốc 0V).

– Cần phân biệt hai dạng giá trị điện thế đối với một tín hiệu rời rạc (tín hiệu digital) là tín hiệu xung (dạng tổng quát) và tín hiệu số (dạng hẹp hơn). Để làm rõ việc này, ta hãy xét ví dụ đối với các khóa CMOS (hình 1.1a) trong biểu diễn nhị phân (mục 1.2). Dải điện thế sử dụng của mạch là từ 0V đến +15V. Các giá trị điện thế rời rạc nhận được trong khoảng này gọi là một xung điện áp xuất hiện tại lối vào hoặc lối ra của mạch. Tuy nhiên, tại lối vào, chỉ các giá trị điện thế từ mức 0V đến mức $V_{\text{ILmax}} = 1,5\text{V}$ hoặc từ mức $V_{\text{IHmin}} = 3,5\text{V}$ đến mức +15V được công nhận là tín hiệu số. Các mức $+1,5\text{V} < V_I < +3,5\text{V}$ chỉ là xung nhưng không được công nhận là tín hiệu số. Tương tự, tại lối ra, các mức giá trị $V_{\text{OLmax}} = 0,1\text{V} < V_O < V_{\text{OHmin}} = 4,9\text{V}$ không được công nhận là tín hiệu số. Điều này tương tự đối với các khóa TTL cho trên hình 1.1b.



Hình 1.1

– Tại lối ra của một cổng số, một trong hai mức logic cao (mức 1) hay thấp (mức 0) được chọn làm mức tích cực. Đây là mức mà tải của nó ở phía sau (sẽ nối trực tiếp vào) yêu cầu, mức còn lại được gọi là mức không tích cực.

1.2. KHÁI NIỆM VỀ MÃ VÀ HỆ ĐẾM

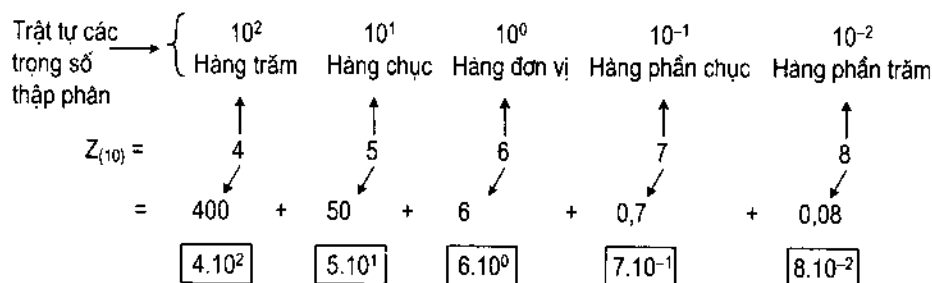
Để biểu diễn một tín hiệu dưới dạng số cần phải sử dụng một hệ thống các số đếm (gọi là một hệ đếm). Bản chất của quá trình mã hóa là chọn một hệ đếm xác định để biểu diễn tín hiệu. Khi đó, một tín hiệu số xác định được biểu diễn bởi một tập hợp các ký số gọi chung là một từ mã. Tính chất quan trọng nhất của một hệ thống số đếm là sử dụng một dãy các ký tự quy ước, thường là các ký tự số tự nhiên, để thể hiện giá trị một con số biểu diễn giá trị tương ứng của tín hiệu số. Giá trị của con số đã được biểu diễn được xác định qua giá trị từng ký số và vị trí của nó theo quy luật trọng số tăng dần khi dịch từ phải qua trái trong dãy số của con số này, mỗi phép dịch tăng một cấp lũy thừa của số đếm.

Các hệ đếm quan trọng nhất là hệ 10 (thập phân), hệ 2 (nhị phân), hệ 8 (OCTAL) và hệ 16 (HEXA).

1.2.1. Hệ đếm thập phân (hệ 10)

Hệ đếm thập phân sử dụng 10 ký số 0, 1, 2..., 9 đầu tiên trong dãy số tự nhiên. Cấu trúc một con số trong hệ thập phân được mô tả trên biểu đồ hình 1.2. Ở đây, vị trí được gọi là các trọng số thập phân 10^k tương ứng.

Ví dụ: $(z)_{10} = 456,78$ được biểu diễn cấu trúc trên hình 1.2.



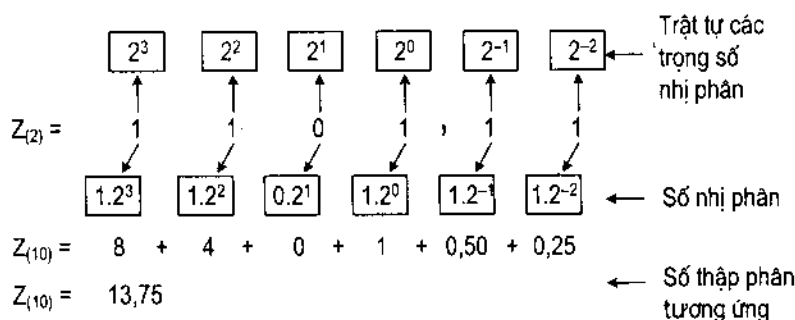
Hình 1.2

Giá trị con số được xác định bằng tổng các tích giá trị của mỗi ký số với trọng số vị trí thập phân của nó. Quy tắc này được áp dụng cho mọi hệ thống số đếm bất kỳ, ví dụ, ở hệ đếm cơ số 5 dùng 5 số tự nhiên đầu tiên 0, 1, 2, 3, 4 và một con số bất kỳ trong hệ đếm 5 chỉ được biểu diễn bởi các ký số 0 → 4 và vị trí của nó. Ví dụ:

$$432,1_{(5)} = 4.5^2 + 3.5^1 + 2.5^0 + 1.5^{-1} = 117,2_{(10)}$$

1.2.2. Hệ đếm nhị phân (hệ đếm 2)

Để xây dựng hệ đếm nhị phân chỉ dùng hai số tự nhiên đầu tiên 0 và 1. Như vậy, ký số của hệ nhị phân chỉ là 0 hoặc 1. Cũng tuân theo quy tắc vị trí, phần nguyên và phần lẻ được phân cách nhờ dấu phẩy (.). Có thể biểu diễn một số nhị phân theo tổ chức cấu trúc hình 1.3.



Hình 1.3

Như vậy, mỗi vị trí bên trong con số nhị phân thể hiện cấp trọng số nhị phân của vị trí đó theo quy luật tăng dần theo cấp số 2 khi dịch vị trí từ phải qua trái.

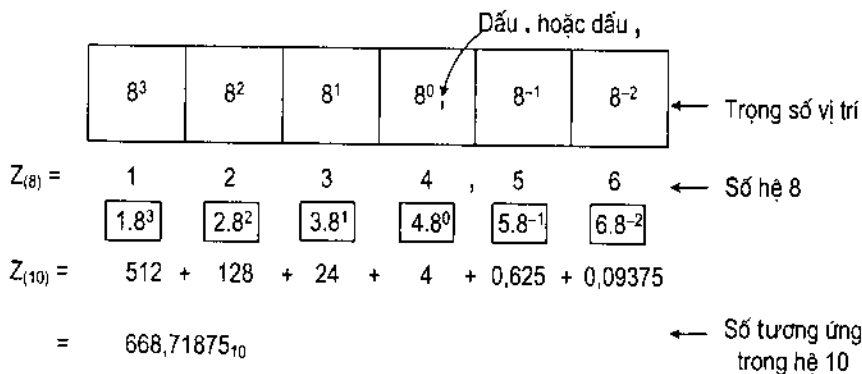
Ký số tận cùng bên phải có trọng số nhị phân thấp nhất gọi là bit có nghĩa nhỏ nhất (LSB – Least Significant Bit). Ký số tận cùng bên trái có trọng số nhị phân cao nhất gọi là bit có nghĩa lớn nhất (MSB).

Hình 1.3 đồng thời biểu thị quy tắc biến đổi một số $(z)_2$ trong hệ nhị phân thành một số $(z)_{10}$ trong hệ thập phân tương ứng, và có thể mở rộng cho dãy trước dấu phẩy đến n bit, sau dấu phẩy đến m bit bất kỳ.

1.2.3. Hệ đếm 8 (hệ OCTAL)

Hệ đếm 8 sử dụng 8 số tự nhiên đầu tiên 0, 1, 2,..., 7 và cũng theo luật vị trí để xác định trọng số 8^k tương ứng của mỗi vị trí có ký số đã

sử dụng. Quy tắc biểu diễn và biến đổi một con số trong hệ 8 thành số trong hệ 10 được cho trên hình 1.4. Ví dụ cho số: $1234,56_{(8)}$.



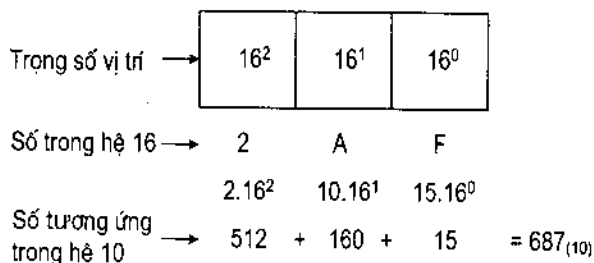
Hình 1.4

1.2.4. Hệ đếm 16 (HEXA)

Hệ đếm 16 sử dụng 16 ký tự và ký số là các số tự nhiên từ 0 đến 9 và 6 chữ cái in đầu tiên trong bảng chữ cái A, B, C, D, E và F.

Vị trí các số tự nhiên và ký tự thể hiện trọng số 16^k của con số khi chuyển nó về hệ đếm 10. Chú ý rằng, 6 chữ cái đầu tiên là mã quy ước của các số tự nhiên tiếp sau 10 số đầu: $A \equiv 10$, $B \equiv 11$, $C \equiv 12$, $D \equiv 13$, $E \equiv 14$ và $F \equiv 15$. Quy tắc biểu diễn một số trong hệ đếm 16 và quy tắc biến đổi nó về dạng số trong hệ đếm 10 được thể hiện qua hình 1.5.

Ví dụ: Với $z_{(16)} = 2AF_{(16)}$



Hình 1.5

Chú ý rằng, mỗi chữ cái dùng trong hệ đếm 16 tương ứng với biểu diễn một nhóm bit trong hệ đếm 2:

$A \equiv 10_{(10)} \equiv 1010_{(2)}$	$D \equiv 13_{(10)} \equiv 1101_{(2)}$
$B \equiv 11_{(10)} \equiv 1011_{(2)}$	$E \equiv 14_{(10)} \equiv 1110_{(2)}$
$C \equiv 12_{(10)} \equiv 1100_{(2)}$	$F \equiv 15_{(10)} \equiv 1111_{(2)}$

Nhờ đó, từng nhóm 4 bit có dạng trên luôn có thể viết thu gọn về một chữ cái tương ứng với nó làm cho cách biểu diễn một con số ở hệ 2 được viết thu gọn theo từng nhóm 4 bit kết hợp với mã BCD sẽ xét dưới đây.

1.2.5. Mã nhị phân BCD (Binary Coded Decimals)

Để biểu diễn một số thập phân trong mã BCD tự nhiên (còn gọi là mã BCD 8421) trước tiên, cần chuyển các số tự nhiên thập phân từ 0 đến 9 sang biểu diễn nhị phân tương ứng theo bảng 1.1. Sau đó, tổ chức chúng theo từng nhóm 4 bit biểu diễn vị trí của mỗi nhóm theo trọng số của cấu trúc hệ 10. Các chú ý quan trọng khi sử dụng mã BCD 8421 là:

+ Mỗi số thập phân tự nhiên từ 0 đến 9 được biểu diễn bởi một nhóm có đầy đủ 4 bit. Ví dụ: $3_{(10)} = 0011_{(2)}$ hay $5_{(10)} = 0101_{(2)}$.

+ Sáu tổ hợp 4 bit $1010_{(2)} = 10_{(10)} \dots 1111_{(2)} = 15_{(10)}$ không được sử dụng vì đã tương ứng với hàng chục trong hệ 10.

Ví dụ: số $357_{(10)}$ khi biểu diễn thành mã BCD 8421, áp dụng bảng 1.1 có:

$$3_{(2)} = 0011$$

$$5_{(2)} = 0101 \Rightarrow 357_{(10)} = 0011\ 0101\ 0111_{(BCD8421)}$$

$$7_{(2)} = 0111$$

Bảng 1.1

Số thập phân	Số nhị phân BCD 8421			
	2^3	2^2	2^1	2^0
	8	4	2	1
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1

+ BCD chỉ là một loại mã (biểu diễn số) hoàn toàn không phải là một hệ thống số đếm như các hệ 10, hệ 2, hệ 8 hay hệ 16 đã xét (tổ chức cấu hình của mã pha trộn hai cách biểu diễn hệ 2 và hệ 10).

+ Khi chọn trọng số vị trí ở các cột đối với mã BCD khác đi có khả năng tạo ra vô số các dạng mã BCD với nhiều quy luật đặc biệt của các mã như thể hiện trên bảng 1.2 với các loại mã BCD 7421, BCD 5121 hay BCD 2421.

Bảng 1.2. Các loại mã BCD và nhị phân khác

Số hệ 10	Mã BCD								Mã thừa 3	Một dạng mã Gray	Mã Jonhson	BCD 8421													
	8	4	2	1	7	4	2	1			5		1	2	1	2	4	2	1	J ₄	J ₃	J ₂	J ₁	J ₀	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	0	1	1	1	1	0
2	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	1	0	1	1	0	1
3	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	1	1	1	0	0
4	0	1	0	0	0	1	0	0	0	1	1	1	0	1	0	0	0	1	1	1	1	1	0	1	1
5	0	1	0	1	0	1	0	1	1	0	0	0	1	0	1	1	0	1	1	1	1	1	0	1	0
6	0	1	1	0	0	1	1	0	1	0	0	1	1	1	0	0	0	1	0	1	1	1	0	0	1
7	0	1	1	1	1	0	0	0	1	0	1	0	1	1	0	0	0	1	0	0	1	1	0	0	0
8	1	0	0	0	1	0	0	1	1	0	1	1	1	1	0	0	1	1	0	0	0	0	1	1	1
9	1	0	0	1	1	0	1	0	1	1	1	1	1	1	0	1	1	0	0	0	0	0	1	1	0

Từ bảng 1.2 có thể nhận xét các quy tắc để xây dựng các loại mã khác nhau trong cột tương ứng dựa trên mã gốc ban đầu là BCD 8421.

Ví dụ: Với mã BCD 7421 chỉ khác mã BCD 8421 gốc ở cột trọng số cao nhất là mọi số tự nhiên từ 0 đến 9 được biểu diễn lại theo trọng số 7421.

$$\text{Ví dụ: } 9_{(10)} = 1001_{(\text{BCD8421})} = 1010_{(\text{BCD7421})}$$

Mã Gray có các từ mã liên tiếp chỉ khác nhau 1 bit giá trị.

Tương tự, cho mã Eiken BCD 2421, ví dụ:

$$8_{(10)} = 1000_{(\text{BCD8421})} = 1111_{(\text{BCD2421})}$$

Mã thừa 3 có được nhờ cấu tạo từ mã BCD 8421 trong đó mỗi trạng thái số tự nhiên được cộng thêm $3_{(10)} = 0011_{(8421)}$.

Ví dụ:

$$3_{(10)} = 0000_{(\text{BCD8421})} + 0011_{(8421)} = 0011_{(\text{thừa 3})}$$

$$9_{(10)} = 1001_{(\text{BCD8421})} + 0011_{(8421)} = 1100_{(\text{thừa 3})}$$

1.2.6. Các phép biến đổi mã

Việc biến đổi các cách biểu diễn số trong các hệ đếm (hay các mã) khác nhau được thực hiện theo các quy tắc tương ứng trên các hình 1.2 đến hình 1.5 và được dựa trên các bảng 1.1, bảng 1.2 đã đưa ra.

1. Chuyển đổi các số giữa hệ 2 và hệ 10

– Khi biến đổi một số từ hệ 2 sang hệ 10, có thể sử dụng bảng 1.3a cho các số chỉ có phần nguyên hay bảng 1.3b cho các số có cả phần lẻ, thực hiện cộng kết quả các cột từ phải qua trái để có kết quả cuối cùng.

Bảng 1.3a

Số thập phân	Số nhị phân											
	2^{10}	2^9	2^8	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	
	1024	512	256	128	64	32	16	8	4	2	1	
174				1	0	1	0	1	1	1	0	
				128 + 32 + 8 + 4 + 2								

Bảng 1.3b

Số thập phân	Số nhị phân																
	...	2^{10}	2^9	2^8	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	2^{-1}	2^{-2}	2^{-3}	2^{-4}	...
	...	1024	512	256	128	64	32	16	8	4	2	1	$\frac{1}{2}$	$\frac{1}{4}$	$\frac{1}{8}$	$\frac{1}{16}$	...
909,625		0	1	1	1	0	0	0	1	1	0	1	1	0	1		
		512+256+128 + 8 + 4 + 1 + $\frac{1}{2}$ + $\frac{1}{8}$															

Chú ý: Bảng 1.3b có thể mở rộng cả về hai phía cho phần nguyên và phần lẻ.

– Khi thực hiện biến đổi một số từ hệ 10 sang hệ 2 $A_{10} \rightarrow A_2$ ta có thể sử dụng bảng 1.3b theo chiều từ trái sang phải kèm với một cột lập hiệu số liên tiếp như trên bảng 1.4 cho các số thập phân chỉ có phần nguyên.

Bảng 1.4

Số thập phân	Số nhị phân										
	2^{10}	2^9	2^8	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
660	1024	512	256	128	64	32	16	8	4	2	1
	0	1	0	1	0	0	1	0	1	0	0
190	0	0	0	1	0	1	1	1	1	1	0

$$\begin{array}{r}
 660 \\
 - 512 \\
 \hline
 148 \\
 - 128 \\
 \hline
 20 \\
 - 16 \\
 \hline
 4 \\
 - 4 \\
 \hline
 0
 \end{array}
 \qquad
 \begin{array}{r}
 190 \\
 - 128 \\
 \hline
 62 \\
 - 32 \\
 \hline
 30 \\
 - 16 \\
 \hline
 14 \\
 - 8 \\
 \hline
 6 \\
 - 4 \\
 \hline
 2 \\
 - 2 \\
 \hline
 0
 \end{array}$$

– Khi cần chuyển đổi phần lẻ thập phân sang số nhị phân (đứng sau dấu phẩy) có thể thực hiện theo quy tắc "nhân 2 trừ 1" như sau:

+ Đặt phần lẻ A_{10} muốn biến đổi ở tận cùng bên trái (bảng 1.5) nhân nó với 2 được $2A_{10}$.

+ Nếu kết quả $2A_{10} \geq 1$ thì trừ tích số cho 1: $2A_{10} - 1$ đồng thời đặt lại số 1 đầu tiên phần lẻ sau dấu phẩy.

Nếu kết quả $2A_{10} < 1$ thì đặt số 0 ở vị trí này.

+ Nhân phần dư $2A_{10} - 1$ hay $2A_{10}$ ở một trong hai bước trên với 2 để tiếp tục tìm ký số thứ 2 sau dấu phẩy...

+ Quá trình tiếp diễn cho tới khi đạt tới số bit lẻ theo yêu cầu (nằm sau dấu phẩy cần bao nhiêu bit biểu diễn).

Bảng 1.5

Số lẻ thập phân $A_{10} = 0,8325$	$2A_{10} = 1,665$ $2A_{10} - 1 = 0,665$	$2.0,665 = 1,330$ $1,33 - 1 = 0,33$	$2.0,33 = 0,66$ $0,66 < 1$	$2.0,66 = 1,32$ $1,32 - 1 = 0,32$
Số lẻ nhị phân A_2	1	1	0	1

Từ bảng 1.5 với số lẻ thập phân $A_{10} = 0,8325$ khi đổi sang số lẻ nhị phân lấy tới 4 bit sau dấu phẩy sẽ bằng $A_2 = 0,1101$

Tương tự, cho ví dụ số: $A_{10} = 0,3125 \Leftrightarrow A_2 = 0,0101$

Khi số thập phân có cả 2 phần nguyên và phần lẻ, cần kết hợp cả hai bảng 1.4 và 1.5 để thực hiện biến đổi sang số nhị phân tương đương.

2. Chuyển đổi giữa các số trong hệ 8 và hệ khác

– Chuyển đổi A_8 sang A_{10} cần dùng quy tắc trọng số vị trí theo hệ thức trên hình 1.4 và hình 1.2. Kết hợp với bảng 1.6 chiều từ trái qua phải:

$$\text{Ví dụ: } A_8 = 375_8 = 3.8^2 + 7.8^1 + 5.8^0$$

$$= 3.64 + 7.8 + 5.1 = 253_{10}$$

$$A_8 = 74,6_8 = 7.8^1 + 4.8^0 + 6.8^{-1} = 60,75_{10}$$

– Chuyển đổi A_{10} sang A_8 cần thực hiện việc chia 8 liên tiếp để lấy phần dư (tương tự như quy luật đã làm với A_{10} sang A_2) theo bảng 1.6 chiều từ phải qua trái.

Bảng 1.6

Số hệ 10	Số hệ 8 (0, 1, 2, 3, 4, 5, 6, 7)							
	8^4	8^3	8^2	8^1	8^0	8^{-1}	8^{-2}	8^{-3}
	4096	512	64	8	1	$\frac{1}{8}$	$\frac{1}{64}$	$\frac{1}{512}$
253,75		0	3	7	7	6	0	0

$$\begin{array}{r}
 253,75 \\
 - 192 \\
 \hline
 61,75 \\
 - 56 \\
 \hline
 5,75 \\
 - 5 \\
 \hline
 0,75 \\
 - 0,125 \\
 \hline
 0
 \end{array}$$

– Khi cần chuyển đổi một số từ hệ 8 sang hệ 2 ($A_8 \rightarrow A_2$) hoặc ngược lại, thực hiện quy tắc chuyển từng ký số của A_8 sang một nhóm gồm 3 ký số nhị phân tương đương theo bảng 1.7 hoặc ngược lại:

Bảng 1.7

Ký số của A_8	0	1	2	3	4	5	6	7
Nhóm A_2	000	001	010	011	100	101	110	111

Kết quả là với A_8 có n ký số (nhận trị từ 0 tới 7) thì A_2 sẽ có 3n bit tương ứng:

$$\text{Ví dụ: } A_8 = 475_8 = 100\ 111\ 101_2 = A_2$$

$$A_8 = 1674_8 = 001\ 110\ 111\ 100_2 = A_2$$

$$A_2 = 101\ 110\ 011_2 = A_8 = 563_8$$

$$A_2 = 001\ 010\ 110_2 = A_8 = 126_8$$

Chú ý:

+ Việc chuyển đổi từ A_2 sang A_8 đặc biệt thuận lợi cho việc ghi tốc ký một số nhị phân quá dài nhờ bản ghi ở hệ A_8 .

+ Trong nhiều trường hợp, việc biến đổi mã dạng $A_{10} \Leftrightarrow A_8 \Leftrightarrow A_2$ thường thuận lợi (nhanh) hơn so với việc làm trực tiếp $A_{10} \Leftrightarrow A_2$.

3. Chuyển đổi một số từ hệ 16 sang hệ khác hoặc ngược lại

– Sử dụng quy tắc trên hình 1.5 cho phép biến đổi theo hai chiều (bảng 1.8) các số trong hệ 16 và hệ 10:

Bảng 1.8

Số thập phân	Số hệ 16 (0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F)					
	16^3	16^2	16^1	16^0	16^{-1}	16^{-2}
	4096	256	16	1	$\frac{1}{16}$ (0,0625)	$\frac{1}{256}$ (0,00390625)
934	→	2 (2.16 ²)	A (10.16 ¹)	6 (6.16 ⁰)	0	0
854, 671875	←	3 3.256	5 5.16 ¹	6 6.16 ⁰	A 0,625 (10.0,0625)	C 0,046875 (12.0,0039)

– Khi cần chuyển đổi số từ hệ 16 sang hệ 2 hoặc ngược lại, ta dùng bảng 1.9 biểu diễn các ký tự của mã 16 thành từng nhóm 4 bit trong mã nhị phân.

Bảng 1.9

Ký tự của hệ 16	0	1	2	3	4	5	6	7
Nhóm bit ở hệ 2 tương đương	0000	0001	0010	0011	0100	0101	0110	0111
Ký tự của hệ 16	8	9	A	B	C	D	E	F
Nhóm bit ở hệ 2 tương đương	1000	1001	1010	1011	1100	1101	1110	1111

Ví dụ: $z_{16} = 9E3_{16} = 1001\ 1110\ 0011_2 = z_2$

$$z_2 = \underbrace{0011}_3 \underbrace{0110}_6 \underbrace{1001}_9 \underbrace{1110}_{E_{16}} = 369E_{16}$$

(Chú ý nhóm từng 4 bit từ phải sang trái, nhóm bit cao nhất nếu chưa đủ 4 bit thì thêm 0 bên trái cho đủ thành một nhóm).

1.2.7. Cách biểu diễn một con số trong một hệ đếm

- Một con số được biểu diễn trong hệ đếm gồm 4 phần cơ bản:
- + Phần dấu đầu tiên bên trái (dấu + hay - trong hệ thập phân).
- + Phần nguyên đứng trước dấu phẩy (,) hoặc dấu chấm (.).
- + Dấu phẩy (,) hoặc dấu chấm (.) phân cách giữa phần nguyên và phần lẻ.

- + Phần lẻ nằm sau dấu phẩy.

Ví dụ: $-125,258_{10}$.

- Để thuận tiện cho việc thao tác tính toán, thường sử dụng phương pháp dấu phẩy tĩnh: đặt dấu phẩy sau chữ số có nghĩa tận cùng bên phải, gọi là cách biểu diễn ở dạng chỉ có phần nguyên:

$$425,758 = 42578, (10^{-3})$$

- Cũng có thể đặt dấu phẩy trước chữ số có nghĩa tận cùng bên trái, gọi là cách biểu diễn ở dạng chỉ có phần lẻ:

$$354,26 = ,35426 (10^3)$$

- Để biểu diễn một số có dấu dương hay âm trong hệ nhị phân, có thể sử dụng cách bổ sung vào số đã cho một ký số mào đầu (tận cùng bên trái trước bit có ý nghĩa lớn nhất MSB) theo quy định:

Ký số 1 biểu diễn (thông báo) số nhị phân đứng sau nó là một số âm.

Ký số 0 biểu diễn số nhị phân đứng sau nó là một số dương.

Đây là các biểu diễn dấu kèm theo trị số thật của số đang xét.

Ví dụ:

Các bit thông báo dấu

$$\begin{aligned} +151_{10} &= \boxed{0} / 1001\ 0111_2 \\ -240_{10} &= \boxed{1} \ 1111\ 0000_2 \end{aligned}$$

- Có thể dùng một số bù hai để biểu diễn một số nhị phân dấu âm. Mã bù 2 của một số nhị phân có được nhờ tạo mã bù 1 (bằng cách đảo giá trị tương ứng ở mọi vị trí các bit) sau đó kết quả này được cộng thêm 1, ví dụ ta có số ban đầu:

$$+45_{10} = (0\ 101101)_2$$

↘
Mã của trị số thật

Bước 1. Tạo mã bù 1 của 45_{10} (1010010)₂.

Bước 2. Cộng thêm 1 vào kết quả

$$(1\ 010011)_2 = -45_{10}$$

↘
Mã bù 2 của trị số thật

1.3. CÁC PHÉP TÍNH SỐ HỌC TRONG HỆ NHỊ PHÂN

1.3.1. Phép cộng trong hệ nhị phân

– Phép cộng hai số nhị phân phải tuân theo các quy tắc sau:

+ Cộng các ký số cùng cột. (cùng trọng số).

+ Cộng lần lượt từ phải sang trái (lần lượt từ LSB đến MSB).

+ Kết quả phép cộng phải tuân theo nhóm quy tắc (1.5)

$$0 + 0 = 0$$

$$0 + 1 = 1 + 0 = 1$$

$$1 + 1 = 10 \text{ (bằng 0 nhớ 1 để cộng tiếp cột bên trái liền kề)}$$

$$1 + 1 + 1 = 11 \text{ (bằng 1 nhớ 1 để cộng tiếp ở cột trái liền kề)}$$

(1.5)

Khi phép cộng không nhớ dùng hai quy tắc dòng trên.

Khi phép cộng có nhớ dùng thêm hai quy tắc dòng dưới trong nhóm hệ thức (1.5).

– Khi cộng nhiều số nhị phân cần thực hiện cộng dần từng nhóm hai số:

Ví dụ:

$\begin{array}{r} + 101101_2 (45_{10}) \\ 1111_2 (15_{10}) \\ \hline 111100_2 (60_{10}) \end{array}$	$\begin{array}{r} + 11,011_2 (3,375_{10}) \\ 10,110_2 (2,750_{10}) \\ \hline 110,001_2 (6,125_{10}) \end{array}$
--	--

1.3.2. Phép trừ nhị phân

– Phép trừ nhị phân tuân theo các quy tắc sau:

+ Trừ các ký số nhị phân cùng cấp (cùng cột).

+ Trừ lần lượt từ phải (LSB) sang trái (MSB).

+ Kết quả phép trừ tuân theo nhóm quy tắc (1.6):

$$0 - 0 = 0$$

$$1 - 0 = 1$$

$$1 - 1 = 0$$

$$10 - 1 = 1 \text{ (vay 1 ở cấp cao liền kề)}$$

(1.6)

Lưu ý, khi ở cột 2^k xảy ra trường hợp $0 - 1$, cần áp dụng quy tắc vay 1 của số bị trừ ở cấp 2^{k+1} theo quy tắc dòng cuối cùng của các hệ thức 1.6.

Nếu áp dụng quy tắc vay cho một số nhị phân dạng:

$$\underbrace{1\ 00 \dots 00}_{n \text{ số } 0} - 1 = 0 \underbrace{11 \dots 11}_{n \text{ số } 1}$$

Kết quả là một số nhị phân gọi là mã bù 1 của số bị trừ.

Hai số nhị phân gọi là bù 1 của nhau khi ở mọi cột tương ứng trị của chúng là nghịch đảo của nhau. Ví dụ, số bù 1 của $z = 110010$ là $z_{(\text{bù } 1)} = 001101 = \bar{z}$.

Để ý tới bảng 1.7 ta thấy cột đầu cho mã BCD 8421 và cột cuối cho mã $\overline{\text{BCD8421}}$ là mã bù của nhau.

Ví dụ: Thực hiện phép trừ hai số

$$\begin{array}{r} \overset{\text{vay}}{\text{~}} \\ 1100\ 0111_2\ (199_{10}) \\ - 110\ 1101_2\ (109_{10}) \\ \hline 0101\ 1010_2\ (90_{10}) \end{array}$$

– Có thể thực hiện phép trừ hai số nhị phân $A_2 - B_2$ bằng cách thực hiện phép cộng $A_2 + (-B_2)$. Số nhị phân B_2 được đảo dấu khi thiết lập mã bù 2 cho nó bằng cách tạo $(\bar{B}_2 + 1)$.

Chú ý ở đây cả A_2 và B_2 phải biểu diễn ở dạng số nhị phân có bit dấu (0 cho A_2 và 1 cho B_2).

1.3.3. Phép nhân hai số nhị phân

– Xét với hai số nhị phân không dấu, quy tắc thực hiện phép nhân cho trên hệ thức (1.7).

$$\left. \begin{array}{l} 0.0 = 0 \\ 1.0 = 0.1 = 0 \\ 1.1 = 1 \end{array} \right\} \quad (1.7)$$

Tương tự như trong hệ thập phân, thực hiện nhân liên tiếp từng cột của một thừa số số nhân với tất cả các cột của thừa số kia của số bị nhân (từ phải qua trái). Hai hàng kết quả của hai cột liên tiếp nhau phải đặt dịch trái một cột, cuối cùng cộng các hàng tích số lại sẽ nhận được tích đầy đủ.

Ví dụ:

$$\begin{array}{r}
 \times 1010 \leftarrow \text{Số bị nhân} \\
 \times 101 \leftarrow \text{Số nhân} \\
 \hline
 + 1010 \\
 + 0000 \\
 + 1010 \\
 \hline
 110010 \leftarrow \text{Tích đầy đủ}
 \end{array}
 \left. \begin{array}{l} \\ \\ \end{array} \right\} \begin{array}{l} \text{Các hàng tích} \\ \text{bán phần} \end{array}$$

Chú ý: Khi nhân hai số có phần lẻ dùng phương pháp dấu phẩy tĩnh đưa chúng về biểu diễn chỉ có phần nguyên và làm như trên.

Kết quả cuối cùng dấu phẩy được dịch trái từ LSB đi $(m + n)$ cột nếu số bị nhân có m bit lẻ và số nhân có n bit lẻ sau dấu phẩy.

1.3.4. Phép chia hai số nhị phân

Có thể phân thành hai trường hợp đặc trưng:

1. Khi số bị chia lớn hơn (nhiều bit hơn) số chia: $A_2 > B_2$.

Bước 1. Chọn một nhóm bit cao của A_2 có số bit bằng số bit của B_2 , thương số sẽ bằng 1 khi chia được và bằng 0 khi không chia được.

Bước 2. Khi kết quả bước trên là 1 (chia được) cần thực hiện phép trừ để lấy phần dư giữa nhóm bị chia và số chia. Sau đó, lần lượt hạ từng cột kế tiếp của A_2 , mỗi lần hạ một cột xuống phần dư, thương số được điền tiếp kết quả 1 nếu chia được hoặc kết quả 0 nếu không chia được vào bên phải (khi nhóm bị chia mới vẫn còn nhỏ hơn B_2 là không chia được). Bước hai được làm liên tiếp cho tới LSB của A_2 được hạ xuống phần dư cuối cùng.

Bước 3. Nếu phép chia không hết (còn phần dư) sau khi đã hạ LSB của A_2 , sẽ thực hiện lấy thêm phần lẻ (sau dấu phẩy) của thương. Muốn vậy, cần thêm các số 0 vào sau phần dư cuối cùng và tiếp tục các bước 1 và bước 2 cho tới lúc chia hết hay cho tới lúc đến số lẻ cuối cùng của thương theo yêu cầu.

2. Khi số bị chia A_2 nhỏ hơn số chia B_2 ($A_2 < B_2$)

Cách thực hiện tương tự như 3 bước khi $A_2 > B_2$, với kết quả thương số chỉ gồm các phần lẻ sau dấu phẩy. Muốn vậy, mỗi lần thêm số 0 vào A_2 cần ghi vào thương một ký số 0 sau dấu phẩy cho tới khi A_2 đã "lớn hơn" B_2 .

Ví dụ 1:

$$\begin{array}{r}
 \begin{array}{r}
 101' \ 101 \\
 - 101 \\
 \hline
 0001' \\
 - 0000 \\
 \hline
 10' \\
 - 00 \\
 \hline
 101' \\
 - 101 \\
 \hline
 000
 \end{array}
 \begin{array}{l}
 101 \\
 \hline
 1001 \leftarrow \text{Thương số}
 \end{array}
 \end{array}$$

000 ← Phần dư

Ví dụ 2:

$$\begin{array}{r}
 \begin{array}{r}
 110' \ 000 \\
 - 101 \\
 \hline
 0010'0'0' \\
 - 101 \\
 \hline
 0110' \\
 - 101 \\
 \hline
 001
 \end{array}
 \begin{array}{l}
 101 \\
 \hline
 1001,1 \leftarrow \text{Thương số} \\
 \text{lấy đến 1} \\
 \text{bit lẻ sau} \\
 \text{dấu phẩy}
 \end{array}
 \end{array}$$

001 ← Phần dư

1.4. ĐẠI SỐ LOGIC (ĐẠI SỐ BOOLE)

1.4.1. Cơ sở của đại số logic

– Đại số logic được hiểu là những cơ sở toán học xây dựng cho tập hợp các đối tượng chỉ nhận hai trạng thái có hoặc không có, các mệnh đề đúng hoặc sai. Các đối tượng có thuộc tính này được biểu diễn bằng các biến logic, ký hiệu bằng các chữ cái A, B, C hay X, Y, Z,... Khi trạng thái của đối tượng là tồn tại (có) ta gán cho biến logic tương ứng giá trị quy ước là 1 và ký hiệu A (hoặc X...); còn khi trạng thái của đối tượng là không tồn tại (không có) ta gán cho biến logic thể hiện nó giá trị quy ước là 0 và ký hiệu là $\bar{A}$ (hoặc $\bar{X}$...). Có những trạng thái luôn luôn tồn tại được quy ước là hằng số 1, luôn luôn không tồn tại là hằng số 0. Như vậy, trong tập hợp tồn tại hai hằng số 1 và 0.

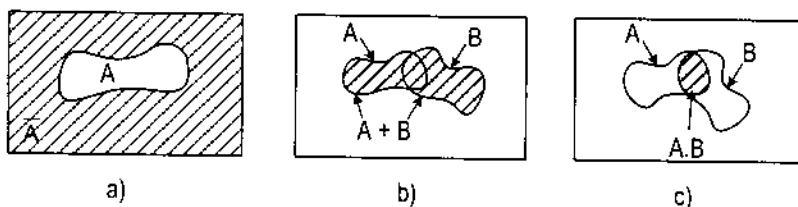
– Tác động lên các biến logic có 3 phép toán cơ sở:

+ Phép phủ định đối với một biến logic A, còn gọi là phép đảo (NOT) là khi tác động tới A, A sẽ nhận giá trị đảo (của giá trị có trước khi tác động) là $\bar{A}$. Ký hiệu phép đảo là dấu "-" trên ký hiệu biến.

+ Phép cộng logic, còn gọi là phép hoặc (OR) được ký hiệu bằng dấu "+" giữa hai hoặc nhiều biến số hạng, ví dụ: A + B hoặc A + B + C... Kết quả được gọi là một tổng.

+ Phép nhân logic, còn gọi là phép và (AND) được ký hiệu bằng dấu "." giữa các biến thừa số (ví dụ A.B). Kết quả được gọi là một tích.

Các phép toán cơ sở có thể mô tả theo giản đồ Venn trong lý thuyết tập hợp (hình 1.6).



Hình 1.6. Giản đồ Venn mô tả 3 phép toán logic cơ bản:

a) Phép phủ định logic (phép đảo - NOT);

b) Phép cộng logic (phép hoặc - OR);

c) Phép nhân logic (phép và - AND)

1.4.2. Các tính chất quan trọng của tập hợp các biến logic

Khi tác động ít nhất một trong ba phép toán logic cơ bản vừa định nghĩa lên các biến logic một hoặc nhiều lần (liên tiếp hay đồng thời) sẽ nhận được kết quả là các hàm logic. Do tính chất các biến là biến trạng thái nên hàm logic cũng là hàm trạng thái tức là cũng chỉ nhận một trong hai giá trị 1 (có) hay 0 (không có). Khi hai (hay nhiều) hàm logic nhận được do tác động từ nhiều cách khác nhau cho kết quả giống nhau thì các hàm này được gọi là tương đương nhau và ký hiệu dấu "=" giữa các kết quả này.

– Các tính chất cơ bản:

+ Tính hoán vị của phép cộng và của phép nhân:

$$A + B = B + A; A.B = B.A \quad (1.8)$$

+ Tính kết hợp:

$$A + B + C = (A + B) + C = A + (B + C) \quad (1.9)$$

$$A.B.C = (A.B).C = A.(B.C)$$

+ Tính phân phối:

$$A.(B + C) = A.B + A.C \quad (1.10)$$

+ Các quy tắc của phép phủ định:

$$\overline{(\overline{A})} = A; \overline{(\overline{\overline{A}})} = \overline{A} \quad (1.11)$$

+ Các quy tắc của phép cộng:

$$A + A = A; A + \overline{A} = 1 \quad (1.12)$$

$$A + 0 = A; A + 1 = 1$$

+ Các quy tắc của phép nhân:

$$A.A = A; A.\bar{A} = 0 \quad (1.13)$$

$$A.1 = A; A.0 = 0$$

+ Tính hấp thụ:

$$A(A+B) = A \quad (1.14)$$

+ Tính nhất quán:

$$\text{Nếu } A+B = B \text{ thì } A.B = A \quad (1.15)$$

+ Tính dán:

$$A + \bar{A}B = A + B \quad (1.16)$$

+ Luật DeMorgan lập hàm phủ định của một hàm:

$$\overline{(A+B)} = \bar{A}.\bar{B}; \overline{(A.B)} = \bar{A} + \bar{B} \quad (1.17)$$

Có thể dễ dàng chứng minh các tính chất trên khi sử dụng đồ thị Venn của lý thuyết tập hợp hay sử dụng các khóa trạng thái A và B trong một mạch, phép cộng là mắc song song và phép nhân là mắc nối tiếp các khóa. Khi nối mạch các khóa có trị 1 (có dòng) khi ngắt mạch các khóa có trị 0 (không có dòng điện).

1.4.3. Phương pháp biểu diễn hàm logic

Có 3 cách thông dụng biểu diễn hàm logic:

– Biểu diễn hàm logic bằng bảng chân lý (bảng trạng thái) bằng cách liệt kê mọi tổ hợp giá trị có thể của biến và trị tương ứng của hàm trong một bảng. Với hàm 2 biến, bảng gồm 3 cột và 4 hàng, hàm 3 biến có 4 cột và 8 hàng (bảng 1.10a, b), hàm n biến có (n + 1) cột và 2^n hàng.

Bảng 1.10. Bảng trạng thái của hàm 2 biến $F_1(a)$ và hàm 3 biến $G_2(b)$

Biến		Hàm F_1
A	B	
0	0	0
0	1	1
1	0	1
1	1	1

a)

Biến	A	0	0	0	0	1	1	1	1
	B	0	0	1	1	0	0	1	1
	C	0	1	0	1	0	1	0	1
Hàm G_2		1	1	1	0	1	0	0	0

b)

– Biểu diễn hàm logic bằng biểu thức gồm các ký hiệu hàm, ký hiệu biến các phép toán tác động. Có hai dạng cơ bản trong cách biểu diễn này:

+ Biểu thức ở dạng tổng các tích các biến.

Khi mỗi số hạng của tổng có đủ mặt các biến ta có dạng đầy đủ và mỗi số hạng được gọi là một mintec ký hiệu m_i ($i = 0, 1, 2, \dots, n - 1$).

Hàm có n biến sẽ có tối đa 2^n mintec khác nhau.

Khi có ít nhất một số hạng không đủ biến thì dạng hàm là không đầy đủ.

+ Biểu thức có dạng tích các tổng các biến (đầy đủ và không đầy đủ).

Khi ở dạng đầy đủ mỗi thừa số của tích được gọi là một maxtec ký hiệu M_j , có tối đa 2^n maxtec cho hàm n biến ($j = 0, 1, 2, \dots, n - 1$).

Bảng 1.11 đưa ra các m_i và M_j của hàm 2 biến và hàm 3 biến tương ứng. Cần chú ý quy ước biến nhận trị 0 ký hiệu là $\bar{A}$ và nhận trị 1 ký hiệu là A và trong cùng một hàng tổng các chỉ số ($i + j$) luôn bằng $(n - 1)$.

Bảng 1.11. Các mintec m_i và maxtec M_j của hàm 2 biến (a) và 3 biến (b)

Biến		Mintec m_i	Maxtec M_j
A	B		
0	0	$m_0 = \bar{A}\bar{B}$	$M_3 = \bar{A} + \bar{B}$
0	1	$m_1 = \bar{A}B$	$M_2 = \bar{A} + B$
1	0	$m_2 = A\bar{B}$	$M_1 = A + \bar{B}$
1	1	$m_3 = AB$	$M_0 = A + B$

a)

Biến			Mintec m_i	Maxtec M_j
A	B	C		
0	0	0	$m_0 = \bar{A}\bar{B}\bar{C}$	$M_7 = \bar{A} + \bar{B} + \bar{C}$
0	0	1	$m_1 = \bar{A}\bar{B}C$	$M_6 = \bar{A} + \bar{B} + C$
0	1	0	$m_2 = \bar{A}B\bar{C}$	$M_5 = \bar{A} + B + \bar{C}$
0	1	1	$m_3 = \bar{A}BC$	$M_4 = \bar{A} + B + C$
1	0	0	$m_4 = A\bar{B}\bar{C}$	$M_3 = A + \bar{B} + \bar{C}$
1	0	1	$m_5 = A\bar{B}C$	$M_2 = A + \bar{B} + C$
1	1	0	$m_6 = AB\bar{C}$	$M_1 = A + B + \bar{C}$
1	1	1	$m_7 = ABC$	$M_0 = A + B + C$

b)

Từ bảng 1.11 đã cho, biểu thức của hàm F_1 hay G_2 trong bảng 1.10a và b có dạng:

$$F_1 = A + B = \bar{A}B + A\bar{B} + AB = m_1 + m_2 + m_3 = \sum m(1, 2, 3)$$

$$G_2 = \bar{A}\bar{B}\bar{C} + \bar{A}\bar{B}C + \bar{A}B\bar{C} + A\bar{B}\bar{C} = m_0 + m_1 + m_2 + m_4 = \sum m(0, 1, 2, 4)$$

Nếu viết theo các maxtec ta có:

$$\bar{F}_1 = \prod M(1, 2, 3) = M_1 \cdot M_2 \cdot M_3 = (A + \bar{B})(\bar{A} + B)(\bar{A} + \bar{B}) \text{ hay } F_1 = M_0$$

$$\bar{G}_2 = \prod M(0, 1, 2, 4) = M_0 \cdot M_1 \cdot M_2 \cdot M_4$$

$$= (A + B + C)(A + B + \bar{C})(A + \bar{B} + C)(\bar{A} + B + C) \text{ hay } G_2 = \prod M(3, 5, 6, 7)$$

– Biểu diễn hàm logic bằng phương pháp hình học (bìa Cacho).

Trong cách biểu diễn Cacho, hàm logic có n biến được biểu diễn thành một bảng gồm 2^n các ô vuông, mỗi ô vuông tương ứng với một m_i (hay M_j).

Các tổ hợp biến phải xếp theo thứ tự mã vòng Gray (bảng 1.2), nghĩa là 2 ô nằm kề nhau chỉ được phép có một biến khác trị số và các ô có nhiều hơn một biến khác trị số không được phép nằm cạnh nhau.

Bảng 1.12 đưa ra cấu trúc tổng quát của bìa Cacho cho hàm 2, 3 và 4 biến tương ứng với vị trí các ô (các mintec) xác định.

Với một hàm cụ thể (ký hiệu hàm ghi ở góc trên bên trái của bìa Cacho) ô nào ứng với mintec có giá trị 1 thì ghi 1 tại ô đó, ô nào ứng với $m_i = 0$ thì bỏ trống (hoặc ghi giá trị 0).

Bảng 1.12. Bìa Cacho của các hàm logic 2 biến (a), 3 biến (b) và 4 biến (c)

F A B		0	1
		$\overline{A}\overline{B} m_0$	$\overline{A}B m_2$
0		$\overline{A}\overline{B} m_0$	$\overline{A}B m_2$
1		$A\overline{B} m_1$	$AB m_3$

a)

G ₂ AB C		00	01	11	10
		$\overline{A}\overline{B}\overline{C} m_0$	$\overline{A}\overline{B}C m_2$	$\overline{A}B\overline{C} m_6$	$\overline{A}BC m_4$
0		$\overline{A}\overline{B}\overline{C} m_0$	$\overline{A}\overline{B}C m_2$	$\overline{A}B\overline{C} m_6$	$\overline{A}BC m_4$
1		$A\overline{B}\overline{C} m_1$	$A\overline{B}C m_3$	$AB\overline{C} m_7$	$ABC m_5$

b)

G ₁ AB CD		00	01	11	10
		$\overline{A}\overline{B}\overline{C}\overline{D} m_0$	$\overline{A}\overline{B}\overline{C}D m_4$	$\overline{A}\overline{B}C\overline{D} m_{12}$	$\overline{A}\overline{B}CD m_8$
00		$\overline{A}\overline{B}\overline{C}\overline{D} m_0$	$\overline{A}\overline{B}\overline{C}D m_4$	$\overline{A}\overline{B}C\overline{D} m_{12}$	$\overline{A}\overline{B}CD m_8$
01		$\overline{A}\overline{B}C\overline{D} m_1$	$\overline{A}\overline{B}CD m_5$	$\overline{A}B\overline{C}\overline{D} m_{13}$	$\overline{A}B\overline{C}D m_9$
00		$\overline{A}\overline{B}C\overline{D} m_1$	$\overline{A}\overline{B}CD m_5$	$\overline{A}B\overline{C}\overline{D} m_{13}$	$\overline{A}B\overline{C}D m_9$
01		$\overline{A}B\overline{C}\overline{D} m_3$	$\overline{A}B\overline{C}D m_7$	$AB\overline{C}\overline{D} m_{15}$	$AB\overline{C}D m_{11}$
01		$\overline{A}B\overline{C}\overline{D} m_3$	$\overline{A}B\overline{C}D m_7$	$AB\overline{C}\overline{D} m_{15}$	$AB\overline{C}D m_{11}$

c)

Ví dụ với hai hàm F_1 và G_2 có bảng trạng thái cho trên bảng 1.10, bìa Cacho tương ứng của chúng có dạng bảng 1.13.

Bảng 1.13

F ₁ A B		0	1
			1
0			1
1		1	1

a)

G ₂ AB C		00	01	11	10
		1	1		1
0		1	1		1
1		1			

b)

1.4.4. Rút gọn (tối thiểu) hàm logic

– Rút gọn hàm logic có mục đích làm cho biểu thức hàm ở dạng đơn giản và dễ thực hiện nhất. Muốn vậy, cần đạt được các yêu cầu:

- + Số lượng các số hạng (hay thừa số) là tối thiểu.
- + Số lượng các biến có mặt trong mỗi số hạng (hay thừa số) là ít nhất.
- + Số các phép toán logic tác động giữa các biến là ít nhất.

– Có nhiều cách để rút gọn hàm logic, hai trong số đó là dùng các tính chất của các hệ thức từ (1.8) đến (1.17) để thu gọn hàm và dùng quy tắc Cacno áp dụng cho bìa Cacno: "Nếu có 2^k các mintec (các ô) có trị 1 nằm liên kề nhau trên bìa Cacno thì có thể thay thế chúng chỉ bằng 1 ô lớn (1 nhóm) với số biến giảm đi k". Khi áp dụng quy tắc Cacno cần chú ý vài đặc điểm sau:

- + Các ô liên kề được tính cả ở các biên hàng và biên cột.
- + Số lượng các nhóm độc lập (không chứa nhau) là ít nhất có thể.
- + Số lượng các mintec trong một nhóm là nhiều nhất có thể.
- + Vài ô trị 1 có thể tham gia vào nhiều nhóm khác nhau.

Ví dụ 1: Xét hàm G_2 có bìa Cacno trên bảng 1.13. Dạng đầy đủ của G_2 là $G_2 = m_0 + m_1 + m_2 + m_4 = \sum m(0, 1, 2, 4)$ gồm 4 mintec đủ mặt 3 biến A, B, C. Trên bìa Cacno của G_2 có thể thực hiện được 3 nhóm, mỗi nhóm có 2 mintec:

$$X_1 = (m_0 + m_1); X_2 = (m_0 + m_2) \text{ và } X_3 = (m_0 + m_4)$$

Sau khi rút gọn G_2 có dạng (bảng 1.14):

$$G_2 = X_1 + X_2 + X_3 = \overline{A}B + \overline{A}C + \overline{B}C$$

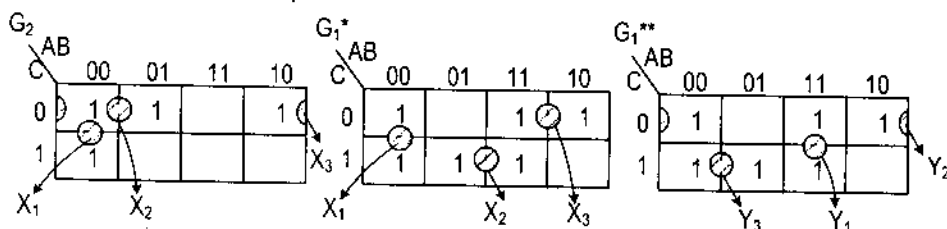
với mỗi số hạng X_i chỉ gồm hai biến:

$$X_1 = m_0 + m_1 = \overline{A}B\overline{C} + \overline{A}BC = \overline{A}B(C + \overline{C}) = \overline{A}B$$

$$X_2 = m_0 + m_2 = \overline{A}B\overline{C} + \overline{A}B\overline{C} = \overline{A}C(B + \overline{B}) = \overline{A}C$$

$$X_3 = m_0 + m_4 = \overline{A}B\overline{C} + A\overline{B}\overline{C} = \overline{B}C(A + \overline{A}) = \overline{B}C$$

Bảng 1.14



Ví dụ 2: Hãy rút gọn hàm G_1 có biểu thức $G_1 = \sum m(0, 1, 3, 4, 6, 7)$.
 Bìa Karnaugh của G_1 có dạng trên bảng 1.14 :

Và có hai khả năng rút gọn:

a) $G_1^* = X_1 + X_2 + X_3$, với:

$$X_1 = m_0 + m_1 = \overline{A}\overline{B}$$

$$X_2 = m_3 + m_7 = BC$$

$$X_3 = m_4 + m_6 = A\overline{C}$$

$$\Rightarrow G_1^* = \overline{A}\overline{B} + BC + A\overline{C}$$

b) $G_1^{**} = Y_1 + Y_2 + Y_3$, với:

$$Y_1 = m_6 + m_7 = AB$$

$$Y_2 = m_0 + m_4 = \overline{B}\overline{C}$$

$$Y_3 = m_1 + m_3 = \overline{A}C$$

$$\Rightarrow G_1^{**} = AB + \overline{B}\overline{C} + \overline{A}C$$

Hàm G_1 đầy đủ có hai dạng rút gọn tương đương nhau: $G_1^* = G_1^{**} = G_1$

1.4.5. Các hàm logic cơ bản

Hàm 1 biến gồm 4 hàm cơ bản:

$$G_2 = 0 \text{ (hằng số 0); } F_0 = \overline{A} \text{ (hàm NOT)}$$

$$G_1 = 1 \text{ (hằng số 1); } G_0 = A \text{ (hàm YES)}$$

Hàm 2 biến gồm 10 hàm cơ bản:

$$F_1 = A + B \text{ (hàm OR)}$$

$$F_2 = A.B \text{ (hàm AND)}$$

$$F_3 = \overline{A + B} \text{ (hàm NOR)}$$

$$F_4 = \overline{A.B} \text{ (hàm NAND)}$$

$$F_5 = A.\overline{B} \text{ (hàm cấm A – INHIBITION)}$$

$$F_6 = \overline{A}.B \text{ (hàm cấm B – INHIBITION)}$$

$$F_7 = \overline{A}.B + A.\overline{B} = A \oplus B \text{ (XOR – hàm cộng modul 2 nhị phân)}$$

$$F_8 = \overline{A}.\overline{B} + A.B = \overline{A \oplus B} \text{ (XNOR – hàm tương đương)}$$

$$F_9 = A + \overline{B} \text{ (IMPLICATION – hàm kéo theo A)}$$

$$F_{10} = \overline{A} + B \text{ (IMPLICATION – hàm kéo theo B)}$$

1.5. HỆ HÀM ĐẦY ĐỦ

Hệ hàm đầy đủ là một nhóm các hàm cơ bản mà từ chúng có thể tạo ra mọi hàm khác. Trên bảng 1.15 có 4 hệ hàm đầy đủ là:

- Hệ gồm các hàm F_0 (NOT) F_1 (OR) và F_2 (AND).
- Hệ gồm các hàm F_7 (XOR) và F_5 (INHIBITION) hoặc F_7 và F_6 .
- Hệ chỉ có hàm F_3 (NOR).
- Hệ chỉ có hàm F_4 (NAND).

Các tính chất của F_3 hoặc F_4 được sử dụng trong bài toán rút gọn hàm logic để cho việc thực hiện hàm bằng các cổng điện tử số được chuẩn hóa cao và đơn giản (số phép toán là ít nhất).

Bảng 1.15. Bảng chân lý các hàm cơ bản hai biến vào

Biến		Hàm cơ bản									
A	B	F_1	F_2	F_3	F_4	F_5	F_6	F_7	F_8	F_9	F_{10}
0	0	0	0	1	1	0	0	0	1	1	1
0	1	1	0	0	1	0	1	1	0	0	1
1	0	1	0	0	1	1	0	1	0	1	0
1	1	1	1	0	0	0	0	0	1	1	1

1.6. CÂU HỎI VÀ BÀI TẬP

- Phân biệt tín hiệu tương tự và tín hiệu số bằng cách nào?
- Cấu trúc cơ bản của các hệ đếm 10, hệ đếm 2, hệ đếm 8 và hệ đếm 16.
Hãy tự viết lại các ký tự được sử dụng trong các hệ đếm trên.
- Đổi các số thập phân sau sang số nhị phân và số nhị thập phân.
85, 73, 164, 200, 1980, 1945.
- Số thập phân 1 triệu cần biểu diễn bằng bao nhiêu bit nhị phân?
- Đổi các số sau từ hệ 2 sang hệ 10, sau đó sang hệ 2 – 10 (BCD8421)
101 101; 101 0010; 1001 1100; 0110 001110.

6. Hãy đổi các số sau từ hệ 16 sang hệ 10 và sang hệ BCD8421:

5F	7AB	8BDF	24AE
3C	4E6	2A9F	A6F2

7. Hãy đổi các số sau từ hệ 8 sang hệ 10 và hệ 2:

47	563	1246	2009
----	-----	------	------

8. Hãy nêu các quy tắc của phép cộng, phép trừ nhị phân và thực hiện các phép toán sau, sau đó chuyển sang hệ 10 và kiểm tra lại kết quả:

- | | |
|-------------------------------------|-------------------------|
| a) $1010 + 0111 + 1100$ | d) $110101 - 100110$ |
| b) $101010 + 01010101 + 110110$ | e) $10101110 - 1101101$ |
| c) $10010110 + 10100101 + 11000011$ | |

9. Thực hiện phép nhân và kiểm tra lại các kết quả trong hệ 10:

- a) 101101×101
- b) 110110×1010

10. Thực hiện phép chia sau và kiểm tra lại các kết quả trong hệ 10:

- a) $11100100 : 100$
- b) $10010110 : 101$
- c) $1100 : 111100$

11. Nêu các tính chất cơ bản của đại số logic.

12. Ba cách biểu diễn hàm logic và mối liên hệ giữa các cách biểu diễn này?

a) Cho hàm logic có dạng $F = \sum m(0, 1, 4, 5, 7)$. Hãy viết bảng chân lý và bìa Cacho của hàm, rút gọn hàm dùng các quy tắc và tính chất của hàm logic.

b) Cho hàm $F = \sum m(0, 1, 4, 5, 2, 8, 10, 11, 14, 15)$. Hãy viết bảng chân lý, bìa Cacho và tìm biểu thức thu gọn của hàm.

13. Hãy chứng minh tính chất đầy đủ của hàm $F_3 = \overline{A+B}$ (hàm NOR) và tính chất đầy đủ của hàm $F_4 = \overline{A.B}$ (hàm NAND) bằng cách xây dựng mọi hàm cơ bản còn lại từ hàm NOR hay từ hàm NAND.

14. Hãy biểu diễn hàm F trong bài tập 12 a và b bằng chỉ các hàm NOR và chỉ các hàm NAND.

BÀI 2. CÁC PHẦN TỬ LOGIC CƠ BẢN

Mạch điện tử số chỉ hoạt động với hai trạng thái khác nhau tại lối ra: hoặc có điện trở thấp ứng với trạng thái điện thế thấp tại lối ra hoặc có điện trở cao ứng với trạng thái điện thế cao tại lối ra. Tính chất này hoàn toàn phù hợp với trạng thái của các biến và hàm logic và cho phép ta thực hiện được các hàm logic, đặc biệt là các hàm cơ bản, nhờ các mạch điện cổng. Trong bài này, 6 cổng cơ bản quan trọng nhất được đề cập tới là NOT, OR, NOR, AND, NAND và XOR.

2.1. CÁC THÔNG SỐ QUAN TRỌNG CỦA MẠCH ĐIỆN CỔNG (phần tử logic cơ bản)

1. Công nghệ sử dụng để chế tạo mạch: TTL, CMOS hay một công nghệ đặc biệt nào khác (ECL, IIL, DTL, RTL, NMOS, PMOS...).
2. Trở kháng tại lối ra của mạch khi ở mức 1 (điện thế cao) và khi ở mức 0 (điện thế thấp).
3. Số lượng các cổng logic khác có thể nối đồng thời tới đầu ra của nó (xác định cho các cổng cùng loại) gọi là hệ số tải của cổng đang xét.
4. Mức nguồn một chiều dùng để cấp cho mạch hoạt động.
5. Công suất tiêu thụ lớn nhất xác định theo số tranzito, điện trở khi ở chế độ bão hòa (dòng lớn nhất).
6. Khả năng chống nhiễu xác định bởi biên độ điện áp lạ (nhiều) tối đa tác động tại lối vào mà không làm thay đổi trạng thái ra của nó trong điều kiện kém thuận lợi nhất.
7. Thời gian trễ cần thiết để chuyển (lật) trạng thái tại lối ra.
8. Tần số làm việc cao nhất (tại đó vẫn còn phân biệt được hai trạng thái logic).

2.2. CỔNG OR (công thực hiện hàm logic $F_1 = A + B$)

– Bảng trạng thái và ký hiệu quy ước của cổng OR cho trên hình 2.1a, b.

Đồ thị xung điện áp theo thời gian minh họa cho trên hình 2.1c.

Ta có nhận xét sau:

$F_1 = 1$ khi có ít nhất một đầu vào của nó có trị 1.

$F_1 = 0$ khi mọi đầu vào của nó có trị 0.

Các tính chất này có thể mở rộng cho một số lượng hữu hạn các biến đầu vào như 3, 4, 5 hoặc nhiều đầu vào hơn.

Biến vào		Hàm ra
A	B	$F_1 = A + B$
0	0	0
0	1	1
1	0	1
1	1	1

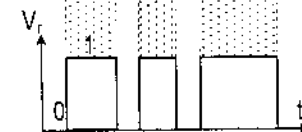
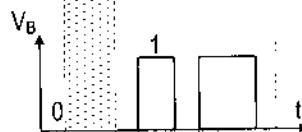
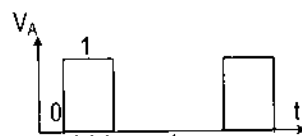
a)



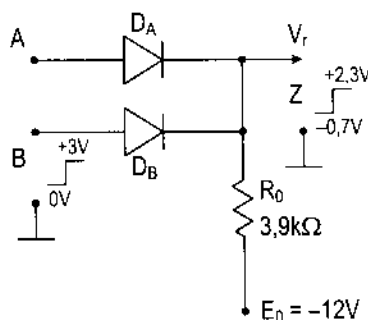
hoặc



b)



c)



d)

$V_A(V)$	$V_B(V)$	$V_F(V)$
0	0	-0,7
0	3	+2,3
3	0	+2,3
3	3	+2,3

e)

Hình 2.1a) Bảng chân lý; b) Ký hiệu quy ước;

c) Đồ thị minh họa hoạt động của cổng OR;

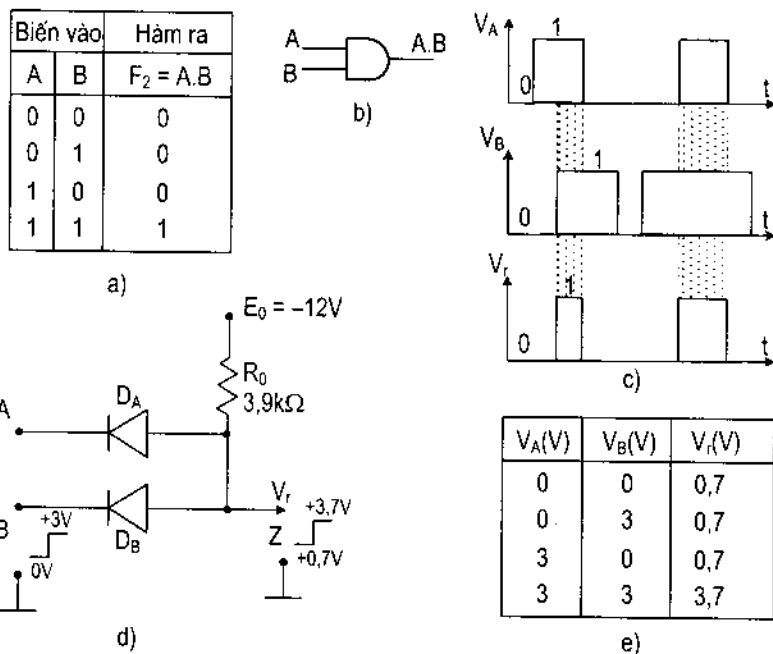
d) Mạch điện cổng OR dùng điốt; e) Bảng giá trị điện thế vào và ra

– Mạch điện cổng OR dùng điốt cho trên hình 2.1d và bảng kết quả các điện áp vào và ra cho trên hình 2.1e.

– Các cổng OR trên thực tế được chế tạo trong một IC SSI: 7432 (TTL), 4071 gồm bốn cổng mỗi cổng hai lối vào hoặc 4072 (CMOS) gồm hai cổng mỗi cổng có bốn lối vào.

2.3. CỔNG AND (cổng thực hiện hàm logic $F_2 = A.B$)

– Bảng trạng thái, ký hiệu quy ước và đồ thị xung điện áp minh họa của cổng AND cho trên hình 2.2.



Hình 2.2

Ta có nhận xét sau:

$F_2 = 1$ chỉ khi mọi đầu vào của nó có trị 1.

$F_2 = 0$ khi có ít nhất một đầu vào của nó có trị 0.

Các tính chất này cũng đúng cho một cổng AND có một số lượng hữu hạn (lớn hơn 2) đầu vào.

– Mạch điện cổng AND dùng các diốt và bảng giá trị các kết quả điện áp tại các lối vào và lối ra của mạch cho trên hình 2.2 d, e.

Theo bảng hình 2.2e có tất cả 4 tình huống khác nhau ở đầu vào. Khi $V_A = V_B = 3V$ cả hai diốt đều dẫn, qua nguồn $E_0 = 12V$ và $R_0 = 3,9k\Omega$, điện áp tại lối ra lúc này: $V_r = V_A + V_D = 3V + 0,7V = 3,7V$. Khi có ít nhất một lối vào ở mức thấp điện áp ra sẽ bị ghim ở mức $V_D = +0,7V$ và do đó ta nhận được bảng kết quả 2.2e.

– Ví dụ trên thực tế các cổng AND trong vi điện tử SSI 7408 (TTL) 4081 (CMOS) gồm 4 cổng AND hai lối vào hoặc 7411 (TTL) gồm 3 cổng AND ba lối vào, 4082 (CMOS) gồm 2 cổng AND mỗi cổng có bốn lối vào.

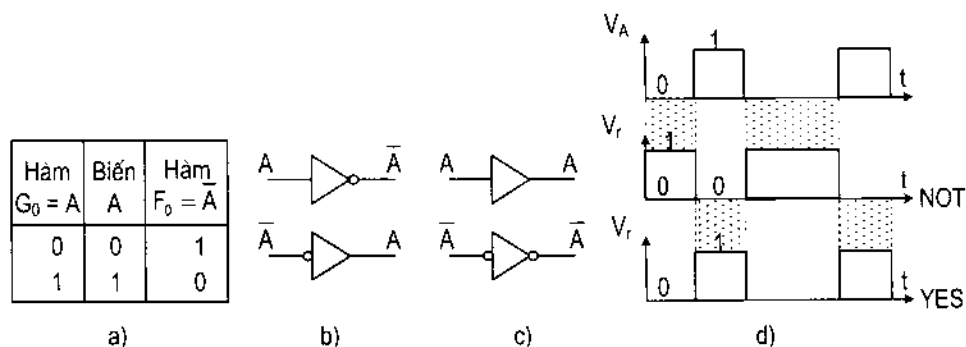
2.4. CỔNG NOT (cổng thực hiện hàm logic $F_0 = \bar{A}$) VÀ CỔNG YES (hàm $G_0 = A$)

– Bảng trạng thái, ký hiệu quy ước và đồ thị thời gian minh họa của cổng NOT và cổng YES được cho trên hình 2.3a, b, c và d.

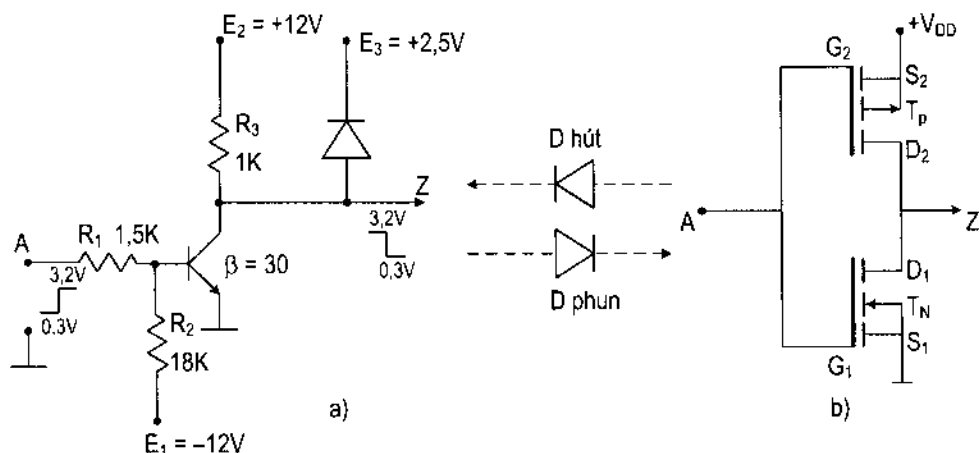
Ta có nhận xét là đối với cổng NOT trạng thái giá trị của hàm ra luôn ngược lại (đảo) với giá trị đầu vào:

Nếu đầu vào có mức 1 tác động thì điện thế đầu ra sẽ xuống mức 0 còn khi đầu vào có mức 0 tác động thì điện thế đầu ra sẽ lên mức 1.

– Mạch điện thực hiện cổng NOT được cho trên hình 2.4.



Hình 2.3a) Bảng trạng thái; b) Ký hiệu logic chuẩn và ký hiệu logic tương đương của cổng NOT; c) Cổng YES; d) Giải đồ thời gian xung điện áp minh họa của cổng NOT và YES



Hình 2.4. Các dạng cổng NOT:
a) Điốt – Tranzito – logic (DTL); b) CMOS

+ Trong mạch hình 2.4a, tranzito làm việc ở chế độ khóa. Nếu V_A ở mức thấp thì T ngắt mạch V_z ở mức cao ($z = 1$). Khi lối vào chuyển lên mức cao, cỡ 3,2V ($A = 1$) thì T nối mạch ở chế độ bão hòa, dòng qua R_3 đủ lớn để điện thế lối ra V_z về mức thấp ($z = 0$).

Nguồn hỗ trợ $E_1 = -12V$ để đảm bảo $T_{ngắt}$ tin cậy khi $A = 0$.

Nguồn $E_2 = +12V$; $E_3 = +2,5V$ có tác dụng giữ cho mức ra ở cao có giá trị mong muốn bằng $E_3 + V_D = 3,2V$.

Tại cổng ra nếu bổ sung thêm 2 điốt (đường nét đứt) sẽ đảm bảo ở hai chế độ:

Nếu $z = 0$, Đ hút hoạt động, dòng từ tải qua Đ hút vào tranzito nối mạch (tương ứng với ký hiệu hình 2.3b).

Còn khi $z = 1$, Đ phun hoạt động, dòng từ E_2 qua R_3 phun qua Đ ra tải (tranzito đang ngắt), chế độ này tương ứng với ký hiệu hình 2.3c.

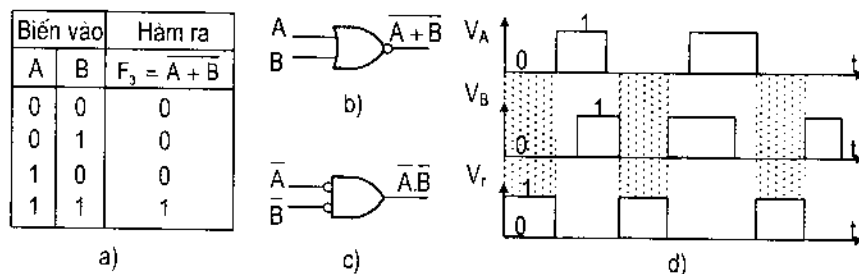
Mạch hình 2.4b dùng cặp tranzito CMOS làm việc kiểu Push – Full (đẩy – kéo) ở chế độ khóa. T_p ở đây đóng vai trò là điện trở tải động R_D của T_N . Khi $A = 0$, T_N ngắt, do đó, $V_{GS2} \approx |-V_{DD}| > |V_{p2}|$ nên T_p nối mạch $z = 1$ còn khi $A = 1$ (mức xung dương $V_A > V_{p1}$).

T_N nối mạch làm $V_{GS2} = 0$. T_p ngắt mức ra ở thấp ($z = 0$). Ở đây, $|V_{p2}|$ và V_{p1} là các điện áp ngưỡng mở của T_p và T_N tương ứng.

– Thực tế sáu cổng NOT độc lập được chế tạo trong một chip IC cổng SSI: 7404 (TTL), 74HC04 (CMOS tốc độ cao) hay 4049, 4069 (CMOS), sáu cổng YES trong một chip IC 4050 (CMOS).

2.5. CỔNG NOR (cổng thực hiện hàm logic $F_3 = \overline{A+B}$)

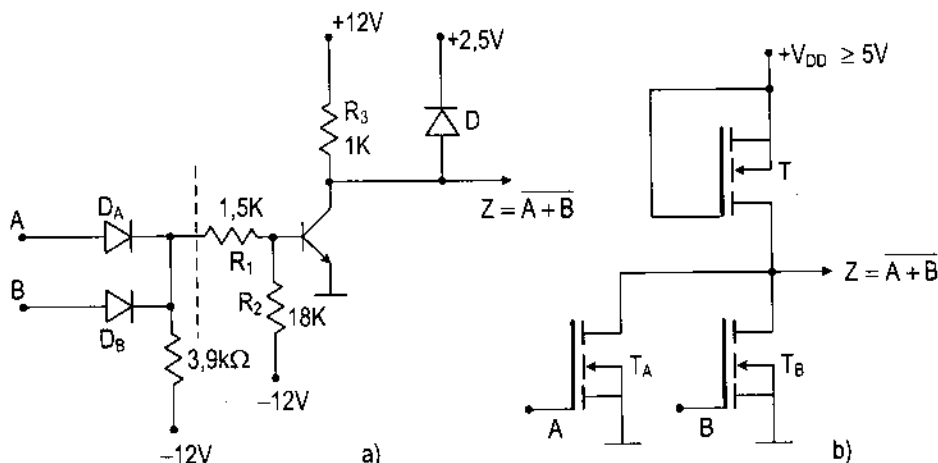
– Bảng trạng thái, ký hiệu quy ước và đồ thị thời gian minh họa hoạt động của cổng NOR cho trên hình 2.5.



Hình 2.5a) Bảng trạng thái; b) Ký hiệu logic chuẩn; c) Ký hiệu logic tương đương; d) Đồ thị thời gian minh họa hoạt động của cổng NOR

Ta có nhận xét là $F_3 = 1$ chỉ khi các lối vào đồng thời có giá trị 0. $F_3 = 0$ khi có ít nhất một lối vào có giá trị 1. Điều này có thể mở rộng cho cổng NOR có nhiều đầu vào hơn.

– Mạch điện cổng NOR có thể xây dựng từ việc kết hợp liên tiếp 1 cổng OR và 1 cổng NOT như cho trên hình 2.6a theo công nghệ DTL hay trên hình 2.6b theo công nghệ NMOS.

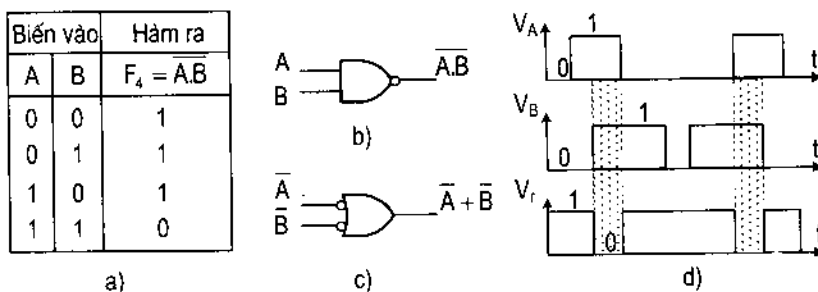


Hình 2.6a) Mạch điện cổng NOR kết hợp OR + NOT;
b) Mạch điện cổng NOR dùng tranzito NMOS

– Thực tế 4 cổng NOR hai đầu vào được làm trong một IC SSI ví dụ 7425 (TTL) hay 4001 (CMOS)...

2.6. CỔNG NAND (cổng thực hiện hàm logic $F_4 = \overline{A.B}$)

– Bảng trạng thái, ký hiệu quy ước và đồ thị thời gian minh họa hoạt động của cổng NAND được cho trên hình 2.7.



Hình 2.7a) Bảng chân lý; b) Ký hiệu logic chuẩn; c) Ký hiệu logic tương đương; d) Đồ thị thời gian minh họa hoạt động của cổng NAND

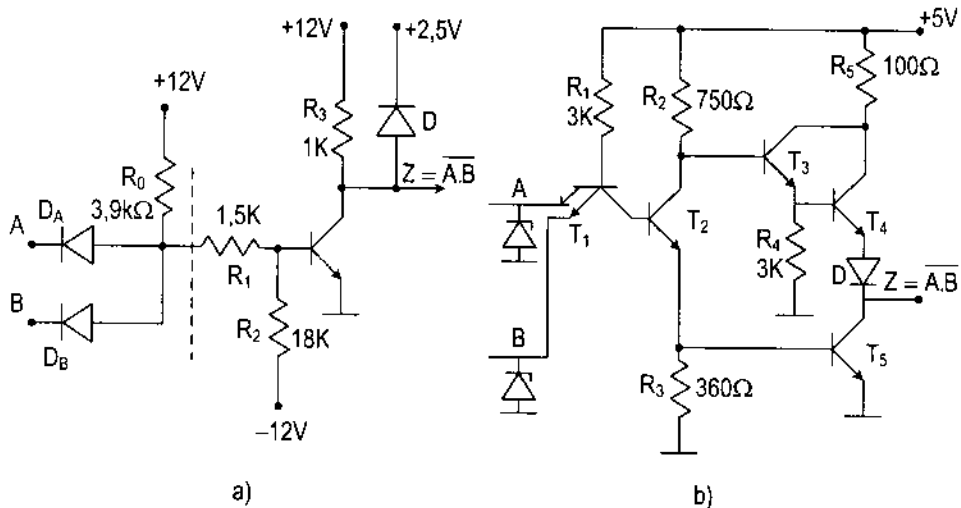
Ta có nhận xét:

$F_4 = 0$ chỉ khi mọi đầu vào của nó có giá trị 1 (có thể mở rộng cho cổng có nhiều hơn hai đầu vào).

$F_4 = 1$ khi có ít nhất một đầu vào có giá trị 0.

Có thể thực hiện cổng NAND nhờ việc ghép nối tiếp một cổng AND với một cổng NOT (hình 2.8a) hoặc theo công nghệ TTL (hình 2.8b).

– Mạch điện hình 2.8b hoạt động tuân theo bảng trạng thái hình 2.7a T_1 và R_1 có chức năng một cổng AND với hai đầu vào là các emitor.



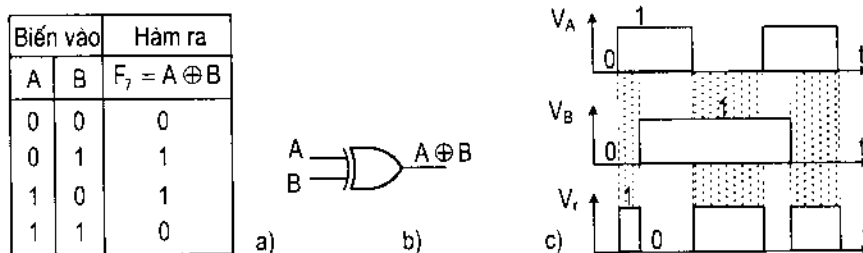
**Hình 2.8a) Mạch cổng NAND ghép nối tiếp 2 cổng AND và NOT;
b) Mạch cổng NAND công nghệ TTL**

Khi các lối vào đều ở mức cao ($A = B = 1$) T_1 ngắt mạch làm điện thế Colectơ của T_1 ở mức cao làm T_2 nối mạch cặp T_3, T_4 ngắt, T_5 nối và lối ra ở mức thấp ($z = 0$). Còn khi có ít nhất một lối vào xuống mức thấp ($A.B = 0$) T_1 nối mạch đưa T_2 về trạng thái ngắt, điện áp trên R_3 bằng 0, T_5 ngắt, điện thế bazơ của T_3 ở mức cao (dòng qua R_2 thấp vì T_2 ngắt) làm T_3 và T_4 nối mạch: lối ra ở mức cao ($z = 1$).

– Các cổng NAND thực tế được chế tạo trong một IC loại SSI (cỡ nhỏ), ví dụ loại 4 cổng, mỗi cổng có 2 đầu vào 7400 (TTL) hay 4011 (CMOS) hoặc 1 cổng 8 lối vào 7430 (TTL) 2 cổng, mỗi cổng có 4 lối vào 4012 (CMOS)...

2.7. CỔNG XOR (công thức hiện hàm logic $F_7 = \overline{A}B + A\overline{B} = A \oplus B$)

– Bảng chân lý, ký hiệu quy ước và đồ thị thời gian của cổng XOR được cho trên hình 2.9.



Hình 2.9a) Bảng chân lý; b) Ký hiệu quy ước;
c) Đồ thị điện thế minh họa chức năng của cổng XOR

Ta có nhận xét là $F_7 = 1$ chỉ khi trị hai biến vào khác nhau.

$F = 0$ khi hai biến vào có cùng trị số.

Cổng XOR thực hiện chức năng của phép cộng modun trong hệ 2, thực hiện cộng không nhớ theo quy tắc:

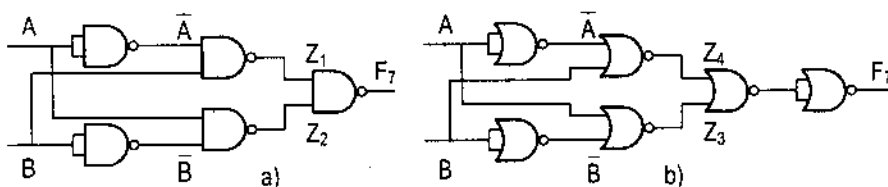
$$0 + 0 = 0; 1 + 1 = 0; 1 + 0 = 0 + 1 = 1$$

– Có thể thực hiện được cổng XOR từ các cổng NAND hoặc từ các cổng NOR bằng cách viết lại biểu thức logic của hàm F_7 dưới dạng

$$\text{khác như sau: } F_7 = \overline{\overline{A}B + A\overline{B}} = \overline{\overline{A}B} \cdot \overline{A\overline{B}} = \overline{z_1} \cdot \overline{z_2} \quad (2.1)$$

$$\text{hoặc: } F_7 = \overline{\overline{A}B + A\overline{B}} = \overline{\overline{A}B} \cdot \overline{A\overline{B}} = \overline{\overline{A}B} \cdot \overline{A\overline{B}} = \overline{z_3} \cdot \overline{z_4} = z_3 + z_4 \quad (2.2)$$

Từ hệ thức (2.1) F_7 được tạo ra từ 5 cổng NAND ghép liên tiếp và từ hệ thức (2.2) từ 6 cổng NOR ghép liên tiếp như trên hình 2.10.

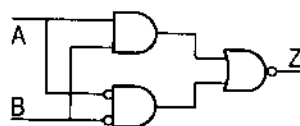


Hình 2.10

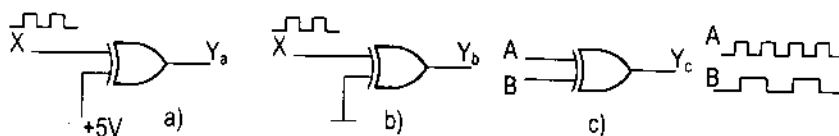
Các cổng XOR trên thực tế được làm trong một IC SSI: 7486 (TTL) hoặc 4030, 4070 (CMOS) gồm 4 cổng XOR 2 lối vào, loại có tốc độ cao 74HC7276 (CMOS)...

2.8. CÂU HỎI VÀ BÀI TẬP

- Hãy nêu các chỉ tiêu và tham số kỹ thuật cơ bản đối với một cổng logic.
Ý nghĩa và lý do cần phải quan tâm tới các tham số trên là gì?
- Hãy tự viết bảng chân lý của các cổng logic cơ bản cùng các ký hiệu của chúng (các cổng thực hiện hàm F_0 đến hàm F_{10}).
- Vẽ đồ thị điện thế theo thời gian trên đồ thị hàng dọc minh họa bảng chân lý đã viết cho từng hàm.
- Hãy viết bảng chân lý cho hàm AND, OR, NAND, NOR có 3 biến vào và có 4 biến vào, vẽ ký hiệu quy ước chuẩn và logic tương đương của chúng.
- Hãy chứng minh các tính chất sau của hàm XOR:
 - Luật hoán vị $A \oplus B = B \oplus A$.
 - Luật kết hợp $A \oplus B \oplus C = A \oplus (B \oplus C)$.
 - Luật phân bố $A(B \oplus C) = AB \oplus AC$.
 - Luật đối chỗ nhân quả:
Nếu $A \oplus B = C$ thì $A \oplus C = B$ và $B \oplus C = A$.
- Hãy viết biểu thức và bảng chân lý của hàm $F_8 = \overline{F_7}$ (XNOR). Vẽ đồ thị thời gian hàng dọc minh họa.
Xây dựng cấu trúc logic thực hiện hàm F_8 chỉ từ các cổng NAND và chỉ từ các cổng NOR.
- Hãy chứng minh $\overline{(A \oplus B)(C \oplus D)} = \overline{AB} + \overline{AB} + \overline{CD} + CD$.
Vẽ sơ đồ logic của hàm $X = A \oplus B \oplus C \oplus D$.
- Tìm hàm ra của cấu trúc hình 2.11, hãy đưa Z về dạng rút gọn và nhận xét.
- Hãy xác định dạng sóng Y đầu ra trong các mạch cổng XOR hình 2.12.



Hình 2.11



Hình 2.12

BÀI 3. PHẦN TỬ LOGIC THÔNG DỤNG

3.1. MẠCH SO SÁNH

Mạch so sánh có nhiệm vụ tìm ra các kết quả của phép so sánh hai số nhị phân là lớn hơn, nhỏ hơn hay bằng nhau.

Khi thực hiện thuật toán so sánh cần theo quy tắc:

1. Chỉ so sánh các số trong cùng một hệ đếm và chúng có cùng cấp cao nhất trong hệ đếm đó.

2. Thực hiện so sánh lần lượt bắt đầu từ cấp cao nhất.

a) Nếu kết quả ở nhịp này là bằng nhau, phải thực hiện so sánh tiếp tục ở cấp thấp hơn liền kề.

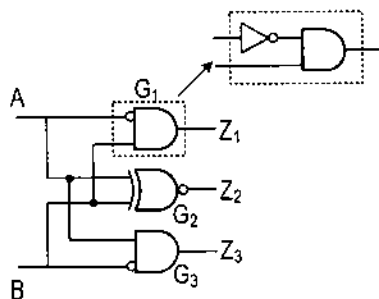
b) Nếu kết quả ở nhịp này là hơn hay kém thì phép so sánh kết thúc, kết quả riêng của nhịp này cũng là kết quả chung của phép so sánh hai số ban đầu.

– Theo quy tắc trên, bài toán so sánh hai số nhị phân cùng cấp quy về bài toán so sánh 2 bit nhị phân cùng vị trí trọng số.

Hình 3.1 đưa ra bảng chân lý gồm 2 đầu vào và 3 đầu ra kết quả cho bộ so sánh 2 bit A_i với B_i ở cấp 2^i bất kỳ.

Bảng 3.1. Bảng chân lý bộ so sánh 2 bit

Biến vào		Hàm ra		
A_i	B_i	$Z_3(>)$	$Z_2(=)$	$Z_1(<)$
0	0	0	1	0
0	1	0	0	1
1	0	1	0	0
1	1	0	1	0



Hình 3.1. Cấu trúc mạch logic bộ so sánh 2 bit

Từ bảng chân lý này, các hàm ra có phương trình logic như sau:

$$\left. \begin{aligned} z_1 &= \overline{AB} \\ z_2 &= A \oplus B = \overline{AB} + AB \\ z_3 &= A\overline{B} \end{aligned} \right\} \quad (3.1)$$

Đối chiếu với bảng 1.14 cho các hàm logic cơ bản có $z_1 = F_6$, $z_2 = F_8$ và $z_3 = F_5$.
Chú ý ký hiệu các cổng cấm G_1 , G_2 và cổng hàm tương đương G_2 .

– Khi cần so sánh hai số nhị phân n bit:

$$A_{(n)} = A_{n-1} \dots A_0$$

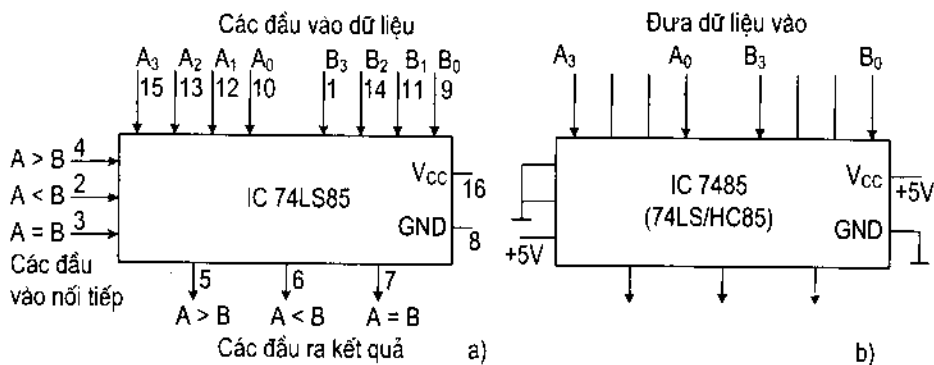
$$B_{(n)} = B_{n-1} \dots B_0$$

Theo quy tắc trên kết quả bằng nhau ($A = B$) chỉ có được sau khi đã lần lượt so sánh các bit từ cấp 2^{n-1} đến cấp 2^0 . Còn kết quả hơn hay kém có được (xuất hiện lần đầu tiên) sau k nhịp với $1 \leq k \leq n$ tùy từng trường hợp. Đây là phương pháp so sánh nối tiếp và nhìn chung có tốc độ chậm, đặc biệt với kết quả bằng nhau.

Để tăng tốc độ làm việc của phép so sánh, có thể dùng phương pháp kết hợp so sánh song song với nối tiếp bằng cách chia thành từng nhóm bit: nhóm bit cao được so sánh đồng thời ở một nhịp sau đó tiếp tục so sánh đồng thời nhóm bit thấp (nếu nhịp trước có kết quả bằng nhau).

– Nhóm vi điện tử IC MSI 7485/74LS85/74HC85 được chế tạo để so sánh đồng thời (song song) 4 bit nhị phân không dấu.

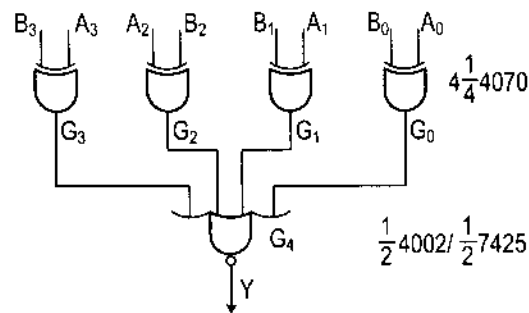
Hình 3.2a đưa ra cấu trúc bố trí chân và các đầu vào, đầu ra chức năng của IC và hình 3.2b cho phép thực hiện so sánh song song hai số nhị phân 4 bit $A = A_3A_2A_1A_0$ với $B = B_3B_2B_1B_0$ khi đưa các dữ liệu tới các đầu vào tương ứng (các số thập phân là chỉ số chân của các lối vào, ra).



Hình 3.2a) Bố trí chân của IC 74LS85;
b) Cách nối IC thực hiện việc so sánh 4 bit song song

Với phép so sánh hai số nhị phân có số bit nhiều hơn, cần thực hiện ghép nối tiếp các nhóm đầu ra của IC cấp thấp tới đầu vào nối tầng của IC ở nhóm so sánh cấp cao. Việc so sánh được thực hiện theo cả hai phương thức vừa song song (trong một nhóm 4 bit) vừa nối tiếp (giữa nhóm cấp cao và nhóm cấp thấp).

– Mạch logic thực hiện phép so sánh song song 4 bit với 2 kết quả khác nhau hay bằng được cho trên hình 3.3 có cấu tạo gồm 4 cổng XOR: G_0, G_1, G_2, G_3 tiếp nhận từng cặp bit cùng cấp cần so sánh tương ứng. Tiếp nhận kết quả tại lối ra của chúng là một cổng NOR. G_4 có 4 lối vào. Trạng thái ra của G_4 ở mức thấp ($Y = 0$) thể hiện kết quả $A = A_3A_2A_1A_0 \neq B = B_3B_2B_1B_0$. Còn khi $Y = 1$ thì kết quả là $A = A_3A_2A_1A_0 = B = B_3B_2B_1B_0$ hay $A = B$ (ví dụ, nếu $A = 1001$; $B = 1011$ thì $Y = 0$).



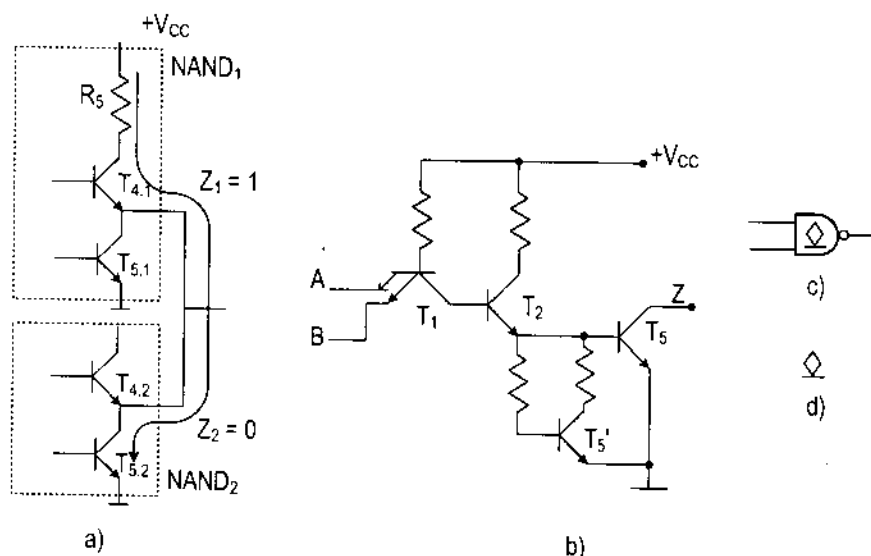
Hình 3.3. Mạch logic so sánh hai trạng thái bằng nhau hay khác nhau

Nếu A là một dữ liệu mẫu (đã chuẩn) có độ tin cậy cao và B là dữ liệu cần xác định xem có bị lỗi (khác mẫu) hay không sau khi thực hiện truyền B từ nơi này đến nơi kia. Cấu trúc dạng hình 3.3 có thể được mở rộng ra cho nhiều nhóm bit hơn (ví dụ $A_{15} \rightarrow A_0$ và $B_{15} \rightarrow B_0$) để phát hiện lỗi nếu có của B. Khi $Y = 1$ thì B không có lỗi bit còn khi $Y = 0$ thì trong B đã có bit bị lỗi (so với nguyên bản gốc A).

3.2. MẠCH CỔNG COLECTƠ ĐỂ HỞ (OPEN COLLECTOR GATES – OC)

– Với các mạch điện cổng NAND loại TTL (hình 2.8b), dù tín hiệu tại đầu ra tích cực ở mức cao hay tích cực ở mức thấp thì điện trở đầu ra của mạch đều nhỏ (vài trăm Ω đến vài $k\Omega$) dẫn tới hệ quả không thể đấu hai cổng NAND song song với nhau như trên hình 3.5a để

thực hiện hiệu ứng hàm AND được. Khi lối ra của $NAND_1$ có mức $Z_1 = 1$, của $NAND_2$ có $Z_2 = 0$ sẽ xuất hiện một dòng điện lớn chảy từ nguồn $+V_{cc}$ qua $T_{4,1}$ đến $T_{5,2}$ làm mức thấp Z_2 của $NAND_2$ tăng lên và hỏng mức cao của $NAND_1$, đây là hiệu ứng xung đột cao thấp giữa hai lối ra khác trị.



Hình 3.4a) Mạch ra của 2 cổng NAND TTL nối trực tiếp với nhau khi $z_1 = 1$ và $z_2 = 0$, hiện tượng xung đột cao – thấp;
b) Cổng NAND để hở collector (dạng thu gọn);
c) Tín hiệu quy ước của OC – NAND;
d) Mã ký hiệu dùng chung cho các cổng có collector để hở

– Các mạch cổng NAND hở collector (OC) có collector (đầu ra) của T_5 để hở lơ lửng (hình 3.4b) cho phép giải quyết được vấn đề trên, có khả năng đấu song song (nối chung đầu ra) n cổng loại này với m tải là các đầu vào của cổng khác. Điện trở tải khi đó được lựa chọn từ ngoài tùy theo các hệ số n và m ở trên và nằm trong dải:

$$R_{t\grave{a}i\min} \leq R_{t\grave{a}i} \leq R_{t\grave{a}i\max}$$

với:

$$R_{t\grave{a}i\max} = \frac{V_{cc} - V_{OH\min}}{n.I_{OH} + m.I_{IH}} \quad (3.2)$$

Trong đó: $V_{OH\min}$: mức ra cao ngưỡng của cổng OC;

$n.I_{OH}$: tổng dòng rò của T_5 khi ngắt;

$m.I_{IH}$: tổng dòng vào của các tải.

$$R_{t\grave{a}i\min} = \frac{V_{cc} - V_{OL\max}}{I_{tM} - m.I_{IL}} \quad (3.3)$$

Trong đó: $V_{OL\max}$: mức ra thấp ngưỡng của cổng OC;

I_{tM} dòng cực đại cho phép của tải;

I_{IL} dòng ngắn mạch lối vào mỗi phụ tải.

– Các cổng OC được chế tạo trong IC gồm nhiều cổng độc lập chung một vỏ và chung nguồn nuôi. Ví dụ: 7409/74LS09 có 4 AND 2 lối vào loại OC;

7406/74F06 có 6 cổng NOT loại OC làm việc với điện áp cao;

5401/7401; 5403/7403 có 4 cổng NAND loại OC hai lối vào;

54LS136/74LS136 có 4 cổng XOR loại OC;

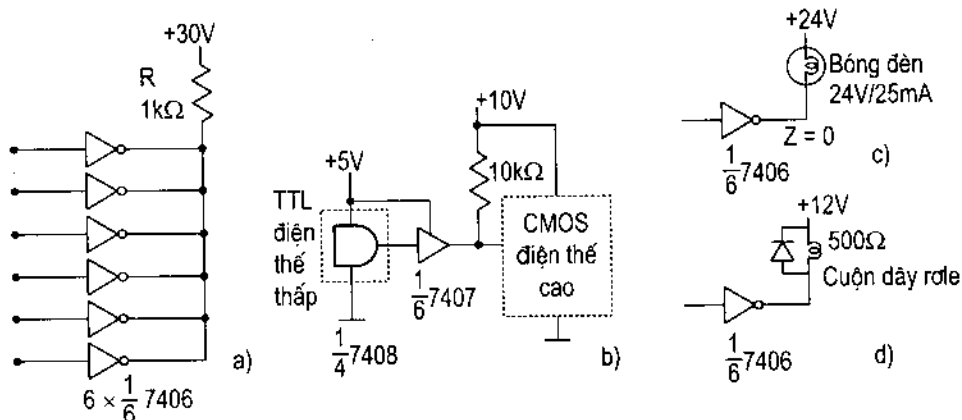
54LS266/75LS266 có 4 cổng XNOR loại OC;

54LS22/74LS22 có 2 cổng NAND OC với 4 lối vào mỗi cổng;

5417/7417; 5416/7416 có 6 cổng NOT OC điện áp cao;

5407/7407: có 6 cổng YES điện áp cao.

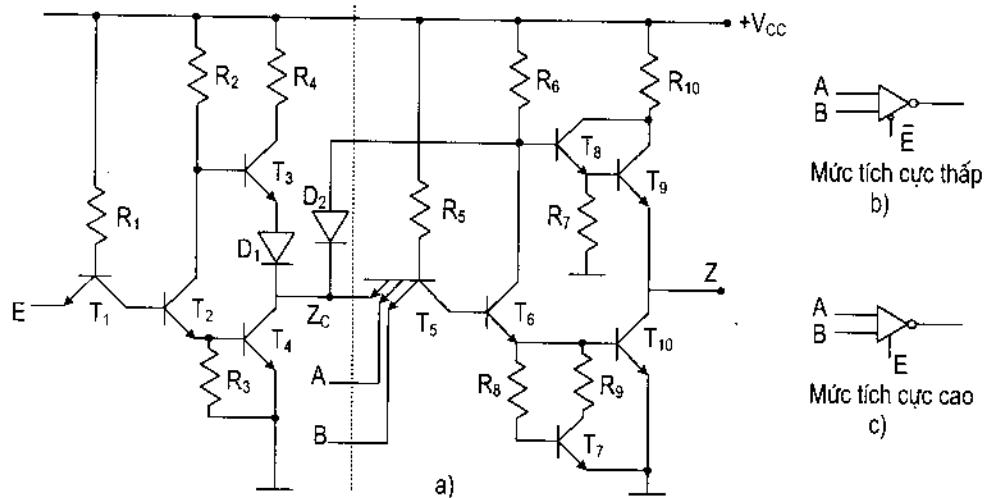
Nhóm IC 7406, 7407, 7417 loại OC, có thể làm việc với điện áp ra cao (tới 30V) là một điều kiện thuận lợi cho việc ghép tải có mức nguồn cao. Ví dụ trên mạch hình 3.5a, 6 cổng NOT của IC7406 được nối chung đầu ra tới nguồn 30V qua điện trở tải $1k\Omega$ chọn ngoài để hạn chế dòng ở mức 30mA khi áp ra ở mức logic thấp (cỡ 0,4V) thuận lợi cho việc đệm ghép nối tiếp giữa các cổng TTL (mức nguồn thấp) với cổng CMOS (mức nguồn cao) như thể hiện trên hình 3.5b. Các mạch hình 3.5c và d cho phép ghép trực tiếp tải là loại bóng đèn 24V/25mA hoặc cuộn dây rơle 12V vào mạch ra của cổng NOT – OC.



Hình 3.5. Một số ứng dụng của cổng hở colectơ

3.3. MẠCH CỔNG LOGIC BA TRẠNG THÁI (TRI - STATE - GATES - TS)

– Mạch điện nguyên lý của một cổng NAND – TS được cho trên hình 3.6a và ký hiệu quy ước trên hình 3.6b, c cho hai loại mức tích cực của lối vào điều khiển E.



Hình 3.6a) Mạch điện nguyên lý của cổng NAND – TS;
b) Ký hiệu quy ước cho loại dùng $\bar{E}$ (mức tích cực thấp);
c) Ký hiệu quy ước cho loại dùng E (mức tích cực cao)

Đầu ra z của loại cổng TS có 3 trạng thái khác nhau: hai trạng thái bình thường như các cổng logic khác là mức cao $z = 1$ và mức thấp $z = 0$. Trạng thái thứ 3 là lối ra có trở kháng cao (Hi – z).

Mạch điện gồm 2 khối lớn: bên phải đường nét đứt là một cổng NAND₂ có 2 lối vào A và B (dạng mạch TTL hình 2.8b). Ở đây, với các cổng NAND thế hệ sau có cải tiến nhờ bổ sung tranzito T₇ làm nhanh quá trình phóng điện cho T₁₀ lúc chuyển từ nổi sang ngắt và kích cho T₁₀ mở nhanh lúc chuyển từ ngắt sang nổi khi T₆ chuyển sang nổi mạch. Phần bên phải đường nét đứt là một cổng NOT TTL (hay có thể gọi là cổng NAND₁ TTL có một cửa vào E), ở đây có bổ sung D₁ để đảm bảo việc T₃ ngắt tin cậy lúc T₂ và T₄ chuyển sang nổi mạch và do yêu cầu dòng điện không lớn nên T₃ không mắc kiểu Darlington.

– Mạch hình 3.6a hoạt động như sau:

+ Khi lối vào E ở mức thấp ($E = 0$) tại lối ra của nó có mức cao $z_c = \overline{E} = 1$ đây là mức cho phép cổng NAND₂ bên phải được phép hoạt động để tiếp đón các dữ liệu A, B của nó, nghĩa là tại lối ra có $z = \overline{A.B}$, chú ý rằng khi $z_c = 1$ D₂ ngắt mạch.

+ Khi E ở mức cao ($E = 1$), $z_c = \overline{E} = 0$ là tín hiệu điều khiển làm T₅ nổi điện thế colectơ của T₅ xuống thấp làm T₆, T₇ và T₁₀ ngắt do $z_c = 0$ D₂ dẫn làm mức bazơ của T₈ xuống thấp và bị ghim ở ngưỡng xấp xỉ 1V; T₈ và T₉ cũng ngắt.

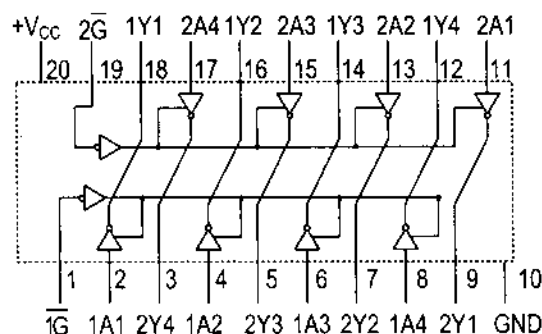
Từ lối ra nhìn vào, mạch ở trạng thái trở kháng cao và không thể có một liên hệ nào với các đường truyền tín hiệu hay với tải. Kết quả chung ta có: $z = \overline{A.B.E}$.

Mạch vừa phân tích ứng với ký hiệu quy ước ở hình 3.6b. Ngoài ra, có loại điều khiển theo mức $E = 1$ khi đó hàm ra có dạng $z = \overline{A.B.E}$.

– Các IC cổng 3 trạng thái cho phép đấu song song nhiều đầu ra với nhau mà không làm giảm tốc độ hoạt động chung, khi đó, yêu cầu chỉ có một cổng được phép hoạt động theo sự điều khiển của đầu vào E, các cổng còn lại ở trạng thái trở kháng cao và không tham gia vào hoạt động đang xảy ra.

– Các cổng logic khác, đặc biệt là nhóm cổng NOT và YES tương tự như cổng NAND – TS đã phân tích có thể được cấu tạo để hoạt động ở chế độ ba trạng thái dùng làm các cổng đệm vào/ra cho nhóm dữ liệu 4 bit hay 8 bit (loại 74LS240 hay 74LS245).

Hình 3.7 đưa ra cấu hình logic của IC 74LS240 gồm 8 cổng NOT – TS dùng làm bộ đệm với vị trí bố trí các chân với 8 dữ liệu vào $1A_1 \rightarrow 1A_4$ và $2A_1 \rightarrow 2A_4$ và 8 lối ra tương ứng.



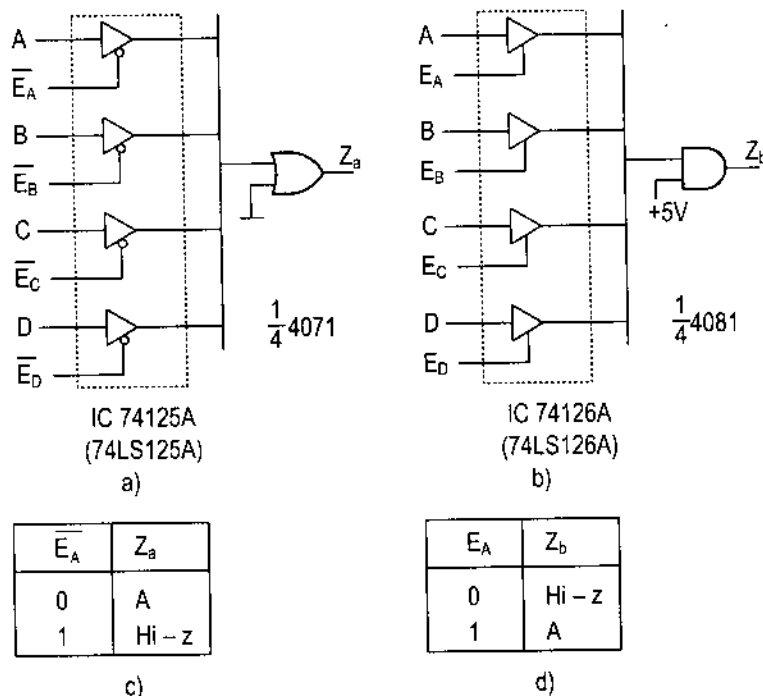
Hình 3.7. Cấu hình logic của IC 74LS240 8 cổng NOT – TS

Cấu hình loại 74LS245 gồm 2 nhóm; mỗi nhóm 8 cổng YES – TS (thực hiện hàm lặp có lối ra ba trạng thái) để kiểm soát hướng dữ liệu vào/ra 8 bit song song ứng với các cặp chân: 2 – 18; 3 – 17; 4 – 16; 5 – 15; 6 – 14; 7 – 13; 8 – 12 và 9 – 11. Việc điều khiển chọn hướng nhờ mã điều khiển 2 bit đưa vào các chân 1 ($S/\overline{R}$) và 19 ($\overline{CE}$).

3.4. CÂU HỎI VÀ BÀI TẬP

1. Các quy tắc thực hiện việc so sánh hai số (nhị phân).
2. Hãy tự viết bảng trạng thái của một bộ so sánh 2 bit nhị phân cùng cấp và vẽ lại mạch điện sau khi đã viết các phương trình logic các hàm ra.
3. Hãy sử dụng IC 74LS85 để thực hiện việc so sánh 2 số nhị phân viết trong hệ 16:
 $z_1 = 9D_{16}$ và $z_2 = B6_{16}$
 - a) Chuyển z_1 và z_2 từ hệ 16 sang hệ 2.
 - b) Vẽ mạch thực hiện gồm 2 IC 74LS85 mắc nối tiếp nhau.
 - c) Điền số chân thích hợp vào mạch điện logic đã vẽ, nối nguồn $+V_{cc}$ và đất.
 - d) Gán các dữ liệu vào z_1 và z_2 (trong hệ 2) và xác định kết quả tại các lối ra (chân số 5, 6, 7 của nhóm bit cao).
4. Cho dữ liệu mẫu $z_0 = 5A3F_{16}$ và một dữ liệu nhận được trên kênh truyền $z_x = 0101\ 1010\ 0011\ 0111$.
 - a) Hãy vẽ mạch so sánh kiểm tra lỗi của z_x .
 - b) Điền dữ liệu z_0 và z_x tương ứng vào mạch đã vẽ, xác định giá trị hàm ra Y và kết luận.
 Nếu $z_x \neq z_0$ hãy chỉ ra bit bị lỗi trong z_x .
5. Bản chất của hiện tượng xung đột cao thấp là gì?
 Khi nào thì xảy ra hiện tượng trên và hậu quả là gì?
 Tại sao các cổng logic có OC khắc phục được tình trạng trên?
6. Hãy nêu các ứng dụng quan trọng của cổng NOT OC và cổng đệm YES OC trong việc phối hợp ghép với tải có sử dụng các mức điện áp cao.

7. Vấn đề gặp phải khi cổng kích thích là loại TTL và cổng tải là loại CMOS là gì? Biện pháp khắc phục.
8. Hãy nêu các trạng thái ra của một cổng logic 3 trạng thái trong các trường hợp sau và lập bảng chân lý tương ứng:
 - a) Đối với cổng NAND có 3 lối vào dữ liệu điều khiển bằng mức 0.
 - b) Đối với cổng NOR có 2 lối vào điều khiển bằng mức 0.
 - c) Đối với cổng NOT điều khiển bằng mức 1 hoặc bằng mức 0.
 - d) Đối với cổng XOR điều khiển bằng mức 0.
9. Cho mạch điện 3 trạng thái hình 3.8a và b dùng các IC cổng tương ứng 74LS125A và 74LS126A có bảng trạng thái cho trên hình 3.8c và d. Hãy xác định trạng thái hàm logic ở lối ra Z_a và Z_b ứng với các trường hợp sau:



Hình 3.8. Bài tập 9

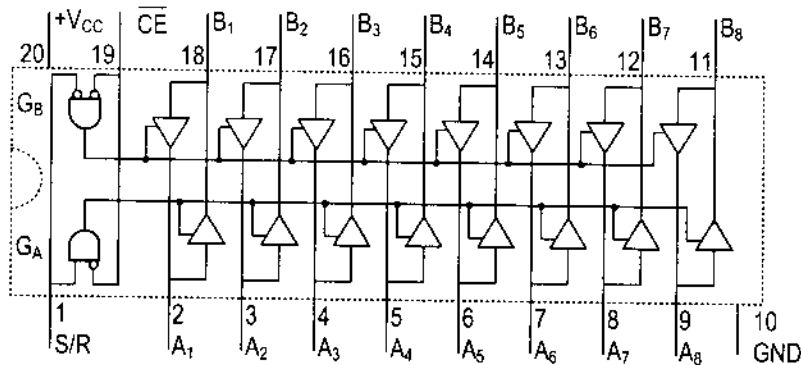
a) Cho nhóm tác động vào mạch hình 3.8a:

$ABCD = 1011$ khi $E_A E_B E_C E_D = 0111$

$ABCD = 1110$ khi $E_A E_B E_C E_D = 1101$

$ABCD = 1100$ khi $E_A E_B E_C E_D = 1110$

- b) Cho nhóm dữ liệu tác động vào mạch hình 3.8b:
 $ABCD = 1001$ khi $E_A E_B E_C E_D = 0100$
 $ABCD = 0110$ khi $E_A E_B E_C E_D = 1000$
 $ABCD = 0111$ khi $E_A E_B E_C E_D = 0010$
10. Cho IC đếm 3 trạng thái 74LS245 có sơ đồ trên hình 3.9, thời gian trễ khi truyền dữ liệu $A \leftrightarrow B$ là 8ns.



Hình 3.9. Bài tập 10

$\overline{CE}$: Chip Enable; S/R: Send/Receive

- a) Theo sơ đồ logic đã cho, nêu nhiệm vụ các đầu vào dữ liệu và đầu vào điều khiển của IC.
- b) Hãy viết bảng chức năng thể hiện hoạt động của IC theo các giá trị mã điều khiển $\overline{E}$ và S ở lối vào.
- 11 a) Sử dụng IC 74LS245 trong việc truyền dữ liệu nối tiếp từng bit, kiểu song song từng nhóm 2, 3... 8 bit hoặc dùng đồng thời vừa nối tiếp vừa song song có được không? Kiểu nào đúng?
- b) Hãy vẽ giản đồ xung theo thời gian của các tín hiệu S/R và $\overline{E}$ khi cần thực hiện truyền dữ liệu theo các cách sau đây:
 – Hướng $A \rightarrow B$: 2 nhóm 8 bit xuất hiện lần lượt từng nhóm, sau đó hướng $B \rightarrow A$: 1 nhóm 8 bit và cuối cùng 3 nhóm 8 bit lần lượt hướng $A \rightarrow B$.

BÀI 4. MẠCH MÃ HÓA (CODE)

4.1. VẤN ĐỀ CHUNG

– Mã hóa có bản chất là việc sử dụng các văn tự và ký hiệu để biểu thị một đối tượng hay các nhóm đối tượng nào đó. Đây là một công việc thường xuyên và đã tồn tại từ xa xưa khi con người đặt tên cho các địa danh, đặt tên cho trẻ sơ sinh hay tên số nhà, đường phố, số áo thi đấu của vận động viên thể thao...

Trong kỹ thuật điện tử số, mã hóa được hiểu là quá trình thực hiện biểu diễn số các tín hiệu dưới dạng một hệ đếm nhất định. Mạch điện tử số có hai trạng thái (có hoặc không có điện thế hoặc dòng điện) dễ dàng có thể mô tả trong hệ nhị phân, do vậy, mã nhị phân là quan trọng nhất trong ngôn ngữ máy còn mã thập phân là quan trọng nhất trong đời sống con người.

– Mã nhị phân 1 ký tự (1 bit) có 2 trạng thái 0 và 1 tương ứng biểu diễn được hai tín hiệu. Với mã nhị phân 2 bit có 4 trạng thái (00, 01, 10 và 11) tương ứng với 4 tín hiệu. Tổng quát với n bit nhị phân, ta có thể biểu diễn được 2^n tín hiệu khác nhau. Như vậy, với N tín hiệu ta cần dùng n bit biểu diễn sao cho: $2^n \geq N$. (4.1)

– Mạch mã hóa là mạch logic thực hiện việc mã hóa.

4.2. BỘ MÃ HÓA NHỊ PHÂN

– Bộ mã hóa nhị phân là mạch logic dùng n bit nhị phân để mã hóa $N = 2^n$ tín hiệu khác nhau. Để cụ thể, ta chọn $N = 8 = 2^3$ tín hiệu vào.

Tại một trong tám đầu vào của mạch xuất hiện một trong tám tín hiệu là các ký tự $Y_0, Y_1, Y_2, Y_3, Y_4, Y_5, Y_6$ và Y_7 . Ví dụ trong hệ đếm 8 là trật tự các số tự nhiên từ 0 đến 7, ta cần 3 ký tự (3 bit) nhị phân biểu diễn tám tín hiệu lối vào này theo hệ thức (4.1). Như vậy, mạch mã hóa có 3 đầu ra tương ứng A (LSB), B và C (MSB), bộ mã hóa thực hiện quá trình biến đổi một trong tám đường vào, thành ba đường ra.

– Quá trình mã hóa yêu cầu bắt buộc ở mỗi thời điểm chỉ cho phép một tín hiệu vào tác động. Ví dụ ở mức 1: $Y_1 = 1$ xuất hiện, Quan hệ logic của $Y_0 \dots Y_7$ là không cùng nhau, tức là chúng phải là một dãy nối tiếp không cho phép đồng thời có từ 2 tín hiệu vào có giá trị 1 xuất hiện.

– Bảng chân lý của bộ mã hóa từ hệ 8 sang hệ 2 với hai cách: chọn mã nhị phân lối ra ở dạng mã tự nhiên 8421 và ở dạng mã vòng Gray được cho trên bảng 4.1.

Bảng 4.1. Bảng chân lý của bộ mã hóa 1 từ 8 thành 3 đường

a) Hàm ra dạng nhị phân 8421

Tín hiệu vào	Các hàm ra		
	C (2 ²)	B (2 ¹)	A (2 ⁰)
Y_0 (0)	0	0	0
Y_1 (1)	0	0	1
Y_2 (2)	0	1	0
Y_3 (3)	0	1	1
Y_4 (4)	1	0	0
Y_5 (5)	1	0	1
Y_6 (6)	1	1	0
Y_7 (7)	1	1	1

b) Hàm ra dạng nhị phân Gray

Tín hiệu vào	Các hàm ra		
	C (2 ²)	B (2 ¹)	A (2 ⁰)
Y_0 (0)	0	0	0
Y_1 (1)	0	0	1
Y_2 (3)	0	1	1
Y_3 (2)	0	1	0
Y_4 (6)	1	1	0
Y_5 (7)	1	1	1
Y_6 (5)	1	0	1
Y_7 (4)	1	0	0

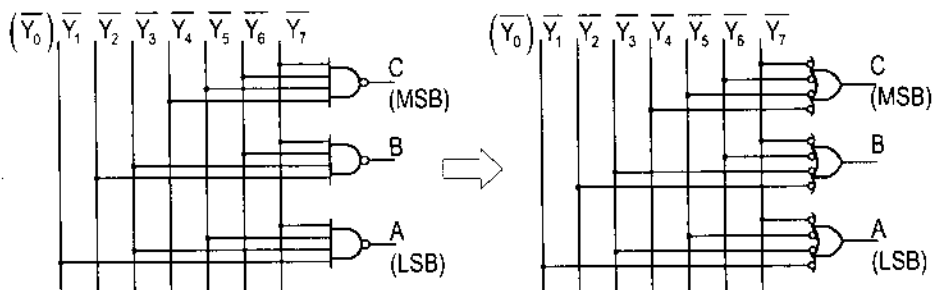
– Theo cách ở bảng 4.1a các hàm ra C, B, A biểu diễn theo các biến vào Y_i có dạng:

$$\left. \begin{aligned} C &= Y_4 + Y_5 + Y_6 + Y_7 = \overline{Y_4} \cdot \overline{Y_5} \cdot \overline{Y_6} \cdot \overline{Y_7} \\ B &= Y_2 + Y_3 + Y_6 + Y_7 = \overline{Y_2} \cdot \overline{Y_3} \cdot \overline{Y_6} \cdot \overline{Y_7} \\ A &= Y_1 + Y_3 + Y_5 + Y_7 = \overline{Y_1} \cdot \overline{Y_3} \cdot \overline{Y_5} \cdot \overline{Y_7} \end{aligned} \right\} \quad (4.2)$$

Theo cách thứ hai từ bảng 4.1b có:

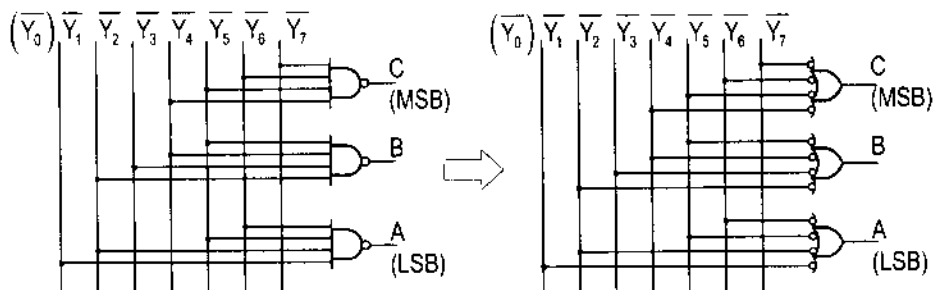
$$\left. \begin{aligned} C &= Y_4 + Y_5 + Y_6 + Y_7 = \overline{Y_4} \cdot \overline{Y_5} \cdot \overline{Y_6} \cdot \overline{Y_7} \\ B &= Y_2 + Y_3 + Y_4 + Y_5 = \overline{Y_2} \cdot \overline{Y_3} \cdot \overline{Y_4} \cdot \overline{Y_5} \\ A &= Y_1 + Y_2 + Y_5 + Y_6 = \overline{Y_1} \cdot \overline{Y_2} \cdot \overline{Y_5} \cdot \overline{Y_6} \end{aligned} \right\} \quad (4.3)$$

– Theo hệ thức (4.2) có thể xây dựng mạch điện cho bộ mã hóa từ hệ 8 sang hệ 2, thể hiện trên hình 4.1.



Hình 4.1. Mạch logic thực hiện mã hóa từ hệ 8 sang hệ 2 tự nhiên

– Theo các hệ thức (4.3) có mạch mã hóa theo mã ra Gray dạng hình 4.2.



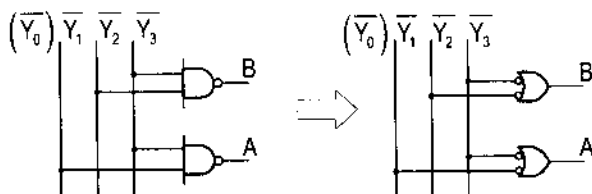
Hình 4.2. Mạch logic thực hiện bộ mã hóa hệ 8 sang hệ nhị phân Gray

– Bộ mã hóa một trong bốn lối vào thành 2 lối ra có bảng chân lý là một phần của bảng 4.1a phần đánh đậm (chấm chấm) với 4 tín hiệu vào là Y_0, Y_1, Y_2, Y_3 và 2 bit ra là B và A.

Hệ hàm ra:

$$\left. \begin{aligned} B &= Y_2 + Y_3 = \overline{Y_2} \cdot \overline{Y_3} \\ A &= Y_1 + Y_3 = \overline{Y_1} \cdot \overline{Y_3} \end{aligned} \right\} \quad (4.4)$$

Từ hệ (4.4) có mạch logic thực hiện hình 4.3.



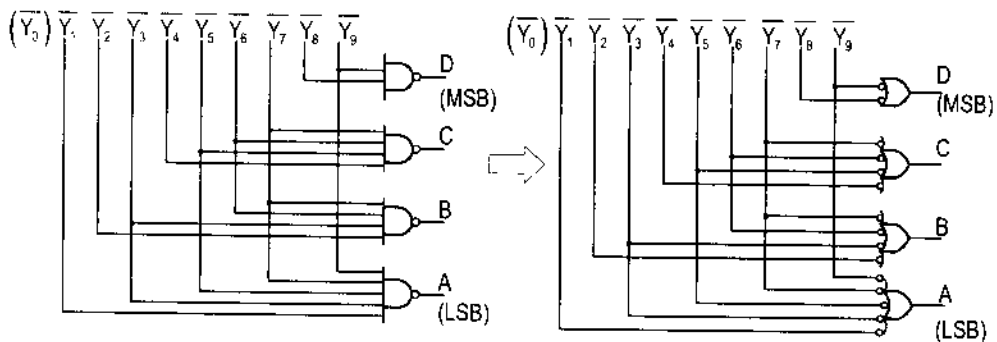
Hình 4.3. Mạch điện thực hiện bộ mã hóa một trong bốn lối vào thành hai lối ra

4.3. MÃ HÓA BCD8421 (Bộ mã hóa một trong 10 đường vào thành 4 đường ra)

Đây là mạch điện chuyển một ký tự thập phân tại lối vào trong hệ 10 (0, 1, 2, 3, 4, 5, 6, 7, 8 hoặc 9) thành số nhị phân tương ứng ở lối ra (4 bit). Áp dụng công thức (4.1), ta có $n = 4$ tức là phải dùng 4 bit nhị phân DCBA để biểu diễn các ký tự thập phân. Cần đảm bảo điều kiện quan hệ logic của các biến vào $Y_0 \dots Y_9$ là không đồng thời. Từ bảng chân lý tương tự như bảng 4.1a bổ sung thêm một cột hàm ra D và hai hàng cuối cho các biến vào Y_8 và Y_9 ứng với DCBA = 1000 và DCBA = 1001, hệ hàm ra tương ứng:

$$\left. \begin{aligned} D &= Y_8 + Y_9 = \overline{Y_8} \cdot \overline{Y_9} \\ C &= Y_4 + Y_5 + Y_6 + Y_7 = \overline{Y_4} \cdot \overline{Y_5} \cdot \overline{Y_6} \cdot \overline{Y_7} \\ B &= Y_2 + Y_3 + Y_6 + Y_7 = \overline{Y_2} \cdot \overline{Y_3} \cdot \overline{Y_6} \cdot \overline{Y_7} \\ A &= Y_1 + Y_3 + Y_5 + Y_7 + Y_9 = \overline{Y_1} \cdot \overline{Y_3} \cdot \overline{Y_5} \cdot \overline{Y_7} \cdot \overline{Y_9} \end{aligned} \right\} \quad (4.5)$$

Mạch logic thực hiện hệ hàm 4.5 được cho trên hình 4.4.



Hình 4.4. Mạch điện bộ mã hóa BCD8421

4.4. BỘ MÃ HÓA ƯU TIÊN

– Thực tế, với các bộ mã hóa thường gặp trường hợp quan hệ logic không đồng thời của các biến vào không được đảm bảo, nghĩa là có nhiều biến vào (đại lượng vào) cùng ở mức tích cực. Trong điều kiện này mạch vẫn hoạt động và chỉ thực hiện việc mã hóa tín hiệu vào nào có cấp ưu tiên cao nhất, các tín hiệu ở mức tích cực khác đã bị từ

chối (hay bị bỏ qua). Bộ mã hóa dạng này được gọi là có tính ưu tiên: Khi đầu vào Y_i đã ở tích cực thì mọi đầu vào $Y_{j < i}$ ở tích cực đồng thời với Y_i sẽ bị bỏ qua (bị cấm).

– IC 74147 là một bộ mã hóa ưu tiên có bảng chân lý cho trên bảng 4.2. IC có 9 lối vào từ $\bar{Y}_1$ đến $\bar{Y}_9$ và 4 lối ra $\bar{D} \bar{C} \bar{B} \bar{A}$ đều ở mức tích cực thấp (mức 0) thực hiện việc mã hóa ưu tiên một số thập phân thành số bù nhị thập phân tự nhiên ($\overline{\text{BCD8421}}$).

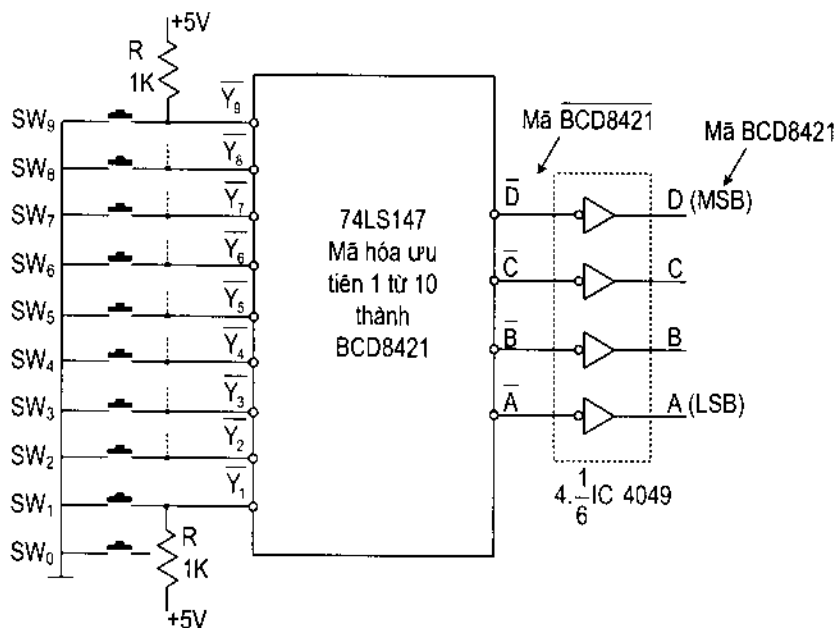
Bảng 4.2. Bảng trạng thái của bộ mã hóa ưu tiên BCD8421

Các biến vào										Các hàm ra			
$\overline{Y}_9$	$\overline{Y}_8$	$\overline{Y}_7$	$\overline{Y}_6$	$\overline{Y}_5$	$\overline{Y}_4$	$\overline{Y}_3$	$\overline{Y}_2$	$\overline{Y}_1$	$\overline{Y}_0$	$\overline{D}$	$\overline{C}$	$\overline{B}$	$\overline{A}$
1	1	1	1	1	1	1	1	1	0	1	1	1	1
1	1	1	1	1	1	1	1	0	x	1	1	1	0
1	1	1	1	1	1	1	0	x	x	1	1	0	1
1	1	1	1	1	1	0	x	x	x	1	1	0	0
1	1	1	1	1	0	x	x	x	x	1	0	1	1
1	1	1	1	0	x	x	x	x	x	1	0	1	0
1	1	1	0	x	x	x	x	x	x	1	0	0	1
1	1	0	x	x	x	x	x	x	x	1	0	0	0
1	0	x	x	x	x	x	x	x	x	0	1	1	1
0	x	x	x	x	x	x	x	x	x	0	1	1	0

Mạch điện bộ mã hóa dùng bàn phím được cho trên hình 4.5 có bổ sung thêm 4 cổng NOT (ví dụ $\frac{4}{6}$ IC 4049) để lối ra nhận được mã BCD8421.

Mạch điện hoạt động như sau, ở trạng thái ban đầu, khi cấp nguồn, các lối vào $\bar{Y}_9 = \dots = \bar{Y}_1 = 1$ do được nối tới mức nguồn 5V qua $R = 1k\Omega$. Các lối vào ở trạng thái không tích cực dẫn tới tại lối ra $\overline{\text{DCBA}} = 1111$ qua 4 cổng NOT có $\text{DCBA} = 0000$. Vậy trạng thái mã số 0 được thiết lập tự động, khi một khóa bàn phím (ví dụ SW_7) được nhấn, $\bar{Y}_7 = 0$ mạch mã hóa hoạt động và cho ra $\overline{\text{DCBA}} = 1000$ và qua các cổng NOT có $\text{DCBA} = 0111$, tức là chỉ số 7 của $\bar{Y}_7$ đã được chuyển thành mã BCD8421: $(7)_{10} = (0111)_{\text{BCD8421}}$.

Trường hợp có nhiều khóa bàn phím cùng nhấn (do sai nhầm) sẽ có vài lối vào ở mức 0 (ví dụ đồng thời có SW_7 , SW_5 và SW_2 thì $\bar{Y}_7 = \bar{Y}_5 = \bar{Y}_2 = 0$) tuy nhiên chỉ $\bar{Y}_7 = 0$ được ưu tiên chọn cho việc mã hóa.



Hình 4.5. Mạch mã hóa khóa bàn phím thập phân thành BCD 8421

Từ bảng chân lý (bảng 4.2) viết được phương trình logic cho các hàm ra.

Sau đó dùng các quy tắc đại số (biểu thức từ 1.8 đến 1.17) để rút gọn hàm ra. Để đơn giản, ta ký hiệu lại các đầu vào bằng các ký tự thập phân tương ứng: Y_i thay bằng i (i từ 0 đến 9, ví dụ Y_9 thay bằng ký tự 9 và cần chú ý rằng các mã ký tự $0 \rightarrow 9$ chỉ có hai trạng thái 0 hay 1).

Kết quả là:

$$\bar{D} = \bar{9} + \bar{98} = \bar{9} + \bar{8} \rightarrow D = \overline{\bar{9} + \bar{8}} \quad (4.6)$$

$$\begin{aligned} \bar{C} &= 98\bar{7} + 987\bar{6} + 9876\bar{5} + 98765\bar{4} \\ &= 98\bar{7} + 98\bar{6} + 98\bar{5} + 98\bar{4} \\ C &= \overline{98\bar{7} + 98\bar{6} + 98\bar{5} + 98\bar{4}} \end{aligned} \quad (4.7)$$

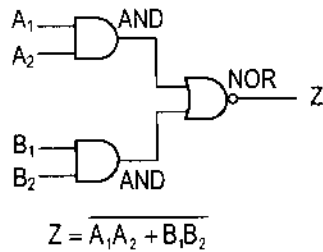
$$\begin{aligned} \bar{B} &= 98\bar{7} + 987\bar{6} + 987654\bar{3} + 9876543\bar{2} \\ &= 98\bar{7} + 98\bar{6} + 9854\bar{3} + 9854\bar{2} \end{aligned}$$

$$B = \overline{987} + \overline{986} + \overline{98543} + \overline{98542} \quad (4.8)$$

$$\overline{A} = \overline{9} + \overline{987} + \overline{98765} + \overline{9876543} + \overline{987654321}$$

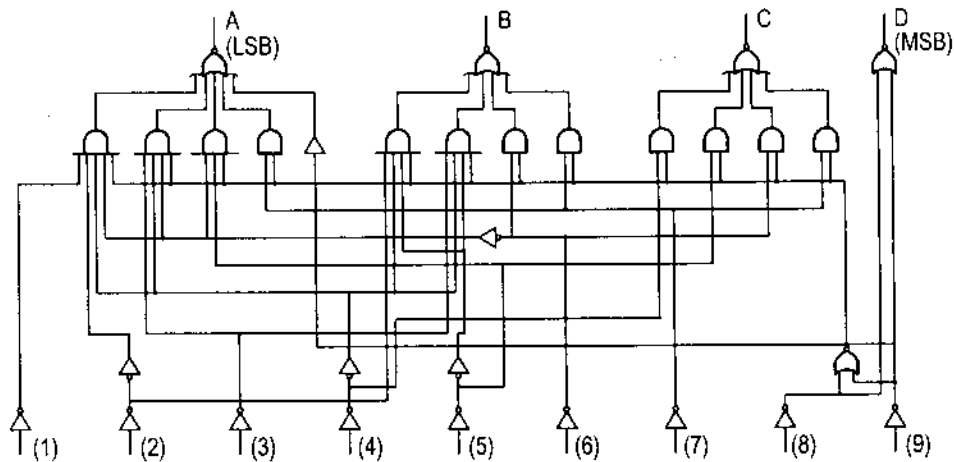
$$A = \overline{9 + 87 + 865 + 8643 + 86421} \quad (4.9)$$

Từ các hệ thức (4.6) đến (4.9) có thể dùng các cổng NOR – AND (và – hoặc đảo) (hình 4.6) để thực hiện mạch logic của bộ mã hóa ưu tiên chế tạo dưới dạng IC MSI74147 (hình 4.7).



Hình 4.6. Cổng logic chuẩn NOR – AND

Đầu ra tín hiệu BCD8421



Đầu vào tín hiệu cần được mã hóa dạng số thập phân

Hình 4.7. Cấu trúc mạch logic bên trong của IC MSI 74LS147

4.5. MỘT SỐ LOẠI MÃ NHỊ THẬP THÔNG DỤNG

4.5.1. Các loại mã nhị thập phân (BCD) thông dụng

Mã BCD sử dụng 4 bit nhị thập phân hợp thành một từ mã biểu diễn một tín hiệu hay thể hiện một hàm ý xác định nào đó và chỉ sử

dùng mô tả các ký tự trong hệ 10. Các dạng mã BCD thông dụng được cho trong bảng 4.3.

Bảng 4.3. Bảng các mã BCD thường gặp

Loại mã Số thập phân	8421	Dư 3	2421 (A) (Eiken)	2421 (B)	5211	5121	7421	Vòng dư 3	Dịch phải (Johnson)	8421
0	0000	0011	0000	0000	0000	0000	0000	0010	00000	1111
1	0001	0100	0001	0001	0001	0001	0001	0110	10000	1110
2	0010	0101	0010	0010	0100	0010	0010	0111	11000	1101
3	0011	0110	0011	0011	0101	0011	0011	0101	11100	1100
4	0100	0111	0100	0100	0111	0111	0100	0100	11110	1011
5	0101	1000	0101	1011	1000	1000	0101	1100	11111	1010
6	0110	1001	0110	1100	1001	1001	0110	1101	01111	1001
7	0111	1010	0111	1101	1100	1010	1000	1111	00111	1000
8	1000	1011	1110	1110	1101	1011	1001	1110	00011	0111
9	1001	1100	1111	1111	1111	1111	1010	1010	00001	0110

Bảng 4.3 cho phép nhận xét các quy luật để xây dựng từng loại mã, chúng chia thành hai nhóm: nhóm có trọng số vị trí và nhóm không có trọng số vị trí.

– Một số có tính đối xứng tạo ra từng cặp từ mã bù của nhau như mã Eiken 2421B. Các cặp từ mã (9, 0) (8, 1) (7, 2) (6, 3) (5, 4) là mã bù của nhau, tức là: $\overline{9_{10}} = \overline{1111}_{2421} = 0_{10} = 0000$

hoặc: $\overline{5_{10}} = \overline{1011}_{2421} = 0100_{2421} = 4_{10}$

– Mã dư 3 có được nhờ lấy mã của số thập phân trong hệ 10 cộng thêm 3 để biểu diễn thành một từ mã, nghĩa là một từ mã dư 3 có giá trị thập phân quy đổi theo mã BCD8421 cao hơn ba đơn vị so với ký tự thập phân thực mà nó đại diện.

– Mã vòng dư 3 có thuộc tính của mã vòng Gray là khoảng cách giữa hai từ mã liên tiếp nhau chỉ khác nhau một giá trị nhị phân (khác nhau 1 bit).

– Các loại mã trong nhóm có trọng số được xác định theo vị trí và trọng số của vị trí bit như cách làm với mã 8421.

Ví dụ: Với loại mã 2421(B) bit đầu tiên bên trái biểu diễn 2^1 , các bit sau lần lượt là 2^2 2^1 và 2^0 , do vậy, 1 từ mã trong biểu diễn 2421 được quy đổi sang hệ thập phân như sau:

$$1101_{2421(B)} = 1.2^1 + 1.2^2 + 0.2^1 + 1.2^0 = 7_{10}$$

$$\text{hoặc từ mã: } 1111_{2421(B)} = 1.2^1 + 1.2^2 + 1.2^1 + 1.2^0 = 9_{10}$$

Ta lấy ví dụ khác với loại mã 5121: bit đầu tiên biểu thị số 5 thập phân.

Các bit sau thể hiện 2^0 2^1 2^0 tương ứng, do vậy:

$$\text{từ mã: } 1111_{5121} = 5 + 1.2^0 + 1.2^1 + 1.2^0 = 9_{10}$$

$$\text{hay từ mã: } 0111_{5121} = 0 + 1.2^0 + 1.2^1 + 1.2^0 = 4_{10}$$

4.5.2. Mã ISO (International Standardization Organization)

Mã ISO do cơ quan tiêu chuẩn hóa quốc tế đưa ra là loại mã 8 bit chủ yếu dùng để truyền tin. Mã ISO gồm 10 số thập phân ($0 \rightarrow 9$); 26 chữ cái tiếng Anh và 20 dấu ký hiệu gồm tổng cộng 56 ký tự được cho trên bảng 4.4. Trên bảng 4.4 có 7 bit b_7 đến b_1 , bit thứ 8 là một bit để biến một từ mã bất kỳ thành một từ mã có số lượng bit là một số chẵn với mục đích phát hiện lỗi truyền tin.

Bảng 4.4. Bảng mã ISO

Ký tự $b_7b_6b_5$ $b_4b_3b_2b_1$	001	001	010	011	100	101	110	111
0000	NUL			0		P		
0001				1	A	Q		
0010				2	B	R		
0011				3	C	S		
0100				4	D	T		
0101			\$	5	E	U		
0110			%	6	F	V		
0111				7	G	W		
1000	BS		(	8	H	X		
1001	HT	EM	)	9	I	Y		
1010	LF		*	:	J	Z		
1011			+		K			
1100			,		L			
1101	CR		-	=	M			
1110			.		N			4
1111			/		O			DEL

Đặc điểm các ký tự trong bảng 4.4 như sau:

1. Các chữ số đều có chung 3 bit cao $b_7b_6b_5 = 011$
2. Các chữ cái đều có chung 2 bit $b_7b_6 = 10$
3. Các chữ số thập phân được chuyển thành mã ISO kiểu mã nhị phân 8421 của 4 bit tương ứng $b_4b_3b_2b_1$.
4. Thứ tự các chữ cái từ A đến Z (từ 1 đến 26) cùng là mã nhị phân 8421 của $b_5b_4b_3b_2b_1$.

4.5.3. Mã ASCII (American National Standard Code for Information Interchange)

Mã ASCII là cụm từ viết tắt mang ý nghĩa là mã tiêu chuẩn quốc gia của Hoa Kỳ để trao đổi tin tức (bảng 4.5).

Bảng 4.5. Bảng mã ASCII

Ký tự $b_7b_6b_5$ $b_4b_3b_2b_1$	000	001	010	011	100	101	110	111
0000	NUL	DLE	SP	0	@	P	\	p
0001	SOH	DC1	!	1	A	Q	a	q
0010	STX	DC2	"	2	B	R	b	r
0011	ETX	DC3	#	3	C	S	c	s
0100	EOT	DC4	\$	4	D	T	d	t
0101	ENQ	NAK	%	5	E	U	e	u
0110	ACK	SYN	&	6	F	V	f	v
0111	BEL	ETB	'	7	G	W	g	w
1000	BS	CAN	(	8	H	X	h	x
1001	HT	EM	)	9	I	Y	i	y
1010	LF	SUB	*	:	J	Z	j	z
1011	VT	ESC	+	;	K	[	k	{
1100	FF	FS	,	<	L	\	l	?
1101	CR	GS	-	=	M	]	m	}
1110	SO	RS	.	>	N	↑	n	~
1111	SI	US	/	?	O	↓	o	DEL

Mã ASCII được dùng cho các thiết bị thông tin và máy tính, là loại mã 8 bit nhị phân có cấu tạo gồm 7 bit biểu thị các tín tức và bit thứ 8 (b_0) dùng để bổ sung kiểm tra tính chẵn lẻ qua đó phát hiện lỗi truyền tin.

Bảng 4.6 giải thích ý nghĩa ký hiệu các chữ cái trong mã ASCII.

Bảng 4.6

Ký hiệu	Ý nghĩa	Ký hiệu	Ý nghĩa
NUL	Số không, không, vô hiệu	BS	Lùi một khoảng ký tự
SOH	Bắt đầu của tiêu đề	HT	Kê bảng hướng ngang
STX	Bắt đầu của văn bản	LF	Chuyển dòng
ETX	Kết thúc của văn bản	VT	Kê bảng hướng dọc
EOT	Kết thúc truyền tin	FF	Điều khiển chạy giấy
ENQ	Hỏi	CR	Quay về đầu dòng
ACK	Thừa nhận	SO	Dịch ra (Shift out)
BEL	Chuông	SI	Dịch vào (Shift in)
DLE	Chuyển mã (Data link escape)	EM	Hết giấy
DC1	Điều khiển thiết bị 1	SUB	Trừ
DC2	Điều khiển thiết bị 2	ESC	Chuyển mã
DC3	Điều khiển thiết bị 3	FS	Dấu phân cách (File separator)
DC4	Điều khiển thiết bị 4	GS	Dấu phân cách gói (Group separator)
NAK	Phủ định	RS	Dấu phân cách ghi (Record separator)
SYN	Đồng bộ	US	Dấu phân cách đơn vị (Unit separator)
ETB	Kết thúc truyền gói tin	SP	Khoảng trống ký tự
CAN	Bỏ qua	DEL	Hủy bỏ

4.6. CÂU HỎI VÀ BÀI TẬP

- Hãy nêu bản chất của quá trình mã hóa. Cho vài ví dụ thực tế minh họa.
- Các bước cơ bản để thực hiện việc mã hóa N đối tượng (N tín hiệu) khi dùng các ký tự trong hệ đếm hai làm cơ sở thực hiện.
- Hãy viết bảng chân lý cho bộ mã hóa 1 trong 4 thành 2 đường.
 - Mở rộng bảng chân lý cho bộ mã hóa 1 trong 16 thành 4 đường.

4. a) Hãy viết hệ các hàm ra cho bộ mã hóa chuyển các số từ hệ 16 thành hệ 2 (1 trong 16 thành 4 đường) và tìm cách rút gọn các hàm ra bằng cách dùng các quy tắc của đại số logic (hệ thức 1.8 đến 1.15).
b) Mạch điện logic thực hiện bộ mã hóa trên gồm bao nhiêu cổng logic cơ bản và bao gồm những loại cổng nào?
5. Bộ mã hóa ưu tiên có đặc điểm gì khác bộ mã hóa thông thường?
Trên hình 4.5, hãy xác định trạng thái tại đầu ra $\overline{DCBA}$ khi cùng lúc các khóa bàn phím SW_5 , SW_3 và SW_8 được nhấn.
6. Cần mã hóa số thập phân có 3 decac hàng trăm, hàng chục và hàng đơn vị. Hãy dựa trên cấu trúc mạch hình 4.4 xây dựng mạch điện thực hiện và xác định giá trị các trạng thái ra $\overline{D_0}\overline{C_0}\overline{B_0}\overline{A_0}$, $\overline{D_1}\overline{C_1}\overline{B_1}\overline{A_1}$ và $\overline{D_2}\overline{C_2}\overline{B_2}\overline{A_2}$ tương ứng với các số lỗi vào.
a) 4 5 6 b) 7 9 3 c) 8 2 1
(ở đây, chỉ số $\overline{D_0}\overline{C_0}\overline{B_0}\overline{A_0}$ chỉ mã $\overline{BCD8421}$ cho cấp 10^0 ...)
7. Cần xây dựng một bộ mã hóa ưu tiên 1 từ 8 thành 3 đường và mã hóa ưu tiên 1 từ 4 thành 2 đường:
a) Hãy viết bảng chân lý cho hai loại trên.
b) Viết phương trình các hàm ra cho hai trường hợp.
Rút gọn các hàm ra (nếu được).
8. a) Vẽ mạch logic xây dựng bộ mã hóa ưu tiên 1 từ 4 thành 2 đường.
b) Vẽ mạch logic của bộ mã hóa ưu tiên 1 từ 8 thành 3 đường dùng các cổng NOR, AND. (Ghi chú: có thể xem bài tập 12 hình 12.22).
9. Hãy nêu các đặc điểm quan trọng của các loại mã:
a) BCD thường gặp: 8421, 2421, 5121, thừa 3, vòng dư 3.
b) Tổ chức cơ bản của mã ISO và các đặc điểm chính.
c) Các phân bố các số thập phân và các chữ cái trong mã ASCII.

BÀI 5. MẠCH GIẢI MÃ (DECODE)

5.1. KHÁI NIỆM

– Mạch giải mã có nhiệm vụ biến đổi một tín hiệu số tại lối vào thường ở dạng một số nhị phân 8421 hay BCD thành một tín hiệu tương ứng (trong hệ thập phân ở lối ra mà nó đã biểu diễn trong quá trình thực hiện mã hóa). Trong trường hợp tổng quát mạch giải mã có nhiệm vụ hồi phục lại tín hiệu hay ký tự gốc ban đầu từ tín hiệu đã được tạo ra trong quá trình mã hóa.

– Bảng chân lý của một bộ giải mã luôn là bảng chân lý đã có của bộ mã hóa tương ứng trong đó các biến vào được chuyển đổi thành các hàm ra và ngược lại, hàm ra được đổi lại thành biến tác động ở lối vào.

– Do tính chất không xuất hiện đồng thời của các biến vào (quan hệ logic là không cùng nhau) đối với bộ mã hóa, khi chuyển đổi sang bộ giải mã tính chất này được bảo toàn đối với các hàm ra là không tồn tại hai hay nhiều hơn hai đầu ra có cùng mức logic tích cực, khi đồng thời tác động tới các lối vào của nó một tổ hợp biến nhị phân đã được mã hóa từ trước.

– Đối với các IC MSI thực hiện chức năng giải mã, để mở rộng phạm vi ứng dụng chúng còn tồn tại một nhóm bit điều khiển để chọn vỏ. Nghĩa là nhờ nhóm bit này chỉ một IC được phép hoạt động "giải mã" phù hợp với mã lệnh, các IC giải mã khác còn lại do không đúng mã lệnh sẽ bị cấm hoạt động. Nhờ cách cấu tạo này, trong một bộ giải mã có hai nhóm cổng vào: vào lệnh chọn vỏ và vào mã dữ liệu cần giải mã. Thường nhóm bit cao của từ mã cần giải mã thực hiện luôn nhiệm vụ điều khiển vỏ. Ví dụ với một từ mã 6 bit $A_5A_4A_3A_2A_1A_0$, nếu sử dụng 3 bit cao $A_5A_4A_3$ để chọn địa chỉ vỏ sẽ điều khiển được một trong 2^3 IC hoạt động và nhóm bit thấp $A_2A_1A_0$ được đưa đồng thời vào 2^3 IC đã có, chức năng giải mã đã được mở rộng $2^3 \cdot 2^3 = 2^6$ tức là chọn một trong 64 đầu ra lên trạng thái tích cực.

5.2. MẠCH GIẢI MÃ HAI LỖI VÀO CHỌN MỘT TRONG BỐN LỖI RA

Bảng chân lý bộ giải mã 1 trong 4 cho trên bảng 5.1.

Bảng 5.1. Bảng chân lý bộ giải mã chọn 1 trong 4

Biến vào		Hàm ra			
$A_1 (2^1)$	$A_2 (2^0)$	Y_0	Y_1	Y_2	Y_3
0	0	1	0	0	0
0	1	0	1	0	0
1	0	0	0	1	0
1	1	0	0	0	1

Hệ hàm logic lối ra đơn giản có dạng:

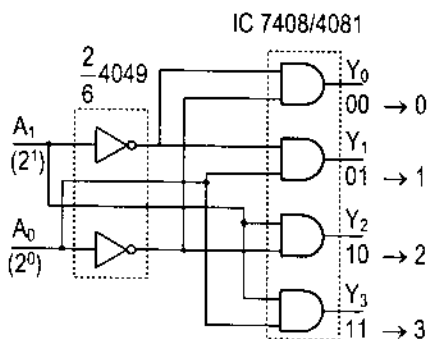
$$Y_0 = \overline{A_1} \overline{A_0}; Y_1 = \overline{A_1} A_0; Y_2 = A_1 \overline{A_0}; Y_3 = A_1 A_0; \quad (5.1)$$

Nếu viết các hàm ra ở mức tích cực thấp thì:

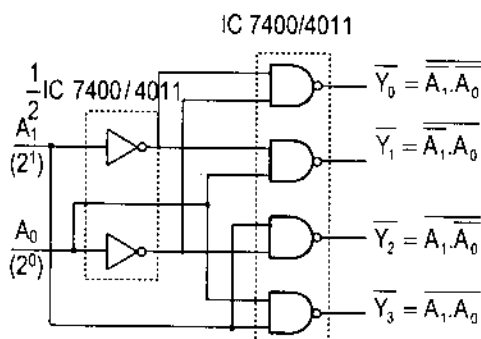
$$\overline{Y_0} = \overline{\overline{A_1} \overline{A_0}}; \overline{Y_1} = \overline{\overline{A_1} A_0}; \overline{Y_2} = \overline{A_1 \overline{A_0}}; \overline{Y_3} = \overline{A_1 A_0}; \quad (5.2)$$

Từ hệ (5.1) ta có mạch logic của bộ giải mã cho trên hình 5.1.

Từ hệ (5.2) ta có mạch logic thực hiện cho trên hình 5.2 kinh tế và dễ thực hiện hơn mạch hình 5.1.



Hình 5.1. Mạch logic của bộ giải mã 1 trong 4



Hình 5.2. Mạch logic thực hiện bộ giải mã 1 trong 4 chỉ gồm 6 cổng NAND

5.3. MẠCH GIẢI MÃ 3 LỖI VÀO CHỌN 1 TRONG 8 LỖI RA

– Bảng chân lý của mạch giải mã chọn 1 trong 8 cho trên bảng 5.2.

Hệ hàm logic ở lối ra có dạng hệ thức (5.3):

$$Y_0 = \overline{A_2} \overline{A_1} \overline{A_0}; Y_1 = \overline{A_2} \overline{A_1} A_0; Y_2 = \overline{A_2} A_1 \overline{A_0}; Y_3 = \overline{A_2} A_1 A_0; \quad (5.3)$$

$$Y_4 = A_2 \overline{A_1} \overline{A_0}; Y_5 = A_2 \overline{A_1} A_0; Y_6 = A_2 A_1 \overline{A_0}; Y_7 = A_2 A_1 A_0;$$

Nếu viết ở dạng các mức ra ở tích cực 0 ta có hệ thức (5.4):

$$\overline{Y_0} = \overline{A_2 A_1 A_0}; \overline{Y_1} = \overline{A_2 A_1 A_0}; \overline{Y_2} = \overline{A_2 A_1 A_0}; \overline{Y_3} = \overline{A_2 A_1 A_0}; \quad (5.4)$$

$$\overline{Y_4} = \overline{A_2 A_1 A_0}; \overline{Y_5} = \overline{A_2 A_1 A_0}; \overline{Y_6} = \overline{A_2 A_1 A_0}; \overline{Y_7} = \overline{A_2 A_1 A_0};$$

Từ các hệ thức trên xây dựng được mạch điện logic của bộ giải mã 1 trong 8 ở dạng hình 5.3 hoặc hình 5.4 cho IC giải mã 1 trong 8 loại MSI 74138/74LS138.

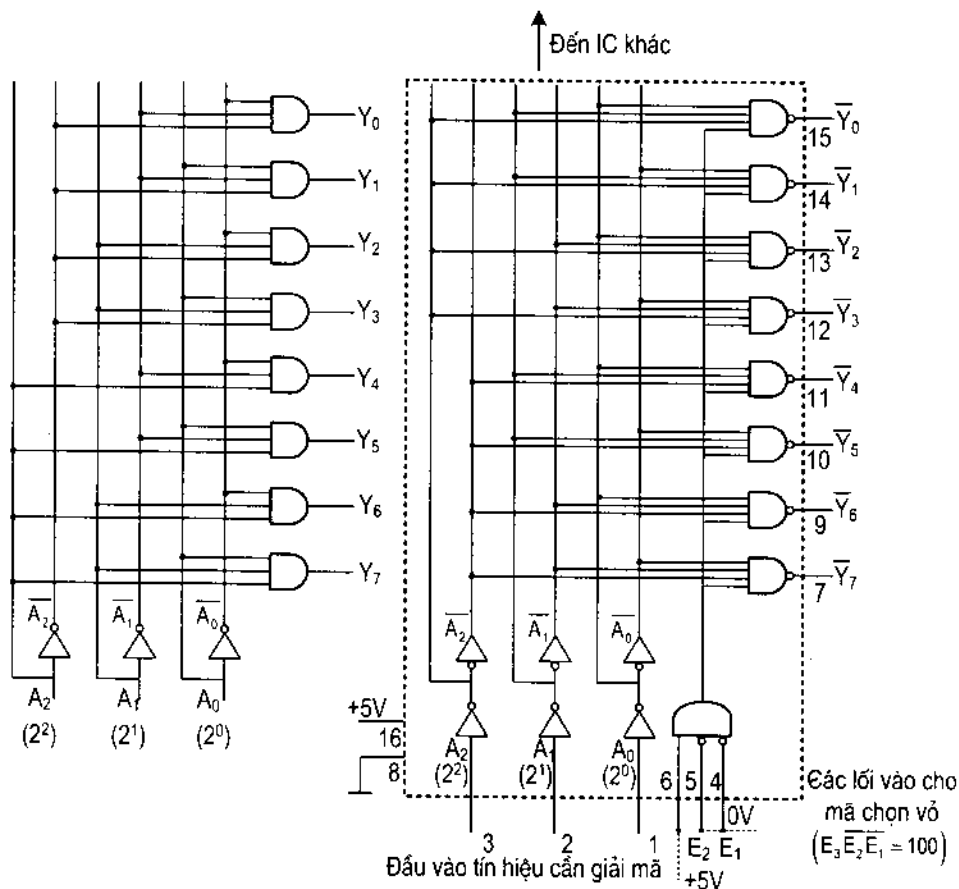
Bảng 5.2. Bảng chân lý bộ giải mã chọn 1 trong 8

Biến vào			Hàm ra							
A_2	A_1	A_0	Y_0	Y_1	Y_2	Y_3	Y_4	Y_5	Y_6	Y_7
0	0	0	1							
0	0	1		1						
0	1	0			1					
0	1	1				1				
1	0	0					1			
1	0	1						1		
1	1	0							1	
1	1	1								1

Mạch hình 5.4 hoạt động theo mã điều khiển $E_3 \overline{E_2} \overline{E_1}$ như sau:

– Khi $E_3 \overline{E_2} \overline{E_1} \neq 100$ toàn bộ các cổng NAND ở lối ra bị cấm (do có mức 0 xuất hiện ở lối ra của cổng AND 3 lối vào). Khi đó, $\overline{Y_0} = \overline{Y_1} = \dots = \overline{Y_7} = 1$ (ở mức không tích cực) với mọi tổ hợp của $A_2 A_1 A_0$.

– Khi $E_3 \overline{E_2} \overline{E_1} = 100$ (như vẽ đường nét đứt tại các lối vào này trên hình) IC được phép hoạt động với chức năng của bộ giải mã chọn 1 trong 8 đối với nhóm $A_2 A_1 A_0$. Nghĩa là $\overline{Y_i} = 0$ khi $(i)_8 = (A_2 A_1 A_0)_2$ và $\overline{Y_{j \neq i}} = 1$ ($i, j = 0, 1, 2, \dots, 7$).



Hình 5.3. Mạch giải mã 3 đường vào 8 đường ra (chọn 1 trong 8)

Hình 5.4. Cấu tạo mạch điện của IC giải mã 74LS138

Nếu mã vào có 6 bit $A_5A_4A_3A_2A_1A_0$, dùng 8 IC đã phân tích với 1 IC cùng dạng để điều khiển (IC "mẹ") có thể mở rộng thành bộ giải mã chọn 1 trong 64 (bài tập 7 trang 70). Khi đó, với IC "mẹ", cần đưa mã vào là nhóm bit cao $A_5A_4A_3$ để giải mã cấp cao, các lỗi vào chọn vô của IC "mẹ" luôn được thỏa mãn $E_3\overline{E_2}\overline{E_1} = 100$, lấy 1 trong 8 tín hiệu cho phép ở các cổng ra của nó điều khiển 1 trong 8 IC "con" hoạt động nhờ các lỗi vào chọn vô của từng IC "con". Lỗi vào dữ liệu của các IC con được đưa đồng thời $A_2A_1A_0$.

5.4. MẠCH GIẢI MÃ 4 ĐƯỜNG VÀO THÀNH 10 ĐƯỜNG RA (chọn 1 trong 10)

Mạch giải mã loại này còn có tên gọi là bộ giải mã BCD thành thập phân.

Bảng chân lý bộ giải mã chọn 1 trong 10 được cho trên bảng 5.3 và mạch logic thực hiện hệ hàm ra của bộ giải mã được cho trên hình 5.5.

Bảng 5.3. Bảng chân lý bộ giải mã chọn 1 trong 10

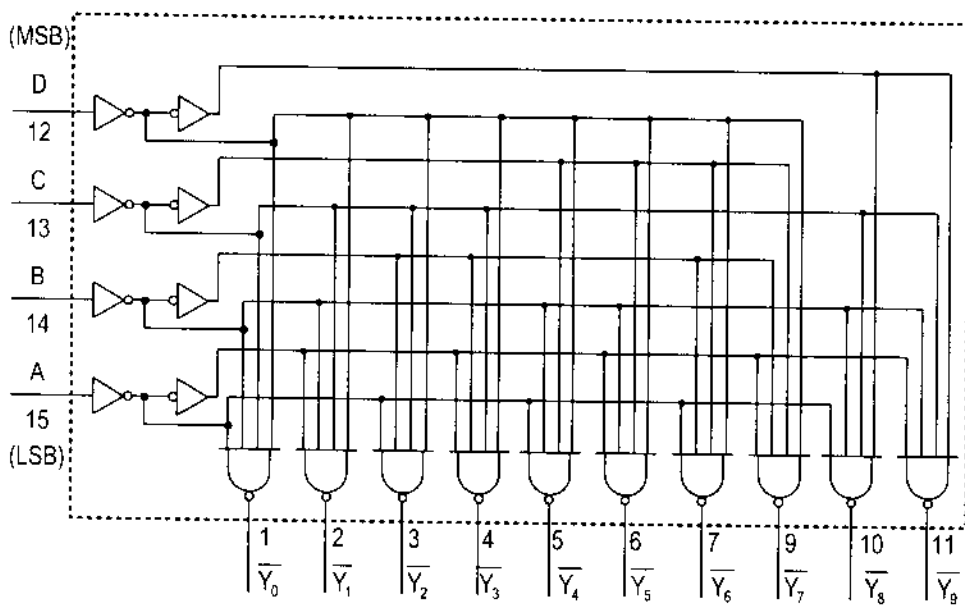
Mã đầu vào				Hàm ra									
D	C	B	A	$\bar{Y}_0$	$\bar{Y}_1$	$\bar{Y}_2$	$\bar{Y}_3$	$\bar{Y}_4$	$\bar{Y}_5$	$\bar{Y}_6$	$\bar{Y}_7$	$\bar{Y}_8$	$\bar{Y}_9$
0	0	0	0	0									
0	0	0	1		0								
0	0	1	0			0					1		
0	0	1	1				0						
0	1	0	0					0					
0	1	0	1						0				
0	1	1	0							0			
0	1	1	1			1					0		
1	0	0	0									0	
1	0	0	1										0

Hàm logic lối ra có dạng:

$$\begin{aligned}
 \bar{Y}_9 &= DA \text{ hay } Y_9 = \overline{DA} & \bar{Y}_4 &= \overline{CBA} \text{ hay } Y_4 = CBA \\
 \bar{Y}_8 &= D\bar{A} \text{ hay } Y_8 = \overline{D\bar{A}} & \bar{Y}_3 &= \overline{CBA} \text{ hay } Y_3 = \overline{\overline{CBA}} \\
 \bar{Y}_7 &= CBA \text{ hay } Y_7 = \overline{CBA} & \bar{Y}_2 &= \overline{CBA} \text{ hay } Y_2 = \overline{\overline{CBA}} \\
 \bar{Y}_6 &= C\bar{B}\bar{A} \text{ hay } Y_6 = \overline{C\bar{B}\bar{A}} & \bar{Y}_1 &= \overline{CBA} \text{ hay } Y_1 = \overline{\overline{CBA}} \\
 \bar{Y}_5 &= CBA \text{ hay } Y_5 = \overline{CBA} & \bar{Y}_0 &= \overline{CBA} \text{ hay } Y_0 = \overline{\overline{CBA}}
 \end{aligned} \tag{5.5}$$

Mạch hình 5.5 là cấu trúc bên trong của IC giải mã 7442 (74LS42 hoặc 74HC42) thực hiện việc giải mã 1 trong 10. Các đặc điểm chú ý thêm là trong mạch có sử dụng 2 cổng NOT liên tiếp để

sửa dạng tín hiệu (xung) cho dốc hơn; có thể dùng 3 đầu vào tín hiệu CBA thực hiện giải mã 1 trong 8 (lúc đó cho D = 0 và buộc $\overline{Y_8}$ và $\overline{Y_9}$ luôn ở mức cao: $\overline{Y_8} = \overline{Y_9} = 1$).



Hình 5.5. Mạch logic bộ giải mã 1 trong 10 dùng IC 7442

5.5. BỘ GIẢI MÃ BCD8421 THÀNH 7 VẠCH DÙNG LED/LCD

– Màn hình LED được sử dụng phổ biến để hiển thị sáng các ký tự thập phân từ 0 đến 9 và đôi khi các ký tự trong hệ 16 (thêm A, B, C, D, E và F). Cấu tạo màn hiển thị LED gồm 7 thanh độc lập nhau có thể nối chung catốt hay nối chung anốt. Loại dùng anốt chung thì được đấu đến +5V và điều khiển mỗi LED bằng mức tích cực 0, có điện trở hạn chế dòng từ 5mA đến cỡ 40mA cho mỗi thanh. Khi đó IC giải mã là loại hở catốt OC để chịu được dòng hút lớn (loại 7446/7447). Loại LED có catốt chung cũng có điện trở (vài trăm Ω) hạn chế dòng và điều khiển vào anốt mỗi thanh nhờ mức tích cực cao (mức 1) thường sử dụng qua một cổng đệm để đủ dòng cung cấp (phun) cho tải là các thanh LED nên có nhược điểm hơn loại trên.

– Bảng chân lý bộ giải mã BCD – 7 vạch được cho trên bảng 5.4.

Bảng 5.4. Bảng chân lý bộ giải mã BCD8421 thành mã 7 thanh

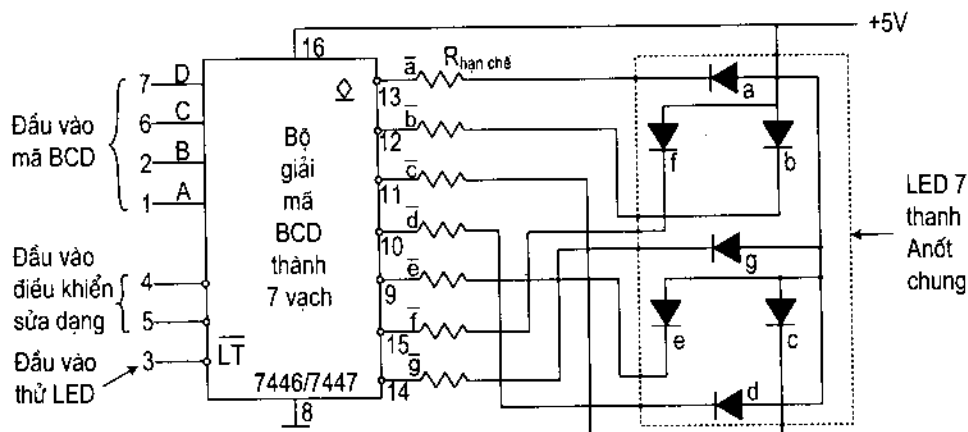
Đầu vào BCD8421				Đầu ra mã 7 thanh							Số thập phân được hiển thị
D	C	B	A	a	b	c	d	e	f	g	
0	0	0	0	0	0	0	0	0	0	1	0
0	0	0	1	1	0	0	1	1	1	1	1
0	0	1	0	0	0	1	0	0	1	0	2
0	0	1	1	0	0	0	0	1	1	0	3
0	1	0	0	1	0	0	1	1	0	0	4
0	1	0	1	0	1	0	0	1	0	0	5
0	1	1	0	0	1	0	0	0	0	0	6
1	1	1	1	0	0	0	1	1	1	1	7
1	0	0	0	0	0	0	0	0	0	0	8
1	0	0	1	0	0	0	0	1	0	0	9

(Viết cho lỗi ra ở mức tích cực thấp)

Trong bảng chân lý (bảng 5.4), các thanh LED loại anốt chung được điều khiển nhờ mức 0 để sáng. Từ bảng chân lý đã có viết được hệ thức (5.6) sau khi rút gọn các hàm:

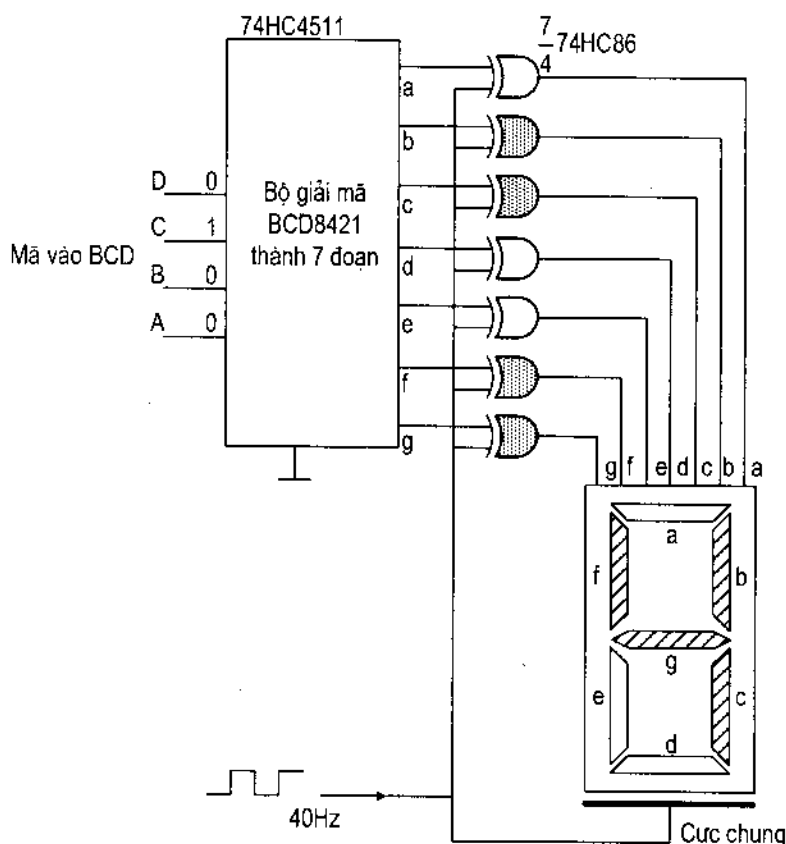
$$\left. \begin{aligned}
 a &= \overline{D + B + CA + \overline{CA}}; & e &= \overline{\overline{CA} + \overline{BA}}; \\
 b &= \overline{\overline{C} + BA + \overline{BA}}; & f &= \overline{D + \overline{CB} + \overline{CA} + \overline{BA}}; \\
 c &= \overline{C + \overline{B} + A}; & g &= \overline{D + \overline{CB} + \overline{BC} + \overline{BA}}; \\
 d &= \overline{D + \overline{CB} + \overline{BA} + \overline{CA} + CBA};
 \end{aligned} \right\} \quad (5.6)$$

Mạch sử dụng IC giải mã 7446/7447, được cho trên hình 5.6.



Hình 5.6. Mạch nối IC giải mã 7446/7447 (loại OC) điều khiển màn hình LED 7 thanh có anốt chung

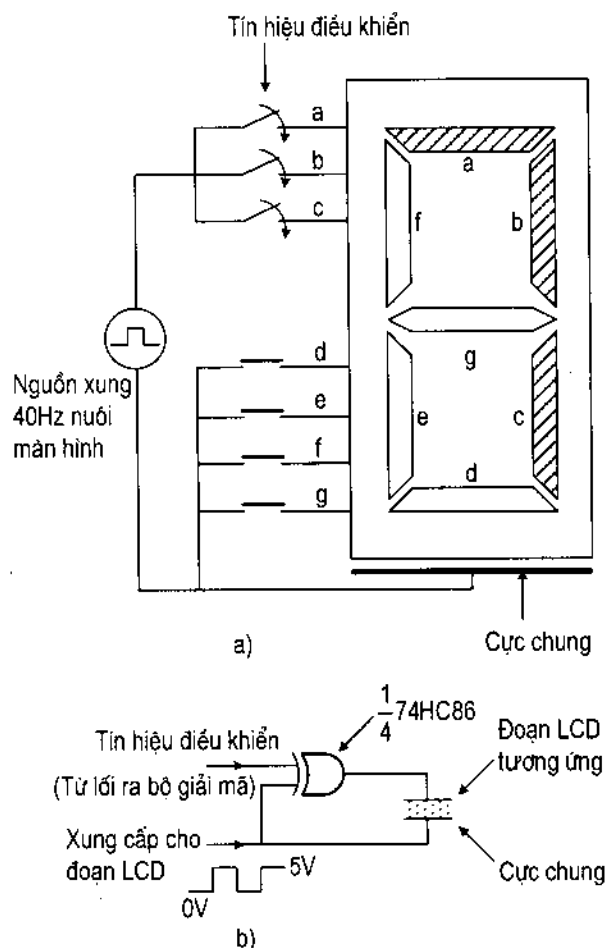
Tín hiệu cần giải mã ở dạng mã BCD8421, các đầu ra $\bar{a} \rightarrow \bar{g}$ ở mức tích cực 0 có thêm các đầu vào điều khiển sử dụng xung vào ($\overline{RBI}$: Ripple Blanking Input) cân bằng đầu vào ($\overline{BI}$: Blanking Input) hoặc sửa dạng xung ra ($\overline{RBO}$: Ripple Blanking Output) và đầu vào thử LED ($\overline{LT}$). Dòng điện tải là dòng hút (vào IC giải mã) nên tương đối lớn và cần dùng loại hở colectơ để chịu được dòng lớn. Giả thiết với mức điện thế trên LED lúc làm việc là 2,7V nguồn cấp là 5V thì điện trở hạn chế cần chọn là $R_{hc} = \frac{5V - 2,7V}{10mA} = 230\Omega$ (hay chọn 220 Ω) trong điều kiện dòng qua mỗi LED là 10mA.



Hình 5.7. Mạch điện bộ giải mã BCD – 7 vạch có tải là màn LCD 7 đoạn (mã vào là 0100, các chuyển mạch XOR b, c, f, g hoạt động hiển thị số 4)

– Khi bảng đèn hiển thị là loại LCD (màn hình tinh thể lỏng có cấu trúc 7 vạch) do đặc điểm nguyên lý hoạt động của màn LCD làm

việc với điện thế xoay chiều (có tần số từ 25Hz tới 60Hz) và dòng tiêu thụ rất nhỏ (tuy độ sáng yếu hơn). Bộ giải mã dùng IC 74HC4511 phối hợp với cổng XOR bộ tạo xung vuông góc tần số 40Hz điều khiển màn hình (hình 5.7). Phương pháp điều khiển màn LCD được cho trên hình 5.8a và b.



Hình 5.8. Màn hình 7 vạch LCD và phương pháp kích thích sáng (trên hình ví dụ là số 7)

5.6. CÂU HỎI VÀ BÀI TẬP

1. Bộ giải mã có nhiệm vụ gì?

Có thể tác động lần lượt các bit nhị phân vào các đầu vào của một bộ giải mã được không? Tại sao?

Có thể cho nhiều đầu ra của một bộ giải mã ở trạng thái logic tích cực được không? Vì sao?

2. Hãy chọn một đáp số đúng trong các mệnh đề sau:

Các lối vào của bộ giải mã chịu tác động các bit vào:

a) Một cách đồng thời; b) Một cách tuần tự; c) Hỗn hợp cả hai cách.

Các lối ra của một bộ giải mã có phương thức xuất dữ liệu kiểu:

a) Song song; b) Nối tiếp; c) Hỗn hợp cả song song và nối tiếp.

3. Cho một từ mã CBA = 101 xuất hiện lần lượt tại các lối vào tương ứng của bộ giải mã 1 trong 8 (nhịp đầu C tác động, sau một chu kỳ đến B tác động, sau hai chu kỳ đến A tác động). Hãy xác định trạng thái ra của bộ giải mã trong suốt thời gian có tác động.

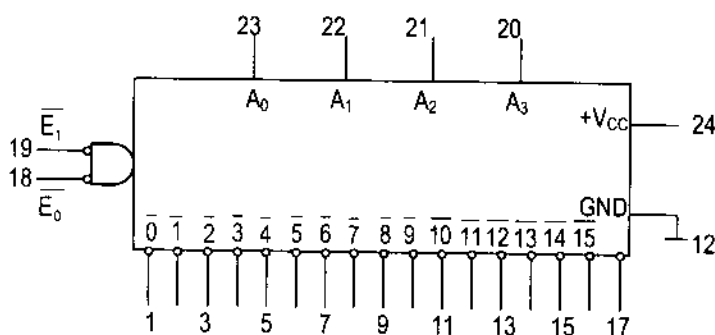
4. Câu hỏi tương tự câu 3 đối với bộ giải mã BCD8421 chọn 1 trong 10 khi tại các lối vào dữ liệu tương ứng $2^3 2^2 2^1 2^0$ lần lượt xuất hiện 1101 trong 4 chu kỳ bit.

5. Cho biết IC MSI 74154 là một bộ giải mã 4 lối vào 16 lối ra (kiểu chọn 1 trong 16). Có thêm hai lối vào điều khiển chọn vỏ $\bar{E}_1 \bar{E}_0$ thể hiện trên hình 5.9.

Khi $\bar{E}_1 \bar{E}_0 = 00$ thì IC được phép làm việc, còn khi $\bar{E}_1 \bar{E}_0 \neq 00$ thì IC bị cấm hoạt động (các lối ra $\bar{0} = \bar{1} = \dots = \bar{15} = 1$).

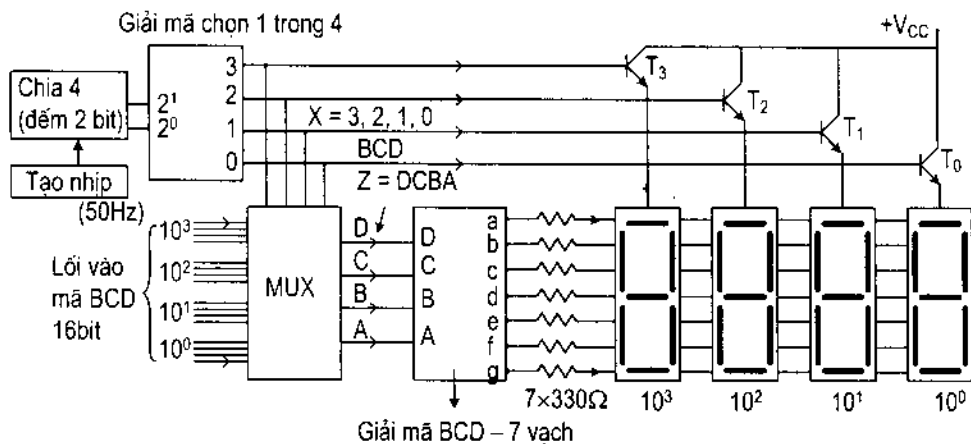
a) Hãy viết trạng thái của các đầu ra khi tại lối vào xuất hiện các từ mã: 1101; 0110 và 1110 (trong điều kiện $\bar{E}_1 \bar{E}_0 = 00$).

b) Hãy dùng 4 hoặc 5 IC đã cho xây dựng cấu trúc của bộ giải mã 1 trong 64 (6 đường vào – 64 đường ra).



Hình 5.9

6. a) Hãy xây dựng một mạch giải mã 4 đường vào $A_3A_2A_1A_0$ 16 đường ra dùng 2 IC 74LS138 đã cho ở hình 5.4 (mạch thực hiện không cần thêm bất kỳ một phần tử nào khác).
 b) Bài toán tương tự khi sử dụng 5 IC 74LS138 để xây dựng một bộ giải mã với 5 bit mã vào $A_4A_3A_2A_1A_0$ và 32 lối ra (bộ giải mã 1 trong 32 đường).
7. Khi xây dựng một bộ giải mã 6 đường vào $A_5A_4A_3A_2A_1A_0$, 64 đường ra $\overline{Y}_0... \overline{Y}_{63}$ có thể sử dụng IC 74LS138 thực hiện nhờ chọn nhóm bit cao $A_5A_4A_3$ để điều khiển chọn vô, nhóm bit thấp của từ mã ($A_2A_1A_0$) để xác định địa chỉ ra cụ thể của một trong các IC đã được chọn.
 a) Hãy xây dựng mạch thực hiện bằng 8 IC và một số cổng logic cơ bản bổ sung.
 b) Xây dựng mạch từ 9 IC đã cho (không cần bổ sung bất kỳ một cổng logic nào).
8. Khi sử dụng bộ giải mã BCD thành 7 thanh LED cần chú ý tới các đặc điểm quan trọng gì?
 Hai phương thức điều khiển LED 7 thanh kiểu anốt chung và kiểu catốt chung có gì khác nhau?
 Khi chọn giá trị điện trở hạn chế (nối tiếp với mỗi thanh LED) tăng lên (ví dụ, tới 500Ω hay $1k\Omega$) hoặc giảm đi (ví dụ tới 50Ω) thì điều gì sẽ xảy ra trong hai trường hợp trên? Giả thiết dòng sáng của một diốt LED $I_{min} = 3mA$ và $I_{max} = 30mA$ điện áp trên LED là 2,7V lúc làm việc.
9. Cho mạch điện có cấu trúc khối trên hình 5.10.
 a) Hãy nêu nhiệm vụ của từng khối trong mạch hình 5.10 đã cho và nhiệm vụ chung. Giải thích phương thức hoạt động của mạch.
 b) Xác định kết quả trên LED khi tại lối vào xuất hiện mã BCD: 1001 0110 0100 0101 và vẽ đồ thị thời gian các xung DCBA tại lối ra của MUX trong một chu kỳ làm việc của nó.
 Tính chu kỳ làm việc của MUX.
 c) Nếu chọn xung nhịp chậm hơn, ví dụ 20Hz rồi sau đó 10Hz có hiện tượng gì xảy ra?



(X: đầu vào địa chỉ chọn nhóm của MUX Z đầu ra mã BCD của nhóm (4bit) được chọn)

Hình 5.10.

10. Kết luận "ứng với mỗi tổ hợp mã vào BCD sẽ có một đầu ra của bộ giải mã BCD thành 7 vạch, như tính chất chung của các bộ giải mã khác, được chuyển đến trạng thái logic tích cực" là đúng hay sai? Vì sao?
 - a) Trạng thái ra nào của IC 7447 sẽ xuất hiện khi tác động lần lượt các tổ hợp mã vào 0101, 1001, 0011 và 1100. Hãy viết mã ra tương ứng cho các hàm ra a, b, c, d, e, f, g và số thập phân được hiển thị trên LED. Nhận xét gì về các kết quả và nêu các trường hợp tương tự như mã vào 1100.
 - b) Hãy viết các ký tự ở mức logic tích cực 0 để LED hiển thị sáng các số thập phân từ 0 đến 9.
11. Muốn mở rộng cho các ký tự A, B, C, D, E và F trong mã 16 hiển thị nhờ LED 7 thanh, các đầu vào của bộ giải mã cần chấp nhận các tổ hợp BCD còn lại là 1010 1011 1100 1101 1110 và 1111 (ký hiệu chung là DCBA).
 - a) Hãy lập bảng chân lý cho 6 ký tự còn lại này thể hiện chỉ thị trên LED vạch dùng anốt chung.
 - b) Hãy viết hệ hàm ra đầy đủ cho các hàm a... g thỏa mãn bảng chân lý đã có ở câu a).

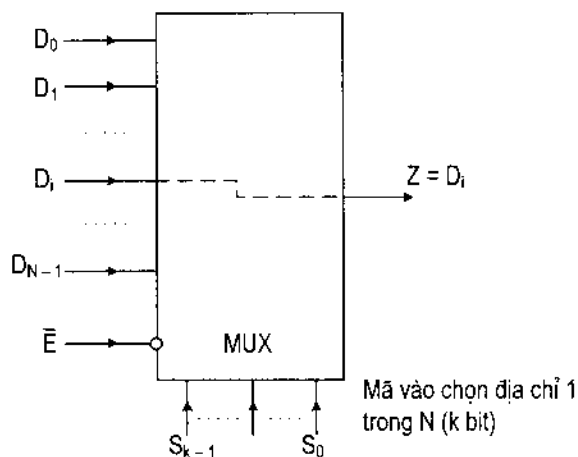
BÀI 6. MẠCH CHỌN DỮ LIỆU

(Dồn kênh – Multiplexer – MUX)

6.1. KHÁI NIỆM

Bộ dồn kênh được định nghĩa là một mạch tổ hợp (hay một thiết bị tổ hợp) có khả năng chọn ra một trong các tín hiệu đang đặt ở đầu vào, đưa tín hiệu này qua nó tới một lối ra duy nhất. Nếu các dữ liệu vào xuất hiện dưới dạng số thì lối ra duy nhất này cũng là dữ liệu số đã được chọn và ta có bộ MUX số. Việc chọn dữ liệu nào được đảm bảo nhờ mã điều khiển đưa tới các lối vào điều khiển. Như vậy, phương thức hoạt động của MUX là vào dữ liệu và vào điều khiển là đồng thời, ra dữ liệu là tuần tự.

Cấu trúc khối của một bộ MUX số tổng quát được thể hiện trên hình 6.1 gồm có N dữ liệu số ở lối vào: D_0 đến D_{N-1} . Muốn vậy, cần có k bit làm mã điều khiển sao cho $2^k \geq N$ (khi mã điều khiển có n bit thì khả năng kiểm soát dữ liệu vào tối đa là 2^n dữ liệu). Tín hiệu $\bar{E}$ để chọn vô. Khi $\bar{E} = 0$ thì bộ MUX được phép làm việc, còn khi $\bar{E} = 1$ thì nó bị cấm làm việc. Như vậy, bộ mã vào S_{k-1} đến S_0 chỉ có hiệu lực khi $\bar{E} = 0$ và khi đó $z = D_i$ với: $(i)_{10} = (S_{k-1} \dots S_0)_2$ ($0 \leq i \leq 2^k - 1$).

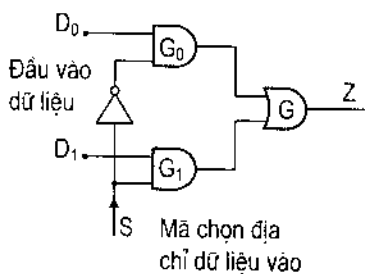


Hình 6.1. Sơ đồ khối chức năng của MUX số

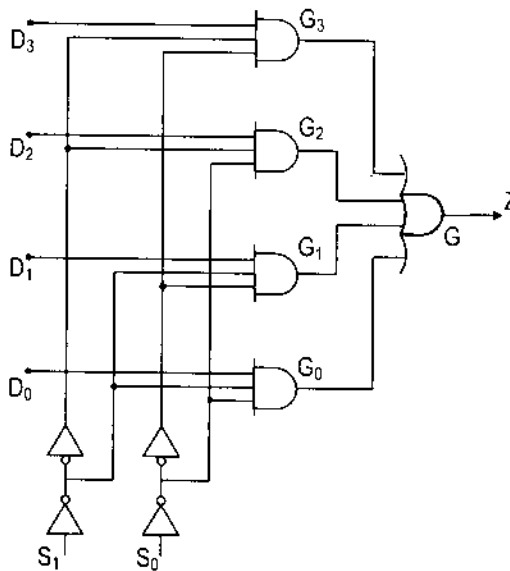
6.2. MẠCH DỒN KÊNH 2 ĐƯỜNG VÀO, 1 ĐƯỜNG RA VÀ MẠCH DỒN KÊNH 4 ĐƯỜNG VÀO, 2 ĐƯỜNG RA

– Trên mạch hình 6.2, khi $S = 0$ thì G_0 mở và D_0 được chọn qua G_0 và G đặt tới lối ra: $z = D_0$. Còn khi $S = 1$ thì G_1 mở và D_1 được chọn qua G_1 và G đặt tới lối ra: $z = D_1$. Hai cổng AND G_0 và G_1 làm việc ngược nhau theo bit địa chỉ S điều khiển.

– Mạch hình 6.3 có nguyên lý hoạt động tương tự. Mã chọn địa chỉ gồm 2 bit S_1S_0 với 4 trạng thái có thể là 00 01 10 và 11 tương ứng sẽ điều khiển lần lượt các cổng AND 3 lối vào, lối vào thứ ba của mỗi cổng có dữ liệu $D_0D_1D_2$ hay D_3 đã chờ sẵn. Hàm ra có được nhờ dấu các đầu ra của AND tới một cổng OR gồm 4 lối vào.



Hình 6.2. Bộ dồn kênh 2 dữ liệu vào

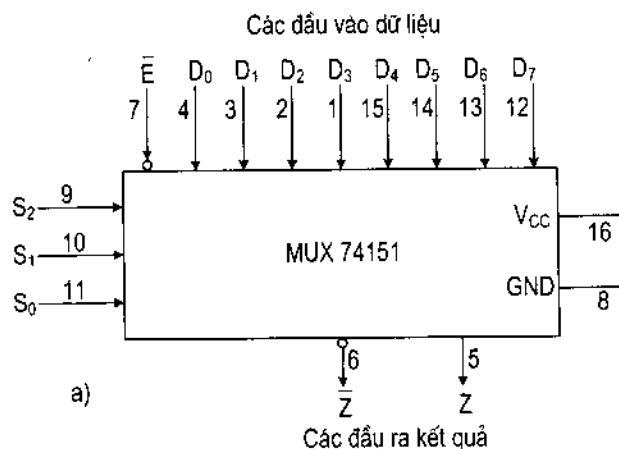


Hình 6.3. Bộ MUX số 4 đầu vào

6.3. BỘ DỒN KÊNH 8 ĐẦU VÀO DỮ LIỆU

Bộ dồn kênh 8 đầu vào dữ liệu dưới dạng IC MSI74151 có sơ đồ logic và bố trí vị trí các chân cho trên hình 6.4a. Bộ dồn kênh IC 74S251 có chức năng tương tự 74151 nhưng có lối ra z và $\bar{z}$ loại cổng

3 trạng thái để kiểm soát dữ liệu ra. Bảng chân lý của 74151 cho trên hình 6.4b và mạch điện logic cấu tạo bên trong của IC 74151 được cho trên hình 6.5.



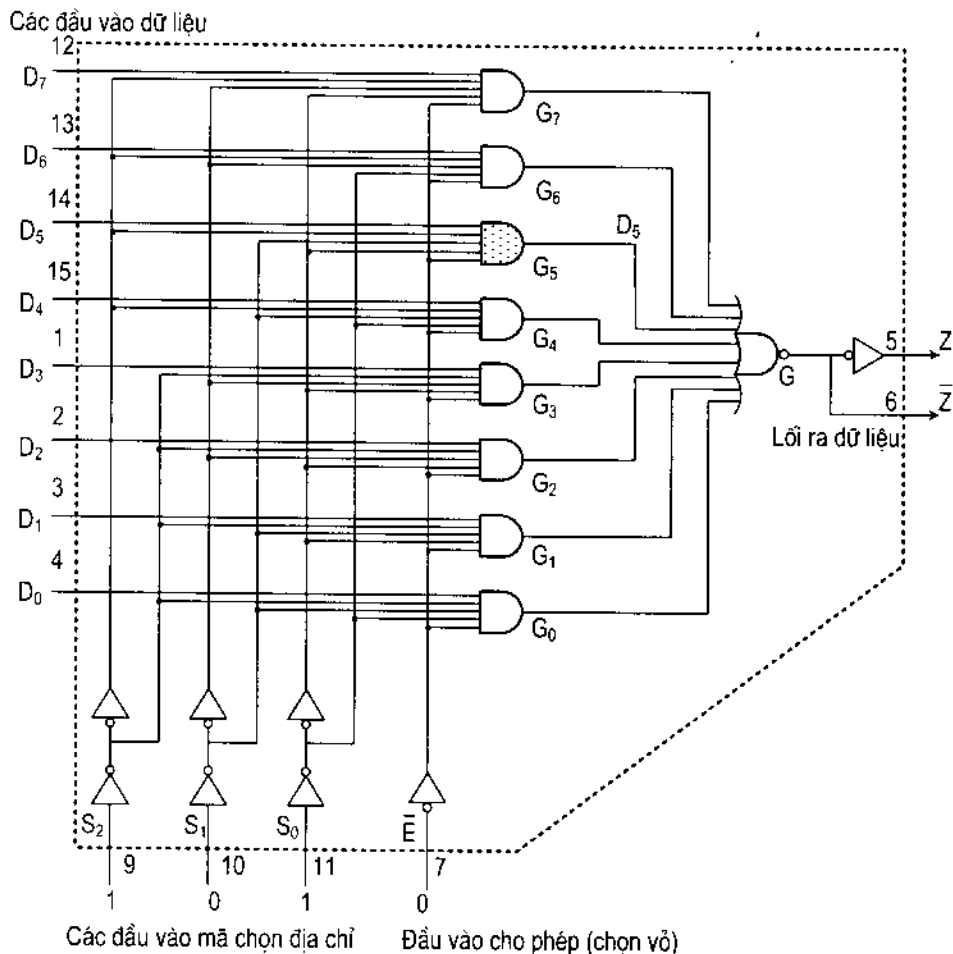
b)

Đầu vào chọn				Đầu ra	
$\bar{E}$	S_2	S_1	S_0	$\bar{Z}$	Z
1	X	X	X	1	0
0	0	0	0	$\bar{D}_0$	D_0
0	0	0	1	$\bar{D}_1$	D_1
0	0	1	0	$\bar{D}_2$	D_2
0	0	1	1	$\bar{D}_3$	D_3
0	1	0	0	$\bar{D}_4$	D_4
0	1	0	1	$\bar{D}_5$	D_5
0	1	1	0	$\bar{D}_6$	D_6
0	1	1	1	$\bar{D}_7$	D_7

Hình 6.4. a) Ký hiệu logic của IC MUX 74151 và bố trí chân vào ra;

b) Bảng chân lý của IC 74151

Theo bảng chân lý 6.4b, khi $\bar{E} = 1$: $\bar{z} = 1$; $z = 0$ hoặc ở trạng thái trở kháng cao Hi_Z với mọi mã địa chỉ chọn $S_2S_1S_0$, IC bị cấm hoạt động. Khi $\bar{E} = 0$, IC được phép hoạt động và lối ra liên hợp z , $\bar{z}$ đóng lại, dữ liệu vào được chọn theo mã $S_2S_1S_0$ tức là $z = D_i$ với $(i)_8 = (S_2S_1S_0)_2$ ($0 \leq i \leq 7$).



Hình 6.5. Cấu tạo mạch logic bên trong của bộ MUX 8 đầu vào dữ liệu loại IC MSI74151

Ví dụ: Khi $\bar{E} = 0$, tất cả các cổng G_0 đến G_7 được mức $E = 1$ cho phép đón nhận dữ liệu D_0 đến D_7 . Nếu khi đó xuất hiện mã chọn địa chỉ dữ liệu $S_2S_1S_0 = 101$ thì duy nhất cổng G_5 hội tụ đủ điều kiện mở vì tất cả các mức vào chọn đều ở 1 và do vậy D_5 qua G_5 đặt tới lối vào của cổng NOR 8 đầu vào, lối ra $z = D_5$.

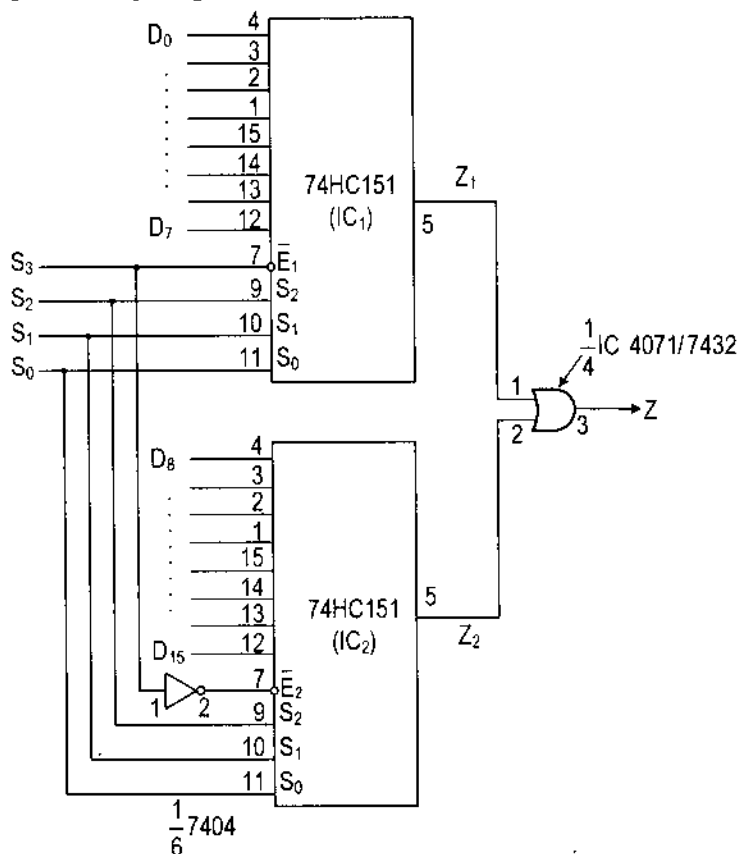
Khi $\bar{E} = 1$ thì $E = 0$ và các cổng G_0 đến G_7 bị khóa, mọi dữ liệu đều bị từ chối kể cả mã chọn địa chỉ và dữ liệu vào.

Lối vào chọn có từng cặp cổng NOT với mục đích sửa dạng tín hiệu.

– Khi số lượng dữ liệu tăng lên, ví dụ $D_0 \rightarrow D_{15}$ có thể dùng 2 IC đã nêu và dùng mã 4 bit chọn $S_3S_2S_1S_0$ trong đó bit S_3 được chọn đưa vào

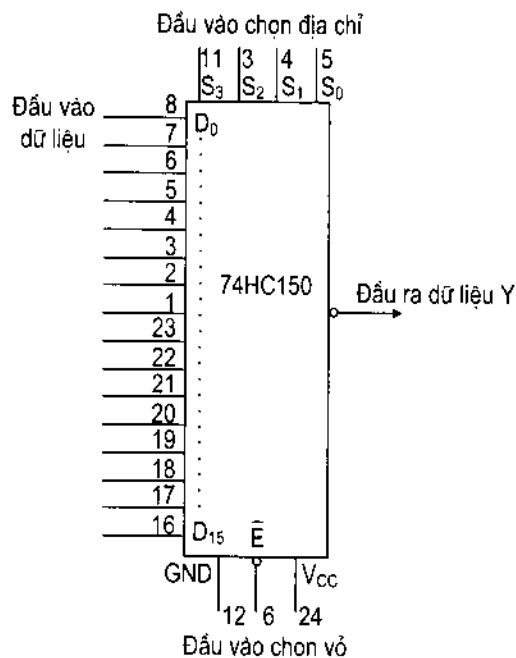
đầu vào $\overline{E_1}$ chọn vỏ: Khi $S = 0$ thì IC₁ được chọn, còn khi $S_3 = 1$ ta bổ sung một cổng NOT trước khi đưa vào đầu vào $\overline{E_2}$ và IC₂ được chọn. IC₁ quản lý nhóm dữ liệu D₀ đến D₇ có mã địa chỉ 0000 đến 0111; IC₂ quản lý nhóm dữ liệu D₈ đến D₁₅ có mã địa chỉ 1000 đến 1111.

Lối ra z_1 và z_2 của IC₁ và IC₂ cần đưa tới 1 cổng OR 2 lối vào để được kết quả $z = z_1 + z_2$ (hình 6.6).



Hình 6.6. Bộ dồn kênh 16 lối vào dùng 2 IC 74151 với 4 dữ liệu mã chọn $S_3S_2S_1S_0$

– Bài toán mở rộng với MUX 16 lối vào hay lớn hơn (ví dụ 32 lối vào) có thể được giải quyết theo phương pháp tương tự như đã trình bày với IC 74151 hoặc chọn loại IC 74150 với 4 mã chọn $S_3S_2S_1S_0$ cho 16 lối vào dữ liệu D₀ → D₁₅ và 1 lối vào chọn vỏ $\overline{E}$, IC 74150 chỉ có một lối ra $\overline{Y}$ duy nhất (hình 6.7). Khi mở rộng thành MUX có 32 lối vào cần dùng 2 IC 74150 với 5 bit chọn $S_4S_3S_2S_1S_0$, lấy S_4 để chọn vỏ (đưa vào cổng $\overline{E}$).



**Hình 6.7. Sơ đồ logic của IC MSI74150
(bộ dẫn kênh 16 lối vào)**

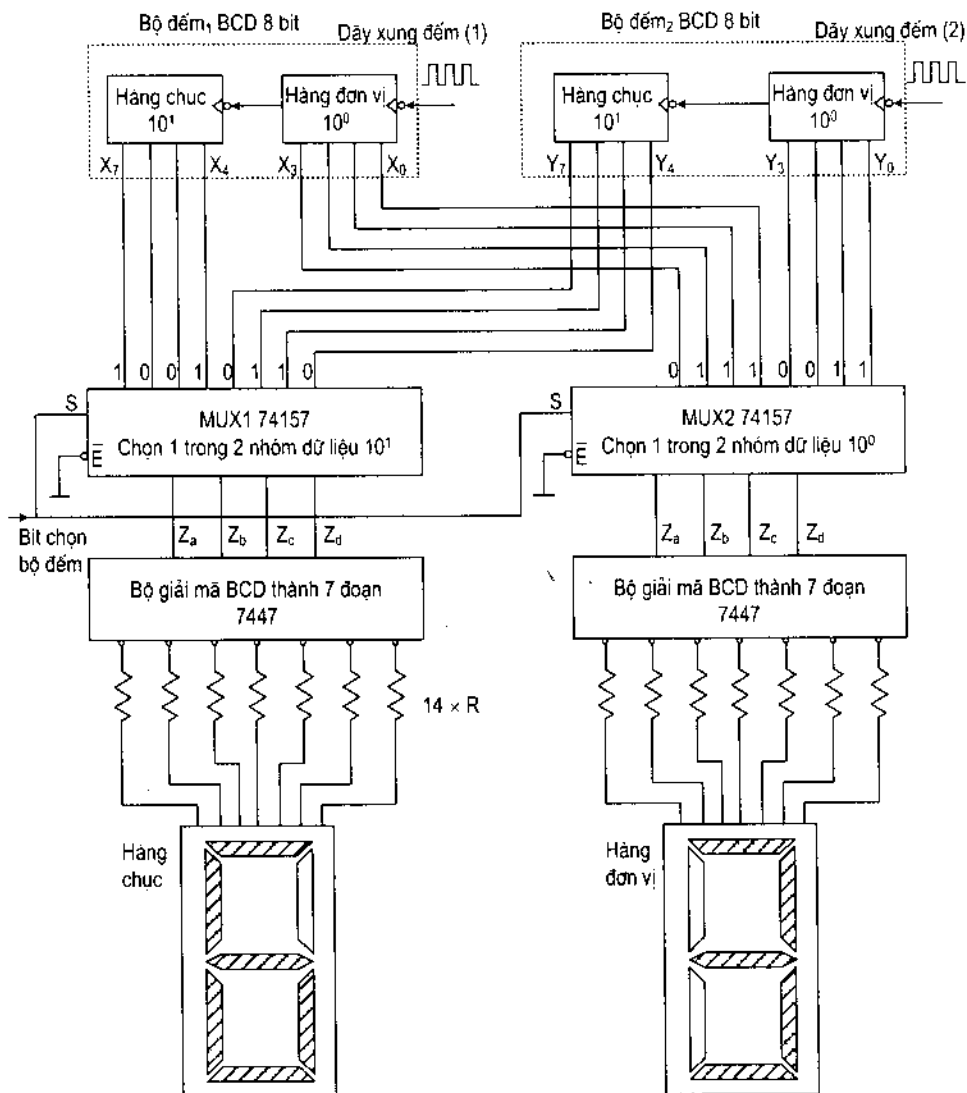
6.4. MỘT VÀI ỨNG DỤNG CỦA BỘ DẪN KÊNH

6.4.1. Định tuyến dữ liệu

Có nhiều dữ liệu muốn định hướng tới một địa chỉ đích duy nhất, khi sử dụng MUX sẽ cho phép chọn dữ liệu nào được đi trước hướng đến đích, các dữ liệu chưa hoặc không được chọn sẽ bị ngăn lại (cấm đi tới đích) vào lúc đó. Ví dụ, có hai luồng dữ liệu ở dạng mã BCD 8 bit thể hiện kết quả của quá trình đếm xung tại các đầu ra song song của hai bộ đếm BCD hàng chục và hàng đơn vị (hình 6.8).

Nhiệm vụ cần giải quyết là phải chọn một trong hai luồng dữ liệu này tới đích là một khối giải mã BCD thành 7 vạch và được hiển thị trên hai màn hình LED kết quả tương ứng với hai số thập phân hàng chục và hàng đơn vị. Hai IC logic MUX 74157 cho phép hiển thị một trong hai kết quả đếm của bộ đếm 1 hay của bộ đếm 2 nhờ mức của tín hiệu chọn S. Khi $S = 1$ dữ liệu 8 bit của bộ đếm 1 sẽ được chọn để hiển thị, còn khi $S = 0$ thì nhóm dữ liệu 8 bit của bộ đếm 2 được chọn. Việc phân chia thời gian khác nhau tới đích của các nhóm dữ liệu sẽ

nâng hiệu quả sử dụng đường truyền do đó giảm tối thiểu các chi phí kinh tế kỹ thuật, điều này càng có ý nghĩa quan trọng khi khối lượng dữ liệu cần định tuyến ngày càng nhiều và tuyến (cự ly) truyền tải dữ liệu càng xa số lượng các đích trung gian càng lớn (đích trung gian dùng để tập kết dữ liệu, phân loại trước khi tiếp tục truyền tải).

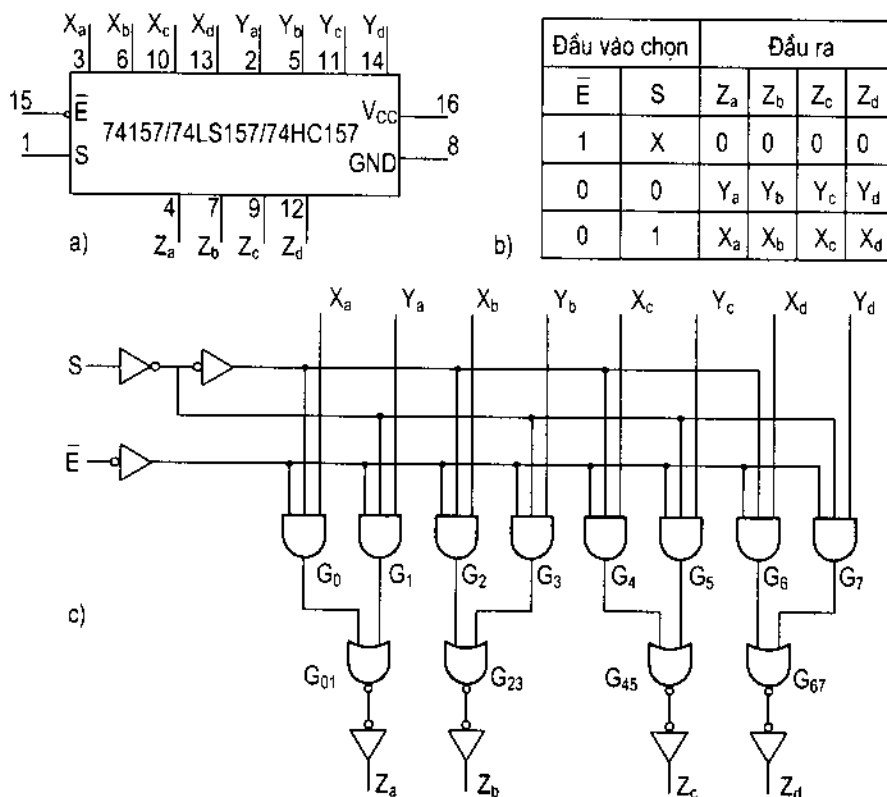


Hình 6.8. Sử dụng MUX 74157 để định tuyến cho hai nhóm dữ liệu 8 bit đầu ra hiển thị thập phân

S = 1: Hiển thị nhóm dữ liệu X; S = 0: hiển thị nhóm dữ liệu Y.

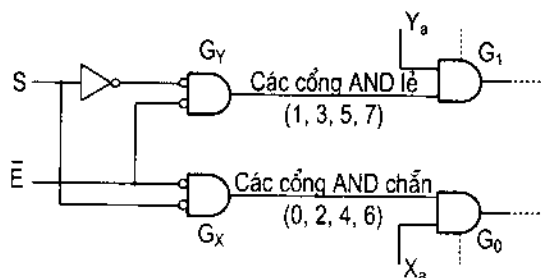
Trên hình đang thể hiện dữ liệu kết quả của nhóm Y (S = 0).

Ký hiệu logic, bảng chân lý và mạch điện cấu tạo của IC MSI 74157 được cho trên hình 6.9.



**Hình 6.9. a) Ký hiệu logic và bố trí chân của IC 74157;
b) Bảng chân lý; c) Mạch logic cấu tạo của IC 74157**

Chú ý trên thực tế 8 cổng AND 3 lối vào được thay thế bằng 8 cổng AND chỉ có hai lối vào và cặp tín hiệu chọn $\bar{E}S$ được đưa vào hai cổng NOR G_Y và G_X như hình 6.10.



Hình 6.10. Đầu vào chọn dùng cặp cổng NOR G_X và G_Y trong cấu tạo mạch logic của IC 74157/74158

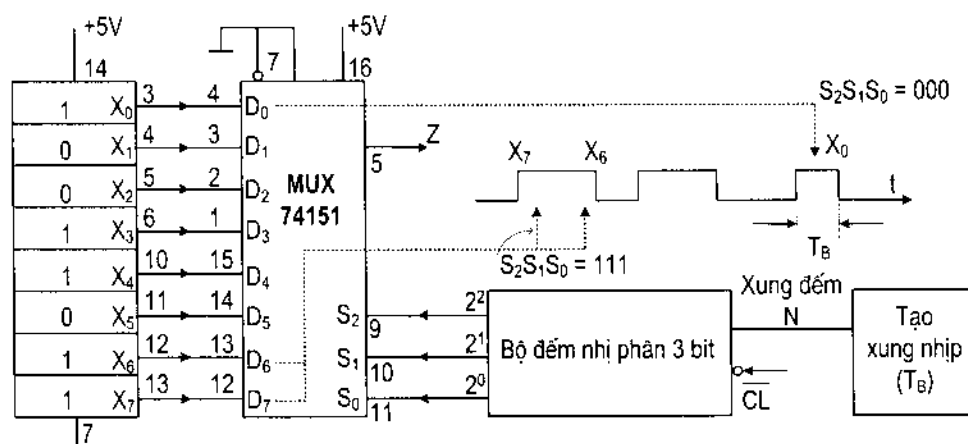
6.4.2. Chuyển đổi một luồng dữ liệu

Một ứng dụng quan trọng và phổ biến của các bộ MUX là thực hiện quá trình chuyển đổi một luồng dữ liệu nhị phân song song (các bit xuất hiện đồng thời) thành luồng dữ liệu truyền nối tiếp nhau (các bit xuất hiện tuần tự).

Luồng dữ liệu bit song song luôn có ưu thế về tốc độ xử lý nhanh nhưng mỗi bit cần riêng một đường truyền nên chỉ phù hợp ở những cự ly truyền dữ liệu ngắn. Do vậy, muốn truyền dữ liệu trên một cự ly thông tin xa, điều cần thiết để có tính kinh tế cao là biến đổi luồng dữ liệu song song thành nối tiếp.

Lúc đó, các bit dữ liệu sẽ xuất hiện tuần tự sau mỗi chu kỳ T_B . Tuy nhiên, để không làm giảm tốc độ truyền tin và làm tăng lưu lượng và qua đó tăng dung lượng thông tin, dãy dữ liệu nối tiếp tại lối ra của MUX được tăng tốc độ truyền lên đáng kể nhờ giảm chu kỳ bit T_B .

Hình 6.11 đưa ra một ví dụ sử dụng bộ dồn kênh 74151 chuyển một dữ liệu song song 8 bit (từ X_0 đến X_7) thành dữ liệu nối tiếp các bit này. Dữ liệu song song được nạp sẵn trong 1 thanh ghi lưu trữ dữ liệu loại 8 bit (ví dụ 74164/74165) và được đưa trực tiếp tới 8 lối vào của 74151. Mã địa chỉ chọn được tạo ra từ một bộ đếm nhị phân 3 bit (chia đôi liên tiếp 3 lần dãy xung nhịp T_B tạo ra chu kỳ của dữ liệu nối tiếp tại lối ra z).



Hình 6.11. Chuyển đổi luồng dữ liệu song song đã có trong một thanh ghi 8 bit (ví dụ 74164) thành dãy bit nối tiếp có chu kỳ T_B nhờ MUX 8 đầu vào 74151

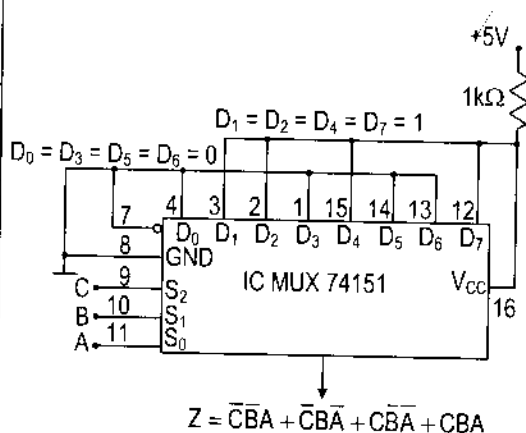
8 trạng thái ra của bộ đếm lần lượt là $S_2S_1S_0 = 000$ đến 111, tương ứng với chỉ số của bit dữ liệu $D_i = X_i$ được chọn lần lượt và xuất hiện tuần tự tại lối ra z.

6.4.3. Tạo hàm logic dạng đầy đủ trực tiếp dùng MUX

Khi coi các đầu vào chọn là các biến logic tương ứng (loại 74150 cho ta tạo hàm 4 biến, loại 74151 cho phép tạo hàm 3 biến). Mỗi đầu vào dữ liệu khi đó được nối thường xuyên tới hoặc mức 1 hoặc mức 0 tùy theo nội dung bảng chân lý. Để làm rõ, xét ví dụ sau: Hàm logic 3 biến ta muốn có bảng chân lý cho trên hình 6.12a.

Đầu vào chọn			Đầu ra	Đầu vào dữ liệu của MUX
C (S_2)	B (S_1)	A (S_0)	Z	
0	0	0	0	D_0
0	0	1	1	D_1
0	1	0	1	D_2
0	1	1	0	D_3
1	0	0	1	D_4
1	0	1	0	D_5
1	1	0	0	D_6
1	1	1	1	D_7

a)



b)

**Hình 6.12a) Bảng chân lý hàm Z;
b) Mạch logic dùng IC 74151 tạo hàm Z**

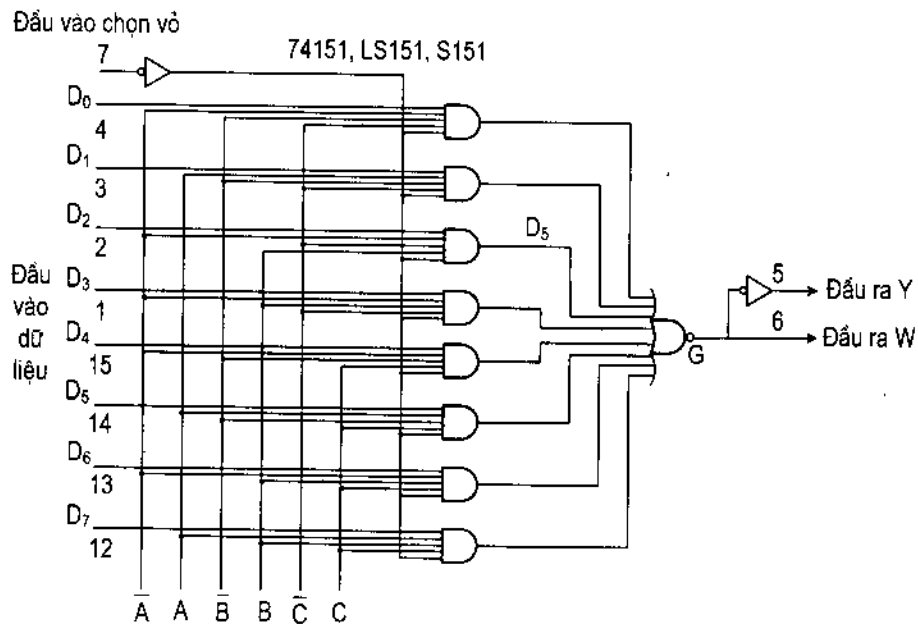
Cách đấu nối các trạng thái logic của lối vào dữ liệu cho trên hình 6.12b.

Các biến logic được đặt tại các đầu vào chọn $S_2 \equiv C$; $S_1 \equiv B$ và $S_0 \equiv A$.

Để tạo một hàm logic 4 biến dạng có các mintec đầy đủ, cần dùng loại MUX 74150 như đã có trên hình 6.7 với phương pháp làm tương tự như ví dụ vừa nêu.

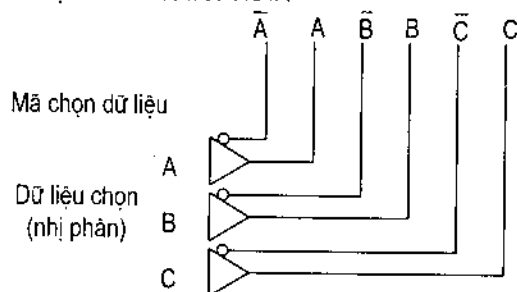
6.5. CÂU HỎI VÀ BÀI TẬP

1. Bộ dồn kênh có nhiệm vụ gì? Bộ dồn kênh số có đặc điểm gì khác so với bộ dồn kênh tổng quát? Chức năng của các đầu vào chọn là gì?
2. Một bộ MUX cần chọn 1 trong 32 hoặc chọn 1 trong 64 đầu vào dữ liệu đặt tới lối ra. Hãy tính tổng số các đầu vào của hai loại MUX này. Nếu được chế tạo dưới dạng một vi điện tử thì chúng sẽ có tổng cộng bao nhiêu chân nối ra ngoài?
3. Hãy xây dựng một bộ MUX gồm 32 dữ liệu vào ($D_0 \rightarrow D_{31}$) dùng các IC loại 74150 như đã cho trên hình 6.7 với bộ mã chọn $S_4S_3S_2S_1S_0$ (gồm 5 bit nhị phân).
Nếu chỉ có loại 74151 sẽ cần bao nhiêu IC? Hãy vẽ mạch thực hiện (cần bổ sung thêm một vài cổng logic cơ bản để chọn vỏ IC làm việc).
4. Có hai luồng dữ liệu mỗi luồng 8 bit cần được qua một bộ MUX đưa tới một đường truyền tín hiệu số. Tại một thời điểm nhất định ở cổng ra của MUX chỉ được phép xuất hiện một trong hai dữ liệu này (dạng song song).
Cấu trúc MUX sẽ có tổng cộng bao nhiêu đầu vào dữ liệu? Có bao nhiêu đầu ra và bao nhiêu đầu vào chọn?
Hãy vẽ cấu trúc khối thực hiện nhiệm vụ trên.
5. Một luồng dữ liệu 4 bit song song muốn được chuyển thành dãy 4 bit nối tiếp. Hãy xây dựng cấu trúc khối và cấu trúc mạch nguyên lý thực hiện bộ MUX loại này (dữ liệu vào là $D_3D_2D_1D_0 = 2^32^22^12^0$ trong hệ 16).
6. Trong bài tập số 5 nếu luồng dữ liệu là 16 bit song song, hãy xây dựng cấu trúc sơ đồ khối thích hợp cho bộ MUX thực hiện nhiệm vụ chuyển đổi song song – nối tiếp này.
7. Hãy dùng IC 74151 (hình 6.13) xây dựng các hàm logic sau:
 - a) $Z = D + A\bar{B} + \bar{C}D$;
 - b) $F_1 = \sum m(1, 2, 3, 4, 5, 6)$;
 - c) $F_2 = \sum m(0, 3, 4, 7, 8, 11, 12, 15)$.
8. Hãy dùng IC 74150 (xem hình 6.7) xây dựng các hàm logic đã cho ở bài tập 7 và nhận xét kết quả cấu trúc mạch giữa hai cách dùng hai loại IC đã làm.

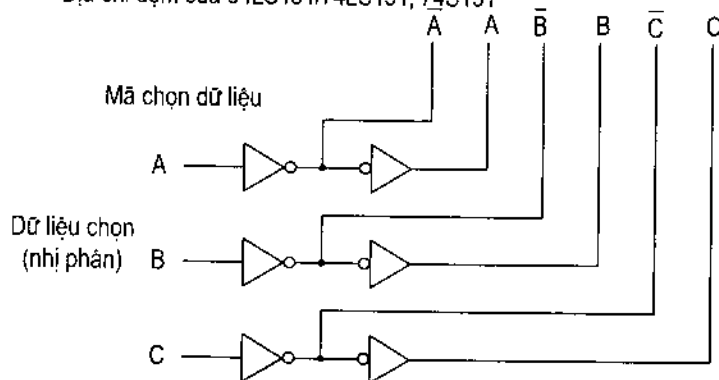


Xem địa chỉ đếm ở dưới:

Địa chỉ đếm của 54151A/74151A



Địa chỉ đếm của 54LS151/74LS151, 74S151



Hình 6.13. Cấu tạo IC MUX 74151

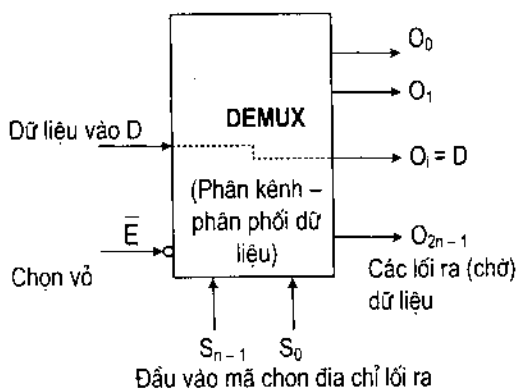
BÀI 7. MẠCH PHÂN KÊNH (DEMUX)

7.1. KHÁI NIỆM

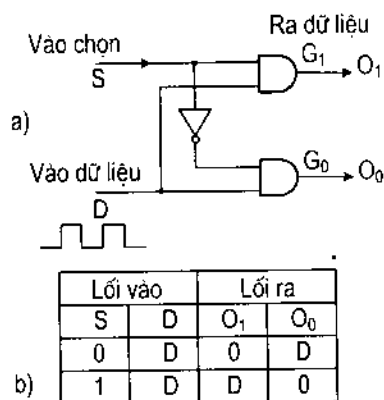
Nhiệm vụ của mạch phân kênh (DEMUX) ngược lại với nhiệm vụ của mạch dồn kênh (MUX) đã xét. Nghĩa là mạch điện của bộ phân kênh có nhiều lối ra chờ sẵn để đón một dữ liệu duy nhất đặt tới lối vào. Theo mã địa chỉ chọn tác động tới các đầu vào điều khiển việc chọn, đầu ra nào có chỉ số mã lối ra của nó trùng với mã chọn thì được quyền tiếp đón dữ liệu vào. Để điều khiển nhiều bộ DEMUX hoạt động qua đó quản lý (mở rộng) được nhiều lối ra hơn, các tín hiệu chọn vô được sử dụng và được coi là nhóm bit địa chỉ chọn bậc cao hơn bổ sung cho nhóm bit chọn đã có sẵn.

7.2. CẤU TRÚC KHỐI CỦA MẠCH PHÂN KÊNH

Cấu trúc khối của một mạch phân kênh được cho trên hình 7.1 và mạch điện thực hiện mạch phân kênh có 2 lối ra (đơn giản nhất) được cho trên hình 7.2a với bảng chân lý 7.2b.

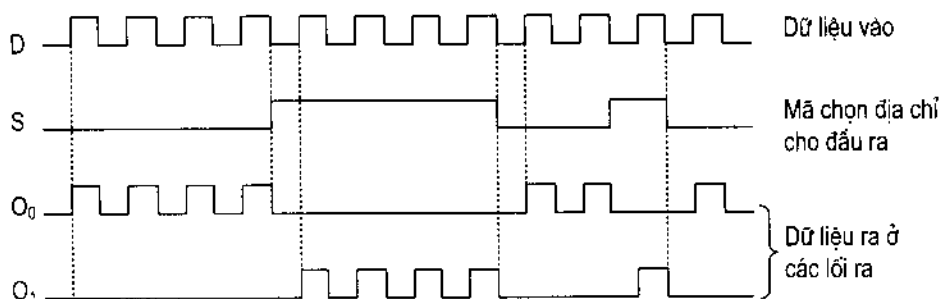


Hình 7.1. Cấu trúc khối tổ chức hoạt động của DMUX



Hình 7.2. Mạch DMUX hai lối ra dùng các cổng logic cơ bản (a) và bảng chân lý (b)

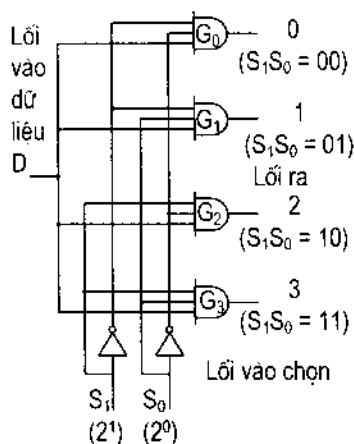
Có thể minh họa hoạt động của mạch hình 7.2b nhờ đồ thị thời gian hình 7.3.



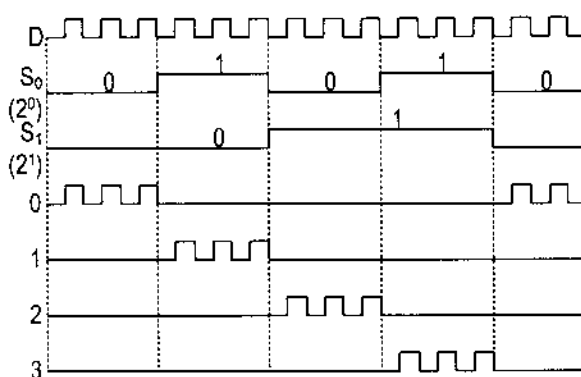
Hình 7.3. Đồ thị thời gian minh họa hoạt động của DMUX hình 7.2b

7.3. MẠCH PHÂN KÊNH 4 LỖI RA

Mạch phân kênh 4 lỗi ra có cấu trúc cho trên hình 7.4 và đồ thị thời gian minh họa hoạt động của mạch cho trên hình 7.5.



Hình 7.4. Cấu trúc DMUX có 4 đường ra



Hình 7.5. Đồ thị thời gian minh họa hoạt động của mạch DMUX hình 7.4

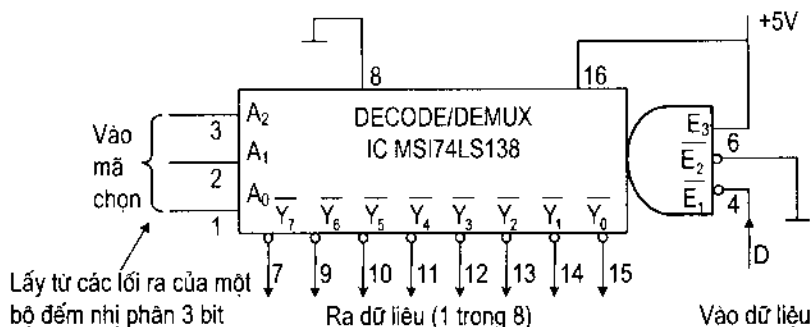
Chú ý rằng mã chọn S_1S_0 tuân theo quy luật nhị phân S_1 cấp 2^1 ; S_0 cấp 2^0 , qua mạch hình 7.4, dòng dữ liệu là dãy xung D được phân phối từng nhóm đến các lỗ ra tương ứng có chỉ số phù hợp với mã chọn S_1S_0 .

7.4. MẠCH PHÂN KÊNH 8 LỐI RA

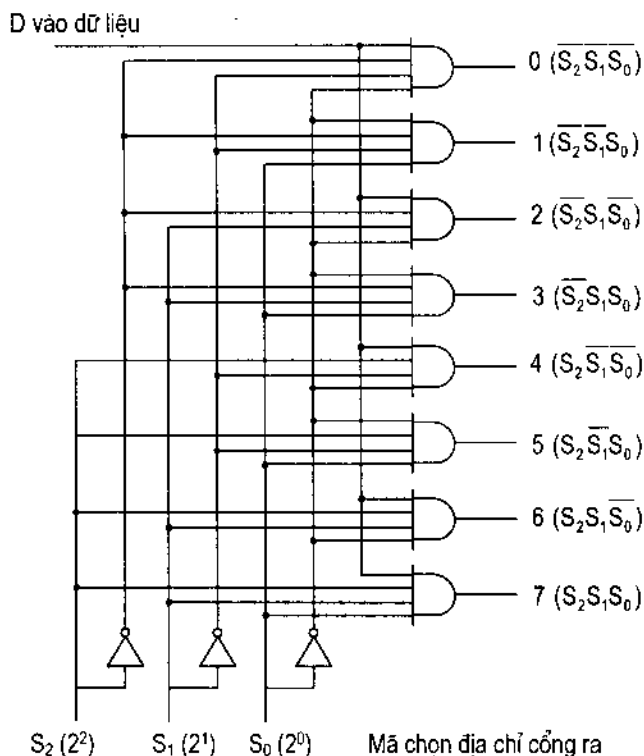
Mạch phân kênh 8 lối ra dùng 3 mã chọn $S_2S_1S_0$ được cho trên hình 7.7 với đồ thị thời gian minh họa cho trên hình 7.8.

Các mạch điện giải mã từ hệ 2 sang hệ 8 (chọn 1 trong 8) có chức năng tương đương mạch DEMUX 8 lối ra. Từ đó, có thể sử dụng các mạch giải mã làm nhiệm vụ DEMUX ví dụ IC 74138 đã cho trên hình 5.6 (bài 5).

Mạch hình 7.6 đưa ra cách sử dụng IC giải mã 1 trong 8 làm nhiệm vụ một bộ DMUX 1 đường vào dữ liệu, chọn 1 trong 8 đường ra dữ liệu.

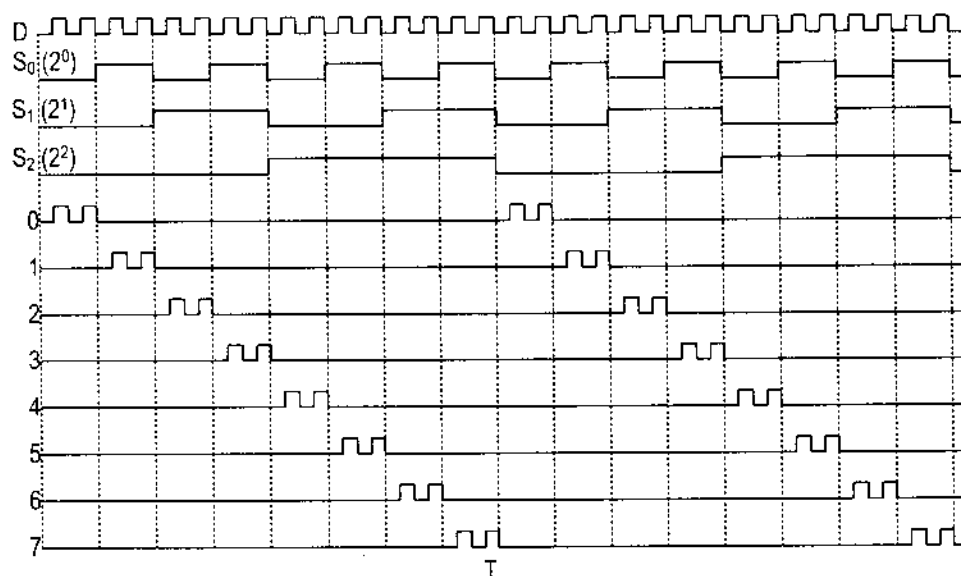


Hình 7.6. Mạch điện dùng IC 74LS138 thực hiện bộ DMUX 8 đường ra



Hình 7.7. Mạch phân kênh 8 đường ra với 3 mã chọn đường

Nếu mã chọn $S_2S_1S_0$ tuần hoàn với chu kỳ lặp lại là T thì sau một chu kỳ, dữ liệu được chọn lại xuất hiện ở cổng ra 0 ứng với mã $\overline{S_2S_1S_0}$.



Hình 7.8. Đồ thị thời gian minh họa hoạt động của mạch hình 7.6.

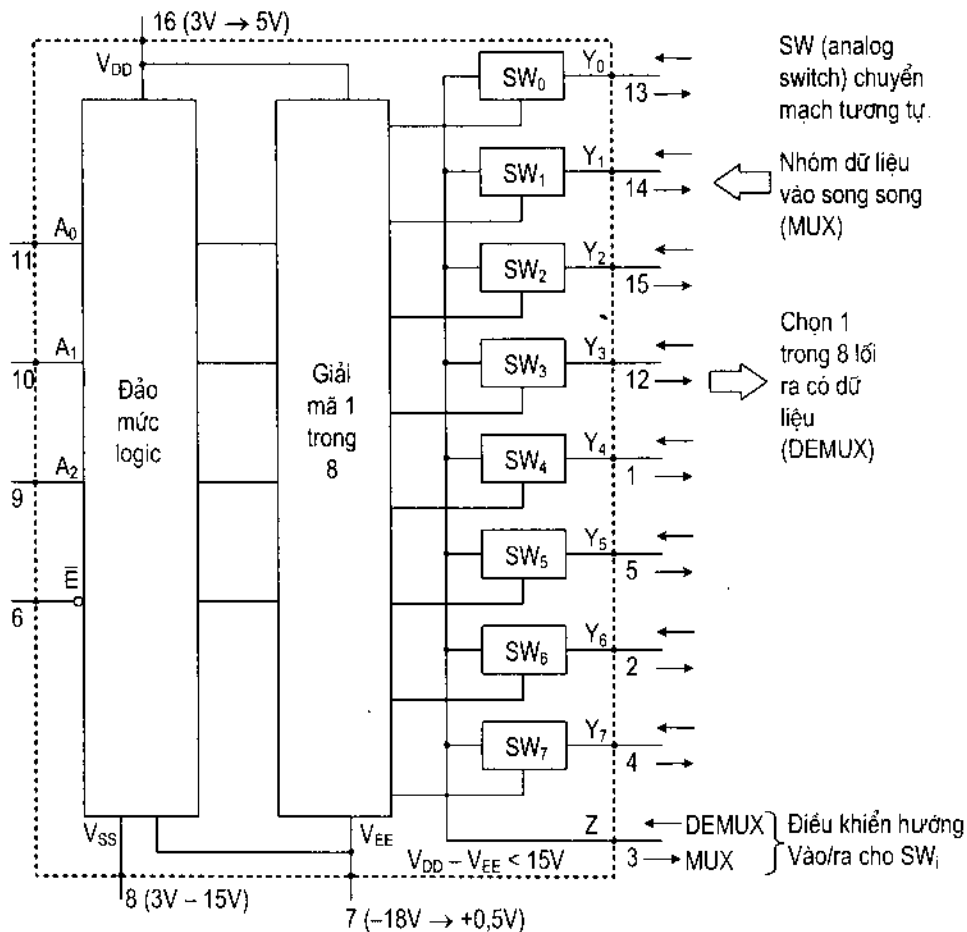
7.5. MẠCH DẪN KÊNH/PHÂN KÊNH TƯƠNG TỰ (MUX/DEMUX) DÙNG CÁC CỔNG IC CMOS

– Nhóm IC CMOS loại HEF4051B/4052B/4053B có thể thực hiện một trong hai nhiệm vụ MUX hoặc DEMUX khi đổi lẫn các lối vào và lối ra dữ liệu cho nhau.

Nếu thực hiện chức năng DEMUX, lối ra duy nhất được sử dụng làm lối vào dữ liệu và nhóm lối vào trước đây chuyển chức năng thành các lối ra chọn (ví dụ 1 trong 8) đón dữ liệu xuất hiện trên nó theo địa chỉ mã chọn.

– Nhóm HEF4051B/4052B là loại có 8 lối vào (MUX) ra (DEMUX) còn nhóm HEF4053B là loại có 2 nhóm lối vào 3 bit và 3 lối ra đồng thời cho mỗi nhóm (kiểu MUX) hoặc ngược lại (kiểu DEMUX).

Sơ đồ khối chức năng của HEF4051B được cho trên hình 7.9, mạch điện của một khóa chuyển mạch analog hai hướng cho trên hình 7.10.



Hình 7.9. Sơ đồ khối cấu trúc chức năng của IC CMOS HEF4051B
(các chữ số thập phân ghi bên ngoài là vị trí chân)

Tín hiệu số sau khi được giải mã 1 trong 8 điều khiển chuyển mạch analog hai chiều IN/OUT (SW_{*i*}) tại lối vào/ra tương ứng. Ở đây, chỉ số (*i*) của SW_{*i*} phù hợp với giá trị nhị phân của mã chọn A₂A₁A₀ trong điều kiện $\bar{E} = 0$ (xem bảng chức năng của IC HEF4051B cho trên bảng 7.1).

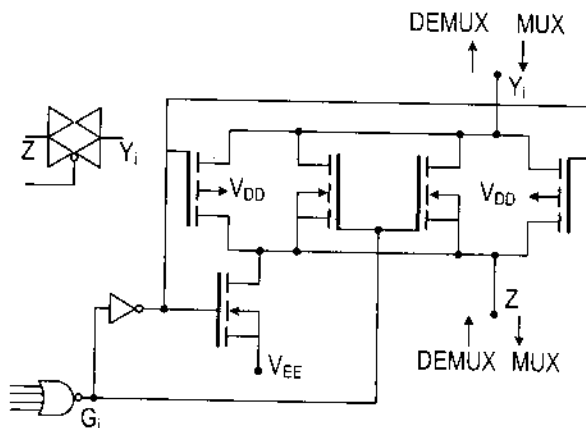
$$(i)_8 = (A_2A_1A_0)_2.$$

Đặc điểm khác nhau căn bản giữa hai nhóm IC loại giải mã (DMUX) 74138 đã phân tích trước và loại CMOS 4051B về chức năng là nhóm linh kiện CMOS có bổ sung các SW kiểm soát luồng dữ liệu giữa các cổng Z – Y_{*i*}. Do đó, có thể dùng đa năng với hai nhiệm vụ không đồng thời: MUX/DEMUX.

Trong các mạch chi tiết, một chuyển mạch SW_i có cấu tạo chi tiết và được ký hiệu thu gọn như trên hình vẽ (7.10).

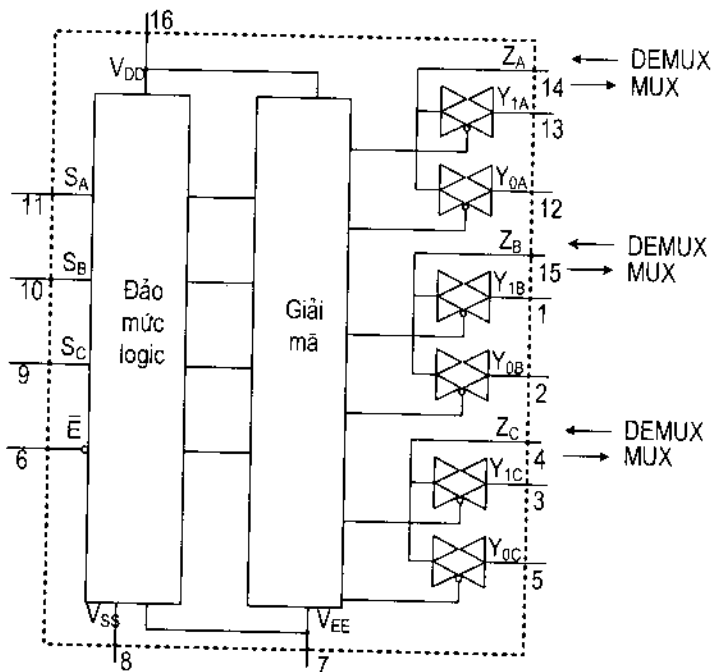
Bảng 7.1. Bảng chức năng của IC HEF 4051B

Mã vào chọn				Kênh được nối
$\bar{E}$	A_2	A_1	A_0	mạch
1	X	X	X	Bị cấm
0	0	0	0	$Y_0 - Z$
0	0	0	1	$Y_1 - Z$
0	0	1	0	$Y_2 - Z$
0	0	1	1	$Y_3 - Z$
0	1	0	0	$Y_4 - Z$
0	1	0	1	$Y_5 - Z$
0	1	1	0	$Y_6 - Z$
0	1	1	1	$Y_7 - Z$



Hình 7.10. Một trong 8 chuyển mạch analog (SW_i) kiểm soát IN/OUT của IC HEF4051B và ký hiệu quy ước của nó

- Hình 7.11 là cấu trúc khối của IC CMOS HEF4053B. Thực hiện MUX/DEMUX hai nhóm dữ liệu 3 bit.

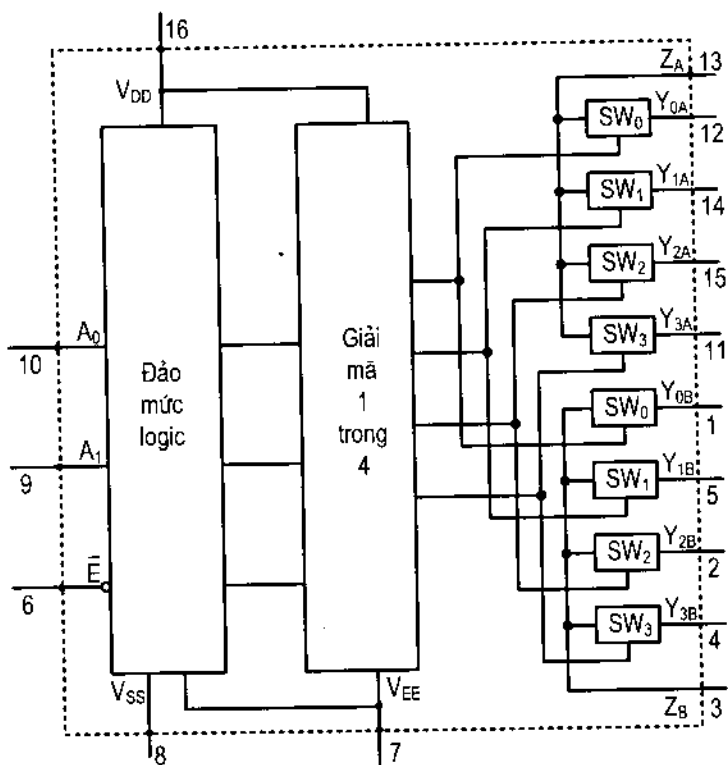


Hình 7.11. Cấu trúc khối của IC HEF4053B

Các đặc điểm của IC HEF4053B:

- Chứa 6 chuyển mạch analog SW kiểm soát kênh vào/ra của dữ liệu theo các cổng vào/ra tương ứng.
- Phương thức làm việc kiểu song song cho các dữ liệu của một nhóm và nối tiếp giữa hai nhóm dữ liệu cho việc MUX dữ liệu.

IC 4052B thuộc nhóm có 8 lối vào chia thành 2 nhóm, 4 cặp bit có 8 chuyển mạch analog kiểm soát vào/ra khi thực hiện chức năng MUX/DEMUX. Cấu trúc chức năng khối được cho trên hình 7.12.



Hình 7.12. Cấu trúc khối chức năng của IC HEF 4052B và bố trí vị trí các chân

Các chức năng cơ bản của IC 4052B và 4053B được thể hiện trên các bảng 7.3 và 7.2 tương ứng. Các kênh vào/ra của nó được nối nhau liên thông theo hai nhóm mã lệnh: nhóm mã lệnh cho phép làm việc $\bar{E}$ và nhóm lệnh chọn địa chỉ đường nối liên thông (địa chỉ chuyển mạch analog) theo mã chọn áp tới lối vào (loại mã tác động đồng thời).

Bảng 7.2. Bảng chức năng của IC 4053B (kỳ tự x chọn lần lượt là A, B và C)

Mã vào		Kênh được nối mạch
$\bar{E}$	S_x	
0	0	$Y_{0x} - Z$
0	1	$Y_{1x} - Z$
1	X	Bị cấm

Bảng 7.3. Bảng chức năng của IC 4052B

Mã vào			Kênh được nối mạch
$\bar{E}$	A_1	A_0	
0	0	0	$Y_{0A} - Z_A; Y_{0B} - Z_B$
0	0	1	$Y_{1A} - Z_A; Y_{1B} - Z_B$
0	1	0	$Y_{2A} - Z_A; Y_{2B} - Z_B$
0	1	1	$Y_{3A} - Z_A; Y_{3B} - Z_B$
1	X	X	Bị cấm hoạt động

7.6. CÂU HỎI VÀ BÀI TẬP

1. Hãy nêu nhiệm vụ của DMUX và giải thích các đặc điểm khác biệt cơ bản giữa MUX và DMUX.

Khi kết luận rằng "Mạch DMUX giống hệt mạch giải mã". Đúng hay sai?

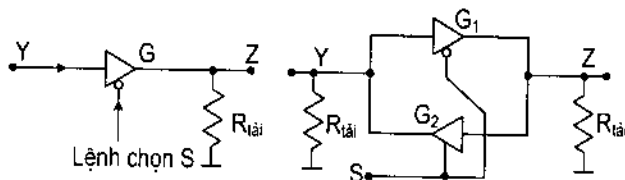
2. Xây dựng một mạch DMUX vào 1 đường dữ liệu, ra 1 trong 4 đường.
 - a) Trường hợp sử dụng các cổng cơ bản NAND và sử dụng các cổng cơ bản NOR.
 - b) Trường hợp sử dụng IC 74138 đã có.

3. Bộ giải mã IC 7442 (hình 5.5) không có đầu vào cho phép E, có thể dùng IC này thực hiện giải mã 1 trong 8 được không? (không sử dụng các lối ra $\bar{Y}_8, \bar{Y}_9$). Có thể dùng lối vào dữ liệu D của IC làm nhiệm vụ đầu vào cho phép được không, nêu cách thực hiện nếu được?

Khi dùng IC 7442 làm nhiệm vụ DMUX (vào 1 dữ liệu, ra 1 trong 8) phải sử dụng các đầu vào dữ liệu và vào chọn của nó như thế nào?

4. Hãy sử dụng phương pháp tương tự như hình 7.6 (thực hiện DMUX 8 đường ra) xây dựng mạch thực hiện DMUX 16 đường ra dùng IC MSI 74154 đã cho trên hình 5.9.

5. Hãy giải thích nhiệm vụ cơ bản của một chuyển mạch analog G theo hai mô hình thu gọn cho trên hình 7.13.



Hình 7.13. Bài tập 5

Ở đây Y/Z là các đầu vào ra dữ liệu (tương tự hoặc số).

S là đầu vào điều khiển có hai giá trị nhị phân.

Mô hình nào ở trên áp dụng được cho mạch hình 7.10.

6. a) Trong mạch hình 7.8 muốn thực hiện việc truyền 1 byte dữ liệu từ dạng song song chuyển sang dạng nối tiếp sau 8 nhịp hoạt động. Hãy thiết lập bộ mã chọn và xây dựng mạch thực hiện (tương tự như mạch hình 6.11).
b) Có thể sử dụng chỉ hai hay ba đường truyền $Z \rightarrow Y_i$ ($i = 0 \rightarrow 7$) trong mạch hình 7.9 được không? Vì sao?
7. Có 1 byte dữ liệu (8 bit) muốn tách nó thành các bit chẵn (0, 2, 4, 6) và các bit lẻ (1, 3, 5, 7) muốn dùng IC HEF 4052 thực hiện có được không?

Nếu được hãy viết bộ mã điều khiển và mạch thực hiện (dùng bộ đếm nhị phân 2 bit).

BÀI 8. MẠCH LOGIC TUẦN TỰ

8.1. KHÁI NIỆM VỀ MẠCH TRIGƠ (Flip Flop – FF)

Theo định nghĩa, mạch lật là loại mạch có đầu ra có thể thay đổi trạng thái một cách đột biến. Một mạch lật có hai trạng thái ra ổn định được gọi là một mạch Trigơ (mạch FF), các trạng thái của FF khi không có sự điều khiển (tác động) đặc biệt nào ở các lối vào thì sẽ không thay đổi theo thời gian.

Một mạch FF thường có hai đầu ra và có ít nhất hai đầu vào thực hiện các chức năng tối thiểu sau:

1. Tại hai lối ra luôn có hai trạng thái ra ký hiệu là Q và $\bar{Q}$, có tính ổn định bền vững, có tính liên hợp (bù logic của nhau), $Q = \overline{(\bar{Q})}$, (khi $Q = 1$ thì $\bar{Q} = 0$ hoặc ngược lại).

2. Các trạng thái ở đầu ra không chỉ phụ thuộc vào các trạng thái tác động ở các đầu vào lúc đang xét mà còn phụ thuộc vào trạng thái trước đó (quá khứ) của chính các lối ra. Tức là phải quan tâm tới câu hỏi: Trước khi tác động các tín hiệu vào, Q đang ở trạng thái nào?

3. Có khả năng tiếp nhận (ghi vào), lưu trữ một giá trị nhị phân 1 hay 0 và khi có yêu cầu sẽ xuất ra (đọc) chính dữ liệu này. Trong thời gian "nhớ" bit dữ liệu không được tác động bất kỳ tín hiệu nào tới lối vào và nhìn chung phải duy trì nguồn một chiều nuôi mạch.

4. Việc mở rộng chức năng của FF được thực hiện dễ dàng nhờ có thêm các đầu vào điều khiển hành vi của FF ngoài các đầu vào dữ liệu đã nói tới với hai mục đích quan trọng là thiết lập cấp tốc một dữ liệu mới quan trọng hơn trong khi dữ liệu cũ (đang nhớ) không còn cần nữa hoặc cần thực hiện các thủ tục nhập/xuất dữ liệu cho FF cùng nhịp với các mạch khác để phối hợp công việc truyền tải xử lý dữ liệu.

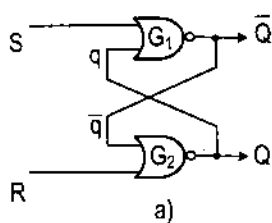
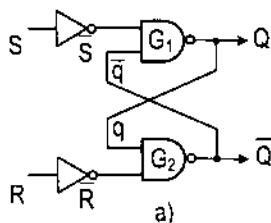
Trong các tính chất đã nêu trên, tính chất 2 là đặc trưng nhất cho dạng mạch FF và từ đó chúng được gọi là mạch logic tuần tự (logic dãy: Sequential logic Circuit) để phân biệt với nhóm mạch logic tổ hợp đã xét ở các bài trước. Tính chất 1 là quan trọng nhất vì loại mạch này đảm bảo khâu lưu trữ dữ liệu nhị phân trong mọi quá trình xử lý tín hiệu số.

Việc phân loại FF thường dựa theo cách gọi tên các đầu vào như RSFF, DFF, MSFF, TFF hay JKFF... sẽ được đề cập ở phần tiếp theo.

8.2. MẠCH RSFF

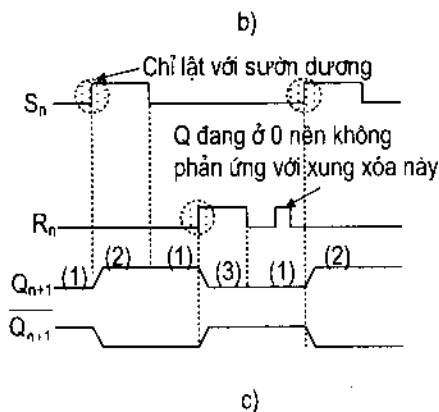
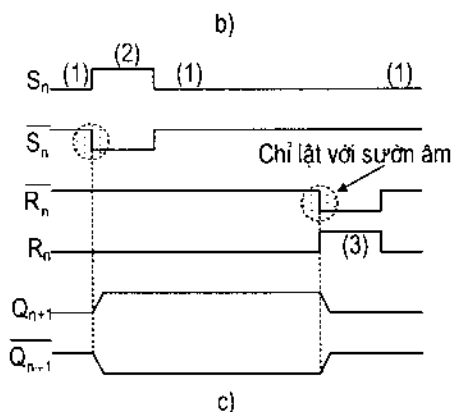
8.2.1. Mạch RSFF không đồng bộ

Mạch RSFF không đồng bộ (không cần xung nhịp điều khiển cho phép hoạt động) có hai cấu trúc cơ bản được cho trên hình 8.1 và 8.2 cùng với các bảng chân lý mô tả chức năng và đồ thị xung minh họa tương ứng.



Đầu vào		Đầu ra	Ý nghĩa lối ra
S_n	R_n	Q_{n+1}	Trạng thái tương lai
0	0	Q_n	Không đổi (1)
0	1	0	Xóa về 0 (Reset) (3)
1	0	1	Lập mức 1 (Set) (2)
1	1	X	Cấm xảy ra ($R_n.S_n = 0$)

Đầu vào		Đầu ra	Ý nghĩa lối ra
S_n	R_n	Q_{n+1}	Trạng thái tương lai
0	0	Q_n	Không đổi (1)
0	1	0	Xóa về 0 (Reset) (3)
1	0	1	Lập mức 1 (Set) (2)
1	1	X	Cấm xảy ra ($R_n.S_n = 0$)



Hình 8.1a) Mạch RSFF dùng hai cổng NAND đầu chéo và hai cổng NOT ở lối vào;
b) Bảng chân lý và ý nghĩa các trạng thái ra;
c) Biểu đồ xung minh họa chức năng (các dòng trên bảng chân lý)

Hình 8.2a) Mạch RSFF dùng hai cổng NOR đầu chéo;
b) Bảng chân lý và ý nghĩa các trạng thái ra (tương tự như bảng 8.1b);
c) Biểu đồ xung minh họa chức năng các dòng trên bảng chân lý)

Chú ý: Ở hình 8.1b: n là chỉ số hiện tại; $n + 1$ chỉ số tương lai; $S_n \cdot R_n = 0$ là điều kiện cấm.

Ở hình 8.1c: phản ứng lật của Q (hay $\bar{Q}$) chỉ xảy ra khi $\bar{S}_n$ hoặc $\bar{R}_n$ có sườn xung đi xuống (từ mức cao về mức thấp).

Đầu ra Q nằm ở lối ra của G_1 .

Ở hình 8.2c: phản ứng lật của Q (hay $\bar{Q}$) chỉ xảy ra khi S_n hoặc R_n có sườn xung đi lên (từ mức thấp lên mức cao).

Đầu ra Q nằm ở lối ra của G_2 .

Ba nhận xét quan trọng về tính chất của RSFF là:

1. Phản ứng lật của trạng thái ra Q chỉ xảy ra khi các biến vào S hay R chuyển trạng thái từ cao xuống thấp ($1 \rightarrow 0$) với loại mạch dùng hai cổng NAND đầu chéo và chuyển từ trạng thái thấp lên trạng thái cao ($0 \rightarrow 1$) với loại mạch làm từ hai cổng NOR đầu chéo.

2. Với loại FF hình 8.1a đầu ra Q lấy tại lối ra của cổng NAND trên G_1 trong khi điều đó là tại lối ra của cổng NOR dưới G_2 với mạch hình 8.2a.

3. Điều kiện cấm chung cho hai loại là $SR = 0$ tức là không được phép xảy ra trạng thái $S = R = 1$ (mạch hình 8.1a cần chú ý có hai cổng NOT tại lối vào, nếu tính lối vào trực tiếp $\bar{S}$ và $\bar{R}$ thì điều kiện cấm là không xảy ra trạng thái: $\bar{S} = \bar{R} = 0$ như trên đồ thị 8.1c minh họa).

– Đầu vào R có tên là đầu vào xóa (đưa Q về 0 – Reset), đầu vào S có tên là đầu vào thiết lập (đưa Q lên 1 – Set).

– Phương trình trạng thái logic của các mạch RSFF xét trực tiếp từ mạch điện là:

$$\left. \begin{aligned} Q_{n+1} &= \overline{\overline{S}_n q_n} = S_n + q_n \\ \overline{Q}_{n+1} &= \overline{\overline{R}_n q_n} = R_n + \overline{q_n} \end{aligned} \right\} \quad (8.1)$$

$$\left. \begin{aligned} \overline{Q}_{n+1} &= \overline{S_n + q_n} = \overline{S_n} \cdot \overline{q_n} \\ Q_{n+1} &= \overline{R_n + q_n} = \overline{R_n} \cdot \overline{q_n} \end{aligned} \right\} \quad (8.2)$$

Nếu viết phương trình từ bảng chân lý, ta có:

$$\left. \begin{aligned} Q_{n+1} &= Q_n \overline{R_n} \overline{S_n} + \overline{R_n} S_n \\ S_n \cdot R_n &= 0 \end{aligned} \right\} \quad (8.3)$$

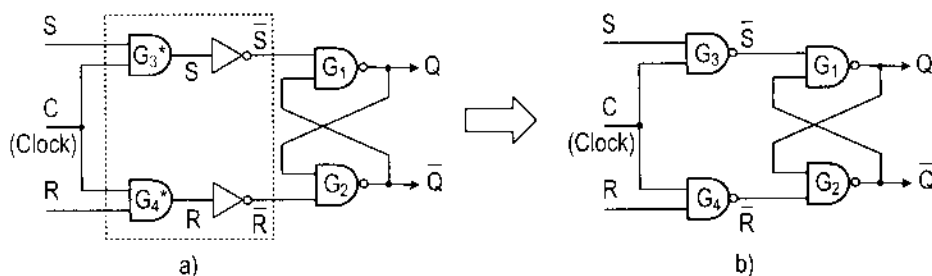
Đơn giản hệ (8.3) bằng cách cộng hai vế và một vài phép biến đổi, nhận được hệ thức cuối cùng (8.4) gọi là phương trình đặc tính của RSFF không đồng bộ:

$$Q_{n+1} = S_n + \overline{R_n} Q_n \quad (8.4)$$

8.2.2. RSFF đồng bộ

– Thực tế, nếu chỉ có trực tiếp hai lối vào S và R thì RSFF được dùng trong tình trạng không biết trước được lúc khởi động (cấp nguồn một chiều cho mạch) Q đang bằng 1 hay 0 và còn cần phải kiểm soát được trạng thái ra hiện tại và làm thay đổi nó theo ý muốn trong một mối liên hệ logic với các phần tử khác. Tức là trạng thái ra chỉ lật khi thỏa mãn điều kiện cần là phải có một tín hiệu chuẩn (tín hiệu cho phép) đưa tới gọi là tín hiệu đồng bộ hay nhịp (clock – C) và điều kiện đủ là có các tín hiệu dữ liệu S hay R ở lối vào. Khi đầu vào cho phép (đầu vào nhịp) không ở mức logic tích cực hay không ở trạng thái logic tích cực thì không thể thay đổi dữ liệu (bit) đã lưu trong RSFF. Chỉ khi đầu vào C chuyển sang tích cực mới có thể nhập và lưu dữ liệu mới FF làm việc theo phương thức trên được định nghĩa là FF đồng bộ.

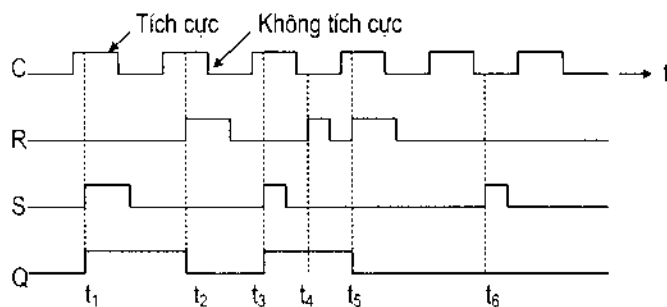
– Mạch điện của một RSFF đồng bộ có được từ ví dụ mạch hình 8.1 có bổ sung thêm hai cổng NAND tại các lối vào S và R (hình 8.3).



**Hình 8.3. Mạch điện logic của một RSFF đồng bộ
(C điều khiển theo mức logic cao)**

Khi $C = 1$: RSFF (không đồng bộ) được phép làm việc.

Khi $C = 0$: Mạch bị cấm hoạt động vì G_3G_4 không tiếp nhận dữ liệu mới do bị cấm.

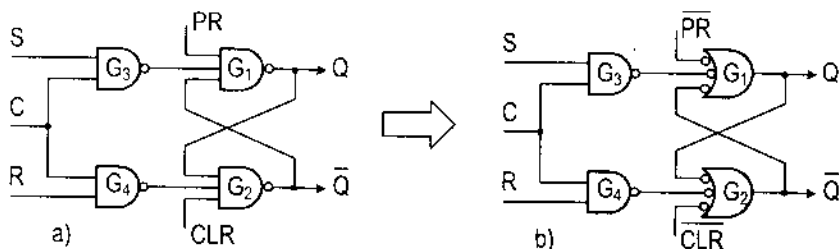


Hình 8.4. Đồ thị thời gian minh họa hoạt động của RSFF chế độ có tín hiệu đồng bộ (cho phép) C loại điều khiển theo mức điện thế cao

Tại các thời điểm t_1, t_2, t_3 do $C = 1$ nên RSFF tiếp nhận dữ liệu S hoặc R. Còn tại t_4 hoặc t_6 do $C = 0$ nên dữ liệu mới không được tiếp nhận RSFF vẫn lưu lại ở trạng thái cũ ngay trước t_4 hoặc ngay trước t_6 (hình 8.4).

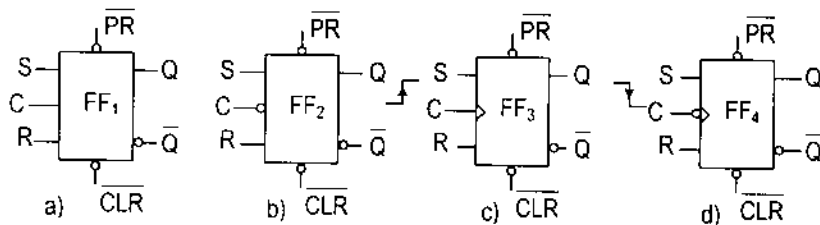
8.2.3. Các đầu vào điều khiển trực tiếp $\overline{PR}$ và $\overline{CLR}$ (thiết lập trực tiếp – Preset và xóa trực tiếp – Clear)

Nếu bổ sung thêm một lối vào nữa cho G_1 và G_2 (đưa trực tiếp ra ngoài được gọi là các lối vào điều khiển trực tiếp) thì điều khiển Q lên 1, khi $\overline{CLR} = 0$ sẽ điều khiển Q về 0. Hành vi này không phụ thuộc vào tại thời điểm áp các mức PR hoặc CLR có hay không có tín hiệu đồng bộ C và không phụ thuộc vào các giá trị tín hiệu dữ liệu S hay R (hình 8.5). Cần chú ý là các mức $\overline{PR}$ hoặc $\overline{CLR}$ phải không đồng thời xuất hiện mức 0.



Hình 8.5. Vị trí các đầu vào xóa $\overline{CLR}$ và đầu vào thiết lập $\overline{PR}$ điều khiển trực tiếp mức tích cực thấp (G_1 và G_2 vẽ theo logic tương đương của NAND 3 lối vào)

– Có hai dạng điều khiển C là theo mức thế áp vào (1 hay 0) hoặc theo sườn xung C đi lên ($0 \rightarrow 1$) hay đi xuống ($1 \rightarrow 0$) đặt tới lối vào C, theo đó có các ký hiệu quy ước chuẩn cho một RSFF với lối vào C và các lối vào, ra như thể hiện trên hình 8.6 (chú ý rằng các ký hiệu đầu vào $\overline{PR}$, $\overline{CLR}$ và C trên hình 8.6 cũng áp dụng cho mọi dạng FF khác sẽ xét tới ở phần sau).



Hình 8.6. Ký hiệu đầu vào điều khiển trực tiếp $\overline{PR}$ và $\overline{CLR}$ (tích cực ở mức 0 áp vào) và ký hiệu đầu vào nhịp với 4 phương thức điều khiển đồng bộ khác nhau:

a) FF_1 : cần mức cao (mức 1) của xung nhịp C áp vào;

b) FF_2 : cần mức thấp (mức 0) của xung nhịp C áp vào;

c) FF_3 : cần sườn đi lên của xung nhịp ($0 \rightarrow 1$);

d) FF_4 : cần sườn đi xuống của xung nhịp ($1 \rightarrow 0$).

– Khi có thêm đầu vào C tích cực ở mức cao. Bảng chân lý hình 8.1 thêm một cột biến vào C như hình 8.7 thể hiện.

Các đầu vào			Đầu ra	Ý nghĩa:
C_n	S_n	R_n	Q_{n+1}	
0	0	0	Q_n	Bảo lưu trạng thái (không cho phép nhập dữ liệu mới chờ ở cổng S và R)
	0	1	Q_n	
	1	0	Q_n	
	1	1	Q_n	
1	0	0	Q_n	Hoạt động như RSFF không đồng bộ
	0	1	0	
	1	0	1	
	1	1	X	

Hình 8.7. Bảng trạng thái của RSFF chế độ đồng bộ (theo mạch hình 8.3)

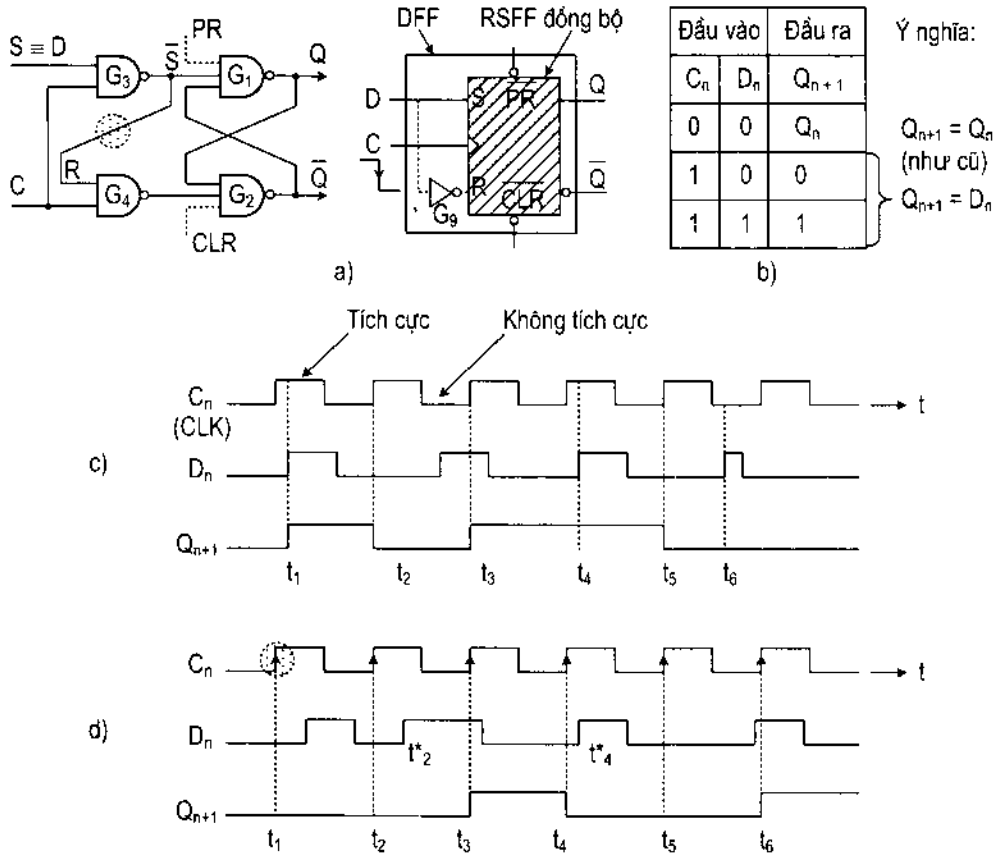
8.3. TRIGƠ TRỄ ĐỒNG BỘ (Delay Flip Flop – DFF)

– Trong các mạch hình 8.3 hoặc 8.5, nếu cho $R = \bar{S}$ ta sẽ nhận được cấu trúc của DFF (hình 8.8a) với bảng chân lý cho trên hình 8.8b và giản đồ xung minh họa chức năng hoạt động trên hình 8.8c và 8.8d.

Tại các thời điểm xung nhịp ở mức cao, dữ liệu D_n được tiếp nhận vào DFF (hình 8.8c) hay tại các thời điểm xung nhịp đi lên ($0 \rightarrow 1$) cho hình 8.8d.

Khi đó $Q_{n+1} = D_n$ (các lối vào $\overline{PR}$ và $\overline{CLR}$ yêu cầu phải thường xuyên ở mức không tích cực $\overline{PR} = \overline{CLR} = 1$).

Như vậy, chức năng quan trọng nhất của DFF là theo sự điều khiển của xung nhịp C_n , tiến hành tiếp nhận và sắp xếp lại dữ liệu D_n về mặt thời gian sao cho đồng bộ (xuất hiện đúng vào những thời điểm chuẩn xác định trước) để phối hợp các thao tác trình tự với dữ liệu khác trong nhiều khối chức năng khác nhau.

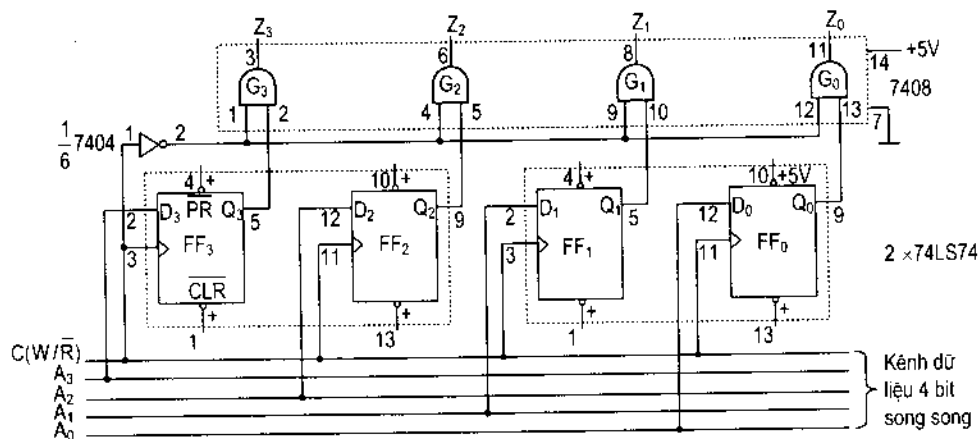


Hình 8.8a) Mạch cấu trúc DFF đồng bộ; b) Bảng chân lý của DFF;
c) Đồ thị thời gian minh họa chức năng của DFF điều khiển theo mức $C_n = 1$, chú ý ở lúc t_6 dữ liệu D bị từ chối vì khi đó $C = 0$, các thời điểm khác do $C = 1$ nên D được tiếp nhận; d) Đồ thị thời gian của DFF lật bằng sườn lên của C_n , chú ý lúc t_2^* dữ liệu bị làm trễ đến t_3 , và lúc t_4^* dữ liệu bị từ chối do không có C_n

– Ứng dụng quan trọng nhất của DFF là tổ chức mạch chốt (Latch) dữ liệu dưới dạng một thanh ghi để lưu giữ dữ liệu, dùng làm một ô nhớ trong các mảng hay ma trận của bộ nhớ bán dẫn ROM, RAM (bài 9 và 10). Hình 8.9 đưa ra một thanh ghi dữ liệu 4 bit dùng 4 DFF dạng đơn giản.

Trên mạch hình 8.9 mỗi DFF là $\frac{1}{2}$ IC 74LS74 và $G_3G_2G_1G_0$ dùng một IC 7408 gồm 4 cổng AND riêng biệt để kiểm soát dữ liệu tại các lối ra song song.

Khi $C = 1$ (cho phép ghi – write khi xung nhịp có sườn đi lên $0 \rightarrow 1$) các dữ liệu được ghi song song từ kênh dữ liệu 4 bit $A_3A_2A_1A_0$ vào các $DFF_3 DFF_2 DFF_1 DFF_0$ tương ứng (chỉ sau một nhịp đi lên của C), lúc này, các cổng $G_3G_2G_1G_0$ bị cấm (cấm đọc) do $\bar{C} = 0$. Khi muốn xuất (đọc dữ liệu đã ghi – read) cần cho C về mức 0 và $\bar{C} = 1$ thể hiện các lệnh cấm ghi và cho phép đọc. Các cổng AND mở cho phép dữ liệu xuất ra $Z_3Z_2Z_1Z_0 = Q_3Q_2Q_1Q_0$.



Hình 8.9. Cấu tạo một thanh ghi (chốt) dữ liệu 4 bit dùng 4 DFF (hai IC 74LS74) với một IC 7408 và một IC 7404 để kiểm soát dữ liệu ra

8.4. MẠCH TRIGƠ SỐ CHỦ – TỎ (Master – Slave Flip Flop – MSFF)

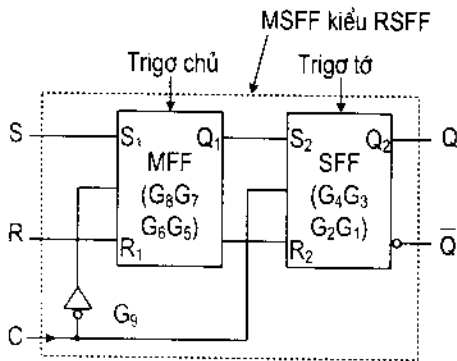
Nếu ghép liên tiếp hai mạch RSFF hình 8.3 theo dạng thể hiện trên hình 8.10 sẽ nhận được một Trơ chủ tớ (MSFF) hay còn gọi là Trơ chính phụ. MSFF cũng có thể được tạo ra từ hai mạch DFF hình 8.8a mô tả trên hình 8.11.

Phương thức làm việc của MSFF được tách ra thành hai nhịp rõ ràng tùy theo mức tín hiệu cho phép C ở tích cực cao hay tích cực thấp để thực hiện chức năng một ô nhớ căn bản.

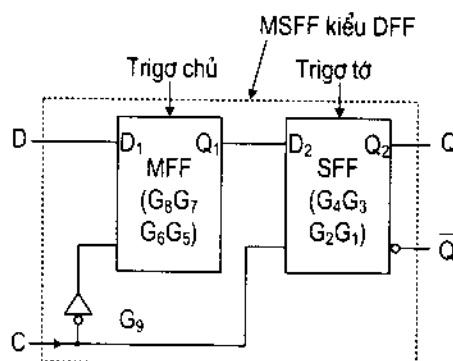
Khi $C = 0$ qua G_9 có $\bar{C} = 1$ vậy MFF nhập (write) dữ liệu vào.

Khi đó SFF bị cấm hoạt động nên không xảy ra thủ tục xuất dữ liệu. Còn khi $C = 1$, SFF được điều khiển làm việc cho phép dữ liệu từ MFF đã ghi truyền sang SFF và xuất hiện ở đầu ra (read), đồng thời, lúc này $\bar{C} = 0$, MFF bị cấm (không cho phép nhập dữ liệu mới khi đang thực hiện xuất). Nhờ MSFF các thủ tục nhập/xuất dữ liệu xảy

ra tuần tự và chỉ cần một xung điều khiển C cho phép và trở nên dễ điều khiển, có độ tin cậy cao, an toàn cho dữ liệu cần ghi/đọc.



Hình 8.10. MSFF có cấu tạo từ hai RSFF làm việc ngược pha nhau nhờ cổng G_9 với một tín hiệu cho phép C



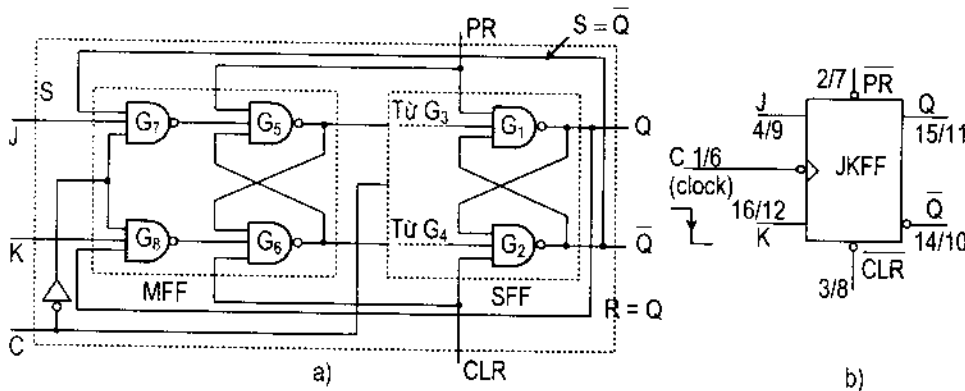
Hình 8.11. MSFF có cấu tạo từ hai DFF làm việc ngược pha (so le) nhờ G_9

8.5. MẠCH TRIGƠ SỐ ĐA NĂNG (JKFF) VÀ TRIGƠ ĐẾM (TFF)

Từ các mạch RSFF hình 8.3 và 8.5 hay 8.10 nếu thực hiện hai thay đổi cơ bản:

1. Nối tắt các lối ra với lối vào kiểu đan chéo ($S = \bar{Q}$ và $R = Q$).
2. Bổ sung thêm hai đầu vào mới: một đầu vào cho G_7 đặt tên là cổng J, một đầu vào cho G_8 đặt tên là cổng K. Khi đó sẽ nhận được trigơ đa năng JKFF.

Hình 8.12 thể hiện cấu tạo của JKFF và ký hiệu quy ước sau khi thực hiện thay đổi mạch.



Hình 8.12. Cấu trúc khối của JKFF (loại Chủ – Tớ) và ký hiệu quy ước tổng quát của JKFF điều khiển bằng sườn đi xuống ($1 \rightarrow 0$) của xung nhịp (ví dụ $\frac{1}{2}$ 74LS76/74LS73 chỉ số thập phân là số thứ tự chân của mỗi JKFF trong kết cấu hình học của IC)

– Bảng chân lý của JKFF cho trên bảng 8.1a dạng thu gọn và 8.1b dạng đầy đủ.

Từ bảng 8.1a có phương trình hàm ra:

$$Q_{n+1} = [J\bar{Q} + \bar{K}Q]_n \quad (8.5)$$

Bảng 8.1. Bảng chân lý JKFF dạng thu gọn (a) và dạng đầy đủ (b)

Đầu vào		Đầu ra
J_n	K_n	Q_{n+1}
0	0	Q_n
0	1	0
1	0	1
1	1	$\bar{Q}_n$

a)

Các đầu vào					Đầu ra		Ý nghĩa/chức năng Ký hiệu các đầu vào, ra
$\overline{PR}$	$\overline{CLR}$	$\overline{Cp}$	K_n	J_n	Q_{n+1}	$\overline{Q_{n+1}}$	
0	1	X	X	X	1	0	Thiết lập không đồng bộ
1	0	X	X	X	0	1	Xóa không đồng bộ (xóa trực tiếp)
0	0	X	X	X	1	1	Không xác định (cấm)
1	1	$\overline{1}$	1	1	$\overline{Q_n}$	Q_n	Lật sau mỗi xung nhịp (TFF)
1	1	$\overline{1}$	1	0	0	1	Xóa (đồng bộ)
1	1	$\overline{1}$	0	1	1	0	Thiết lập (đồng bộ)
1	1	$\overline{1}$	0	0	Q_n	$\overline{Q_n}$	Treo (không thay đổi, như cũ)

b)

– Chú ý dòng thứ 4 khi $J = K = 1$ thì JKFF sẽ lật trạng thái sau mỗi sườn đi xuống của xung nhịp và chức năng này gọi là một Trigo đếm (TFF – khi đó $C = T$).

– Về cấu tạo có thể bổ sung nhiều cổng J cho G_7 và nhiều cổng K cho G_8 tạo ra loại JKFF nhiều lối vào dữ liệu J và K ở chế độ đồng bộ.

– JKFF là loại Flip Flop vạn năng vì từ nó có thể tạo ra các loại FF khác như DFF, RSFF, hay TFF...

Khi cho $K = \bar{J}$ (nhờ thêm một cổng NOT giữa hai lối vào J và K) ta có DFF.

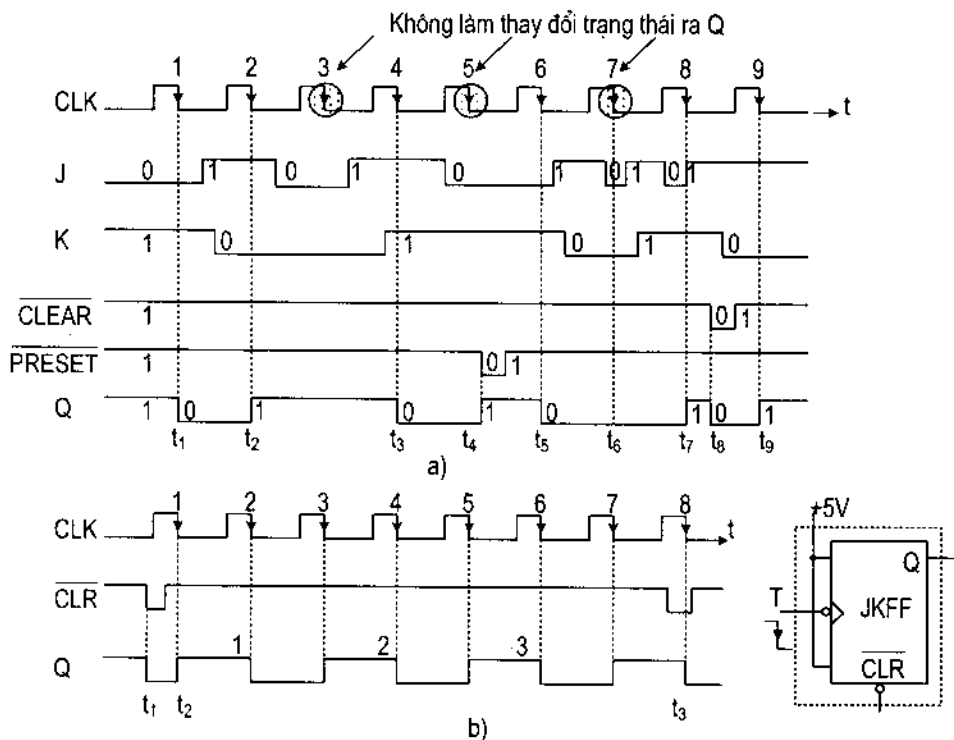
Khi cho $J = S$ và $K = R$ ta có loại RSFF.

– Phương trình logic của TFF có dạng:

$$Q_{n+1} = [T\bar{Q} + \bar{T}Q]_n$$

Trên hình 8.13 mô tả đồ thị xung điện áp minh họa chức năng hoạt động của JKFF (hình 8.13a) và của loại TFF (hình 8.13b).

Các đồ thị có được nhờ vận dụng bảng chân lý 8.1 đã có:



Hình 8.13a) Đồ thị xung mô tả các chức năng của JKFF, xác định trạng thái ra Q theo các trạng thái vào điều khiển và vào dữ liệu;

**b) Đồ thị xung mô tả hoạt động của JKFF khi cho $J = K = 1$
(chức năng của Trùng đếm TFF)**

Trên hình 8.13a, áp dụng bảng trạng thái 8.13 cho các thời điểm $t_1 \rightarrow t_9$ sẽ được đồ thị Q tại lối ra theo các trạng thái vào tương ứng. Ví dụ, lúc $t = t_1$ xung cho phép chuyển $1 \rightarrow 0$, $J = 0$, $K = 1$ vậy $Q = 0$, lúc $t = t_2$ xung cho phép chuyển $1 \rightarrow 0$, $J = 1$; $K = 0$ vậy $Q = 1$.

Lúc $t = t_3$; $J = K = 1$ trạng thái ra lật. Vậy $Q = 0$.

Lúc $t = t_4$ do lối vào không đồng bộ (trực tiếp) $\overline{PR} = 0$ nên $Q = 1$ (không quan tâm tới giá trị các dữ liệu vào J , K và nhịp đồng bộ CLK).

Lúc $t = t_5$ trạng thái như khi $t = t_1$, lúc $t = t_6$ CLK chuyển về 0, $J = K = 0$ vậy Q bảo lưu giá trị đang có là mức 0, lúc $t = t_8$, do $\overline{CLR} = 0$ nên $Q = 0$.

Tương tự cho đồ thị hình 8.13b cho TFF (chỉ dùng trạng thái $J = K = 1$).

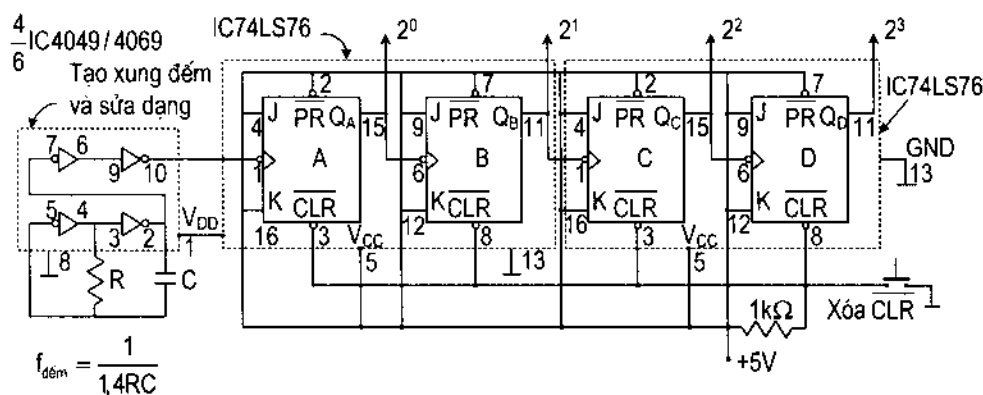
Lúc $t = t_1$ và $t = t_3$ do $\overline{\text{CLR}} = 0$, Q bị xóa về 0.

Lúc $t = t_2$ TFF thực hiện đếm số xung vào lối vào T (chính là xung đồng bộ) bằng cách thực hiện chia đôi số lượng dãy xung này. Có thể kết luận rằng một TFF thực hiện một bộ đếm 2 bằng cách chia đôi số xung đếm ở lối vào.

8.6. BỘ ĐẾM NHỊ PHÂN VÀ BỘ ĐẾM NHỊ – THẬP PHÂN (BCD)

8.6.1. Bộ đếm nhị phân

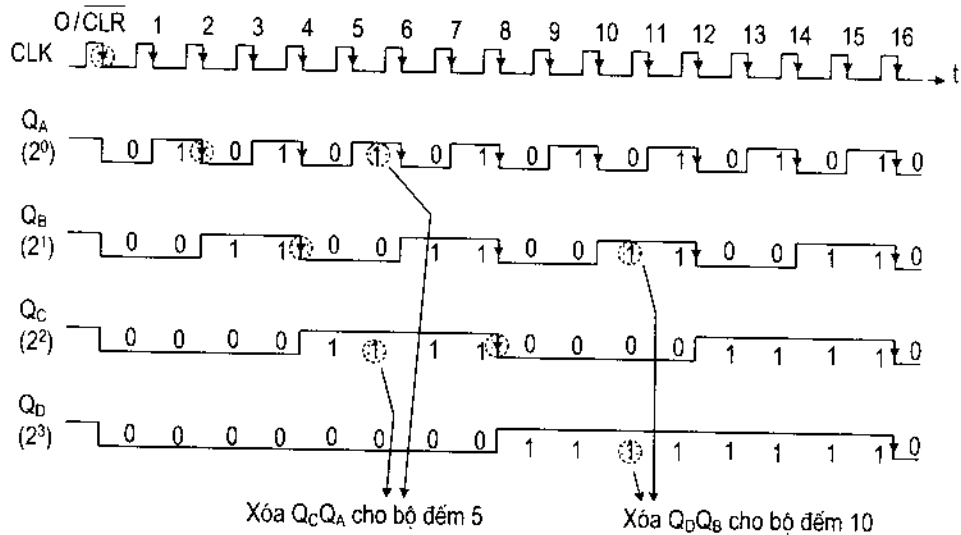
Thực hiện việc ghép liên tiếp các TFF như cho trên hình 8.4, sẽ nhận được một bộ đếm xung kiểu tuần tự hay dãy xung vào đã được chia đôi liên tiếp qua mỗi TFF. Nếu có n Trigo T ghép liên tiếp nhau, số lượng xung đếm được tối đa (gọi là dung lượng của bộ đếm) đúng bằng số lượng tối đa các trạng thái nhị phân mà lối ra của n TFF có được sẽ là $2^n - 1$. Với mạch hình 8.14, dung lượng bộ đếm là $2^4 - 1 = 15$ xung hay mạch được gọi là bộ đếm nhị phân cơ số đếm 16. Bảng trạng thái và đồ thị thời gian minh họa hoạt động được cho trên các bảng 8.2 và hình 8.15.



Hình 8.14. Mạch đếm nhị phân cơ số 16 kiểu tuần tự (nối tiếp hoặc không đồng bộ) dùng 4JKFF nối tiếp kiểu TFF (2 IC 74LS76) IC 4049 được dùng để tạo xung đếm có chu kỳ $T_{\text{dém}} = 1.4RC$

Bảng 8.2. Bảng chân lý của bộ đếm nhị phân cơ số 16 (tối đa)

Số lượng xung vào	Trạng thái ra của các TFF				Số lượng xung vào	Trạng thái ra của các TFF			
	Q _D	Q _C	Q _B	Q _A		Q _D	Q _C	Q _B	Q _A
O/CLR	0	0	0	0	8	1	0	0	0
1	0	0	0	1	9	1	0	0	1
2	0	0	1	0	10	1	0	1	0
3	0	0	1	1	11	1	0	1	1
4	0	1	0	0	12	1	1	0	0
5	0	1	0	1	13	1	1	0	1
6	0	1	1	0	14	1	1	1	0
7	0	1	1	1	15	1	1	1	1



Hình 8.15. Đồ thị xung mô tả hoạt động của mạch đếm nhị phân cơ số 16 hình 8.14 và bộ đếm BCD khi xóa Q_D, Q_B sau xung thứ 10 (hình 8.17)

– Từ bảng chân lý 8.2 của bộ đếm nhị phân có các nhận xét quan trọng sau:

1. Bản chất bộ đếm nhị phân là một bộ biến đổi mã (mã hóa) thập phân thành nhị phân.

2. FF phía sau chỉ chuyển trạng thái khi FF phía trước liên kế chuyển từ 1 → 0 (tất nhiên đồng thời với sườn sau của xung đồng bộ).

3. Hoặc FF phía sau chỉ chuyển khi mọi trạng thái ra của các FF trước nó đã nhận trị 1 (đều ở mức cao).

4. Có thể dùng bộ đếm một cách tuần hoàn sau một số lượng xung vào nhất định (ví dụ sau $k - 1$ xung). Trạng thái kế tiếp tương ứng với xung thứ k được các mạch logic bổ sung sẽ tự động xóa về 0 (tức là với 4FF có $Q_D Q_C Q_B Q_A = 0000$ hay $Q_C Q_B Q_A = 000$ với 3FF hoặc $Q_B Q_A = 00$ với trường hợp chỉ dùng 2 FF). Khi đó sẽ nhận được bảng chân lý của một bộ đếm nhị phân cơ số đếm là k .

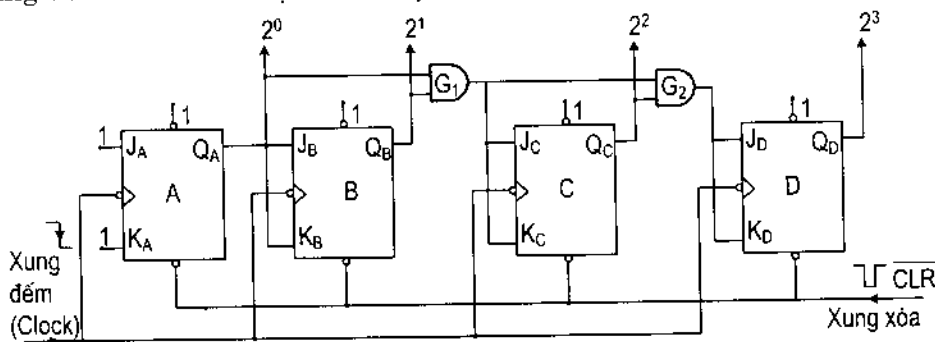
Nhận xét đầu tiên cho phép xây dựng các bộ mã hóa biểu diễn một số thập phân thành số nhị phân với cơ số đếm 2^n .

Nhận xét thứ hai cho phép xây dựng mạch điện của bộ đếm nối tiếp (không đồng bộ) dùng TFF như hình 8.14 đã thể hiện.

Nhận xét thứ ba cho phép xây dựng mạch điện của bộ đếm song song (bộ đếm đồng bộ) có tốc độ làm việc cao hơn vì thời gian trễ nhỏ hơn như được thể hiện trên hình 8.16.

Nhận xét thứ tư cho phép xây dựng các mạch điện của bộ đếm nhị phân với cơ số đếm bất kỳ: Ví dụ, bộ đếm nhị thập phân cơ số 10 như trên hình 8.17 hay bộ đếm nhị phân cơ số 6 như thể hiện trên hình 8.18.

– Để thực hiện các bộ đếm song song, xung đếm được đưa đồng thời vào mọi đầu vào nhịp của TFF (đầu vào $C \equiv T$). Các đầu vào J và K được dùng để điều khiển chọn điều kiện $J = K = 1$ như hình 8.16 thể hiện.



Hình 8.16. Mạch điện nguyên lý bộ đếm nhị phân hệ 16 kiểu đồng bộ:

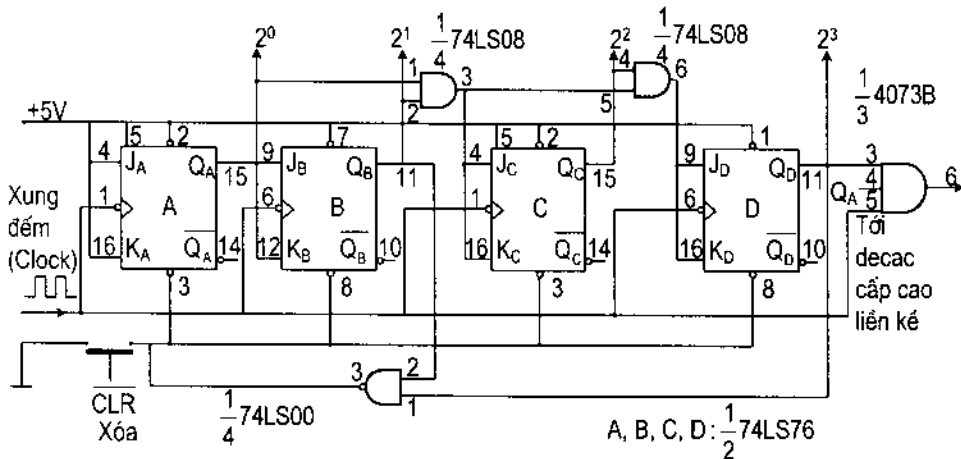
$$G_1 = Q_A \cdot Q_B; Q_A = Q_B = 1 = J_C = K_C; G_2 = Q_A \cdot Q_B \cdot Q_C; Q_A = Q_B = Q_C = 1 = J_D = K_D$$

Mạch đếm song song có ưu điểm quan trọng là thời gian trễ nhỏ đi n lần (n là số FF sử dụng) so với bộ đếm nối tiếp.

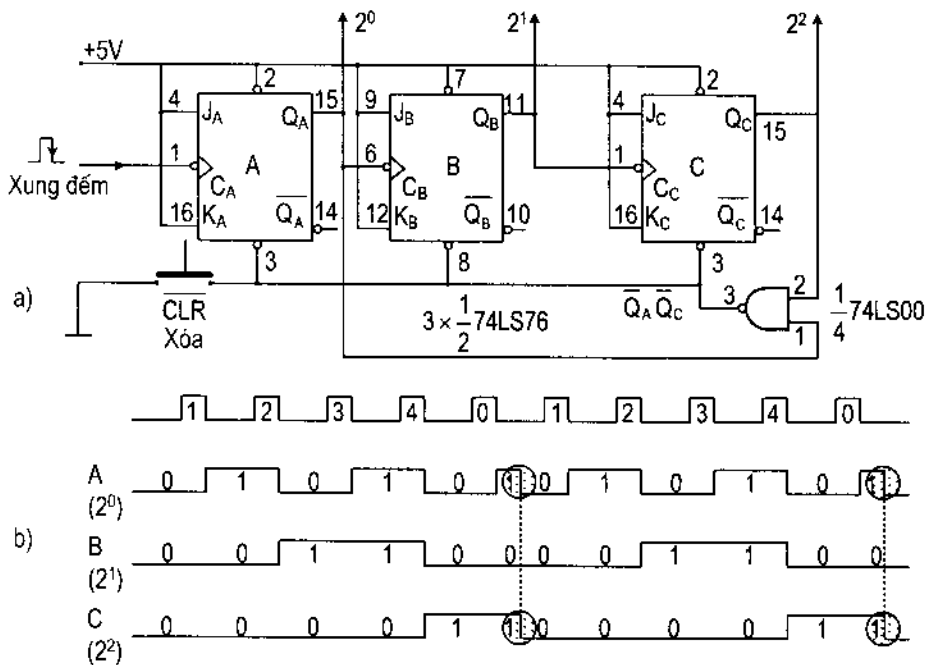
8.6.2. Bộ đếm nhị thập phân (bộ đếm BCD)

– Trên bảng chân lý 8.2, nếu chỉ sử dụng đến trạng thái 9 số xung vào (đường đứt nét) sau sự xuất hiện của xung thứ 10 bộ đếm về lại trạng thái ban đầu $Q_D Q_C Q_B Q_A = 1010 \rightarrow 0000$ tức là phải thực hiện xóa ngay các trạng thái $Q_D = Q_B = 1$.

– Bộ đếm BCD loại đồng bộ được cho trên hình 8.17. Ở mạch hình 8.17, cổng NAND 74LS00 có nhiệm vụ xóa các trạng thái ra của D và B ($Q_D = Q_B = 0$ ngay sau khi xuất hiện các mức $Q_B = 1$ và $Q_D = 1$).



Hình 8.17. Mạch điện bộ đếm BCD đồng bộ (1 decac)



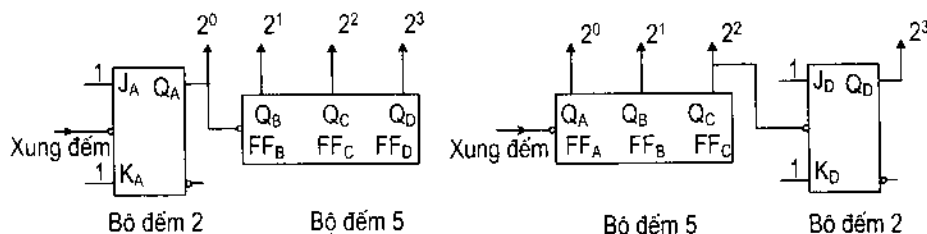
Hình 8.18a) Bộ đếm nhị phân cơ số 5 (gọi tắt là bộ đếm 5)

kiểu nối tiếp dùng 3 JKFF loại $\frac{1}{2}$ IC 74LS76;

b) Giải đồ xung minh họa hoạt động của bộ đếm 5

Chú ý sau xung đếm thứ 5 (ký hiệu 0) cổng NAND lấy mức ra 0 xóa Q_A và Q_C .

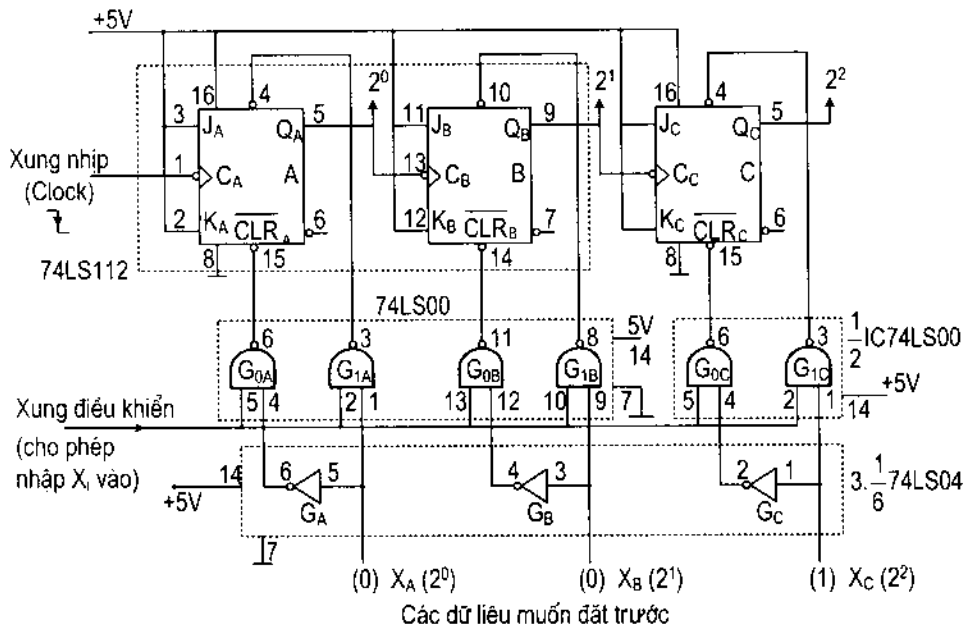
- Có thể xây dựng bộ đếm 10 từ cách ghép liên tiếp hai bộ đếm: Một bộ đếm 2 (dùng 1 FF) và một bộ đếm 5 (dùng 3 FF) như sơ đồ khối trên hình 8.19 (cho kiểu nối tiếp). Đây là kiểu cấu trúc khối của IC 7490 (hình 6.24).



Hình 8.19. Tổ chức bộ đếm nối tiếp (hoặc song song) bằng cách ghép liên tiếp hai bộ đếm 2 và 5.

8.7. BỘ ĐẾM CÓ GIÁ TRỊ BAN ĐẦU THIẾT LẬP TRƯỚC

Bộ đếm 8 kiểu nối tiếp có khả năng thiết lập trạng thái ban đầu trước khi thực hiện đếm dãy xung vào được cho trên hình 8.20 dùng 3 JKFF làm việc ở chế độ TFF ($J = K = 1$). Nếu giá trị thiết lập trước là N_0 thì xung đầu tiên vào sẽ tương ứng với $N_0 + 1...$



Hình 8.20. Bộ đếm 8 có thiết lập trước trạng thái đầu nhờ các đầu vào điều khiển trực tiếp $\overline{CLR}$ và $\overline{PR}$ (của JKFF dùng 74LS112)

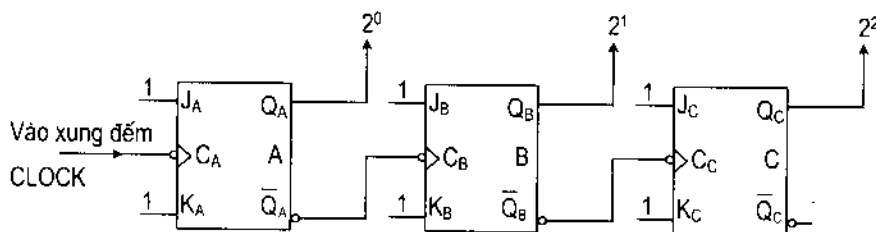
Các dữ liệu muốn nhập trực tiếp vào các FF A, B, C của bộ đếm được áp đồng thời tới các lối vào X_A , X_B và X_C tương ứng, ví dụ $X_C X_B X_A = 100$. Khi có lệnh cho phép nhập (mức logic 1) dữ liệu X_i , ví dụ $X_A = 0$ qua các cổng G_0 và G_1 tương ứng FF_A có $\overline{CLR}_A = 0$ và $\overline{PR}_A = 1$ nên $Q_A = X_A = 0$.

Tương tự, đồng thời khi đó FF_B: $\overline{CLR}_B = 0$ và $\overline{PR}_B = 1$ nên $Q_B = X_B = 0$. Tại các lối vào của FF_C: $\overline{CLR}_C = 1$ và $\overline{PR}_C = 0$ nên $Q_C = X_C = 1$. Dữ liệu X_i đã được nạp song song vào sau một nhịp chuẩn bị (một xung clock). Như vậy, trạng thái xuất phát của bộ đếm bây giờ là $X_C X_B X_A = Q_C Q_B Q_A = 100$ (tương đương như "đã" đếm trước được 4 xung).

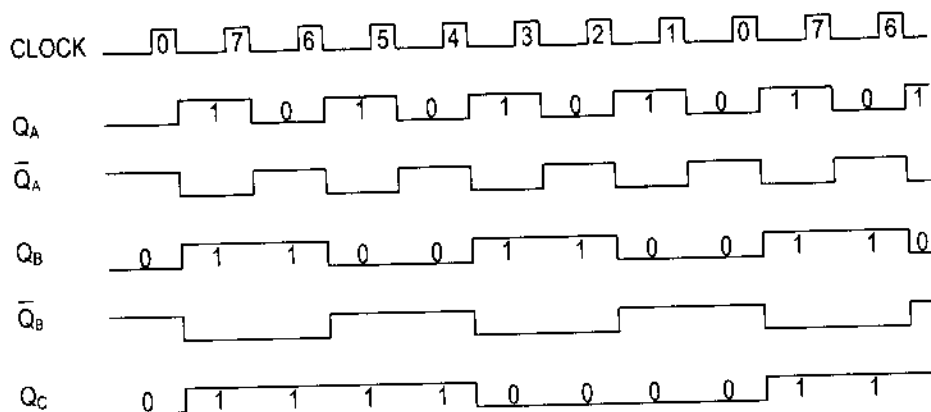
8.8. BỘ ĐẾM HAI HƯỚNG (đếm thuận/ngược: UP/DOWN counter)

– Bộ đếm hướng xuống thực hiện trừ dần số xung ở lối vào đếm được thực hiện nhờ việc ghép lối ra $\overline{Q}$ của TFF trước tới lối vào nhịp của TFF kế phía sau (hình 8.21, đối với loại không đồng bộ). Đồ thị thời gian minh họa cho trên hình 8.22 với bộ đếm đồng bộ cần dùng các cổng ra $\overline{Q}$ tương ứng điều khiển các đầu vào JK của TFF kế tiếp (hình 8.23).

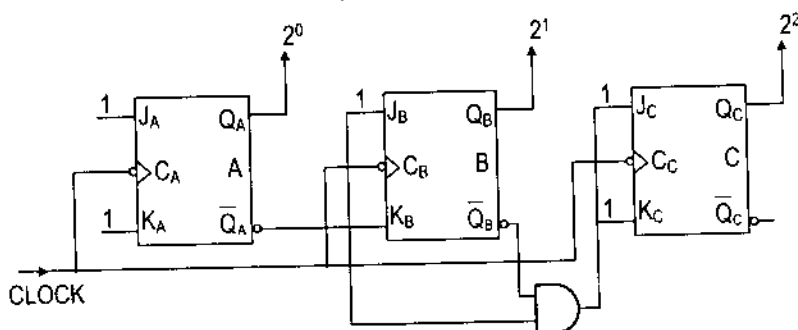
– Nhờ việc chọn cách đấu hoặc từ Q_i tới C_{i+1} hoặc từ $\overline{Q}_i$ tới C_{i+1} sẽ nhận được bộ đếm 2 hướng lên hoặc xuống tùy theo lệnh UP/DOWN kiểu không đồng bộ. Với loại đồng bộ, cách làm tương tự nhưng dùng Q_i hay $\overline{Q}_i$ điều khiển $J_{i+1} K_{i+1}$ (có bổ sung thêm các cổng AND).



Hình 8.21. Mạch nguyên lý bộ đếm 8 hướng xuống (không đồng bộ) điều khiển lật nhờ các cổng $\overline{Q}$



Hình 8.22. Đồ thị xung minh họa hoạt động của bộ đếm hướng xuống không đồng bộ (hình 8.21)



Hình 8.23. Mạch điện nguyên lý bộ đếm 8 hướng xuống kiểu đồng bộ điều khiển lật nhờ các cổng JK

Các JKFF trên hình 8.21 và 8.23 có thể dùng các loại 74LS76 hay 74LS112.

Thực tế, các bộ đếm được chế tạo dưới dạng IC đếm loại MSI (Medium Scale Integration) với cấu hình kết hợp modun để dễ dàng khai thác sử dụng ví dụ như:

IC 7490 có cấu tạo kết hợp một bộ đếm 2 (FFA) và một bộ đếm 5 (FFB, FFC và FFD) đều là các JKFF điều khiển theo sườn đi xuống của xung Clock.

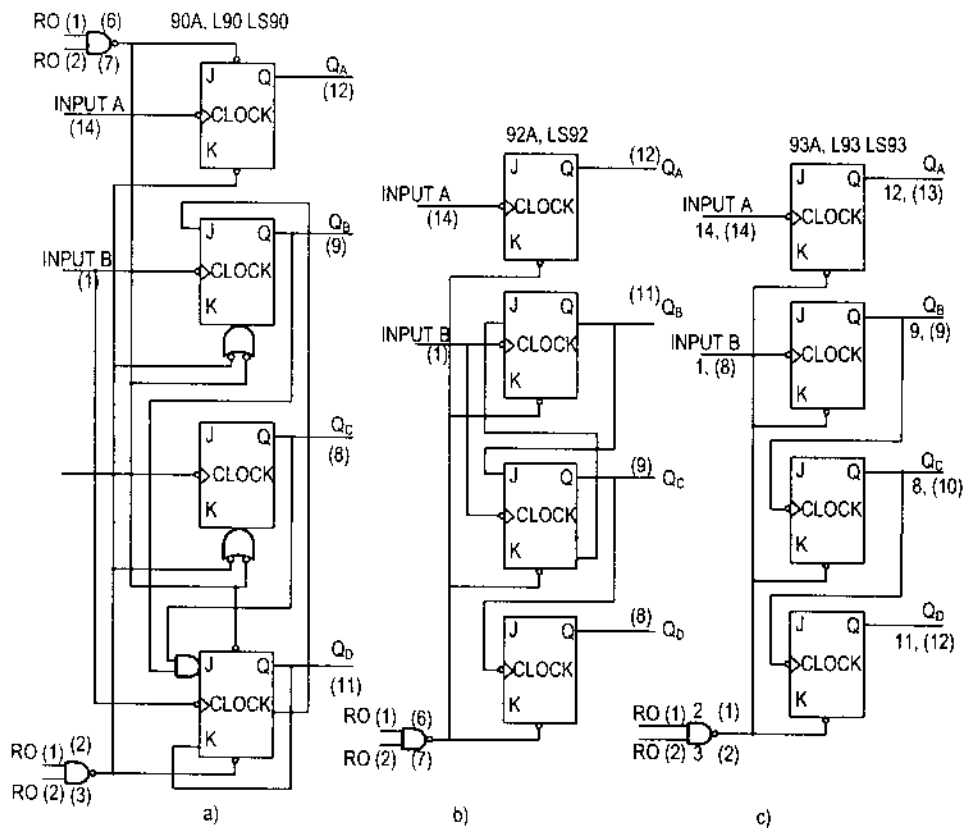
IC 7492 gồm một bộ chia 2 (đếm 2) và một bộ chia 6 (đếm 6) như vậy dùng IC 7490 có thể tạo được bộ đếm 10 và IC 7492 tạo được bộ đếm 12 (bộ đếm thời gian giờ, ngày).

IC 7493 có chứa một bộ đếm 2 và một bộ đếm 8 và như vậy, có thể tạo được bộ đếm 16.

IC 74LS190, 74LS191 cho phép đếm 2 hướng kiểu đếm 10 (với 190) và kiểu đếm nhị phân (với 191) dạng đồng bộ dùng các JKFF loại MSFF.

Bộ đếm thời gian phút, giây (đếm 60) được kết hợp ví dụ từ một IC 74LS90 và 74LS92. Hình 8.24 đưa ra cấu tạo của 3 nhóm IC điển hình kể trên: nhóm 7490A/74L90/74LS90 (hình 8.24a), nhóm 7492A/74LS92 (hình 8.24b) và nhóm 7493/74L93/74LS93 (hình 8.24c).

Tất cả các mạch điện của bộ đếm cơ số N đều thực hiện phép chia tần số dãy xung nhịp N lần tương ứng nên bộ đếm N còn có tên khác là bộ chia tần N.



Hình 8.24

8.9. THANH GHI DỊCH

8.9.1. Nhiệm vụ cơ bản của thanh ghi dịch

1. Chuyển đổi luồng dữ liệu từ dạng nối tiếp thành dạng song song hoặc ngược lại từ song song thành nối tiếp.

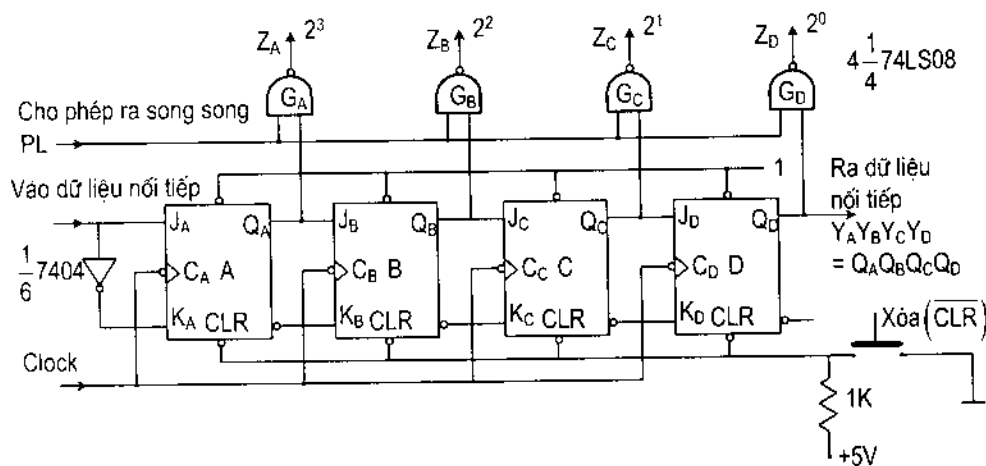
2. Lưu trữ dữ liệu từ dạng nối tiếp sang nối tiếp hay từ song song sang song song.

3. Gia công, xử lý toàn bộ hay từng bit dữ liệu trong quá trình thao tác với dữ liệu hoặc tính toán số học...

Thanh ghi dịch có cấu tạo gồm các MSFF ghép liên tiếp nhau (có bao nhiêu MSFF thì có bấy nhiêu bit). Ưu điểm quan trọng của loại MSFF là giá trị dữ liệu cũ đã lưu trong nó còn cần thiết và có thể được sử dụng trong lúc đang ghi giá trị mới nhờ tách các thao tác ghi/đọc thành hai nhịp độc lập và ngược pha nhau.

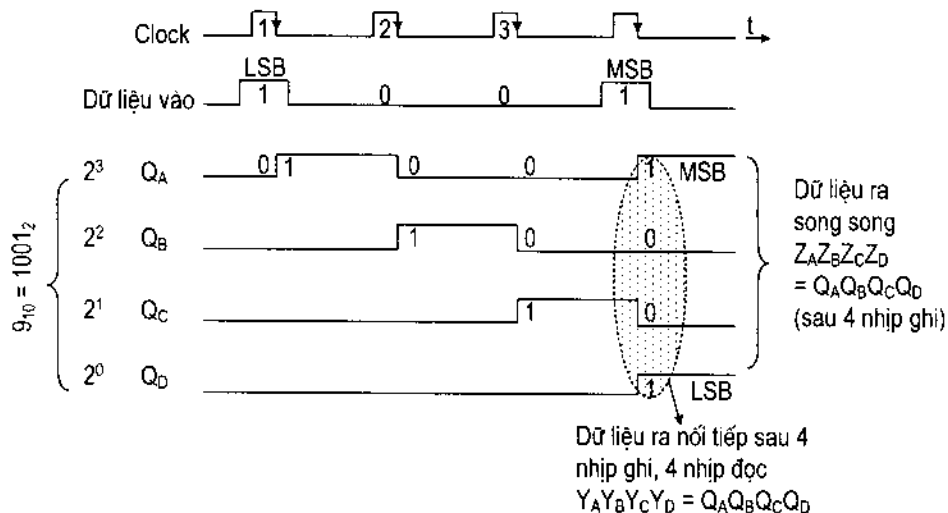
8.9.2. Mạch điện thanh ghi dịch

Mạch điện hình 8.25 đưa ra một thanh ghi dịch vào nối tiếp ra nối tiếp hoặc song song loại 4 bit dùng 4 JKFF.



Hình 8.25. Mạch điện thanh ghi 4 bit dịch phải vào nối tiếp ra nối tiếp hoặc ra song song dùng 4 JKFF ($4 \times \frac{1}{2}$ IC74LS76)

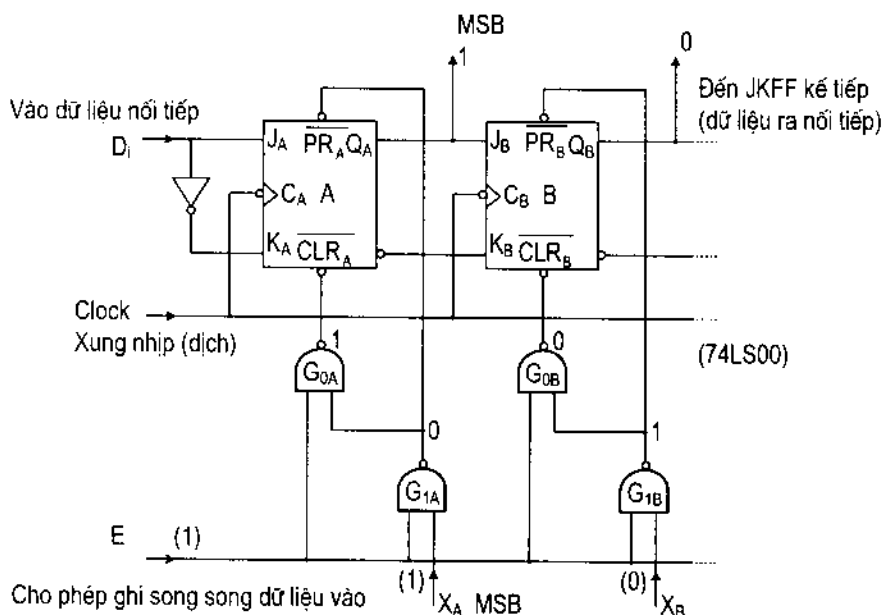
Đồ thị thời gian minh họa hoạt động của mạch thanh ghi dịch 4 bit (hình 8.25) được cho trên hình 8.26. Sau 4 nhịp dữ liệu 4 bit được ghi vào các JKFF tương ứng Q_A, Q_B, Q_C, Q_D , chú ý LSB vào trước MSB sau cùng. Sau đó, có thể đọc dữ liệu ra theo hai cách: nối tiếp sau 4 nhịp tiếp theo với thủ tục vào trước ra trước. $Y_A Y_B Y_C Y_D = Q_A Q_B Q_C Q_D$ hay chỉ sau một nhịp cho phép đọc song song ($Z_A Z_B Z_C Z_D = Q_A Q_B Q_C Q_D$) xuất hiện khi có xung điều khiển cho phép ra song song PL.



Hình 8.26. Giản đồ thời gian minh họa hoạt động của mạch hình 8.25.

8.9.3. Mở rộng chức năng ghi

Có thể mở rộng chức năng ghi song song cho mạch điện hình 8.27 nhờ dùng các đầu vào điều khiển không đồng bộ $\overline{PR}$ và $\overline{CLR}$ của các JKFF.

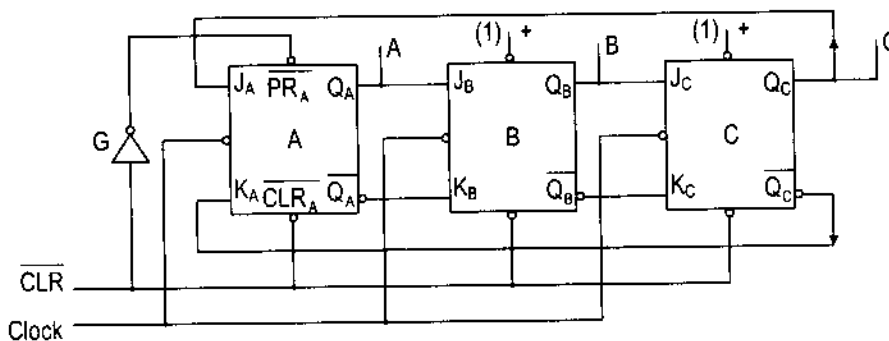


Hình 8.27. Phương pháp ghi song song dữ liệu cho các ô của thanh ghi các đầu vào $\overline{PR}$ và $\overline{CLR}$

Khi đó, sau một nhịp cho phép ghi song song ($E = 1$), các bit dữ liệu $X_A X_B$ được ghi trực tiếp vào các $FF_A, FF_B \dots$ không phụ thuộc vào các trạng thái ở các lối vào nối tiếp D_i hay vào nhịp C khi $E = 0$ thì các cổng NAND bị cấm. Hoạt động của các cổng NAND khi $E = 1$ như sau: $G_{1A} = \overline{E.X_A}$. Nếu $X_A = 1$ thì G_{1A} cho ra mức 0, mức này cấm G_{0A} và kích hoạt $\overline{PR}_A = 0$ do vậy $Q_A = X_A = 1$. Còn với $X_B = 0$ thì G_{1B} cho ra mức 1 (không tích cực) cổng $G_{0B} = \overline{E.G_{1B}}$ mở cho lối ra lên tích cực 0, do vậy $\overline{CLR}_B = 0$ dẫn đến $Q_B = X_B = 0$. Đồng thời với X_A và X_B việc ghi sẽ xảy ra đối với các bit thấp hơn $X_C, X_D \dots$ ở các ô kế tiếp. Sau khi mở rộng chức năng thanh ghi dịch hình 8.25 có bổ sung như hình 8.27 sẽ thực hiện chức năng biến đổi đa năng: vào nối tiếp hoặc song song, ra nối tiếp hoặc song song (4 cách thay đổi dữ liệu về phương thức truyền).

8.9.4. Bộ đếm vòng (thanh ghi dịch từng bit SAR)

Mạch điện của một bộ đếm vòng (hình 8.28) thực chất là một thanh ghi khi ghép hồi tiếp các lối ra của ô cuối cùng tới lối vào tương ứng của ô đầu tiên, ví dụ: $Q_C = J_A, \overline{Q}_C = K_A$ trong bộ đếm vòng dung lượng 3 JKFF hình 8.28.

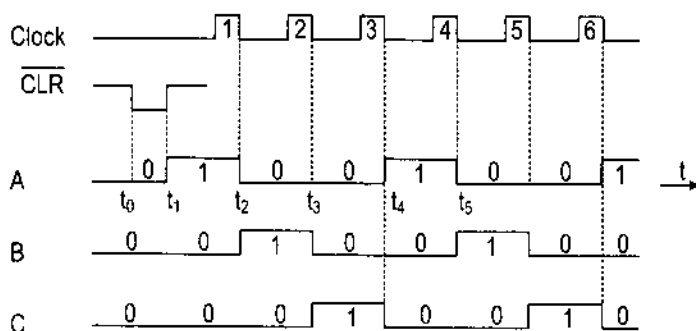


Hình 8.28. Bộ đếm vòng dung lượng 3 dùng 3 JKFF

Theo đồ thị hình 8.29, lúc t_0 do $\overline{CLR} = 0$, $A = B = C = 0$. Lúc $t = t_1$ do $\overline{CLR} = 1$ qua cổng NOT G nhận được $\overline{PR}_A = 0$ thiết lập mức $A = 1$. Lúc $t = t_2$ xung dịch chuyển dữ liệu A dịch phải: $Q_B = B = 1$ ta có: $ABC = 010$ ($J_B = A = 1$; do $C = J_A = 0$ nên A về 0).

Lúc $t = t_3$ xung nhịp dịch dữ liệu tiếp: $Q_C = C = 1$ ta có $ABC = 001$ ($J_C = B = 1$ nên C lên 1).

Lúc $t = t_4$, bộ đếm vào chu kỳ tiếp do chỉ FF_A phản ứng với xung nhịp: $A = 1$, trạng thái $ABC = 100$ được lặp lại.

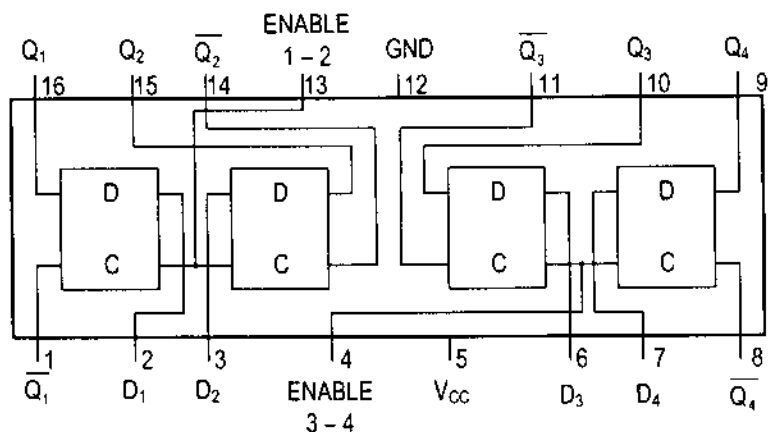


Hình 8.29. Đồ thị thời gian minh họa hoạt động của bộ đếm vòng hình 8.28

8.10. CÂU HỎI VÀ BÀI TẬP

- Hãy nêu các chức năng tối thiểu của một mạch Flip – Flop. Khi có mặt đầy đủ các lối vào, chức năng đó được mở rộng như thế nào? Hãy nêu ví dụ minh họa.
- FF có thể làm việc ở hai chế độ đồng bộ và không đồng bộ, hãy phân biệt sự khác nhau giữa hai phương thức này.
 - Hãy phân biệt cấu trúc RSFF không đồng bộ dùng 2 NAND và 2 NOR đấu chéo, viết bảng chân lý và giải thích ý nghĩa của bảng.
 - Hãy vẽ cấu trúc của RSFF đồng bộ dùng 4 NAND hoặc dùng 4 NOR và mô tả phương thức làm việc của loại này?
- Phân biệt cấu trúc của DFF, JKFF và TFF có nguồn gốc từ RSFF. Hãy viết bảng chân lý của DFF và JKFF. Giải thích ý nghĩa của bảng. MSFF có đặc tính gì là quan trọng nhất, đặc điểm cấu tạo của nó có gì đặc biệt để thể hiện tính chất này?
- Hãy mô tả cấu tạo của thanh ghi 4 bit dùng DFF (hình 8.9).
Hãy phân tích mạch điện hình 8.9: chức năng, nhiệm vụ và phương thức làm việc, nêu ví dụ một ứng dụng của mạch.
- Viết bảng chân lý của một bộ đếm nhị phân cơ số đếm 16 và hãy nêu các nhận xét cơ bản từ bảng chân lý này.
 - Bộ đếm nối tiếp (không đồng bộ) được xây dựng theo quy luật nào? Cách thực hiện một bộ đếm có modun đếm khác bất kỳ như thế nào?

6. a) Để xây dựng một bộ đếm modul 12, hãy viết bảng chân lý và nhận xét bảng này để tìm ra quy luật xây dựng vòng hồi tiếp thực hiện bằng.
b) Câu hỏi tương tự khi xây dựng bộ đếm modul 6.
7. a) Bộ đếm hai chiều (hai hướng) có đặc điểm cấu tạo khác bộ đếm một hướng ở điểm nào? Cổng logic cần bổ sung là loại cổng AND hay OR để thay đổi hướng đếm theo một lệnh duy nhất.
b) Cần bổ sung những gì vào bộ đếm để nó có khả năng thiết lập trước trạng thái ban đầu khi chuẩn bị hoạt động?
8. a) Hãy viết bảng trạng thái của bộ đếm thập phân.
Dựa trên mạch hình 8.17, hãy xây dựng bộ đếm để đếm từ 000 đến 999 xung vào dùng IC 74LS90 (hình 8.24).
b) Từ IC 7493 (hình 8.24c) hãy vẽ mạch nối chân chip để tạo các bộ đếm 4, 8, 10 và 12.
9. a) Hãy phân biệt chức năng và cấu tạo của một thanh ghi và một thanh ghi dịch: đặc điểm giống nhau và khác nhau là gì?
Các phương thức hoạt động cơ bản của một thanh ghi dịch và đặc điểm của mỗi loại.
b) Hãy vẽ mạch điện của một thanh ghi 8 bit dữ liệu vào ra kiểu song song dùng DFF loại IC 74LS74.
10. Cho IC 7475 có cấu tạo logic và bố trí chân như trên hình 8.30.

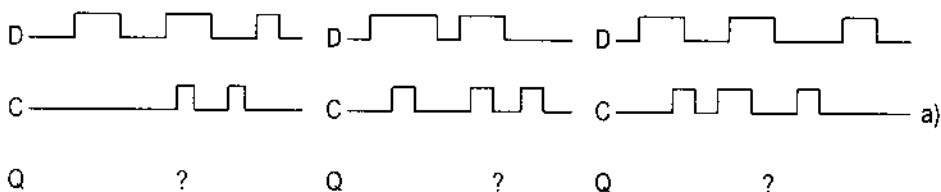


Hình 8.30

a) Có thể thực hiện được bao nhiêu khả năng chốt dữ liệu khi dùng IC 7475 đã cho.

Đầu vào xung nhịp là chân nào (số mấy)? Ở mức tích cực nào?

b) Hãy vẽ đồ thị thời gian của lối ra Q của một DFF trong cấu trúc IC với các trường hợp sau (hình 8.31a).



c) Xây dựng một mạch chốt dữ liệu gồm hai cặp bit A_1A_0 và B_1B_0 vào song song ra song song dùng IC 7475.

11. Cho một thanh ghi 6 bit trên hình 8.31b dùng IC 74LS174.

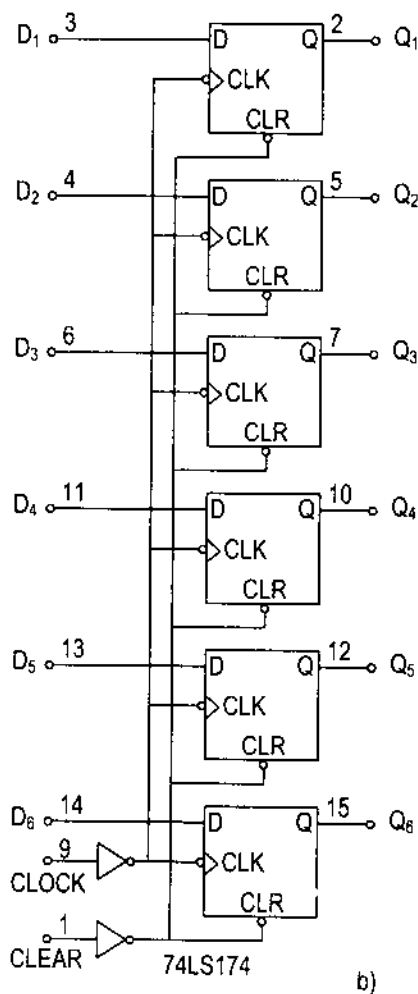
a) Thanh ghi đã cho có thể làm việc ở chế độ không đồng bộ (nối tiếp) được không?

b) Trạng thái ra của mỗi DFF sẽ lật khi nào xung clock ở mức cao, ở mức thấp, đang chuyển từ thấp lên cao hay đang chuyển từ cao xuống thấp?

c) Trong điều kiện nào thì $Q_6Q_5Q_4Q_3Q_2Q_1 = D_6D_5D_4D_3D_2D_1$?

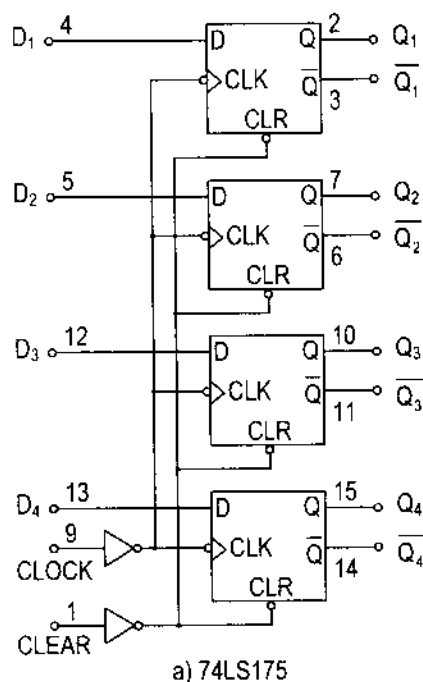
d) Nếu có hai nhóm dữ liệu 3 bit muốn ghi không đồng thời vào thanh ghi này thì có thực hiện được không?

12. Các câu hỏi tương tự như câu 11a, 11b và 11c (với dữ liệu 8 bit)



Hình 8.31

cho các thanh ghi dùng DFF cho trên hình 8.32 (IC 74LS175 và IC 74ALS273).



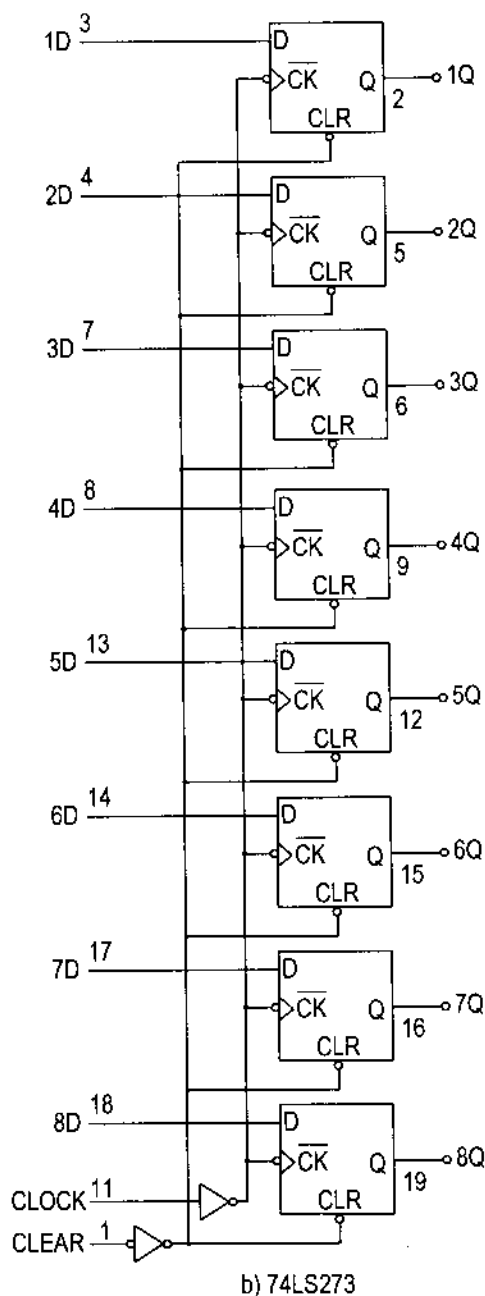
13. Cho mạch logic hình 8.33.

a) Hãy nêu chức năng (nhiệm vụ logic) mạch thực hiện?

Mô tả hoạt động của mạch khi $D_3D_2D_1D_0 = 1001$ và $C = 0$; khi $D_3D_2D_1D_0 = 0110$ và $C = 1$.

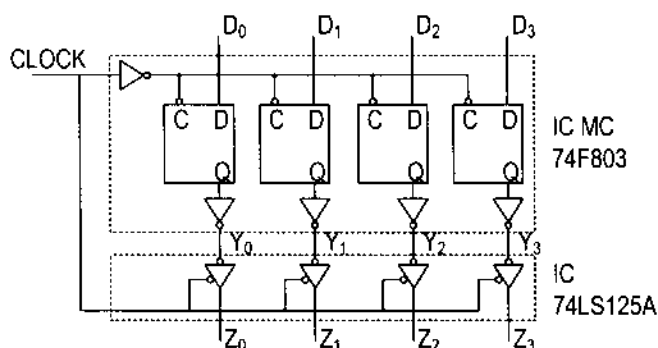
Xác định trạng thái ra trong các trường hợp trên.

b) Các chức năng hoạt động ghi dữ liệu vào và đọc dữ liệu ra có phân biệt được không nếu thay thế IC74LS175 bằng 74LS126? (hình 3.8 bài tập 9 của bài 3).



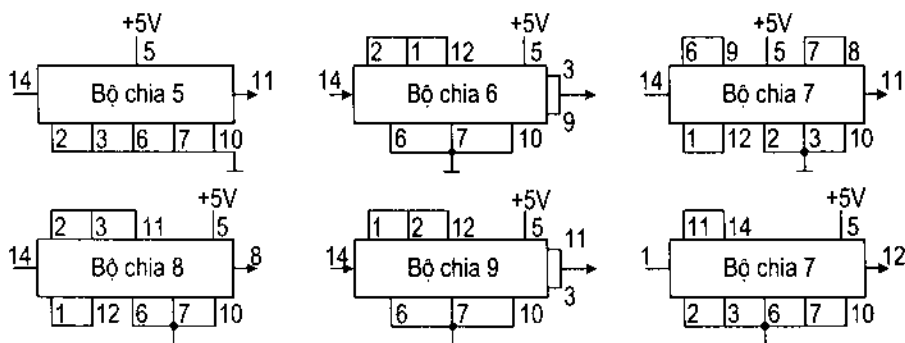
Hình 8.32

c) Cần thay đổi lại cách nối mạch như thế nào để dùng được IC 74LS126 với chức năng tương đương như mạch ban đầu?

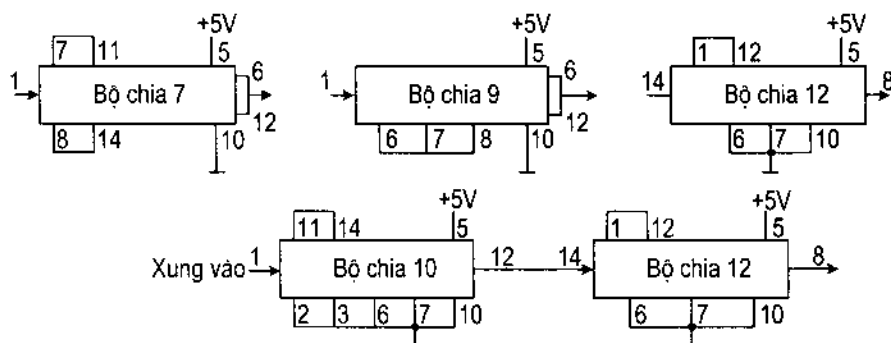


Hình 8.33

14. Hình 8.24 cho cấu trúc IC đếm 74LS90 và 74LS92. Hãy giải thích cách nối các chân của hai loại IC này để thực hiện các bộ chia 5, 6, 7, 8, 9, 10 (cho IC 74LS90 trên hình 8.34) và bộ chia 7, 9, 12, 120 (cho IC 74LS92 trên hình 8.35).



Hình 8.34. Các bộ chia (bộ đếm) dùng IC 74LS90



Hình 8.35. Các bộ đếm dùng IC 7492

BÀI 9. BỘ NHỚ CHỈ ĐỌC (ROM)

9.1. KHÁI NIỆM CHUNG VỀ BỘ NHỚ

9.1.1. Bộ nhớ

Theo định nghĩa, bộ nhớ là một mạch (hay thiết bị) có khả năng lưu giữ thông tin. Trong kỹ thuật số các thông tin mà bộ nhớ có nhiệm vụ lưu giữ ở dạng nhị phân. Muốn sử dụng bộ nhớ cần có ba quá trình cơ bản: kiểm tra thông tin đang được lưu giữ trong nó, nếu thông tin này không còn cần thiết phải xóa chúng đi (làm sạch bộ nhớ); ghi dữ liệu và các thông tin nhị phân cần lưu giữ (mới) vào nó; đọc thông tin đã ghi trong nó để sử dụng.

Thủ tục thực hiện các quá trình trên phải được kiểm soát chặt chẽ tránh mất mát hoặc nhầm lẫn thông tin dữ liệu. Muốn vậy, phải định vị chính xác vị trí từng ô nhớ (từng bit nhị phân) theo một mã địa chỉ duy nhất để tìm kiếm để người sử dụng biết rằng: vị trí nhớ đã xác định đó còn đang trống (chưa được dùng) trong thủ tục ghi vào hay thông tin dữ liệu cần lấy ra đang nằm ở đâu khi làm thủ tục đọc.

– Theo các chức năng nêu trên, có thể phân các bộ nhớ thành hai loại cơ bản là bộ nhớ chỉ cho phép đọc ROM (Read Only Memory) và bộ nhớ cho phép cả ghi vào hoặc đọc ra RAM (Random Access Memory/Read Write Memory).

– Thông số kỹ thuật quan trọng nhất để đánh giá chất lượng của một bộ nhớ gồm có dung lượng nhớ (là khả năng tối đa lưu giữ thông tin tính theo số lượng bit nhị phân có thể lưu giữ trong bộ nhớ) và tốc độ đọc hay tốc độ ghi/đọc khi thực hiện trao đổi thông tin lưu trữ trong nó ra môi trường bên ngoài.

Có thể ước tính dung lượng của một bộ nhớ qua tổng số các đầu vào, đầu ra của nó (xem ví dụ phía sau).

9.1.2. Cấu trúc khối tổng quát của một bộ nhớ

Hình 9.1 mô tả cấu trúc khối của một bộ nhớ gồm 4 khối cơ bản:

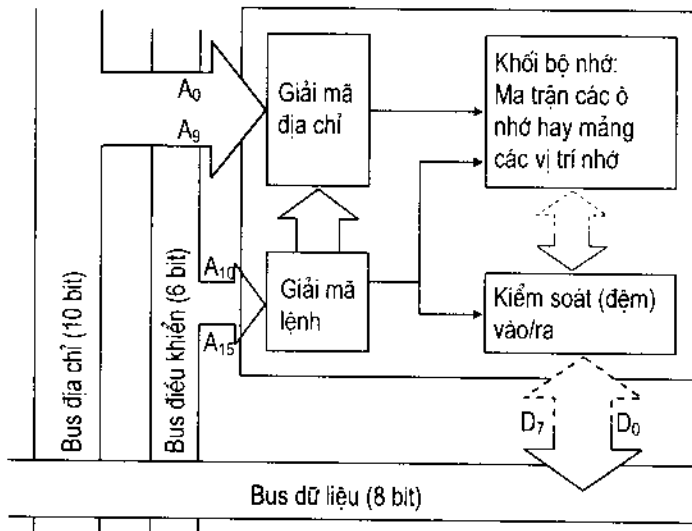
Khối giải mã địa chỉ để tìm chính xác vị trí nhớ cần sử dụng.

Khối giải mã lệnh để xác định chính xác lệnh (thao tác) bộ nhớ phải thực hiện.

Khối ma trận hay mảng (tuyến tính) các vị trí nhớ.

Khối điều khiển thủ tục xuất hay thủ tục xuất nhập dữ liệu.

Như vậy, một bộ nhớ giao tiếp với bên ngoài thông qua 3 kênh thông tin (bus): kênh điều khiển, kênh địa chỉ và kênh dữ liệu. Thông qua các bit lệnh trên kênh điều khiển, kênh địa chỉ đưa các bit mã địa chỉ tới tìm vị trí nhớ sẽ được sử dụng và phối hợp đồng bộ với kênh dữ liệu để "đón" trong thủ tục đọc hay "tiễn" trong thủ tục ghi các bit trên kênh dữ liệu. Thiết bị nhớ phải làm việc đồng bộ với các bộ phận khác trong một hệ thống kỹ thuật số nhờ các tín hiệu điều khiển đồng bộ trên kênh điều khiển.



Hình 9.1. Cấu trúc khối tổng quát của một bộ nhớ và 3 kênh thông tin tiếp xúc với một bộ nhớ

(mũi tên đứt nét là hướng dữ liệu hai chiều cho RAM)

Dung lượng tối thiểu $2^{10} \approx 1K$

Dung lượng tối đa $2^{16} \approx 64K$

Trên hình 9.1, bus địa chỉ có 10 bit ($A_9 \rightarrow A_0$) cho biết có thể quản lý được 2^{10} vị trí nhớ (hay 2^{10} ô nhớ trong ma trận) từ mã địa chỉ $(000)_{16} = (0000000000)_2$ đến mã địa chỉ $(3FF)_{16} = (1111111111)_2 = (1024)_{10}$.

Như vậy, dung lượng tối thiểu của bộ nhớ là $1024 \approx 10^3$ vị trí nhớ. Nếu mỗi vị trí là một hàng gồm 8 ô nhớ (có thể chứa được 1 byte dữ liệu) ta có dung lượng tối thiểu là 1kbyte. Nếu một vài bit điều khiển trong số các bit điều khiển $A_{10} \rightarrow A_{15}$ tham gia thêm vào quá trình giải mã địa chỉ thì dung lượng được mở rộng tương ứng ra 2^k vị trí nhớ (k là số bit điều khiển tham gia bổ sung). Khi tất cả các bit $A_{10} \rightarrow A_{15}$ đều hỗ trợ cho việc giải mã địa chỉ thì dung lượng tối đa của bộ nhớ là $2^6 \cdot 2^{10} = 65.536 \approx 64K$.

9.1.3. Nâng cao chỉ tiêu chất lượng của bộ nhớ

1. Tăng dung lượng của bộ nhớ để lưu giữ và trao đổi được nhiều thông tin hơn.

2. Giảm thời gian xuất nhập (thời gian đọc hay ghi/đọc dữ liệu gọi chung là thời gian trao đổi hay truy cập dữ liệu).

– Tuy nhiên, việc làm tăng dung lượng của bộ nhớ (qua đó mở rộng được năng lực và phạm vi hoạt động của hệ thống số) và làm giảm thời gian truy cập luôn gặp mâu thuẫn: chip IC nhớ có dung lượng càng lớn luôn kèm theo việc tìm kiếm dữ liệu trong nó vất vả và lâu hơn do đó làm chậm tốc độ hoạt động của cả hệ thống.

– Biện pháp có hiệu quả là phân cấp quản lý thông tin dữ liệu cho các bộ nhớ:

Các thông tin (dữ liệu hoặc lệnh) thường và cần thiết dùng ngay trong công tác có liên quan tới các thiết bị số khác được lưu giữ trong một bộ nhớ chính có dung lượng không cần lớn (đến vài chục kbyte) nhưng tốc độ nhập xuất nhanh (đến $10^{-9}s$) và được bố trí gần bộ phận xử lý dữ liệu. Các nội dung chưa cần dùng ngay hoặc dùng không thường xuyên được nhớ trong các bộ nhớ phụ có dung lượng lớn tới $10^{10} \rightarrow 10^{12}$ bit) có tốc độ truy cập chậm (đến $10^{-3}s$). Biện pháp kết hợp là xử lý xuất/nhập luồng dữ liệu kiểu song song khi ghép song song nhiều IC nhớ cỡ nhỏ hoặc cỡ vừa.

– Thời gian thâm nhập bộ nhớ gồm hai phần là thời gian cần thiết để tìm kiếm (định vị) được vị trí nhớ sẽ được sử dụng và thời gian để nhập (ghi) hay xuất (đọc) thông tin từ vị trí này tới kênh dữ liệu.

– Giá thành chi phí tính cho 1 bit nhớ của bộ nhớ là thông số kinh tế kỹ thuật của một bộ nhớ.

9.2. CÁC THUỘC TÍNH CỦA BỘ NHỚ CHỈ ĐỌC ROM

Các bộ nhớ ROM còn được gọi là các bộ nhớ chương trình chứa các tập hợp danh sách các lệnh cho phép khởi động một hệ thống thiết bị số (ví dụ máy tính) và điều khiển các hoạt động cơ bản nhất của hệ thống. Bộ nhớ ROM có các thuộc tính sau:

1. Chỉ cho phép đọc các nội dung đã được ghi sẵn từ trước chứa bên trong nó. Nội dung này do người điều hành thiết kế đã lập trình sẵn và ghi vào nó nhờ một phương pháp đặc biệt. Như vậy, dữ liệu chỉ truyền theo một hướng đi ra.

2. Người sử dụng về nguyên tắc không thể hoặc khó lòng thay đổi nội dung thông tin đã ghi trong ROM.

3. Nội dung đã ghi trong ROM có tính cố định không bị mất đi theo thời gian hay do sự cố mất nguồn năng lượng cấp chung, không bị mất đi sau một lần đọc.

Nhờ các tính chất trên ROM được dùng để:

– Chứa các chương trình khởi động, vận hành máy, các chương trình hỗ trợ việc lập trình hay giám sát điều hành theo một trình tự cố định chặt chẽ.

– Chứa các chương trình cài đặt chuyên dụng cho máy tính để thực hiện một nhóm công việc có giới hạn, được lập lại có quy luật.

– Lưu giữ các bảng biểu và các công cụ (toán học) thường xuyên được sử dụng cho việc xử lý tính toán.

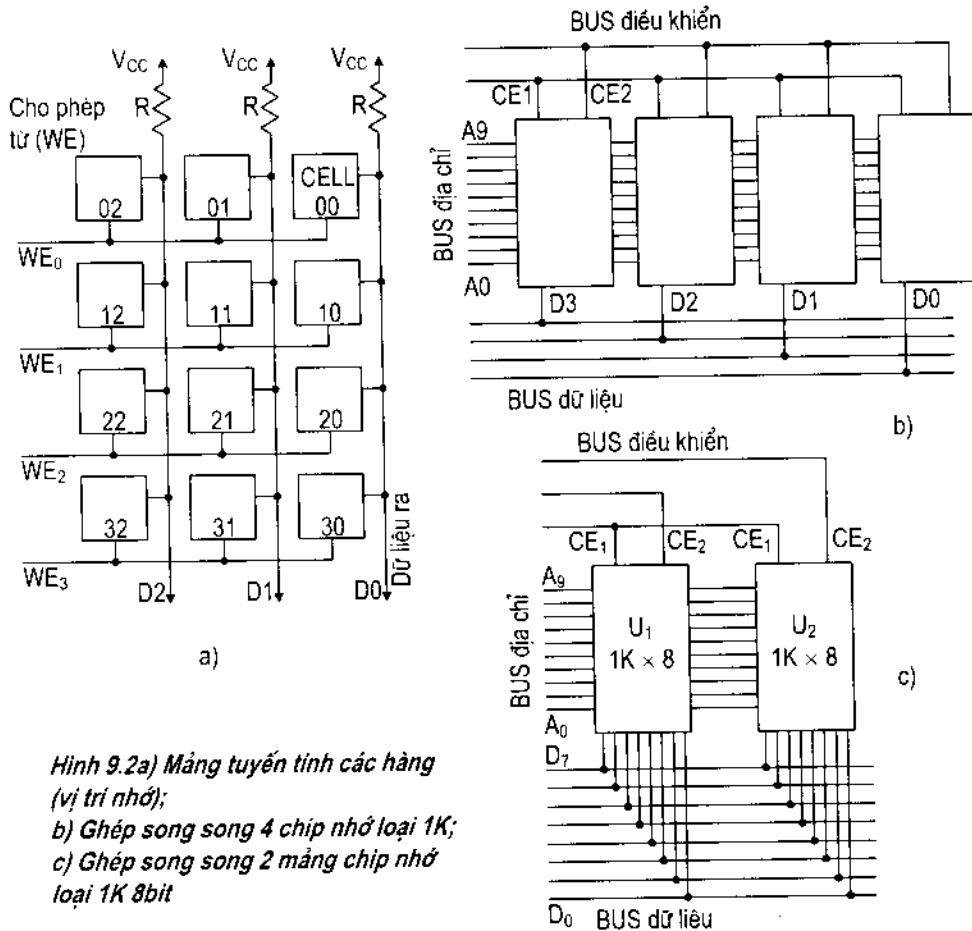
9.3. CẤU TRÚC ROM

Gồm 4 khối như trên hình 9.1 (không có đường nét đứt)

9.3.1. Khối bộ nhớ

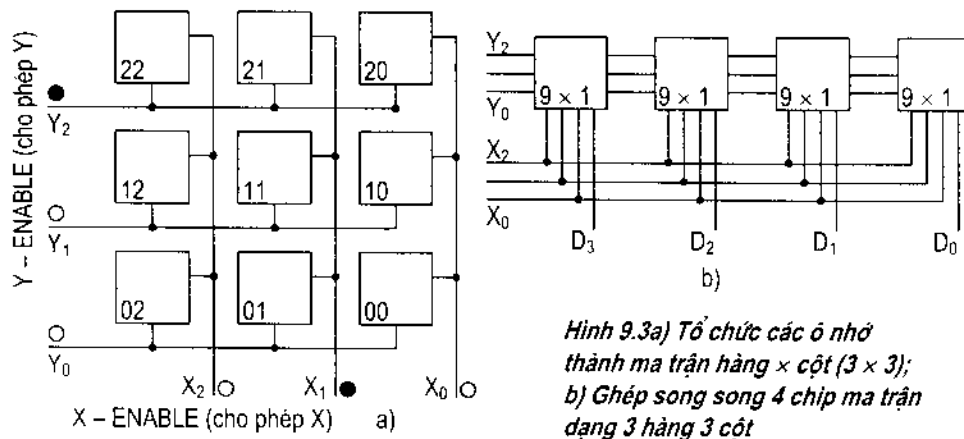
Phần tử cơ bản của một khối bộ nhớ là một ô nhớ nhị phân dùng Flip Flop (ví dụ một DFF). Tập hợp một nhóm các ô nhớ tạo thành

một hàng (còn gọi là một vị trí nhớ), cấu trúc một chip nhớ kiểu mảng tuyến tính bao gồm các hàng các ô nhớ (hình 9.2a) mỗi hàng được gọi là một từ. Ghép nhiều chip nhớ kiểu song song ta được một khối bộ nhớ như trên hình 9.2b có dung lượng $4 \times 1k$ hay trên hình 9.2c có dung lượng $2 \times 8k$.



Hình 9.2a) Mảng tuyến tính các hàng (vị trí nhớ);
b) Ghép song song 4 chip nhớ loại 1K;
c) Ghép song song 2 mảng chip nhớ loại 1K 8bit

Cách tổ chức thứ hai là các ô nhớ sắp xếp theo hàng (Y) và cột X, ở dạng một ma trận như trên hình 9.3a. Chỉ một vị trí giao hai đường cho phép (trên hình 9.3a là vị trí ô nhớ 2.1) mới được quyền xuất giá trị bit đã lưu trên nó. Ghép song song nhiều chip ma trận các ô nhớ hình thành cấu hình khối bộ nhớ. Hình 9.3b là cách ghép 4 chip loại ma trận 3×3 . Tổ chức kiểu ma trận các ô nhớ thích hợp cho các bộ nhớ có dung lượng lớn với n hàng và m cột, dung lượng một ma trận chip gồm $n \times m$ bit nhớ, nếu ghép song song k chip để tạo nên khối bộ nhớ sẽ đạt được dung lượng $k \times m \times n$ bit.

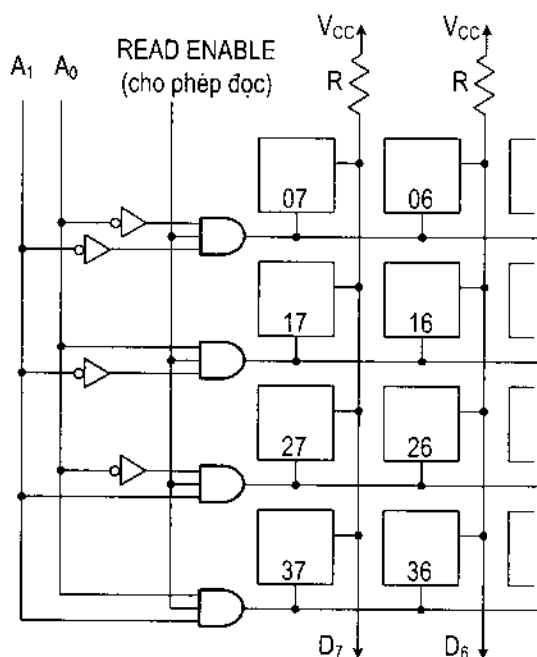


Hình 9.3a) Tổ chức các ô nhớ thành ma trận hàng $\times$ cột (3×3);
b) Ghép song song 4 chip ma trận dạng 3 hàng 3 cột

9.3.2. Khối giải mã địa chỉ

Khối giải mã địa chỉ có nhiệm vụ làm giao diện giữa bus địa chỉ và khối ô nhớ có khả năng chọn 2^n địa chỉ trên một số ít (n) đường truyền.

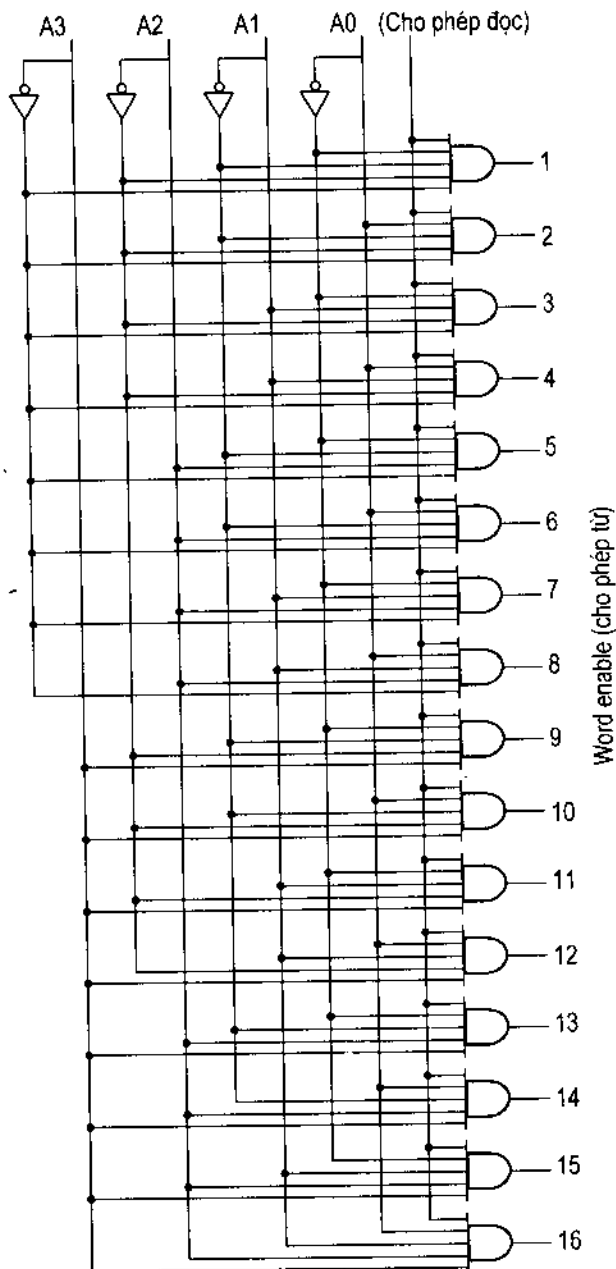
Tùy theo cấu hình các chip nhớ, dạng mạch giải mã địa chỉ cũng có hai loại phù hợp. Hình 9.4 là mạch giải mã cho cấu trúc mảng tuyến tính tương ứng với cấu trúc ô nhớ hình 9.2a (mạch giải mã 1 trong 4 đường truyền) và hình 9.5 là bộ giải mã 1 trong 16 tương ứng với mảng tuyến tính cần có 16 tín hiệu cho phép $WE_0 \rightarrow WE_{15}$ điều khiển một trong các hàng lên tích cực (được chọn làm việc).



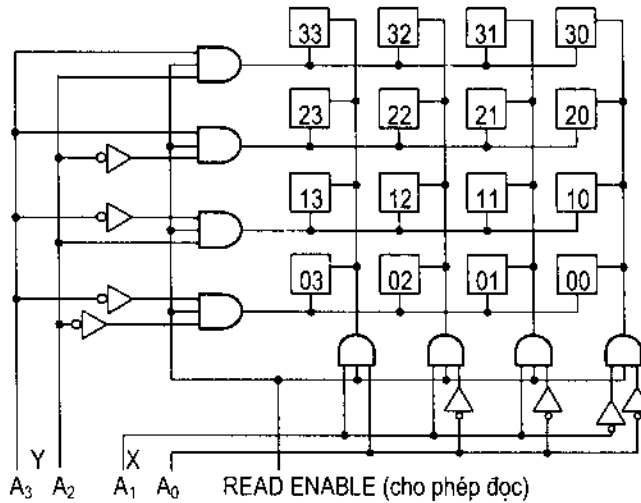
Hình 9.4. Dùng AND và NOT trong bộ giải mã địa chỉ của mảng tuyến tính. Bộ nhớ 4 từ đòi hỏi địa chỉ 2 bit

Ta có nhận xét là khi số bit địa chỉ tăng lên, mạch giải mã sẽ tăng nhanh mức độ phức tạp và số lượng các linh kiện.

– Hình 9.6 là mạch giải mã cho ma trận ô nhớ 16 bit, khi đó ta cần 2 bộ giải mã 1 trong 4 cho địa chỉ hàng và địa chỉ cột. Như vậy, cấu trúc mạch giải mã cho loại tổ chức ô nhớ dạng ma trận đơn giản và kinh tế hơn.



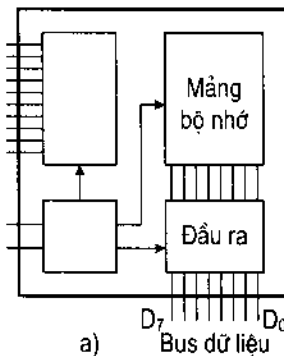
Hình 9.5. Giải mã địa chỉ 4 bit



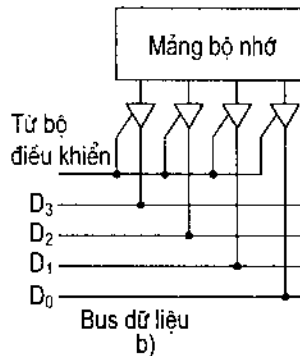
Hình 9.6. Bộ giải mã cho ma trận 16 bit chỉ đòi hỏi 8 chip AND

9.3.3. Mạch điện đầu ra của ROM

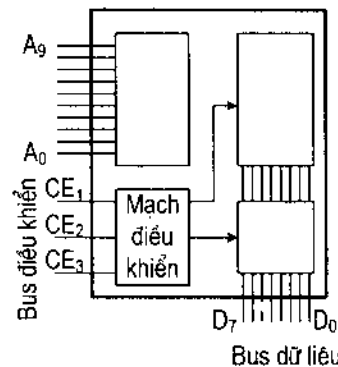
Mạch điện đầu ra của ROM có vai trò là giao diện giữa khối các ô nhớ (hay vị trí nhớ) với bus dữ liệu làm nhiệm vụ kết nối dữ liệu đã được chọn theo địa chỉ xác định tại khối ô nhớ với kênh dữ liệu vào lúc thích hợp. Mạch điện đầu ra điển hình dùng các phần tử ba trạng thái (hình 9.7) với mục đích ngắt việc kết nối giữa khối nhớ và kênh dữ liệu khi bộ nhớ không được chọn làm việc. Các phần tử đệm 3 trạng thái thường là loại hở colectơ (OC) để có hệ số tải cao kích thích kênh dữ liệu với vô số các tải khác cùng nối tới kênh để đón dữ liệu được xuất khỏi ROM.



a)



b)



Hình 9.8. Khối mạch giải mã lệnh để tạo lệnh điều khiển

Hình 9.7. Vị trí của mạch đầu ra (a) và cấu tạo mạch ra dùng các phần tử 3 trạng thái tương ứng (4 bit ra) (b)

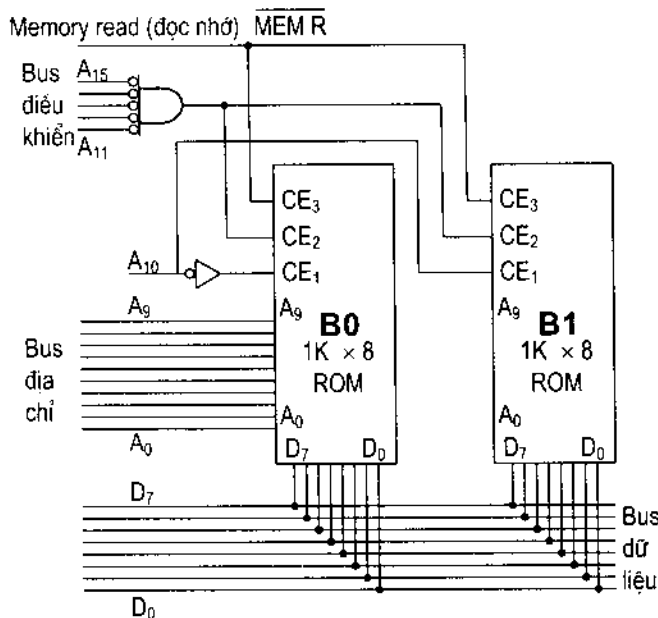
9.3.4. Mạch điều khiển trong ROM có chức năng tạo các tín hiệu điều khiển

Khối mạch điều khiển trong bộ nhớ có các nhiệm vụ chính là (hình 9.9):

- Chọn chip cần đọc khi bộ nhớ có nhiều chip đấu song song nhau (CE_1).
- Định thời gian cho phép thực hiện kết nối ROM với kênh dữ liệu (CE_2).
- Tham gia thêm vào tiến trình giải mã để mở rộng khả năng kiểm soát các ô nhớ nâng cao độ chính xác và tin cậy cho việc chọn địa chỉ (CE_3).

Mạch điện điều khiển có vị trí như trên hình 9.8 và có cấu trúc thể hiện trên ví dụ hình 9.9 với 3 chức năng điều khiển mà tín hiệu CE (chip select) thực hiện:

Bộ nhớ hình 9.9 có chức năng 2 chip nhớ loại 1K (gọi là 2 trang nhớ – pages) hay hai khối. Bus địa chỉ của máy tính có 16 bit ($A_0 \rightarrow A_{15}$), các bit $A_0 \rightarrow A_9$ được nối song song và giải mã ở bên trong mỗi chip. Tuy vậy, chỉ chip nào có xuất hiện đồng thời 3 đầu vào CE_1 , CE_2 , CE_3 ở mức tích cực 0 thì nó mới được phép xuất dữ liệu ra: chip B_0 có 11 bit địa chỉ là: $A_{10}A_9 \dots A_0 = 00 \dots 0$ đến $01 \dots 1$, còn chip B_1 có 11 bit địa chỉ là: $A_{10}A_9 \dots A_0 = 10 \dots 0$ đến $11 \dots 1$.



Hình 9.9. Dùng đầu vào CE để điều khiển việc chọn chip cho ROM

Hai hệ thống địa chỉ (11 bit) trên cho 2 chip B_0 và B_1 với điều kiện các bit địa chỉ cao hơn $A_{11} \rightarrow A_{15}$ đều ở mức 0 để không nhầm lẫn (với khu vực khác).

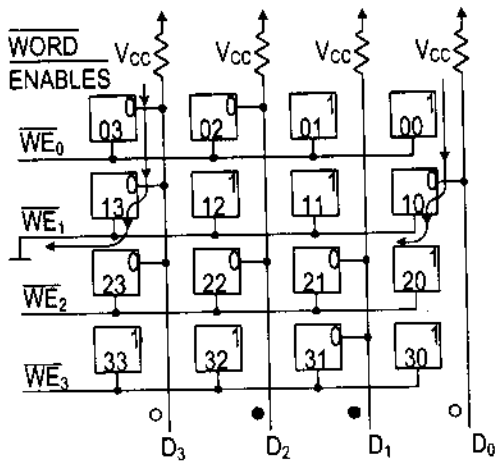
Khi $A_{15}A_{14}A_{13}A_{12}A_{11} = 00000$ thì qua cổng NOR có $CE_2 = 0$. Các giá trị khác $A_{15}A_{14}A_{13}A_{12}A_{11} = 10000$ đến 11111 đều cho $CE_2 = 1$ và chip B_0 và B_1 do đó không được chọn làm việc. Khi đó nếu sử dụng tối đa các bit địa chỉ, bộ nhớ ROM có dung lượng tối đa là $2^6 \cdot 2^{10} \approx 64Kb$, quản lý và chọn 1 trong 64 chip nhớ (từ B_0 đến B_{63}) cho làm việc. Đầu CE_3 dùng để định giờ hoạt động cho ROM (đến một thời điểm đã xác định trước mỗi xuất hiện CE_3 tích cực cho phép ROM gồm 2 chip B_0 hoặc B_1 hoạt động tùy theo giá trị của bit A_{10} .

9.4. CÁC DẠNG ROM THƯỜNG GẶP

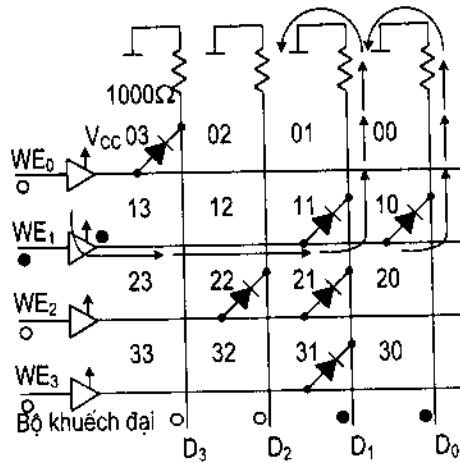
Điểm khác biệt quan trọng của ROM là cấu tạo vật lý các ô nhớ: dùng diốt, dùng BJT hay FET như được thể hiện trên các hình 9.11, 9.12 và 9.13 gọi chung là các ROM lập trình bằng mặt nạ (mask programmed ROM).

9.4.1. Cấu trúc khối của loại ROM lập trình bằng mặt nạ

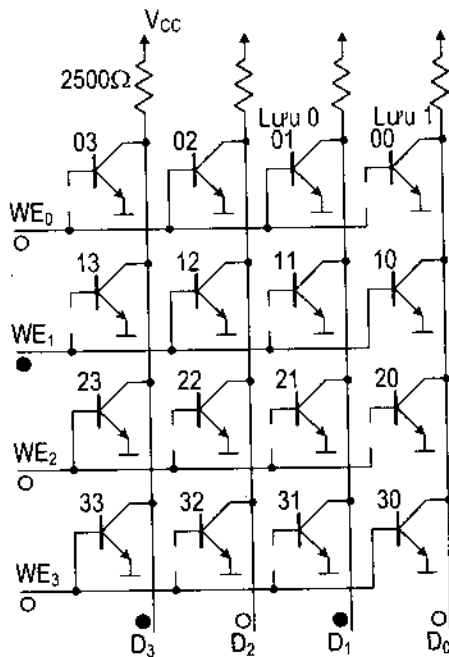
Cấu trúc khối của loại ROM lập trình bằng mặt nạ cho trên hình 9.10. Dữ liệu nhớ được nạp ngay trong quá trình chế tạo (theo ý định của người đặt hàng khi cung cấp danh sách dữ liệu cần lưu trong ROM cho nhà sản xuất). Các ô nhớ trên hình 9.10 nếu được nối dây (đường mũi tên đậm nét) thể hiện đã được nhớ bit 0, còn các ô không được nối thể hiện đã lưu bit 1. Để đọc số đã lưu trên một hàng (ví dụ hàng thứ hai, trên hình 9.10) cho $\overline{WE}_1 = 0$, lúc đó các ô 13 và 10 có dòng qua (được nối dây) nên $D_3 = D_0 = 0$ (ở thế thấp) còn các ô 1.2 và 1.1 không có dòng qua (không nối dây) nên $D_2 = D_1 = 1$ (ở thế cao). Trên hình 9.11 với ROM mặt nạ dùng diốt, do có $WE_1 = 0$ hàng với các ô nhớ 13, 12, 11, 10 được kích hoạt và các số đã lưu 1100 được đọc ra $D_3D_2D_1D_0 = 1100$.



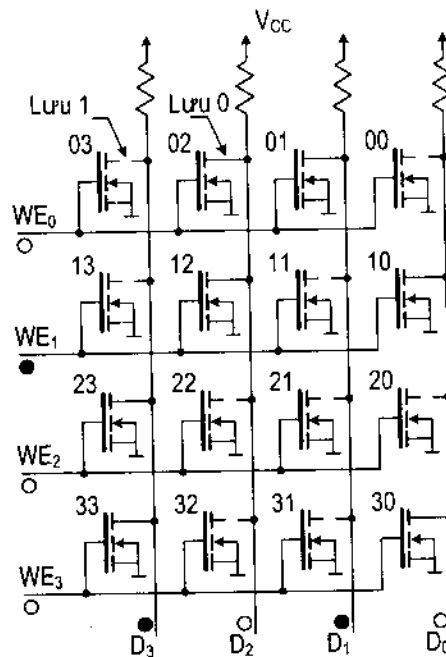
Hình 9.10. Mô hình ROM mật nã với 4 hàng 4 bit dữ liệu
(Đang đọc hàng 13 12 11 10 với dữ liệu ra $D_3D_2D_1D_0 = 0110$)



Hình 9.11. ROM mật nã dùng điốt nối mạch các bit nhớ 1
(Đang đọc hàng 13 12 11 10 do $WE_1 = 1$ giá trị dữ liệu đọc ra là $D_3D_2D_1D_0 = 0011$)



Hình 9.12. ROM mật nã dùng BJT
(Đang đọc dòng 2: $D_3D_2D_1D_0 = 1010$ do $WE_1 = 1$)



Hình 9.13. ROM mật nã dùng FET (NMOS)
(Đang đọc dòng 2: $D_3D_2D_1D_0 = 1010$ do $WE_1 = 1$)

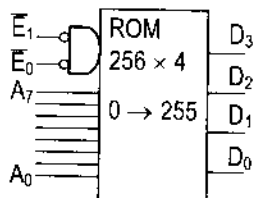
ROM mặt nạ có nhược điểm là thời gian chế tạo tại hãng sản xuất lâu (vài tuần từ lúc đặt hàng để hãng lập trình đến lúc giao hàng), khi cần thay đổi dù rất ít số đã lưu trong chip ROM phải làm lại chip mới và nếu sử dụng với số lượng ít giá thành rất cao cho chi phí sản xuất. Các sản phẩm tiếp sau với ROM cho phép lập trình (programmable – PROM) hay cho phép xóa và lập trình lại (EPROM – Erasable PROM) khắc phục được các nhược điểm nêu trên với đặc điểm là có tính năng động cao hơn hẳn.

9.4.2. ROM cho phép lập trình (PROM)

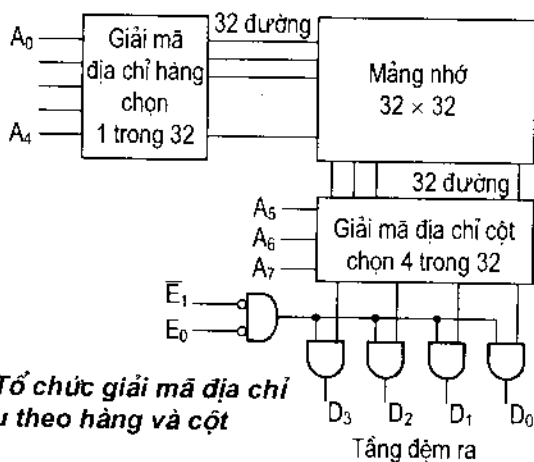
Ưu thế quan trọng nhất của PROM là để người sử dụng nó tự lập trình. Hình 9.17 cho cấu tạo một PROM dùng BJT có nối cầu chì ở cực emitơ của mỗi ô nhớ dùng tranzito lưỡng cực sau khi chế tạo xong. Người sử dụng khi cần sẽ tự nạp dữ liệu vào từng ô nhớ bằng cách làm đứt cầu chì tại ô nhớ lưu trị 1 (nhờ một nguồn điện thế ngoài cỡ +12,5V nối tới các colectơ (theo cột của ma trận) và +5V tới các bazơ (theo hàng của ma trận). Hình 9.17 thể hiện PROM đã được nạp xong với các dữ liệu là: $D_{03} D_{02} D_{01} D_{00} = 1001$; $D_{13} D_{12} D_{11} D_{10} = 1010$; $D_{23} D_{22} D_{21} D_{20} = 0111$; $D_{33} D_{32} D_{31} D_{30} = 0110$.

PROM chỉ cho phép người sử dụng lập trình một lần (không làm lại được, chỉ một sai sót nhỏ (1 bit) là làm hỏng cả chip).

Trên hình 9.14 là ký hiệu logic của IC ROM 74163 có dung lượng nhớ 256×4 (1024 bit tổ chức thành 256 từ 4 bit). Với 8 bit địa chỉ, nhờ mạch giải mã $1/256$ sẽ đọc ra được 256 từ riêng biệt (từ 0 đến 255). Tuy nhiên, việc tổ chức địa chỉ theo một chiều như vậy không thích hợp khi dung lượng tăng. Thực tế, việc tổ chức địa chỉ được tổ chức kiểu hai chiều theo hàng và theo cột như được mô tả trên hình 9.15.



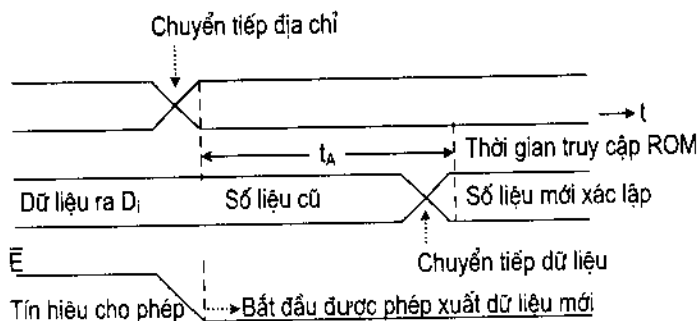
Hình 9.14. Ký hiệu logic của ROM 74163



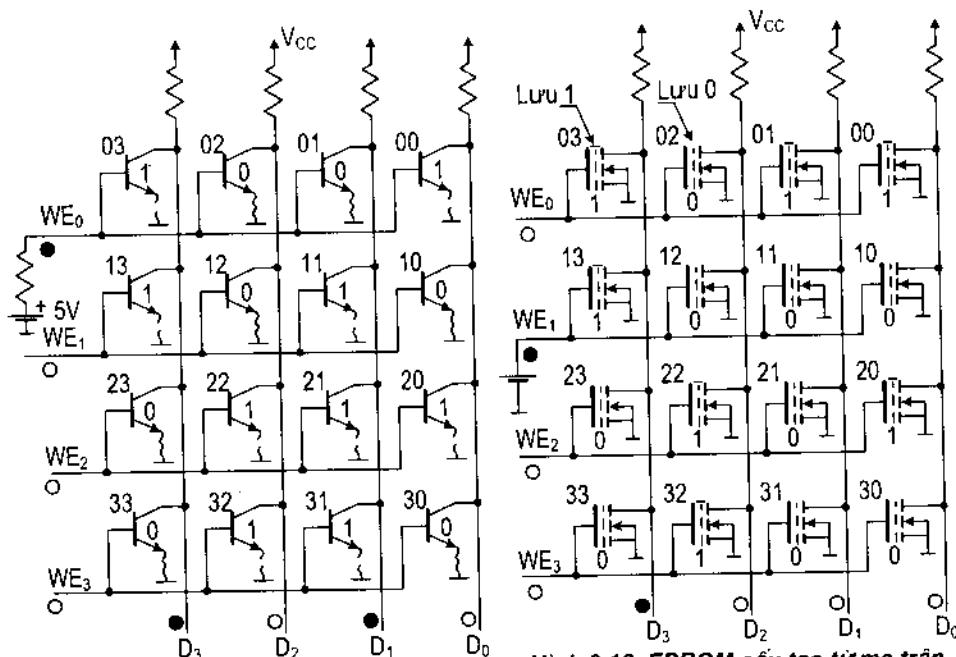
Hình 9.15. Tổ chức giải mã địa chỉ 2 chiều theo hàng và cột

Khi giải mã địa chỉ hai chiều, 5 bit $A_0 \rightarrow A_4$ dùng cho việc chọn hàng (RAS – Row Access Select) 3 bit còn lại A_5, A_6, A_7 dùng cho việc chọn cột (CAS – Column Access Select). Mảng các ô

nhớ 32×32 gồm 4 mảng tuyến tính 32×8 nên có dạng giải mã cột là 4 mạch chọn kênh (MUX) 1 trong 8. Số liệu mỗi lần đọc ra $D_3 \rightarrow D_0$ là một từ nhớ 4 bit đã lưu trong địa chỉ vừa chọn, hình 9.16 đưa ra đồ thị thời gian mô tả quá trình truy cập ROM và cách xác định thời gian truy cập từ lúc chuyển tiếp địa chỉ vừa (giải mã xong) đến lúc chuyển tiếp xong dữ liệu từ cũ sang mới trên kênh dữ liệu.



Hình 9.16. Đồ thị thời gian mô tả quá trình xác lập dữ liệu mới khi được phép (E tích cực thấp)



Hình 9.17. PROM có cấu trúc cực emitter
(Khi làm đứt cấu trúc: lưu bit 1; khi không làm đứt: lưu bit 0)

Hình 9.18. EPROM cấu tạo từ ma trận EMOS kênh N có thêm cực cửa thả nổi
(Khi cực G tích điện âm là ô nhớ lưu mức 1, khi cực G không tích điện âm: ô nhớ lưu bit 0)

9.4.3. Chip ROM cho phép xóa và lập trình lại (EPROM – Erasable PROM)

– Cách lập trình cho EPROM tương tự như cho PROM, tuy nhiên EPROM cho phép xóa và nạp lại khi kết quả cuối cùng (có vài ô nhỏ) chưa đúng chuẩn. Hình 9.18 đưa ra cấu tạo một loại EPROM từ ma trận các ô nhớ loại NMOS (có bổ sung thêm một cực của transistor G_2 giữa cực G_1 và kênh). Khi G_2 được nạp điện tích âm thì FET bị khóa tương ứng với việc lưu (ghi) bit 1 vào ô nhớ này. Trường hợp ngược lại khi G_2 không được nạp, lúc $WE = 1$ (hàng thứ hai trên hình 9.18) FET dẫn điện, tương ứng ô nhớ đã được lưu bit 0. Sau khi nạp điện tích âm cho tất cả các cực G_2 , dữ liệu đã được nạp trong EPROM hình 9.15 là:

$$D_{03} D_{02} D_{01} D_{00} = 1011$$

$$D_{13} D_{12} D_{11} D_{10} = 1000$$

$$D_{23} D_{22} D_{21} D_{20} = 0101$$

$$D_{33} D_{32} D_{31} D_{30} = 0100$$

Trên hình vẽ $WE_1 = 1$ thể hiện hàng thứ hai đang có lệnh được đọc ra, xuất hiện dữ liệu của hàng này $D_{13} D_{12} D_{11} D_{10} = D_3 D_2 D_1 D_0 = 1000$ tại các cổng song song.

– Có hai phương pháp xóa EPROM: Phương pháp dùng tia cực tím mạnh chiếu trong thời gian cỡ 15 phút, qua một cửa sổ trong suốt được chế tạo sẵn, vào ma trận các MOS của chip qua đó giải phóng các electron đã nạp trên G_2 (xóa điện tích âm thể hiện lưu bit 1 trước đó), trả lại quyền hoạt động điều khiển của G_1 với dòng I_D của mỗi MOSFET: chip EPROM đã được xóa (làm sạch toàn bộ) để chuẩn bị nạp dữ liệu mới.

Cách thứ hai là dùng phương pháp tĩnh điện bằng cách: đặt lên G_1 một điện thế dương đủ lớn (cỡ 25V) làm mất lớp điện tử đã nạp vào G_2 trước đây. Kỹ thuật này cho phép xóa chọn lọc một vài từ hay một vài bit và chip nhớ loại này có tên là EEPROM (electrically erasable PROM).

– Lúc nạp EPROM cần điện thế cao hơn mức chuẩn (để đưa điện tích âm vào G_2). Thường chọn $U_{GS} = 25V$ và $U_{DS} = 20V$.

9.5. CÂU HỎI VÀ BÀI TẬP

1. Bộ nhớ có nhiệm vụ gì?

Hãy nêu các đầu vào, đầu ra tối thiểu của một bộ nhớ bán dẫn sau khi vẽ cấu trúc khối của nó và các bus mà nó giao tiếp.

Tại sao có thể dùng một Flip Flop (ví dụ DFF) làm một ô nhớ một bit nhị phân? Các thủ tục tối thiểu để một bộ nhớ hoạt động được là gì?

2. Phân biệt các dạng bộ nhớ theo chức năng:

a) Thuộc tính khác nhau của ROM và RAM.

b) Dung lượng của một bộ nhớ là gì? Cách tính dung lượng của một bộ nhớ dựa trên những đặc điểm nào? Cho ví dụ minh họa.

3. Nói rằng dung lượng của một bộ nhớ là 64kbyte và của một bộ nhớ khác là 512kbit có gì khác nhau? Cần dùng bao nhiêu đường địa chỉ $A_0, \dots, A_k$ cho mỗi loại nhớ trên?

Bộ nhớ chính và bộ nhớ phụ có đặc điểm gì khác nhau?

4. Tại sao lại muốn tăng dung lượng của một bộ nhớ?

Bằng cách nào có thể làm tăng dung lượng?

Thời gian thâm nhập (truy cập) bộ nhớ là gì? Mối liên hệ giữa thời gian truy cập và dung lượng của một bộ nhớ?

5. a) Hãy nêu các thuộc tính cơ bản của ROM.

b) Vẽ cấu trúc khối của một ROM và giải thích nhiệm vụ các khối trong ROM.

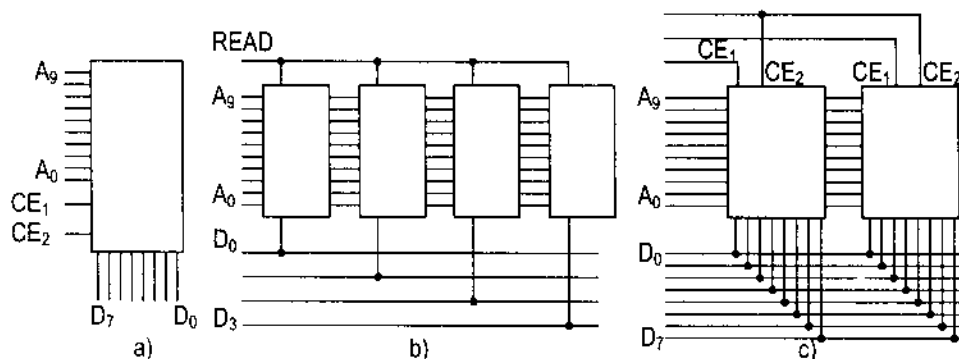
c) Khối mạch điều khiển có những nhiệm vụ gì? Nhiệm vụ nào là quan trọng nhất?

6. Phân biệt hai loại cấu trúc các ô nhớ: cấu trúc mảng tuyến tính và cấu trúc ma trận hàng cột các ô nhớ về đặc điểm cấu tạo, phương pháp giao tiếp giải mã và ưu nhược điểm của mỗi loại.

7. a) Hãy vẽ một ma trận các ô nhớ gồm 4 hàng, 4 cột, cần bao nhiêu bit đánh địa chỉ cho các hàng và cho các cột của ma trận này?

b) Khi nâng cấp lên thành ma trận 64 hàng và 8 cột sẽ cần bao nhiêu bit địa chỉ? Nếu các hàng của ma trận được kết cấu kiểu từng nhóm 8 hàng liên tiếp nhau thì cách dùng các bit địa chỉ hàng có gì khác?

- c) Một ma trận các ô nhớ có N_1 hàng, N_2 cột và N_3 lớp ghép song song với nhau. Vậy tổng số ô nhớ của bộ nhớ là bao nhiêu? Cần bao nhiêu bit địa chỉ quản lý các vị trí ô nhớ khi $N_1 = 64$; $N_2 = 8$ và $N_3 = 16$?
8. Mạch điện đầu ra của ROM có đặc điểm gì?
Tại sao phải dùng phần tử 3 trạng thái có colectơ để hỗ trợ thực hiện quản lý lỗi ra? Vai trò của mạch điều khiển đối với mạch ra.
9. ROM mặt nạ có đặc trưng gì là quan trọng nhất?
ROM mặt nạ dùng các điốt được thực hiện như thế nào? Nêu cách nạp bit 1 và bit 0 trong các trường hợp ROM mặt nạ có cấu trúc vật lý:
a) Dùng ma trận ô nhớ là các điốt.
b) Dùng ma trận các ô nhớ là các tranzito lưỡng cực (BJT) có cấu chỉ ở emitter.
c) Dùng tranzito trường MOSFET.
- 10 a) PROM và ROM có gì khác nhau? Ưu điểm của PROM là gì?
Tại sao nói EPROM có nhiều ưu điểm hơn PROM và ROM mặt nạ?
b) Phương pháp nạp cho PROM có đặc điểm gì?
c) Có khả năng nạp lại cho PROM không? Vì sao?
- 11 Hãy nêu phương pháp nạp dữ liệu cho EPROM và các phương pháp xóa nó. Cửa sổ của EPROM sau khi nạp dữ liệu cần phải che lại vì lý do gì?
- 12 a) Hãy xác định dung lượng của các bộ nhớ trên hình 9.19 a, b, c.
Nêu ý nghĩa các đường vào/ra của chúng.
b) Hãy dùng 2 đường A_9A_8 để chọn vỏ cho các chip nhớ 0, 1, 2, 3 trên hình 9.19b sau đó vẽ lại cấu trúc cho phù hợp.
c) Câu hỏi tương tự cho hình 9.19c với hai bit A_9A_8 . Nếu chọn các bit cao hơn ($A_{10}A_{11}$) có được không? Phân biệt hai trường hợp này.

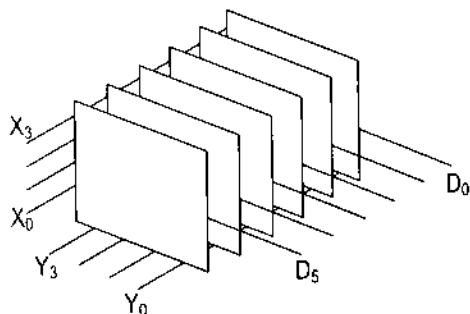


Hình 9.19. Cho bài tập 12

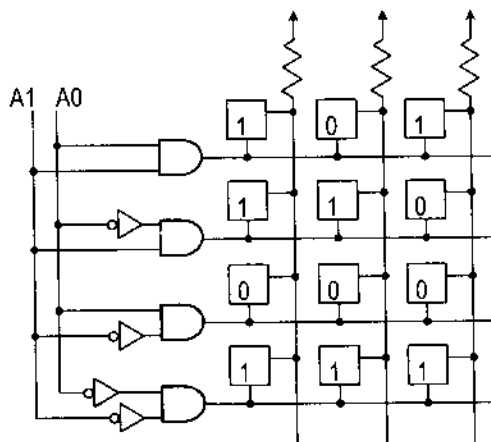
13. Bộ nhớ trên hình 9.20 có dung lượng nhớ là bao nhiêu bit?

Để chọn vị trí nhớ và các hàng trên nó cần mấy bit địa chỉ?

Khi mở rộng các cột gồm N đường: $X_0 \dots X_{N-1}$. Các hàng gồm M đường ($Y_0 \dots Y_{M-1}$) và K mảng ($K_0 \dots K-1$) thì dung lượng của bộ nhớ là bao nhiêu bit với $K = 16$ thì dung lượng là bao nhiêu bit, với $K = 16$ thì dung lượng là bao nhiêu kbyte khi $N = 32$; $M = 8$.



Hình 9.20. Mở rộng dung lượng dùng các mảng chip loại ma trận đầu song song



Hình 9.21. Cho bài tập 14

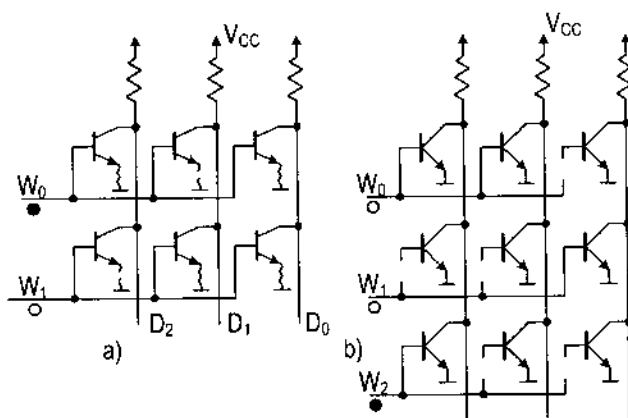
14 a) Ma trận ô nhớ 4 hàng 3 cột của một chip nhớ cho trên hình 9.21.

Hãy viết dữ liệu ra khi mã địa chỉ vào sau 8 nhịp làm việc là 10 11 01 00 01 11 00 10; dữ liệu ra $D_2D_1D_0 = 110$ là đúng hay sai?

Hãy sửa lại cho đúng nếu bị sai.

b) Hãy dùng các cổng ra ba trạng thái để kiểm soát dữ liệu ra.

15. Xác định dữ liệu đã nạp và dữ liệu đọc ra đối với hai ma trận nhớ của ROM cho trên hình 9.22.



Hình 9.22. Hai ma trận nhớ của ROM:

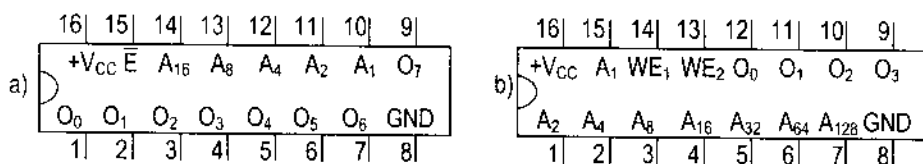
- Mức tích cực cao (giá trị 1);
- Mức tích cực thấp (giá trị 0)

16. Cho các IC PROM

7603 và 7611 với sơ đồ ký hiệu bố trí chân trên hình 9.23.

a) Hãy phân biệt và nêu chức năng các chân bố trí trên hình vẽ và xác định dung lượng nhớ của mỗi loại.

b) Việc nạp dữ liệu cho các IC này là thực hiện đốt cháy cầu chì. Hãy giải thích cách nạp dữ liệu 0 và dữ liệu 1 vào các ô nhớ của chúng.



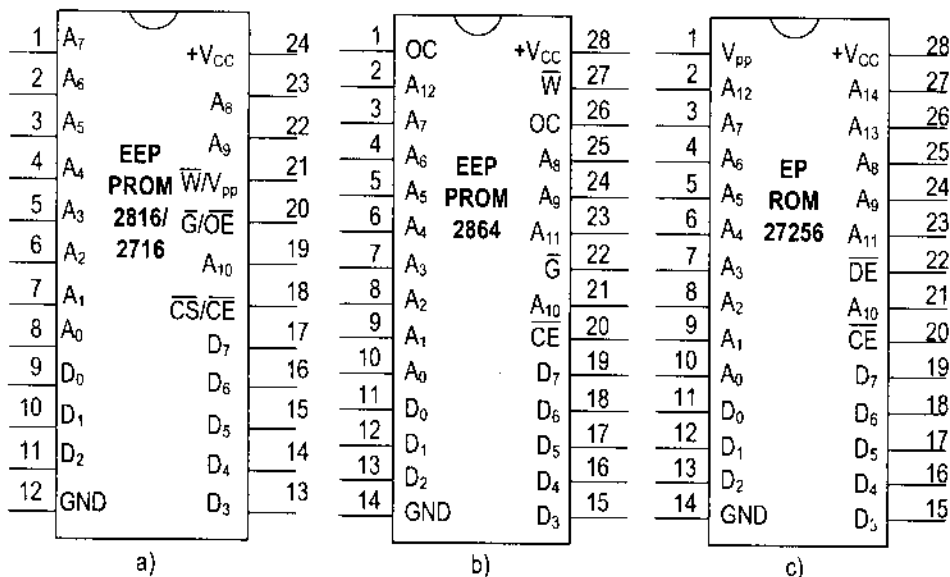
Hình 9.23. Bố trí chân các IC PROM 7603 (a) và 7611 (b)

17. Hình 9.24 đưa ra các sơ đồ bố trí chân của các IC EEPROM loại 2816/2716, 2864 và EPROM 27256 tương ứng.

a) Giải thích nguyên lý thực hiện ghi dữ liệu vào IC và xóa dữ liệu đã ghi trên chúng (chú ý, khi ghi dữ liệu V_{pp} được nối tới nguồn cỡ +25V).

b) Hãy chỉ rõ chức năng các đầu vào, đầu ra của các IC đã cho và tính dung lượng nhớ của mỗi loại.

(chú ý ký hiệu $\bar{G}$ là lệnh cho phép ra, $\bar{W}$ là lệnh cho phép đọc).



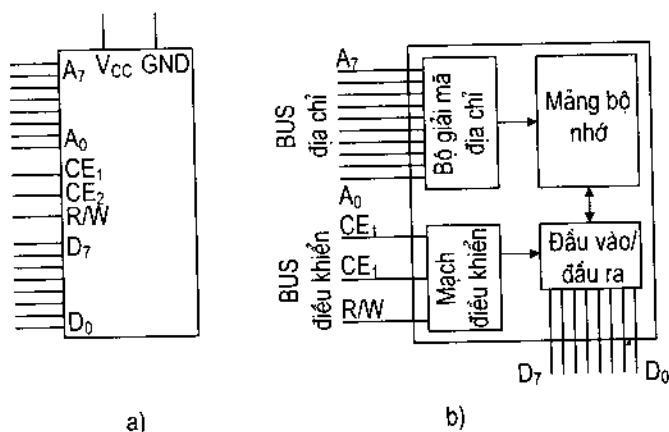
Hình 9.24a) EEPROM 2816/2716; b) EEPROM 2864; c) EPROM 27256

BÀI 10. BỘ NHỚ TRUY CẬP NGẪU NHIÊN (RAM)

10.1. KHÁI NIỆM

Bộ nhớ RAM (Random Access Memory) cho phép ghi để lưu trữ dữ liệu trong một thời gian nào đó, sau đó có thể đọc dữ liệu đã ghi để tiếp tục xử lý. Như vậy, dữ liệu được ghi trong RAM có tính chất tạm thời, cần đề phòng khả năng mất dữ liệu có ích còn trong RAM trong các quá trình xử lý nhờ các nguồn dự phòng nóng.

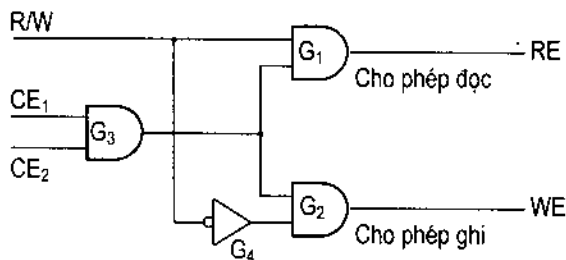
– Cấu trúc khối của RAM cũng gồm 4 khối chính như trong hình 9.1 đã mô tả (hình 10.1a và b).



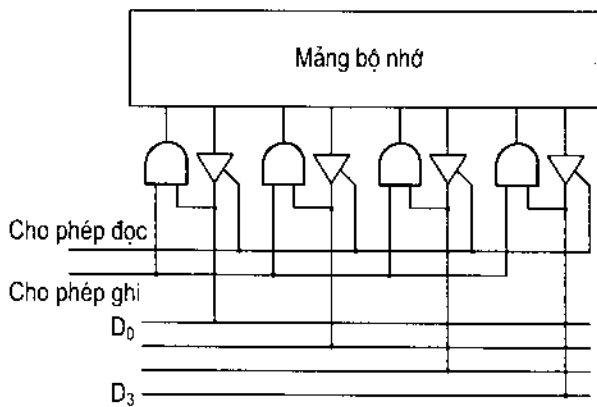
**Hình 10.1a) Các đầu vào ra của một IC RAM;
b) Cấu trúc khối của một RAM**

(Chú ý điểm khác biệt quan trọng là có đầu vào điều khiển đọc/ghi (R/W)).

Mạch điều khiển trong RAM nhất thiết phải có thêm đầu vào điều khiển hai quá trình cơ bản là đọc thông tin ra (Read) và ghi dữ liệu vào (Write) (hình 10.2). Mạch ra của RAM phức tạp hơn ROM là các phân tử ba trạng thái loại hai hướng như thể hiện trên hình 10.3 với bốn bit dữ liệu.



Hình 10.2. Mạch điều khiển đặt bộ nhớ vào một trong ba chế độ: đọc, ghi, hoặc chờ



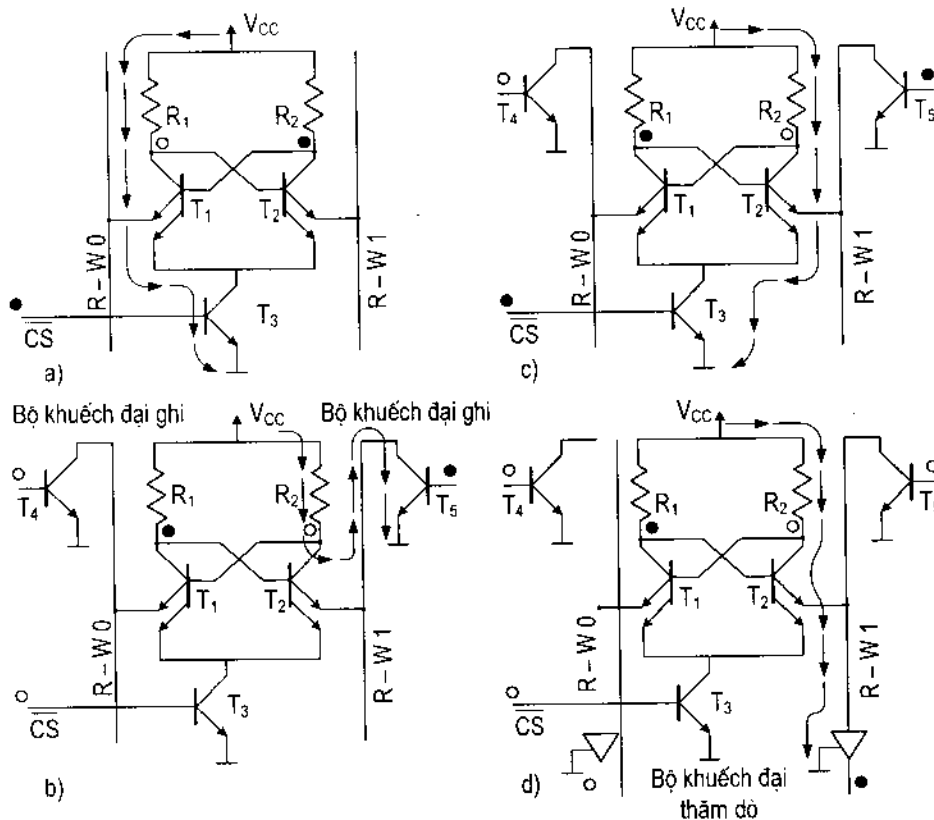
Hình 10.3. Mạch kiểm soát dữ liệu vào/ra của RAM

– Các phần tử ba trạng thái hai hướng cho phép RAM chọn lựa một trong ba chế độ đọc, chế độ ghi hay chế độ trở kháng ra cao (Z – Hi) gọi là chế độ chờ tùy theo mạch điều khiển hình 10.2.

Các thao tác tại mạch vào/ra phải đồng bộ với việc chọn (giải mã) xong địa chỉ dữ liệu tại khối nhớ. Mạch điều khiển hình 10.2 hoạt động như sau: ở chế độ đọc, xung R/W ở mức 1 đồng thời với $CE_1 = CE_2 = 1$. G_1 cho xung cho phép đọc $RE = 1$; G_2 đưa ra mức $WE = 0$ (cấm ghi). Ở chế độ ghi, xung R/W ở mức 0 qua G_4 có $\overline{R/W} = 1$ cùng với $CE_1 = CE_2 = 1$. G_2 cho xung cho phép ghi $WE = 1$, G_1 bị khóa bởi $R/W = 0$ cho ra mức $RE = 0$ (cấm đọc). Tín hiệu CE_1 và CE_2 (chip Select) cho phép chọn RAM làm việc phải xuất hiện đồng thời ở mức tích cực cao (mức 1). Khi $CE_1 \cdot CE_2 = 0$ (có ít nhất một trong hai tín hiệu ở mức không tích cực – mức 0), mạch điều khiển hình 10.2 chuyển RAM sang chế độ chờ (Standby) lúc này $RE = 0$; $WE = 0$ không phụ thuộc vào R/W bằng 1 hay bằng 0.

10.2. MẢNG Ô NHỎ CỦA RAM DÙNG TRANZITO NHIỀU EMITƠ

Khi các chip của RAM dùng DFF NAND loại chủ tớ (hình 8.11) làm một ô nhớ nhị phân, ưu điểm cơ bản là tốc độ nhập/xuất dữ liệu nhanh (cỡ từ 10^{-8} đến 10^{-9} s) tuy nhiên, với số lượng cho mỗi ô nhớ tới gần 20 BJT và hơn 6 điện trở, mạch với dung lượng nhớ lớn sẽ khá phức tạp và tiêu hao nhiều năng lượng. Một giải pháp tốt là sử dụng mỗi ô nhớ chỉ gồm vài BJT mắc kiểu đa hài lưỡng ổn (bistate multivibrator). Cấu trúc mạch dạng này cho trên hình 10.4.



Hình 10.4. FF dùng tranzito nhiều emitơ làm thành một ô nhớ nhị phân ba chế độ: chế độ chờ (a và c) chế độ ghi (b) và chế độ đọc (d)

– Mạch Flip Flop hình 10.4 tạo thành một ô nhớ của mảng nhớ (hình 10.5) có ba chế độ cơ bản: để phân tích các chế độ của FF ta quy ước: ô nhớ đang chứa bit 0 khi T_1 dẫn điện (T_1^+) và T_2 không dẫn (T_2^-), ngược lại ô nhớ ở mức 1 khi T_1^- T_2^+ . Dấu $\bullet$ là mức 1 và dấu $\circ$ là mức 0.

Khi $\overline{CS} = 1$ (hình 10.4a) T_3^+ vì ô nhớ đang chứa bit 0 ($T_1^+T_2^-$) nên mức 0 đang lưu trong FF được giữ lại: chế độ chờ của FF.

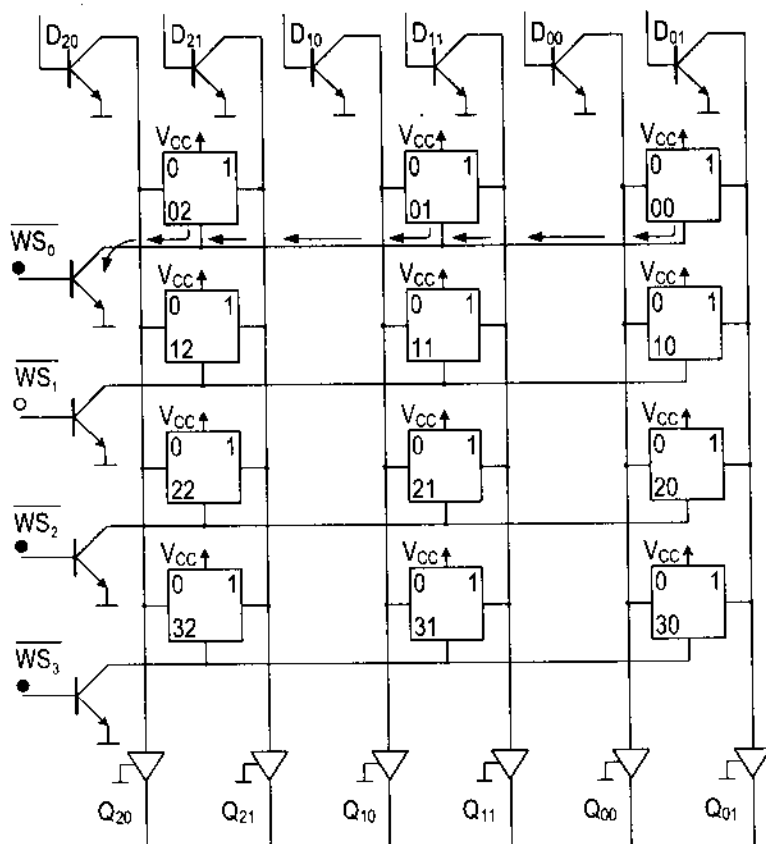
– Ở chế độ ghi (hình 10.4b) ví dụ muốn ghi bit 1 vào ô nhớ T_4 và T_5 làm nhiệm vụ khuếch đại xung ghi, khi T_5^+ (hoặc T_4^+ đường tín hiệu RW sẽ nối 0V ($RW_1 = 0$), đồng thời, lúc đó $\overline{CS} = 0$ T_3^- làm T_2^+ (theo cực emitter thứ hai T_1^- mạch FF về chế độ $T_2^+T_1^-$ (hay ô nhớ đã được ghi 1). Kết thúc quá trình này ô nhớ lại về chế độ chờ $\overline{CS} = 1$ (hình 10.4c).

– Ở chế độ đọc (hình 10.4d) mạch được bổ sung thêm phần tử khuếch đại thăm dò (Sense Amplifier – SA) có đặc điểm là điện trở vào nhỏ và có một đầu nối 0V. Khi có dòng kích thích tới đầu vào của SA, mức ra của nó $SA = 1$. Vậy muốn đọc dữ liệu đang lưu, yêu cầu $\overline{CS} = 0$ làm T_3^- dòng của T_2^+ về qua nhánh có emitter nối với đường $RW_1 = 0$, dẫn tới SA_1 bị kích thích có dòng tại lối vào và lối ra của nó có $SA_1 = 1$. Bit giá trị 1 đang lưu đã được đọc ra.

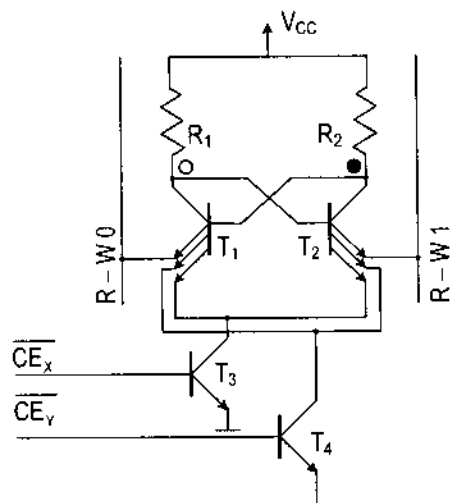
– Ô nhớ hình 10.4 được tổ chức thành ma trận hàng cột, ví dụ như trên hình 10.5 với 4 hàng 3 cột tạo thành một mảng tuyến tính 4 từ, mỗi từ 3 bit nhớ.

Đặc điểm của mạch hình 10.5 là các ô trong cùng một hàng (một từ mã được lưu giữ) có chung tranzito T_3 và đầu vào của T_3 sẽ có tín hiệu điều khiển chọn từ WE_i hay WS_i ($i = 0, 1, 2, 3$). Khi ô nhớ đang ở chế độ chờ dòng các ô trong cùng một từ đều qua T_3^+ và T_1^+ (nếu FF ở 0) hay T_2^+ (nếu FF đang ở 1). Khi cần đọc/ghi một từ nào đó (ví dụ trên hình là từ ứng với các ô 12, 11 và 10) cần đặt $WS_1 = 0$ làm T_3^- , dòng của T_1^+ hay T_2^+ lúc đó sẽ rẽ qua các dây cột tương ứng và thực hiện được thủ tục đọc/ghi đồng bộ với xung cho phép R/W.

– Khi cần thực hiện việc ghi/đọc cho từng bit của ma trận ô nhớ, cần sử dụng mỗi ô nhớ là loại BJT có ba emitter như hình 10.6. Hai cực emitter của T_1 và T_2 dùng để áp tín hiệu $\overline{CE}_X$ và $\overline{CE}_Y$ vào; chỉ khi $\overline{CE}_X = 0$ và $\overline{CE}_Y = 0$ (đồng thời ở tích cực) thì ô nhớ tương ứng được phép tham gia vào các hoạt động ghi/đọc. Khi có ít nhất một tín hiệu $\overline{CE}_X$ hay $\overline{CE}_Y$ ở mức cao thì ô tương ứng đang ở chế độ chờ.

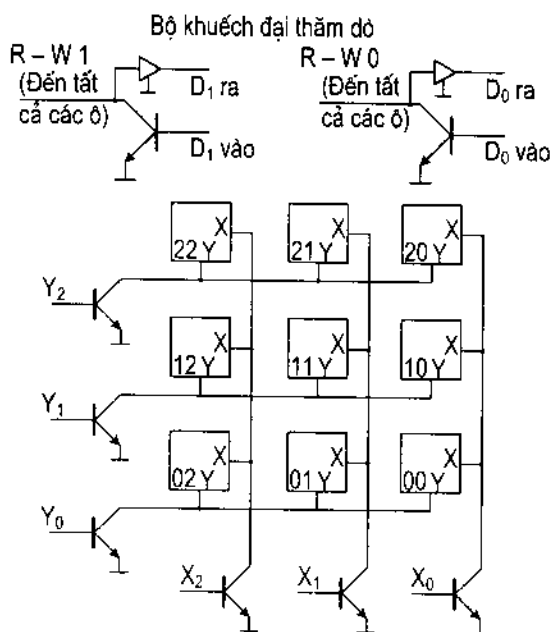


Hình 10.5. Mảng ô nhớ gồm 4 từ 3 bit kiểu chọn từng từ



Hình 10.6. Cấu tạo một ô nhớ dùng BJT có 3 cực emitter kiểu chọn bit (khi $\overline{CE_x} = 0$ và $\overline{CE_y} = 0$ đồng thời)

Mạch điện cấu tạo với kiểu ma trận các ô nhớ 9×1 bit được cho trên hình 10.7. Khi ở chế độ đọc mạch ra của một ô nhớ lấy dữ liệu ra qua bộ khuếch đại thăm dò SA và khi ở chế độ ghi dữ liệu qua một bộ khuếch đại ghi dùng tranzito, các đường RW_1 và RW_0 đưa tới tất cả các ô nhớ như được mô tả trên hình 10.7.

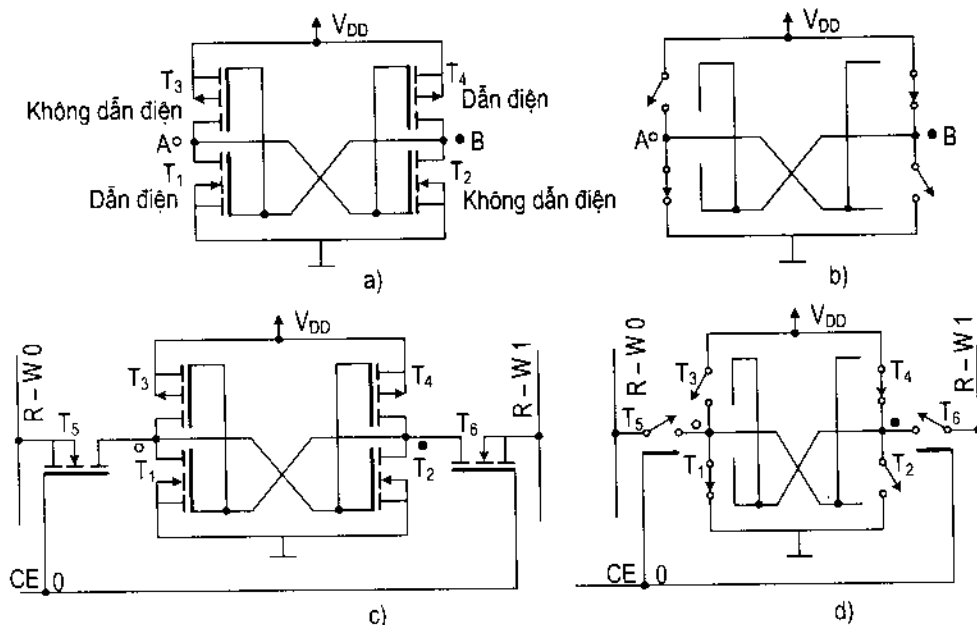


Hình 10.7. Cấu trúc ma trận ô nhớ 9×1 bit

10.3. RAM TĨNH DÙNG CMOS (SRAM – Static Random Access Memory)

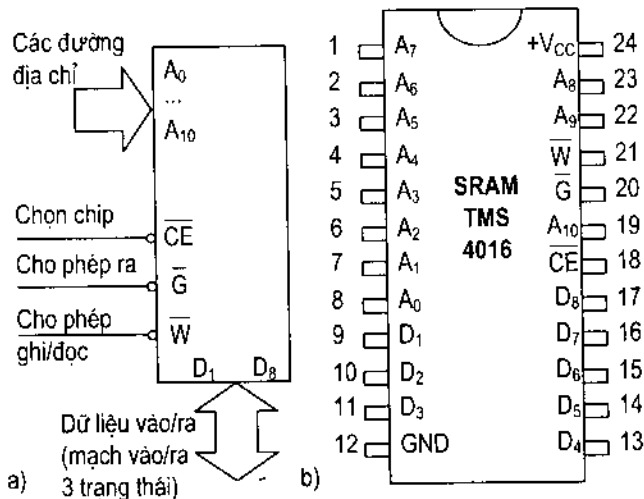
Ô nhớ dùng một mạch Flip Flop theo công nghệ CMOS có ưu điểm là công suất tiêu hao của mỗi ô rất thấp ($10^{-10}W$ ở chế độ lưu dữ liệu) khi hoạt động ghi đọc hoặc khi tần số lật tăng lên thì công suất tăng nhưng cũng đủ nhỏ nên được dùng phổ biến ở các bộ nhớ có dung lượng lớn. Nhược điểm là tốc độ ghi/đọc có xu hướng chậm hơn loại dùng công nghệ TTL.

Hình 10.8a đưa ra cấu trúc mạch của một ô nhớ dùng CMOS với hai cặp MOS bù T_1 T_3 và T_2 T_4 hoạt động ngược pha nhau $T_1^+ T_3^- T_2^- T_4^+$ hay $T_1^- T_3^+ T_2^+ T_4^-$ đang ở trạng thái lưu dữ liệu ($V_B = 1$; $V_A = 0$) và mạch mô tả trạng thái lưu bit 1 hình 10.8b.



Hình 10.8. Cấu trúc một ô nhớ dùng CMOS

Khi đầu vào của ô nhớ giao tiếp với tín hiệu cho phép chọn CE thông qua T_5 và T_6 , ô nhớ được nối tới RW_0 hoặc RW_1 hoạt động ghi/đọc sẽ xảy ra. Nếu $CE = 0$, T_5^- - T_6^- ô nhớ cách ly khỏi các đường dữ liệu (hình 10.8d – trạng thái chờ). Khi $CE = 1$, ô nhớ được chọn cho phép hoạt động ghi/đọc: T_5^+ - T_6^+ , tuy nhiên chỉ những đường có tín hiệu $R/\bar{W} = 0$ thì mức 1



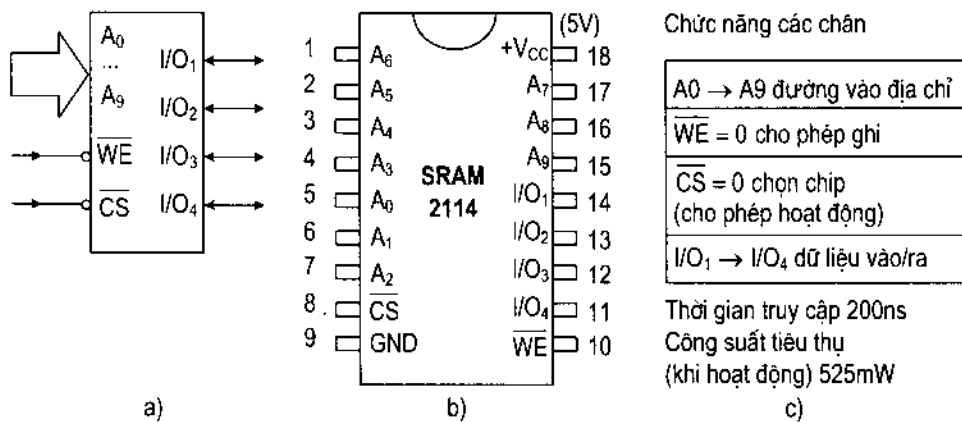
$\bar{W}$	$\bar{CE}$	$\bar{G}$	D_1 đến D_8	Chế độ
0	0	X	Dữ liệu xác định	Ghi
1	0	0	Dữ liệu ra	Đọc
X	1	X	Hi-Z	Cấm IC hoạt động
1	0	1	Hi-Z	Cấm ra

Hình 10.9. IC SRAM TMS4016:

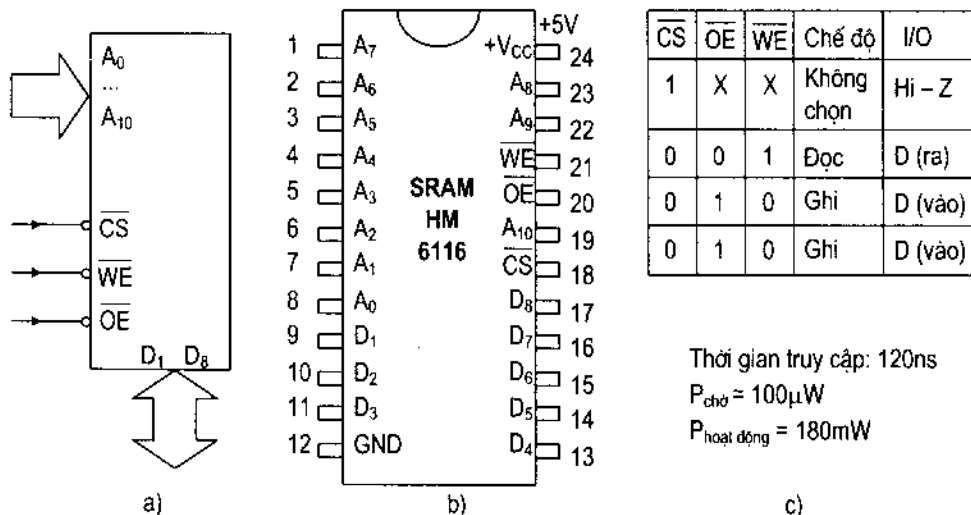
a) Sơ đồ chức năng; b) Bố trí chân; c) Bảng chân lý

mới qua được T_5 hay T_6 (ở ví dụ hình vẽ là qua T_6) tác động tới SA gây ra mức 1 ở lối ra thể hiện hoạt động đọc. Còn khi ghi vào cũng thông qua xung chọn $CE = 1$ với đường dữ liệu ví dụ $\overline{RW}_0 = 1$, qua T_5^+ (vì $CE = 1$) tới điểm A kích thích làm T_2^+ và do vậy dập tắt T_1 (T_1^-) và hoạt động ghi mức 1 vào ô nhớ ($V_A = 1$) đã xảy ra. Mức nhớ nằm lưu tại ô nhớ theo thời gian cùng nguồn cấp V_{CC} . Hình 10.9 giới thiệu sơ đồ chức năng (a) phân bố vị trí chân (b) và bảng chân lý (c) của chip SRAM loại TMS4016.

Hình 10.10 giới thiệu sơ đồ logic, bố trí chân và bảng chức năng của IC SRAM 2114 – 2 và tương tự cho IC SRAM HM 6116 trên hình 10.11.



Hình 10.10. IC SRAM 2114 – 2



Hình 10.11. IC SRAM HM 6116

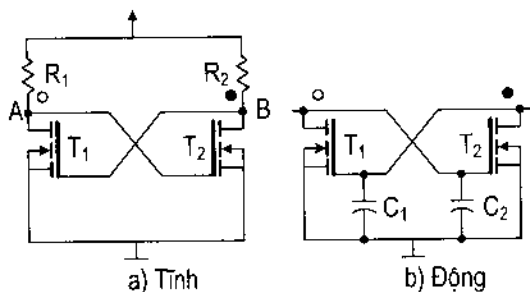
10.4. RAM ĐỘNG (DRAM) DÙNG CMOS

10.4.1. Nguyên lý RAM động

Với trở kháng đầu vào của MOSFET rất lớn có thể chế tạo loại ô nhớ sử dụng sự có mặt hay không có mặt của các điện tích trên cực của G của những MOSFET để lưu giữ thông tin 1 hay 0 trong một khoảng thời gian nhất định. Bộ nhớ với các ô nhớ dạng này được gọi là RAM động hay DRAM (dynamic random access memory). Hình 10.12b mô tả một đơn vị nhớ (một ô nhớ một bit) của DRAM dùng tụ C_{GS} của loại NMOS (T_1 hay T_2) nhớ dữ liệu. Với cách nhớ này, do có dòng điện rò của tụ có xu hướng bị mất điện tích (hay mất dữ liệu) theo thời gian và vì vậy theo định kỳ phải nạp lại điện tích cho C_{GS} gọi là "làm tươi" (refreshing) bộ nhớ.

Trên hình 10.12a (một ô nhớ của SRAM) các tranzito T_1 và T_2 được nối qua R_1 và R_2 hay qua T_3 và T_4 ở hình

10.8 tới nguồn $+V_{DD}$. Muốn lưu lại thông tin (ví dụ trên hình 10.12a là $T_1^+ T_2^-$ hay $V_A = 0, V_B = 1$) ô nhớ phải luôn duy trì nguồn cấp $+V_{DD}$. Ô nhớ của DRAM hình 10.12b không nối trực tiếp với $+V_{DD}$ tuy nhiên mạch có thêm hai tụ điện (cấu tạo của MOSFET) là $C_1 = C_{GS1}$ và $C_2 = C_{GS2}$. Khi hoạt động, các tụ này được nguồn $+V_{DD}$ nạp điện tích tương ứng với mức nhớ bit dữ liệu. Ở trạng thái lưu dòng rò của tụ C_{GS} cỡ $10^{-9} \rightarrow 10^{-10}A$ có xu thế làm hao hụt điện tích nhớ nên định kỳ phải nạp lại (làm tươi) mặc dù chưa dùng dữ liệu này.

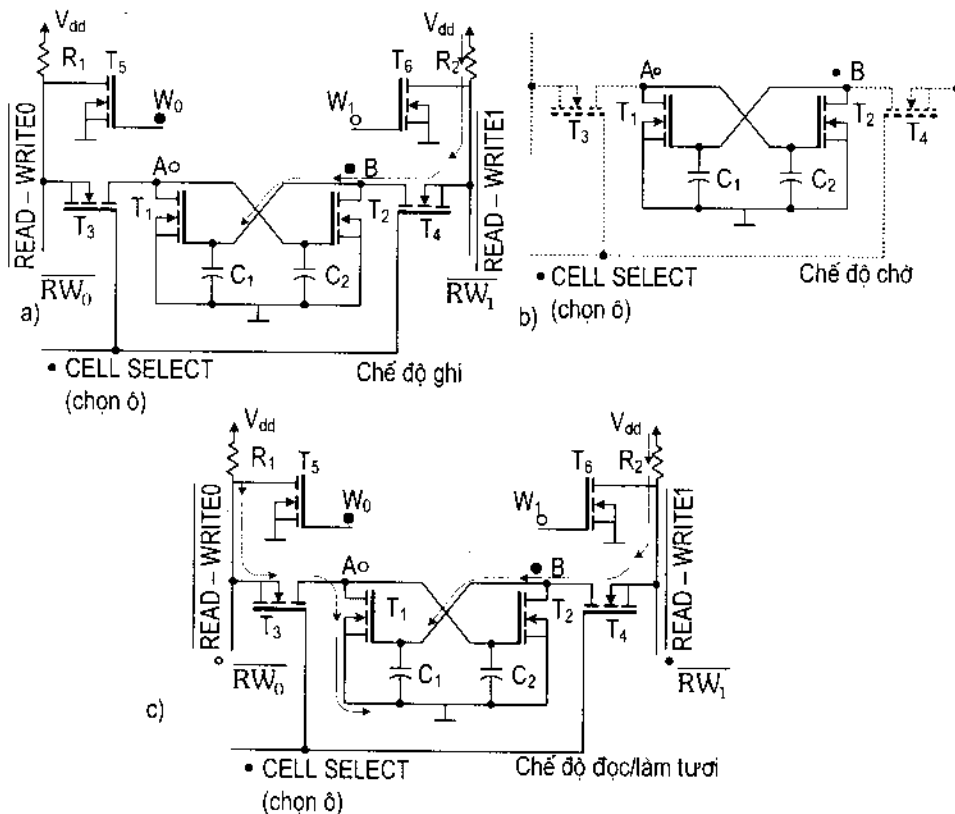


Hình 10.12. Cấu tạo một đơn vị nhớ của SRAM (a) và của DRAM (b)

10.4.2. Hoạt động của một đơn vị nhớ DRAM

DRAM minh họa trên hình 10.13 gồm 3 chế độ: ghi, chờ và đọc kết hợp làm tươi ô nhớ. Cấu tạo một ô nhớ cơ bản gồm T_1 T_2 và C_1 C_2 . Các đường điều khiển chế độ đọc/ghi $\overline{RW_0}, \overline{RW_1}$ được đưa vào ô nhớ qua T_3 , T_4 , đường chọn ô CS (cell select) tác động tới G_3 và G_4 của T_3 , T_4 .

Hình 10.13a mô tả hoạt động của ô nhớ ở chế độ ghi dữ liệu, ví dụ đang có bit giá trị 0 được ghi vào ô nhớ (xác định theo chế độ nối mạch của T_1 T_1^+). Khi đó $CS = 1$ T_3^+ T_4^+ nối ô nhớ tới đường RW_0 và RW_1 . Nếu lúc này $\overline{RW_0}$ ở mức tích cực thấp $\overline{RW_0} = 0$ (bằng cách áp tín hiệu cho phép ghi $W_0 = 1$ vào G_5 của T_5 làm T_5^+ và $\overline{RW_0} = 0$ trong lúc này $W_1 = 0$ nên T_6^-). Điểm A nối 0V làm T_2^- , điện thế dương tại đường RW_1 qua T_4^+ xuất hiện tại điểm B, tụ C_1 được nạp tới mức nguồn $+V_{DD}$ T_4^+ $V_A = 0$ vì được nối qua T_3^+ tới $\overline{RW_0} = 0$.



Hình 10.13. Hoạt động cơ bản của một ô nhớ trong DRAM

Hình 10.13b mô tả hoạt động của ô nhớ ở chế độ chờ: $CS = 0$, T_3^- T_4^- cô lập ô nhớ với các đường vào $\overline{RW_0}$, $\overline{RW_1}$. T_1^+ duy trì điều kiện A nối đất $V_A = 0$. Điện thế đã được nạp trên C_1 tiếp tục làm T_1^+ nối tắt hai cực của C_2 ; T_2^- và mạch ổn định (chờ) ở chế độ T_1^+ T_2^- ứng với trạng thái đang lưu $V_A = 0$, $V_B = 1$.

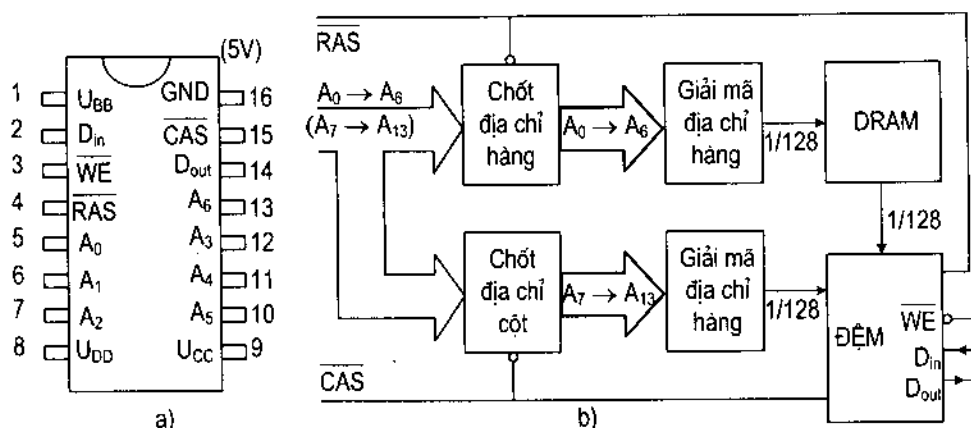
– Khi cần đọc dữ liệu đang lưu, $\overline{CS} = 1$ (để đưa ô nhớ lên trạng thái được chọn) làm $T_3^+ T_4^+$, T_1 đang dẫn điện vì $\overline{RW}_0 = 0$ bit 0 hình thành trên đường $\overline{RW}_0$ cho biết đã đọc dữ liệu 0 ra khỏi ô nhớ. Còn nếu đang lưu bit 1 thì $\overline{RW}_1 = 0$ và bit 1 hình thành trên đường $\overline{RW}_1$ (đọc trị 1) (hình 10.13c).

– Trong thời gian ở chế độ chờ (hình 10.13b) điện tích nhớ trên C_1 hoặc C_2 bị hao hụt cần được nạp lại. Khi cho $W_0 = W_1 = 0$ thì $\overline{RW}_0 = \overline{RW}_1 = 1$. Nguồn $+V_{DD}$ sẽ qua T_3^+ (hay T_4^+) nạp cho C_1 (hoặc C_2). Như vậy, khi hoạt động ở chế độ đọc, ô nhớ đã được kết hợp làm tươi. Với các giá trị tụ C_1, C_2 tương đối nhỏ (cỡ $10^{-13} \rightarrow 10^{-14}F$) điện tích nhớ nhanh bị mất mát, chu kỳ làm tươi ở DRAM thông dụng thường chọn bằng $2ms = 2 \cdot 10^{-3}s$. Như vậy, cứ sau 2ms DRAM lại được "đọc" một lần, mỗi lần kéo dài cỡ $40\mu s$ không phụ thuộc vào việc có cần đọc thực hay chưa.

Khi cần tăng dung lượng của DRAM có thể ghép song song nhiều chip dung lượng nhỏ và kết hợp nhiều nhánh song song qua một bộ giải mã địa chỉ chọn nhánh.

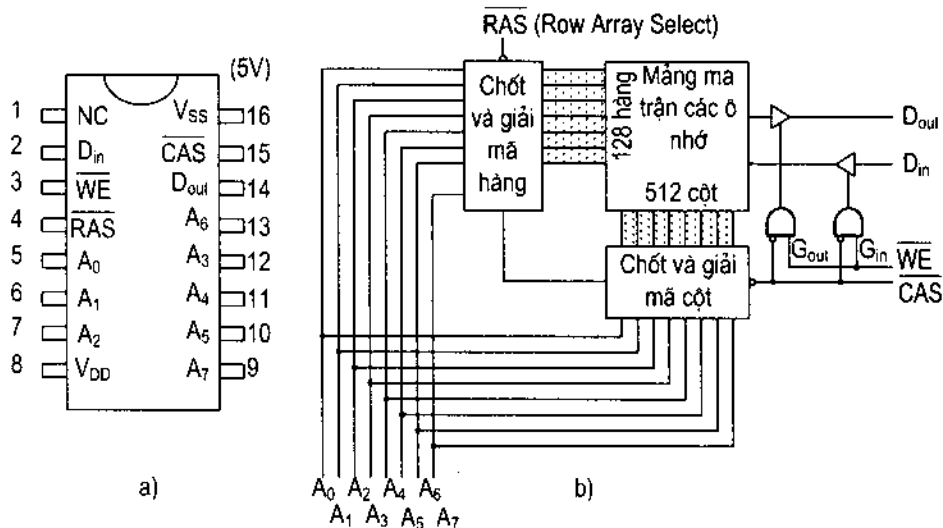
10.4.3. IC DRAM

TMS4116 và MN4164 có sơ đồ bố trí chân và sơ đồ cấu trúc tổng quát cho trên hình 10.14 và hình 10.15.



Hình 10.14. Sơ đồ bố trí chân của IC DRAM TMS 4116 dung lượng $16k \times 1bit$ (a) và sơ đồ cấu trúc khối của IC (b)

(Tổng cộng có $2^{14} = 16.384$ ô nhớ, nhờ tín hiệu chọn kênh $\overline{RAS}$ và $\overline{CAS}$)



Hình 10.15. Sơ đồ bố trí chân của IC DRAM MN4164 loại 65kbit (a) và sơ đồ cấu trúc khối của IC (b)

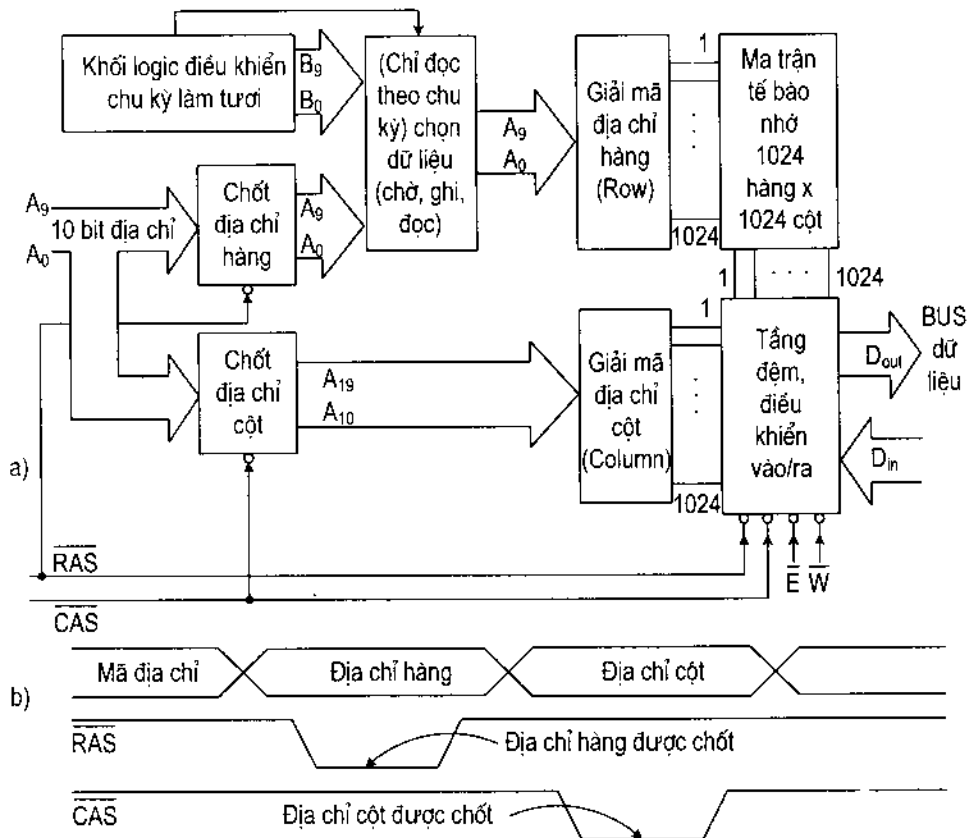
Trong cấu trúc IC hình 10.14 hoặc hình 10.15, hai tín hiệu $\overline{\text{RAS}}$ (Row Array Select) và $\overline{\text{CAS}}$ (Column Array Select) được gọi là tín hiệu chọn kênh sử dụng để điều khiển các đường địa chỉ hàng và địa chỉ cột tương ứng: khi $\overline{\text{RAS}} = 0$ mạch chốt và giải mã hàng hoạt động, còn khi $\overline{\text{CAS}} = 0$, mạch chốt và giải mã cột làm việc. Sau khi kết thúc quá trình giải mã, tín hiệu ra của hai khối giải mã hàng và cột đồng thời tác động lên ma trận các ô nhớ. Cần cấm trạng thái $\overline{\text{RAS}} = \overline{\text{CAS}} = 0$ (đồng thời sẽ làm rối loạn việc xác định địa chỉ ô nhớ), chỉ cần dùng hai nhóm bit $A_0 \rightarrow A_6$ hay $A_0 \rightarrow A_7$ chung cho việc giải mã hàng ($A_0 \rightarrow A_7$) sau đó giải mã cột ($A_8 \rightarrow A_{14}$) hay ($A_0 \rightarrow A_6$) cho hàng và ($A_7 \rightarrow A_{13}$) cho cột với 4116. Như vậy, với IC 4116 với 7 đường địa chỉ $A_0 \rightarrow A_6$ có thể quản lý được 2^{14} ô nhớ (16384 ô nhớ) hay với $A_0 \rightarrow A_7$ (loại 4164) quản lý được $2^{16} = 65536$ ô nhớ. Tín hiệu $\overline{\text{WE}}$ cho phép ghi dữ liệu khi ở mức tích cực (mức 0: $\overline{\text{WE}} = 0$): D_{in} được phép nhập, còn khi $\overline{\text{WE}} = 1$, DRAM được phép xuất (đọc) dữ liệu từ một địa chỉ hàng cột đã xác định từ việc giải mã. Việc kiểm soát vào/ra dùng các phần tử 3 trạng thái hai hướng làm bộ đệm điều khiển nhờ các cổng G_{IN} và G_{OUT} . Ở chế độ chờ công suất tiêu hao của nhóm IC 4164 là 27,5mW, ở chế độ tích cực là 275mW, 128 chu kỳ làm tươi/2ms, tốc độ truy cập 150ns.

10.4.4. Sơ đồ khối cấu trúc DRAM

Hình 10.16 cho một sơ đồ khối của cấu trúc của DRAM có dung lượng $1\text{M} \times 1\text{bit}$ và đồ thị thời gian các tín hiệu địa chỉ lối vào.

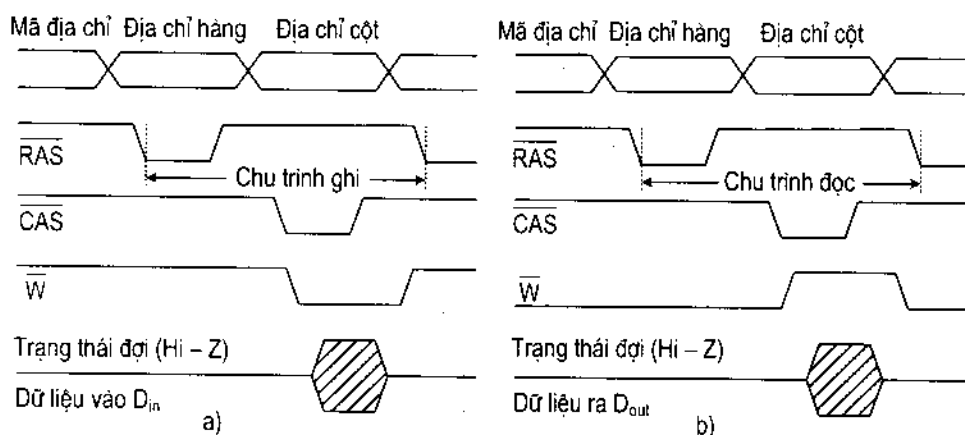
Tổ chức ô nhớ của DRAM kiểu ma trận hàng cột (dạng vuông hay chữ nhật) chọn địa chỉ hoạt động theo kiểu dồn kênh sau khi chốt địa chỉ hàng và chốt địa chỉ cột theo các tín hiệu điều khiển $\overline{\text{RAS}}$ và $\overline{\text{CAS}}$ (hình 10.16a), chúng được giải mã địa chỉ đồng thời để tác động đến vị trí nhớ được chọn hoạt động. Như vậy, với 10bit địa chỉ hàng $A_9 \rightarrow A_0$ do tổ chức kiểu dồn kênh chúng đồng thời là $A_{10} \rightarrow A_{19}$ cho địa chỉ cột đảm bảo dung lượng $1024 \text{ hàng} \times 1024 \text{ cột}$ xác định vị trí cho 1 triệu tế bào nhớ (1048576 ô nhớ).

Mạch logic làm tươi có thể là một khối riêng bên ngoài chip, hiện nay, đa số các chip DRAM có khối này nằm bên trong.



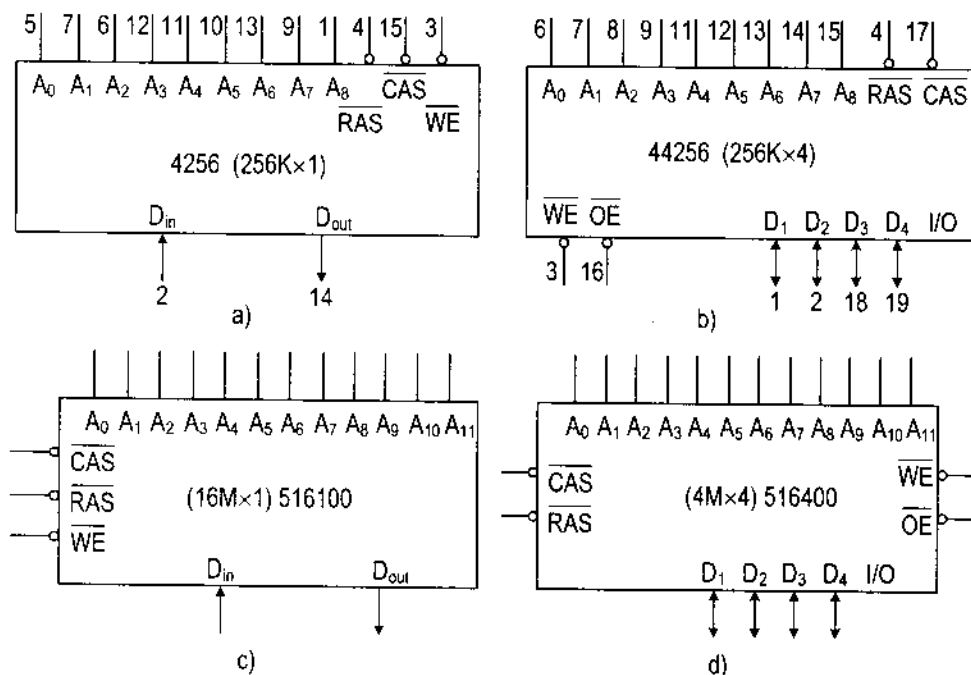
Hình 10.16 a) Cấu trúc khối tổ chức DRAM dung lượng 1Mbit kiểu ma trận vuông các ô nhớ; b) Đồ thị thời gian minh họa quá trình chốt địa chỉ kiểu dồn kênh

Hình 10.17 là đồ thị thời gian minh họa hoạt động đọc và ghi của DRAM.



Hình 10.17. Đồ thị thời gian minh họa chu trình ghi dữ liệu vào DRAM (a) và chu trình đọc dữ liệu đã ghi từ DRAM (b).

Hình 10.18 đưa ra một số ký hiệu logic và bố trí chân của một vài IC DRAM thường gặp.



Hình 10.18. Ký hiệu logic các IC DRAM:

a) 4256 (256K×1); b) 44256 (256K×4); c) 516100 (16M×1); d) 516400 (4M×4)

10.5. CÂU HỎI VÀ BÀI TẬP

1. Hãy chỉ ra điểm khác nhau căn bản giữa ROM và RAM ? Loại bộ nhớ nào có nhiều đầu vào và đầu ra hơn ? Vì sao ?
- 2 a) Dạng RAM có cấu tạo là loại MSFF và có cấu tạo dùng tranzito nhiều emitor có gì khác nhau? Ưu nhược điểm của mỗi loại? Khi cần bộ nhớ có dung lượng lớn loại cấu hình nào thích hợp hơn?
b) Hãy phân tích 3 chế độ ghi, chế độ chờ và chế độ đọc của một ô nhớ RAM dùng tranzito nhiều emitor. Sử dụng loại tranzito có 3 emitor mang lại ưu điểm gì so với các loại có 2 emitor?
3. Hãy nêu đặc điểm để phân biệt SRAM và DRAM về cấu tạo.
Có thể dùng ô nhớ tranzito nhiều emitor cho chip DRAM được không? Vì sao? RAM có dung lượng lớn dùng SRAM hay DRAM thuận lợi hơn? Vì sao?
4. Làm tươi bộ nhớ DRAM là gì? Tại sao lại chỉ phải làm tươi đối với DRAM. Cách thực hiện làm tươi DRAM như thế nào?
- 5 a) Hãy tính dung lượng của các bộ nhớ SRAM 2114 và 6116 trên hình 10.10 và 10.11. Vẽ cấu trúc khối chi tiết của IC SRAM 2114 và giải thích chức năng các khối.
b) Câu hỏi tương tự a) cho các DRAM 4116 và 6164 cho trên hình 10.14 và 10.15.
c) Một DRAM có dung lượng nhớ là 64kbit sẽ có bao nhiêu đầu vào và đầu ra, nêu chức năng của chúng. Chip nhớ này chứa được bao nhiêu byte dữ liệu?
- 6 a) SRAM 2114 có dung lượng 1024 từ $\times$ 4 bit chứa được bao nhiêu byte dữ liệu? Nếu sử dụng hết các đầu vào địa chỉ của nó thì có thể gán địa chỉ cho bao nhiêu ô nhớ?
b) Các địa chỉ sau đây có nằm trong số các địa chỉ quản lý được của SRAM 2114 hay không:
5FF; 3AF; 2AE; 7BD; A6B; E3A và 4C7.
7. Thời gian truy cập của SRAM HM 6116 là 120ns có ý nghĩa gì?
Cần bao nhiêu bit địa chỉ cho SRAM có dung lượng 2048 từ $\times$ 8 bit.
Chip RAM trên có thể nhớ được bao nhiêu từ dài 32 byte?
Nếu ghép song song 8 chip IC loại 6116 thì dung lượng tối đa của bộ nhớ là bao nhiêu? (tính dung lượng theo kbyte).

Làm thế nào để tránh các địa chỉ bị trùng lặp khi ghép 8IC vừa nêu trên?

8. Dùng phương pháp chọn kênh (MUX) đã giảm được bao nhiêu đường địa chỉ trong DRAM 4116? Điều kiện chọn là gì?

Nói rằng dung lượng DRAM 4164 được tăng gấp đôi dung lượng 4116 có đúng không?

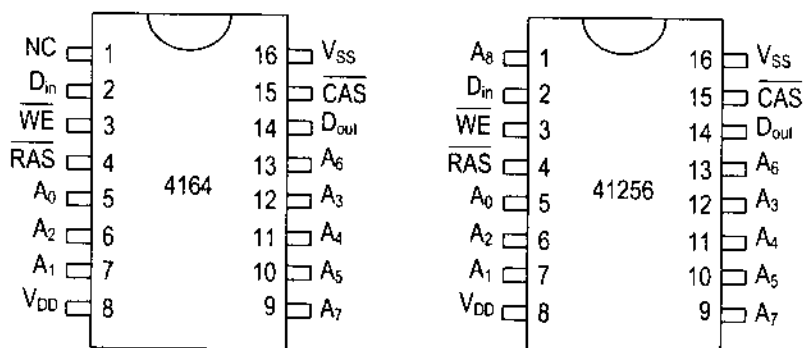
9. Cho 2 chip IC DRAM hình 10.19.

a) Hãy nêu nhiệm vụ của các chân IC đã cho.

b) Tính dung lượng của mỗi loại và giải thích tại sao loại 41256 có dung lượng cao hơn loại 4164?

Từ đặc điểm khác nhau đã giải thích ở câu b) hãy sửa lại cấu trúc khối hình 10.15 cho phù hợp với loại 41256.

c) Làm thế nào để ghép song song 4IC 41256 đạt được dung lượng nhớ 1Mbit?



Hình 10.19. Sơ đồ bố trí chân của nhóm DRAM 4164 và 41256 (cho bài tập 9)

10. Cho mạch điện hình 10.20 sử dụng 2SRAM2114

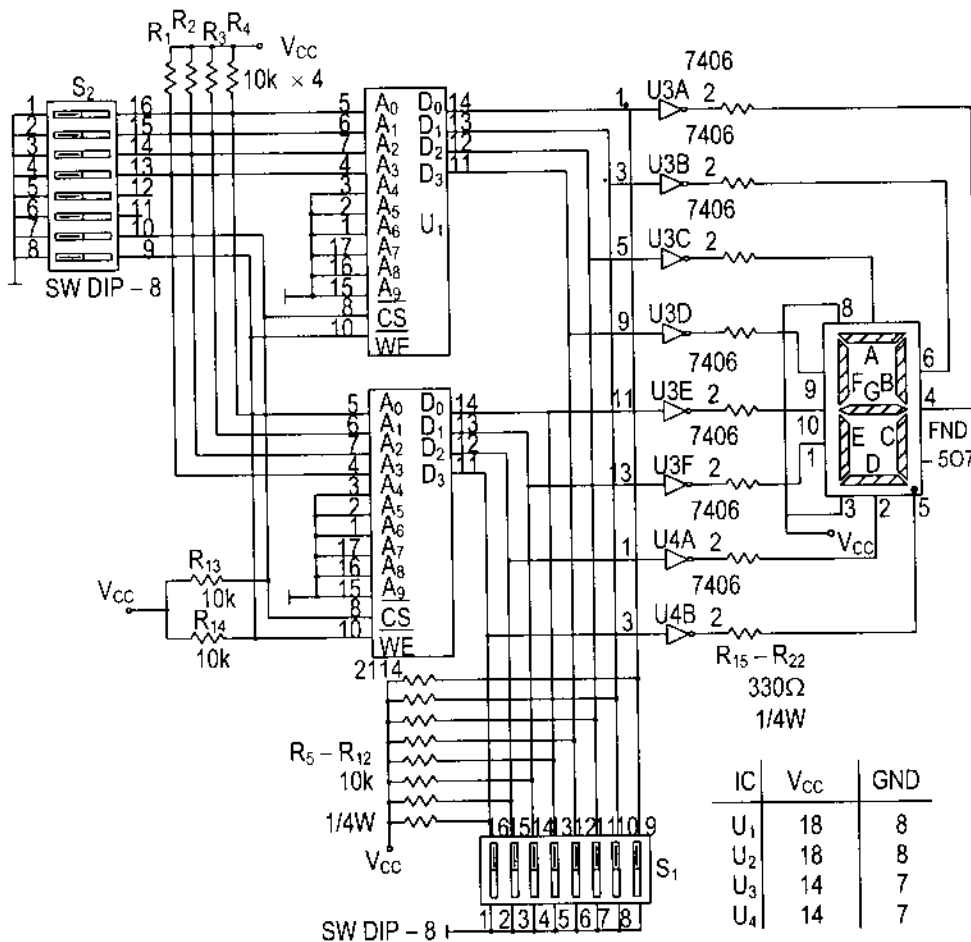
a) Hãy phân tích nhiệm vụ các khối trong mạch điện và qua đó tìm hiểu nguyên lý hoạt động của mạch đã cho.

b) Sử dụng S_2 đặt $A_0A_1A_2A_3 = 0000$ (nhấn phím 1, 2, 3, 4). Trạng thái các SRAM sẽ như thế nào khi cho $\overline{CS} = 0$ và $\overline{WE} = 0$ (nhấn phím 7 và 8 của S_2)?

- 11 a) Hãy nhập các dữ liệu 4 bit trong hệ 16 thông qua các phím của S_2 và đọc các dữ liệu vừa ghi thông qua các phím của S_1 hiển thị trên màn LED 7 vạch (các mã 1010 đến 1111 hiển thị theo bảng mã ASCII với các chữ cái ABCDEF) thực hiện trên mạch hình 10.20.

b) Nếu nối lại đầu vào A_4 của 2SRAM (chân 3) với lối ra chân 12 của S_2 và sau đó cho qua một điện trở $10k$ tới $+V_{CC}$ (như các chân $A_0 \rightarrow A_3$) thì điều gì xảy ra với các kết quả dữ liệu đã làm với bước c.

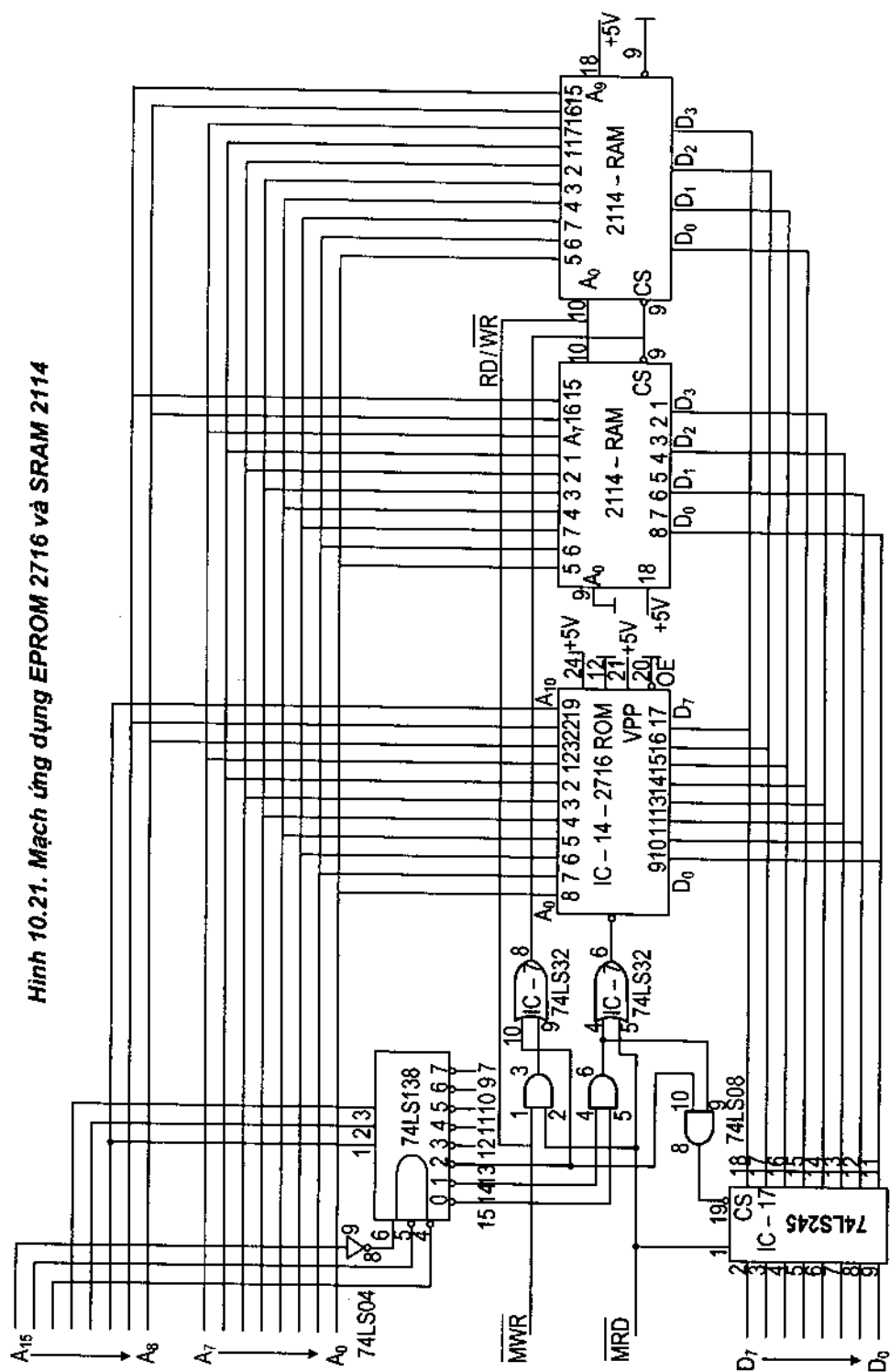
c) Tương tự, tiếp tục với A_5 (chân số 2) làm như với A_4 . Hãy viết lại địa chỉ của các ô nhớ trong RAM ứng với hai trường hợp vừa làm cho A_4 và A_5 .



Hình 10.20. Mạch điện cho bài tập 10 và cho thực hành SRAM

12. Hãy phân tích nhiệm vụ các IC trong mạch điện hình 10.21 có sử dụng EPROM 2716 và các SRAM 2114.

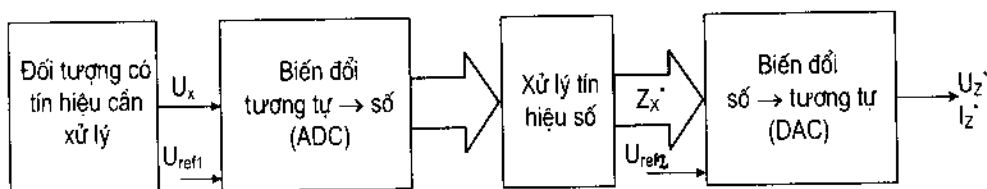
Hình 10.21. Mạch ứng dụng EPROM 2716 và SRAM 2114



BÀI 11. MẠCH CHUYỂN ĐỔI SỐ – TƯƠNG TỰ (DAC)

11.1. KHÁI NIỆM CHUNG

– Phương pháp xử lý số tín hiệu mang lại nhiều ưu thế quan trọng trong quá trình xử lý thông tin như có tính chính xác và hiệu quả cao, tự động hóa và tin học hóa thuận lợi, truyền tải và lưu trữ thông tin với lưu lượng và dung lượng lớn, tốc độ cao và đặc biệt thuận lợi trong việc kết nối với mạng cục bộ hay mạng diện rộng đa dịch vụ số băng rộng. Để thực hiện quá trình xử lý số tín hiệu cần thực hiện hai quá trình cơ bản theo sơ đồ khối hình 11.1 là quá trình biến đổi thuận từ tín hiệu tương tự sang tín hiệu số (ADC) và quá trình biến đổi ngược từ đại lượng số đã xử lý về lại đại lượng tương tự thể hiện kết quả quá trình (DAC), các quá trình này luôn cần một điện áp tham chiếu U_{ref} (điện áp mẫu).



Hình 11.1. Các bước cơ bản trong quá trình biến đổi để xử lý số tín hiệu

– Mạch DAC có nhiệm vụ biến đổi một tín hiệu vào biểu diễn ở dạng rời rạc (dạng tín hiệu số – digital) thành một giá trị điện áp hay dòng điện duy nhất tại lối ra tỷ lệ với trọng số của đại lượng vào. Trong khi đó, theo định nghĩa, mạch ADC có nhiệm vụ biến đổi một tín hiệu vào ở dạng liên tục (analog) theo thời gian thành một đại lượng ra dưới dạng số rời rạc (digital) biểu diễn giá trị vào.

– Quá trình ADC luôn gồm 3 bước cơ bản là:

1. Thực hiện rời rạc hóa đại lượng vào (U_x) bằng cách lấy mẫu giá trị U_x theo một chu kỳ mẫu ($T_{\text{mẫu}}$) xác định: $U_x(k T_{\text{mẫu}}) = U_k$ (k là các số nguyên dương). Quá trình này phải tuân theo định lý lấy mẫu $f_{\text{mẫu}} \geq 2f_{x\text{max}}$ để việc mất mát thông tin do bỏ qua các giá trị U_x không lấy mẫu là nhỏ nhất.

2. Chọn một đơn vị làm tròn (gọi là đơn vị lượng tử) bằng cách chia nhỏ điện áp mẫu U_{ref} và làm tròn các giá trị U_k đã lấy ở bước 1 (chỉ lấy phần nguyên, bỏ phần lẻ). Đây là quá trình lượng tử hóa với sai số làm tròn không vượt quá $\frac{1}{2}$ đơn vị làm tròn.

3. Biểu diễn giá trị đã làm tròn trong một hệ đếm chọn trước, gọi là quá trình mã hóa dữ liệu thường là mã hóa $(U_k)_{10} = (Z_k)_2$ hay $(Z_k)_{2-10}$.

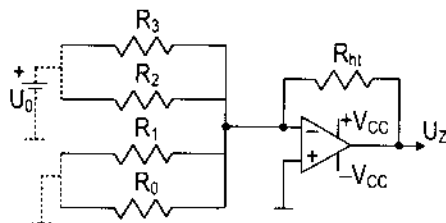
Quá trình DAC chỉ thực hiện được kết quả ra là các giá trị phần nguyên của đơn vị làm tròn (hay là các giá trị bậc thang) mà không lấy lại được đại lượng analog thực sự.

11.2. PHƯƠNG PHÁP DAC DÙNG THANG ĐIỆN TRỞ

11.2.1. Phương pháp dùng mạch cộng điện áp (hình 11.2)

Giả thiết số cần biến đổi ở dạng nhị phân 4 bit $Z = Z_3Z_2Z_1Z_0 = (1100)_2$. Sử dụng mạch cộng áp hình 11.2 có 4 lối vào: Bộ cộng làm việc với 3 điều kiện sau:

1. Đầu vào có R_i được nối tới nguồn điện áp mẫu có giá trị U_0 khi $Z_i = 1$ (ở ví dụ là R_3 và R_2 vì $Z_3 = Z_2 = 1$). Đầu vào R_i sẽ nối tới 0V khi $Z_i = 0$ (ở ví dụ là R_1 và R_0 vì $Z_1 = Z_0 = 0$, xem đường nối đứt nét cho ví dụ). Từ điều kiện này, hệ thức tính điện áp ra có dạng:



Hình 11.2. Mạch DAC 4 bit dùng bộ cộng đảo

$$U_Z = \sum_{i=0}^3 U_{Z_i} \cdot Z_i \quad (11.1)$$

Ở đây $U_{Z_i} = U_0 \left(-\frac{R_{ht}}{R_i} \right)$ theo tính chất của mạch khuếch đại đảo và tính chất tuyến tính của bộ cộng. U_{Z_i} chính là phần đóng góp của $Z_i = 1$ vào trong kết quả U_Z tại lối ra.

2. U_{Z_i} phải thể hiện được trọng số vị trí 2^i của Z_i trong biểu diễn số của Z . Tức là $U_{Z_3} = 2U_{Z_2} = 2^2 U_{Z_1} = 2^3 U_{Z_0}$. Muốn vậy, nếu chọn U_0 và R_{ht} không đổi có độ chính xác cao thì theo hệ thức U_{Z_i} có:

$$R_0 = 2R_1 = 2^2 R_2 = 2^3 R_3 \quad (11.2)$$

3. Gọi sai số của U_{Z_3} là ΔU_{Z_3} (do sai số của điện trở R_3 gây ra), để quá trình biến đổi không bị lỗi bit thì:

$\Delta U_{Z_3} < U_{Z_0}$ tức là sai số lớn nhất phải nhỏ hơn phần đóng góp của $Z_0 = 1$ vào U_Z (là giá trị có nghĩa nhỏ nhất ứng với phần đóng góp của Z_{LSB}).

Từ đây, suy ra hệ thức tương đương:

$$\frac{\Delta R_3}{R_3} < \frac{1}{2^4} \quad (11.3)$$

Chú ý: Trong trường hợp Z có n bit ở lối vào thì các hệ thức (11.1) đến (11.3) có dạng:

$$\left. \begin{aligned} U_Z &= \sum_{i=0}^{n-1} U_{Z_i} \cdot Z_i \\ R_0 &= 2R_1 = \dots = 2^{n-1} R_{n-1} \\ \frac{\Delta R_{n-1}}{R_{n-1}} &< \frac{1}{2^n} \end{aligned} \right\} \quad (11.4)$$

– Sử dụng điều kiện (11.2) tính lại các kết quả U_{Z_i} có:

$$U_Z = U_0 \left(-\frac{R_{ht}}{R_0} \right) \cdot \sum_{i=0}^3 2^i U_Z \Rightarrow \begin{aligned} U_Z &= U_{Z_i} \cdot (Z)_{10} \\ (0 \leq Z &\leq 2^4 - 1) \end{aligned} \quad (11.5)$$

Hệ thức (11.5) cho phép tính trực tiếp Z trong hệ thập phân: $(Z)_{10}$ là biểu diễn của $(Z)_2$ trong hệ thập phân:

$$(Z)_2 = \left(\sum_i 2^i Z_i \right)_{10} \quad \text{trong ví dụ } (Z)_2 = (1100)_2 = (12)_{10}$$

11.2.2. Phương pháp cộng điện áp dùng thang điện trở

– Mạch DAC dùng thang điện trở với 4 bit vào có dạng được cho trên hình 11.3.

Các điều kiện làm việc của mạch 11.3 là:

1. Chuyển mạch điện tử S_i nối tới B_i (với U_0) khi $Z_i = 1$ và tới A_i (với $0V$) khi $Z_i = 0$.

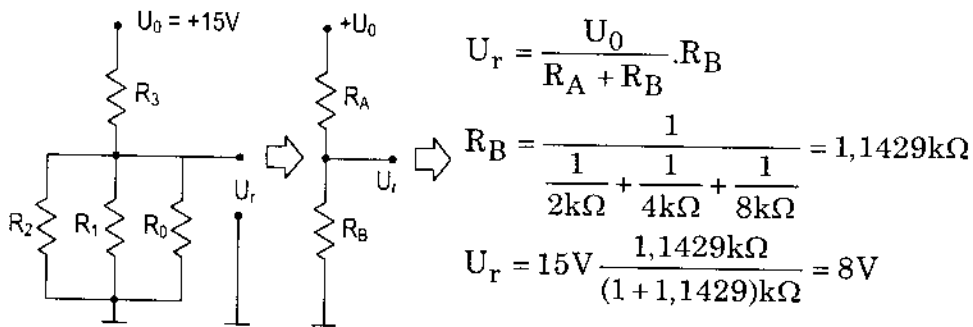
2. Đóng góp của $Z_i = 1$ vào U_r phải thể hiện trọng số 2^i của Z_i , suy ra:

$$R_3 = \frac{1}{2} R_2 = \frac{1}{2^2} R_1 = \frac{1}{2^3} R_0$$

với $R_3 = 1k\Omega$ thì $R_2 = 2k\Omega$; $R_1 = 4k\Omega$; $R_0 = 8k\Omega$.

3. Sai số $\frac{\Delta R_3}{R_3} < \frac{1}{2^4}$ để không mắc lỗi bit do mất thông tin về đóng góp của $Z_0 = 1$ vào U_r .

Từ ví dụ với $Z = 1000$, theo mạch tương đương, có:



Trong trường hợp tổng quát, mức điện áp nhỏ nhất do $Z_0 = 1$ đóng góp là:

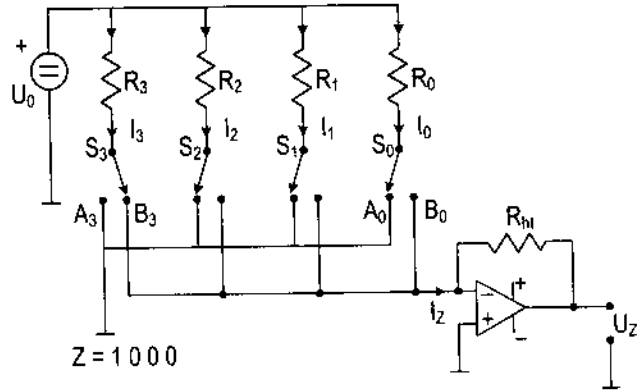
$$U_{Z_0} = \frac{U_0}{2^n - 1} \text{ (n là số bit lối vào)} \Rightarrow U_r = U_{Z_0} (Z)_{10} = \frac{U_0}{2^n - 1} (Z)_{10} \quad (11.6)$$

Hình 11.3. Thang điện trở nhị phân DAC 4 bit (trạng thái S_i ứng với $Z = 1000$)

Mạch hình 11.3 có nhược điểm là hệ thức 11.6 chỉ đúng khi không mắc tải tiếp nhận U_r ($R_{t\text{ai}} \rightarrow \infty$) và khi số bit n tăng lên, điều kiện yêu cầu độ chính xác của R_{n-1} khó thực hiện:

$$\frac{\Delta R_{n-1}}{R_{n-1}} < \frac{1}{2^n} (100\%)$$

Nếu kết hợp thang điện trở nhị phân với một vi điện tử (như thể hiện trên hình 11.4) thì nhược điểm thứ nhất về ảnh hưởng của tải được khắc phục. Điều kiện làm việc của mạch điện hình 11.4 tương tự như trước.



Hình 11.4. DAC 4 bit dùng bộ cộng dòng điện trọng số

1. Chuyển mạch điện tử S_i do Z_i điều khiển sao cho:

S_i nối B_i khi $Z_i = 1$ (ví dụ $Z_3 = 1$).

S_i nối A_i khi $Z_i = 0$ (ví dụ $Z_2 = Z_1 = Z_0 = 0$).

$$\text{Vậy: } I_Z = \sum_{i=0}^3 I_i \cdot Z_i \quad (11.7)$$

$$\text{với: } I_i = \frac{U_0}{R_i}$$

2. Dòng điện trên R_i phải thể hiện vị trí 2^i của Z_i :

$$\left. \begin{aligned} I_3 &= 2I_2 = 2^2I_1 = 2^3I_0 \text{ hay tổng quát } I_{n-1} = 2^{n-1}I_0 \\ \text{và } I_{i+1} &= 2I_i \text{ hay } R_{i+1} = \frac{1}{2}R_i \end{aligned} \right\} \quad (11.8)$$

3. Sai số của dòng điện lớn nhất ứng bit MSB phải nhỏ hơn dòng LSB (dòng có nghĩa nhỏ nhất).

$$\Delta I_3 < I_0 \text{ hay } \Delta I_{n-1} < I_0 \text{ (với } Z \text{ có } n \text{ bit) hay } \frac{\Delta R_{n-1}}{R_{n-1}} < \frac{1}{2^n} \quad (11.9)$$

Từ các điều kiện trên, dễ dàng tìm được hệ thức tính I_Z :

$$I_Z = \frac{U_0}{R_0} \sum_i 2^i \cdot Z_i \Rightarrow I_Z = I_0(Z)_{10} \text{ với } 0 \leq Z \leq 2^n - 1 \quad (11.10)$$

Qua bộ biến đổi I/U có áp ra:

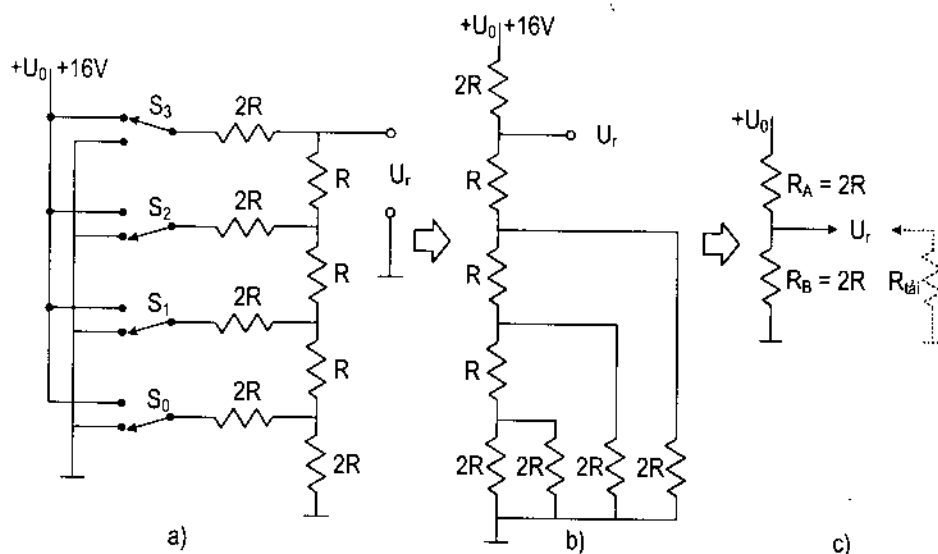
$$U_z = -R_{ht} I_z \quad (11.11)$$

Trong các hệ thức (11.5), (11.6) hay (11.10) $(Z)_{10}$ là các số nguyên trong hệ thập phân có giá trị trong khoảng $0 \leq Z \leq 2^n - 1$ (11.12)

Trong công nghệ IC, điều kiện (11.9) chỉ phù hợp với số bit $n \leq 4$ cho việc chế tạo các điện trở ở bit MSB, đây là nhược điểm quan trọng của các phương pháp đã xét trên.

11.3. MẠCH DAC DÙNG MẠNG ĐIỆN TRỞ (phương pháp dùng ma trận có điện trở tương đương cố định: $R - 2R$)

Phương pháp DAC dùng ma trận (mạng) điện trở $R-2R$ khắc phục được nhược điểm vừa nêu với yêu cầu khắt khe về độ chính xác của điện trở ứng với bit MSB. Mạch điện đơn giản nhất của phương pháp này cho trên hình 11.5a, các chuyển mạch S_i được nối cho ví dụ $Z = 1000$, hình 11.5b và c là mạch tương đương thu gọn của 11.5a.



Hình 11.5 a) Mạch DAC 4 bit dạng mạng điện trở $R-2R$;
b) Mạch tương đương đầy đủ của 11.5a;
c) Mạch tương đương đã rút gọn

Từ mạch 11.5c tính được:

$$U_r = U_0 \frac{R_B}{R_A + R_B} = 16V \cdot \frac{2R}{2R + 2R} = 8V$$

Tổng quát, với số Z bất kỳ, có n bit, điện áp ra bằng:

$$U_r = \frac{U_0}{2^n} (Z)_{10} \quad (0 \leq Z \leq 2^n - 1) \quad (11.13)$$

Để đảm bảo tương thích các mức ra 0 và 1 đối với TTL ($0 \leq U_r \leq 0,4V$ với mức 0 và $3,5V \rightarrow 5V$ với mức 1) cần dùng các cổng hở colectơ với điện trở kéo lên ($1k\Omega$) phối hợp kích thích tải là mạng điện trở như trên hình 11.6a và b.

Để tránh ảnh hưởng của tải (mắc trực tiếp vào lối ra) làm mức ra bị thay đổi gây ra sai số (hình 11.5c), cần dùng một bộ lặp điện áp tại lối ra như đường nét đứt trên hình 11.6b.

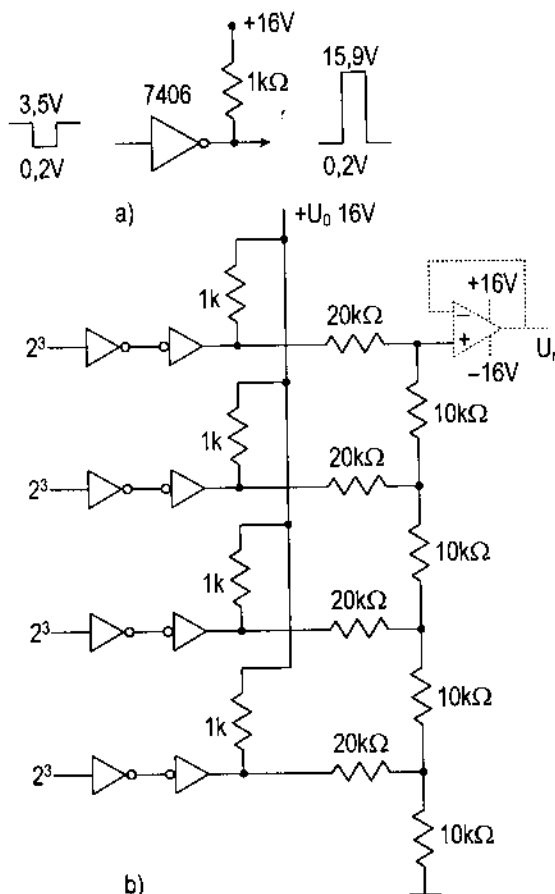
Có thể sử dụng IC như một bộ biến đổi dòng điện thành điện áp (tương tự như dạng mạch DAC hình 11.4) thể hiện trên mạch hình 11.7.

Mạch DAC hình 11.7 cần hai điều kiện làm việc:

1. Chuyển mạch điện tử S_i do Z_i điều khiển nối tới B_i khi $Z_i = 1$ (trong ví dụ là S_3) và nối tới A_i khi $Z_i = 0$ (trong ví dụ là S_2 , S_1 và S_0). Từ đó:

$$I_Z = \sum_{i=0}^3 I_i \cdot Z_i$$

2. Dòng I_i phải mang trọng số 2^i của Z_i , tức là $I_3 = 2I_2 = 2^2I_1 = 2^3I_0$



Hình 11.6 a) Dùng cổng logic TTL hở colectơ để kéo mức ra lên gần 16V cấp cho mạng R-2R; b) Mạch DAC mạng R-2R phối hợp với cổng kích thích loại TTL hở colectơ

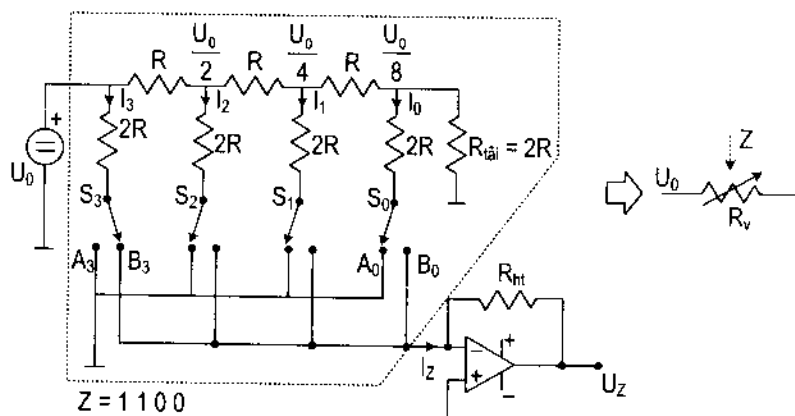
Điều này đạt được nhờ ma trận: mạng $R-2R$ đã chia đôi liên tiếp điện áp chuẩn U_0 . Tức là với n bit vào, nút cuối cùng của mạng có giá trị điện áp là

$$\frac{U_0}{2^{n-1}}; I_{LSB} = I_0 = \frac{U_0}{2^{n-1} \cdot 2R} = \frac{U_0}{2^n \cdot R} \quad (11.14)$$

Từ đây, có hệ thức tính dòng và áp lỗi ra:

$$\left. \begin{aligned} I_Z &= \frac{U_0}{2^n \cdot R} (Z)_{10} = I_0 (Z)_{10} \quad (0 \leq Z \leq 2^n - 1) \\ U_Z &= -R_{ht} \cdot I_Z \end{aligned} \right\} \quad (11.15)$$

Điều kiện về sai số của điện trở không cần thiết khắt khe với các bit cao như khi dùng thang điện trở chính là ưu điểm của phương pháp này, tạo điều kiện thuận lợi cho công nghệ chế tạo IC DAC.



Hình 11.7. DAC loại mạng điện trở $R-2R$ cộng dòng trọng số

11.4. MỘT VÀI DẠNG CHUYỂN MẠCH HAI VỊ TRÍ S_i (chuyển mạch đảo chiều)

Việc sử dụng các loại chuyển mạch 2 vị trí $A_i B_i$ trong các mạch DAC đã xét mang lại hai ưu điểm quan trọng so với chuyển mạch nối/ngắt thông thường là:

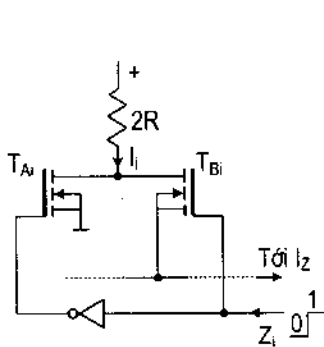
1. Tiếp điểm không phải chịu điện áp cao khi nối/ngắt mạch nên sử dụng sẽ bền hơn.

2. Nếu U_0 là nguồn có nội trở $R_{nguồn}$ (không lý tưởng) thì sai số do $R_{nguồn}$ gây ra là cố định và bằng $(2^n - 1)I_0 \cdot R_{nguồn}$ có thể khắc phục

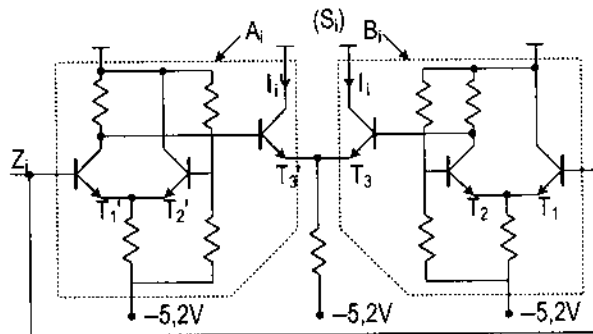
trong kết quả của I_Z (hoặc U_Z). Nếu dùng loại chuyển mạch nối/ngắt sai số này thay đổi theo từng giá trị của biến vào Z và không khắc phục được.

Trên hình 11.8 đưa ra cấu tạo của 1 ô chuyển mạch S_i dùng công nghệ NMOS, Z_i tác động dưới dạng 1 xung điện áp điều khiển S_i . Khi $Z_i = 1$ (xung dương biên độ cao hơn mức U_p của EMOS kênh N). T_{Bi} nối mạch, $\bar{Z}_i = 0$ nên T_{Ai} không hoạt động (ngắt mạch). Dòng tín hiệu I_i sẽ qua T_{Bi} tới mạch ra tham gia vào I_Z . Còn khi $Z_i = 0$ thì $\bar{Z}_i = 1$ và T_{Ai} nối T_{Bi} ngắt, dòng I_i qua T_{Ai} tới 0V.

Mạch hình 11.9 dùng công nghệ ECL cho tốc độ chuyển mạch cao tới ns nhờ các điện trở emítơ của các tranzito.



Hình 11.8. Khóa S_i đảo chiều dùng công nghệ NMOS



Hình 11.9. Khóa S_i đảo chiều dùng công nghệ ECL có tốc độ chuyển mạch cao

Khi $Z_i = 1$ thì T_1^+ đẩy $T_3^+ = 0$; nhánh vì sai $T_1 T_2$: tác động của $Z_i = 1$ làm T_1^+ đẩy T_2^- điện thế colectơ T_2 dâng cao làm T_3^+ : xuất hiện dòng I_i qua R_i (S_i được nối tới B_i).

Khi $Z_i = 0$ thì ngược lại T_1^- làm T_3^+ và xuất hiện dòng $I_i' \neq 0$;

T_1^- làm T_2^+ và T_3^- ; $I_i = 0$ tương đương lúc này S_i nối tới A_i . Trong mạch này các tranzito T_1 , T_2 và T_3 đóng vai trò tiếp điểm A_i và $T_1 T_2 T_3$ – tiếp điểm B_i của chuyển mạch S_i .

11.5. MẠCH DAC DÙNG PHƯƠNG PHÁP MÃ HÓA LIÊN TIẾP (mã hóa Shannon)

Có thể tạo được điện áp ra trên một tụ điện tỷ lệ với số Z nhị phân tác động tới lối vào nhờ phép biến đổi liên tiếp từng bit. Nội dung của phương pháp mã hóa Shannon gồm các điều kiện sau:

1. Quá trình biến đổi phải thực hiện lần lượt từng bit vào từ LSB (trẻ nhất) đến kết thúc ở bit MSB (già nhất). Như vậy, mỗi bit cần một nhịp biến đổi với thời gian tương ứng là $T_{nhịp} = T_B$ (T_B là chu kỳ bit của Z).

2. Mỗi nhịp biến đổi khi thực hiện phải chia đều thành hai bước:

Bước 1: Dùng một nguồn ổn dòng nạp điện cho một tụ mẫu C_0 để điện áp trên tụ khi kết thúc bước 1 tăng lên một lượng U_0 nếu bit đang biến đổi có trị 1.

Tụ C_0 sẽ không được nạp điện nếu bit đang biến đổi có trị 0.

Bước 2: Cho tụ C_0 phóng điện qua một điện trở chính xác có giá trị R_0 , sao cho lúc kết thúc bước 2 điện áp trên C_0 còn lại 50% giá trị có được sau bước 1. Quá trình tiếp diễn cho tới MSB, khi đó điện áp ra:

$$U_Z = U_C(nT_{nhịp}) \approx \frac{U_0}{2^n}(Z)_{10}; 0 \leq Z \leq 2^n - 1 \quad (11.16)$$

với n là số bit của Z .

Cần lưu ý hệ thức (11.16) chỉ đúng khi bit LSB ở nhịp đầu tiên có giá trị 1. Còn nếu có k bit trẻ có trị 0 liên tiếp (tính từ LSB) thì hệ thức (11.16) cần phải hiệu chỉnh bằng cách tăng lên 2^k lần.

Mạch điện thực hiện theo phương pháp mã hóa Shannon cho trên hình 11.10 và đồ thị thời gian biểu diễn biến thiên điện áp trên tụ C_0 cho trên hình 11.11 (còn gọi là đồ thị Shannon – Rack) ví dụ $Z = 0110\ 1001$. Trạng thái các chuyển mạch điện tử S_1 và S_2 trong một nhịp biến đổi là:

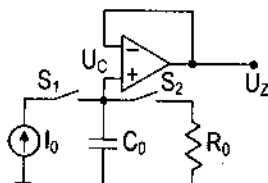
Bước 1: S_1^+ khi $Z_i = 1$

S_1^- khi $Z_i = 0$

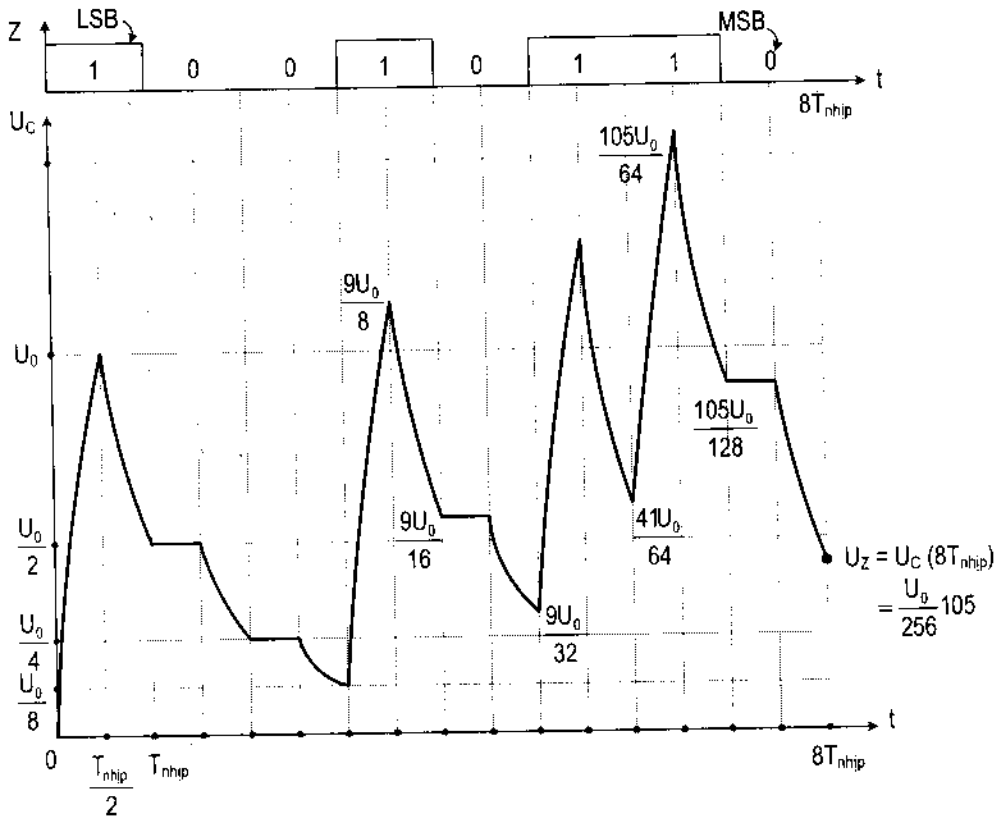
S_2^-

Bước 2: S_1^-

S_2^+



Hình 11.10. Mạch điện DAC theo phương pháp mã hóa Shannon



Hình 11.11. Đồ thị Shannon – Rack cho $Z = 0110\ 1001$

Chú ý rằng từ điều kiện quá trình phóng (bước 2) của tụ C_0 với phương trình cho quy luật biến thiên của $U_C(t)$, ràng buộc điều kiện đầu và điều kiện cuối:

$$U_C(t) = U_C(0) \exp\left(-\frac{t}{R_0 C_0}\right)$$

$U_C(0) = U_0$ (chọn gốc thời gian là lúc C_0 bắt đầu phóng)

$$U_C(t \approx \frac{T_B}{2} = \frac{T_{nhip}}{2}) = \frac{U_0}{2}$$

sẽ tìm được quan hệ giữa $\tau_0 = R_0 C_0$ và $T_B = T_{nhip}$ là:

$$T_B = 0,7 R_0 C_0 \quad (11.17)$$

(đây là giá trị gần đúng vì khi đó $U_C\left(\frac{T_B}{2}\right) = 0,496 U_0 \approx 0,5 U_0$)

Điện áp lỗi ra theo phương pháp Shannon tính theo hệ thức:

$$U_Z = U_C(n.T_B) = \frac{U_0}{2^n} \cdot (Z)_{10} \quad (0 \leq Z \leq 2^n - 1) \quad (11.18)$$

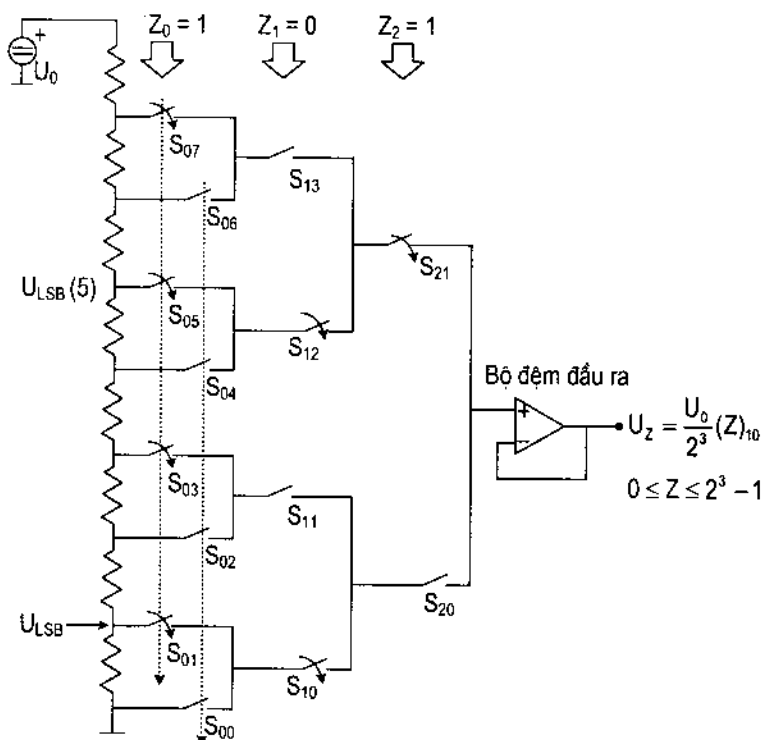
11.6. MẠCH DAC DÙNG PHƯƠNG PHÁP CHỌN KÊNH (ma trận $2^n R$)

Nội dung của phương pháp chọn kênh gồm hai bước chính:

1. Chia đều một nguồn chuẩn U_0 thành 2^n mức sao cho hai mức liên tiếp hơn kém nhau một đơn vị lượng tử U_{LSB} .
2. Dùng một bộ dồn kênh n tầng ghép liên tiếp nhau để chọn một trong các mức đã chia theo địa chỉ xác định của Z đưa đến lối ra, khi đó:

$$U_Z = \frac{U_0}{2^n} \cdot (Z)_{10} = U_{LSB} (Z)_{10} \quad (0 \leq Z \leq 2^n - 1) \quad (11.19)$$

Mạch điện thực hiện cho trên hình 11.12 ví dụ với $Z = Z_2 Z_1 Z_0 = 101$ (3 bit lối vào).



Hình 11.12. DAC theo phương pháp dồn kênh (ma trận $2^n - R$) cho số Z có 3 bit ở lối vào

Bộ chọn kênh trên hình 11.12 dùng các chuyển mạch điện tử có thể ký hiệu chúng là S_{ij} . Chỉ số i của chuyển mạch chỉ rõ nó chỉ chịu

sự điều khiển của bit Z_i trong biểu diễn Z của đại lượng số nhị phân ở lối vào.

Cách điều khiển của Z_i như sau:

- Nếu $Z_i = 1$ nhóm chỉ số j lẻ sẽ được nối mạch đồng thời.
- Nếu $Z_i = 0$ nhóm chỉ số j chẵn sẽ được nối mạch đồng thời.

Vậy các chuyển mạch S_{ij} chẵn (cùng nhóm j chẵn) và S_{ij} lẻ (cùng nhóm j lẻ) sẽ làm việc thành hai nhịp: trong cùng một nhóm làm đồng thời và giữa hai nhóm làm việc ngược pha nhau (theo trị của Z_i). Trong ví dụ với đại lượng vào 3 bit:

$Z = Z_2Z_1Z_0 = 101$, hoạt động của S_{ij} như sau (hình 11.12).

Vì $Z_0 = 1$ nên $S_{01}^+ S_{03}^+ S_{05}^+ S_{07}^+$ (nối) còn $S_{00}^- S_{02}^- S_{04}^- S_{06}^-$ (ngắt).

$Z_1 = 0$ nên $S_{10}^+ S_{12}^+$ còn $S_{11}^- S_{13}^-$ (ngắt).

$Z_2 = 0$ nên S_{21}^+ (nối) còn S_{20}^- (ngắt).

Chỉ có một đường nối duy nhất qua được mọi tầng của bộ chọn kênh là $S_{05}^+ \rightarrow S_{12}^+ \rightarrow S_{21}^+$ đưa mức $\frac{U_0}{2^3} \cdot 5$ từ ma trận chia $2^n - R$ đặt tới đầu ra.

$$U_Z = \frac{U_0}{2^3} \cdot 5 = U_{\text{LSB}}(1 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0)$$

Trong trường hợp tổng quát với n bit vào của Z thì sẽ nhận được hệ thức tương tự như hệ thức 11.19:

$$U_Z = \frac{U_0}{2^n} \cdot (Z)_{10} = U_{\text{LSB}}(Z)_{10} \quad (0 \leq Z \leq 2^n - 1)$$

Để tránh ảnh hưởng của tải tới hoạt động của mạch biến đổi, cần sử dụng một bộ đệm ở cổng ra. Số lượng chuyển mạch S_{ij} sẽ tăng nhanh (theo cấp số nhân) khi số bit của Z ở lối vào tăng nên phương pháp này chỉ thích hợp với công nghệ vi điện tử cỡ lớn LSI hay cực lớn (VLSI).

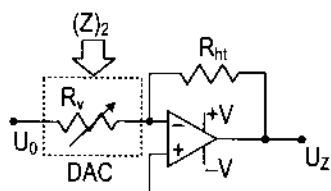
11.7. MỘT VÀI ỨNG DỤNG CỦA CÁC MẠCH DAC

Nếu mô tả thang hay mạng điện trở trên hình 11.4 hay trên hình 11.7 tương đương như một điện trở R_v có giá trị nhảy bậc theo sự điều

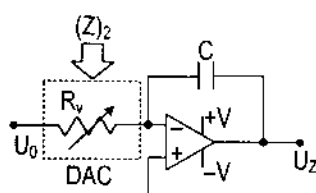
khuyến của Z (hình 11.13) ta nhận được một mạch khuếch đại có hệ số khuếch đại biến thiên nhảy bậc gồm tối đa $(2^n - 1)$ bậc theo sự điều khiển của Z , $U_Z = U_0 \left(-\frac{R_{ht}}{R_v} \right)$. Nguyên lý này được sử dụng trong

cách điều chỉnh có lập trình. Ví dụ, âm lượng hay độ sáng, tối, độ tương phản của tín hiệu hình... vẫn gặp trong các thiết bị điện tử dân dụng thể hệ điều chỉnh kiểu có lập trình phổ biến hiện nay.

Nếu thay thế điện trở hồi tiếp R_{ht} trong hình 11.13 bằng một tụ điện C ta nhận được một mạch tích phân với hằng số thời gian tích phân là $\tau_{tích\ phân} = R_v.C$. Như vậy, dưới tác động của Z , R_v biến thiên nhảy bậc và làm $\tau_{tích\ phân}$ thay đổi theo dạng nhảy bậc, ta có một bộ tích phân lập trình $\tau_{tích\ phân}$ theo một chương trình có ứng dụng trong đo lường, điều khiển tự động (hình 11.14).

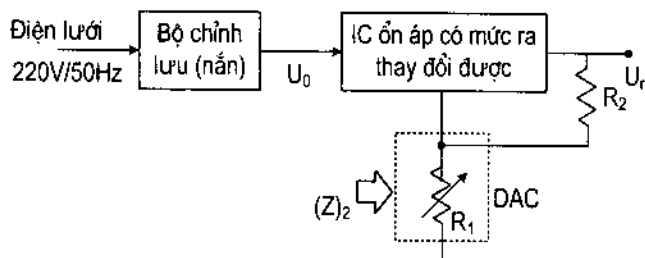


Hình 11.13. Bộ khuếch đại có hệ số khuếch đại biến thiên nhảy bậc theo Z



Hình 11.14. Mạch tích phân có $\tau_{tích\ phân} (\tau_{tích\ phân} = R_v.C)$ biến thiên nhảy bậc theo Z

Trên hình 11.15 có mô tả một bộ nguồn một chiều có mức ra thay đổi kiểu bậc thang (lập trình được) được sử dụng ví dụ trong việc điều khiển các động cơ bước (loại một chiều).



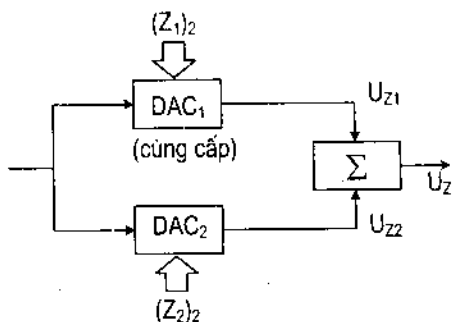
Hình 11.15. Nguồn một chiều có mức ra thay đổi nhảy bậc (lập trình được)

Các bộ DAC còn được sử dụng trong các mạch tạo hàm như phép cộng hai số nhị phân, phép nhân hai số nhị phân (kiểu mạch lai) hay

tạo hàm sin, hàm răng cưa quét tuyến tính, nhờ phương pháp xấp xỉ bậc thang.

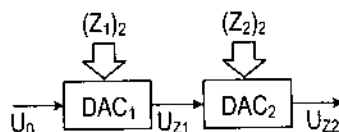
Hình 11.16 mô tả phép cộng hai số nhị phân Z_1 và Z_2 nhờ hai mạch DAC ghép song song.

Hình 11.17 mô tả phép nhân Z_1, Z_2 nhờ hai mạch DAC ghép nối tiếp.



Hình 11.16. Mạch cộng lại

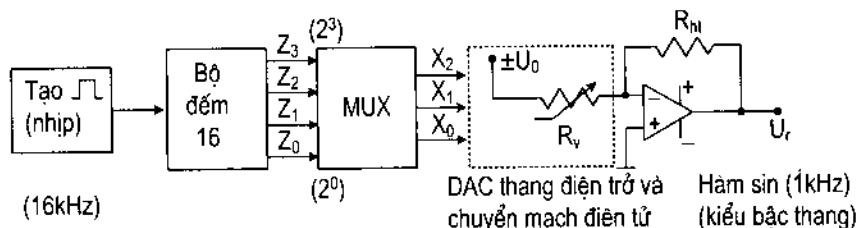
$$U_z = U_{z_1} + U_{z_2}$$



Hình 11.17. Mạch nhân lại

$$U_z = U_{z_1} U_{z_2}$$

Hình 11.18 đưa ra một cấu trúc ứng dụng một bộ DAC để tạo hàm điều hòa hình sin nhờ việc xấp xỉ kiểu bậc thang (một chu kỳ gồm 16 bậc) có sử dụng bộ khuếch đại hình 11.13 để tạo bậc thang.



Hình 11.18. Tạo hàm sin xấp xỉ kiểu bậc thang dùng DAC

Điện trở R_v gồm 4 cặp giá trị được chọn kiểu 1 trong 8 nhờ một chuyển mạch điện tử 8 vị trí do mã $(X_2 X_1 X_0)$ điều khiển làm việc kiểu quét với chu kỳ là $\frac{1}{16}$ ms đưa điện áp mẫu $\pm U_0$ vào khuếch đại

với hệ số khuếch đại bằng $\left(-\frac{R_{ht}}{R_v} \right)$ để tạo các giá trị điện áp ra dạng

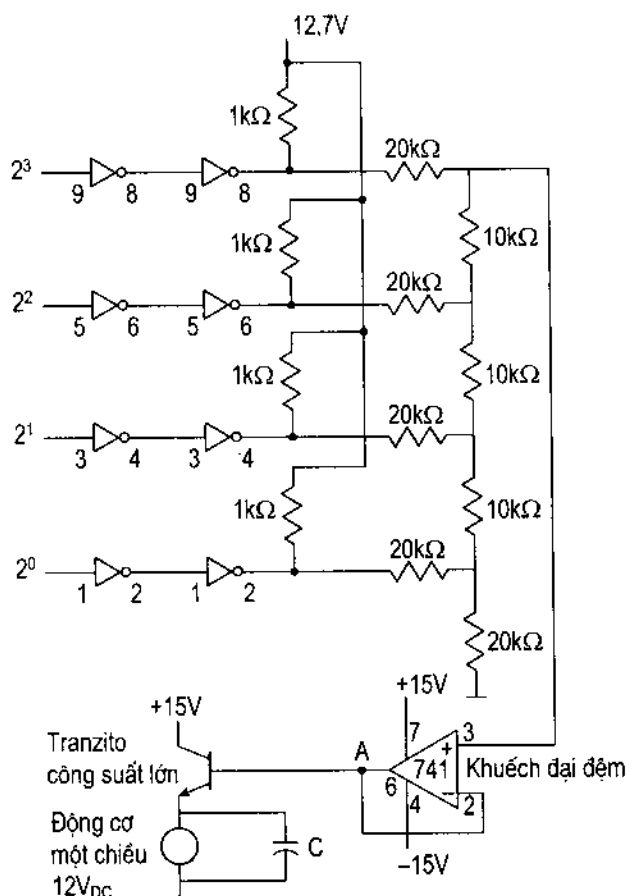
bậc thang gồm 8 bậc dương (lúc đưa $-U_0$ vào) và 8 bậc âm (lúc đưa $+U_0$ vào).

11.8. CÂU HỎI VÀ BÀI TẬP

1. Hãy nêu vị trí và chức năng của một mạch DAC trong quá trình xử lý số tín hiệu. Đại lượng tại đầu ra của mọi DAC có thuộc tính gì?
Có thể tác động các đại lượng số trong bộ đếm hệ 5, hệ 8 hay hệ 12 vào một mạch DAC được không? Nếu được thì có đặc điểm gì cần quan tâm nhất trong việc tổ chức kết cấu và chọn giá trị các linh kiện?
2. Hãy vẽ một mạch điện thực hiện DAC dùng bộ cộng áp cho các số nhị phân lối vào 8 bit. Nêu các điều kiện làm việc của mạch. Hãy viết biểu thức xác định giá trị của điện áp ra.
3. Khi tại lối vào DAC của bài tập 2 có mã BCD 8421 (8bit = 2 decac) tác động mạch đã có ở bài tập 2 cần thay đổi những gì để có thể làm việc được với mã vào BCD8421 này? Có mấy cách thay đổi?
Hãy xác định kết quả tại lối ra U_z trong hai trường hợp:
a) Z ở mã nhị phân tự nhiên $Z = 11000101$
b) Z ở mã BCD8421 $Z = 10010110$
Biết $U_0 = 16V$, $R_{ht} = 1k\Omega$; $R_7 = 4k\Omega$.
4. Hãy vẽ hai mạch DAC 8 bit loại thang điện trở và loại mạng điện trở $R - 2R$ (tương tự mạch hình 11.5a). Khi tác động vào mạch số nhị phân $Z = 01101110$.
a) Tính U_{LSB} và U_r khi $U_0 = 25,6V$.
b) Tính sai số ΔU_0 do $R_{nguồn} = 0,02\Omega$ gây ra cho mỗi loại mạch.
c) Nếu lối vào là số Z 5 bit, ví dụ 11001, có dùng được các mạch DAC đã vẽ không khi chọn $U_0 = 15,5V$, hãy tính U_{LSB} và U_r trong trường hợp dùng được.
5. Cho mạch điện hình 11.19.
a) Hãy phân tích nhiệm vụ các khối linh kiện trong mạch và giải thích nguyên lý hoạt động của mạch đã cho.
b) Khi cho số lối vào thay đổi từ 0000 đến 1111 hãy lập bảng giá trị điện áp tương ứng tại điểm A của mạch.

c) Muốn cho động cơ tải quay với tốc độ khác nhau cần tác động tới lối vào như thế nào? Khi mắc các đầu vào của mạch hình 11.19 tới các đầu ra song song của một bộ đếm 16 (hình 8.15 bài 8) động cơ tải sẽ hoạt động như thế nào?

d) Thay $\frac{4}{6}$ IC7406 bằng loại 7404 có được không? Vì sao? Bộ khuếch đại đệm IC741 ở đây có nhiệm vụ gì?



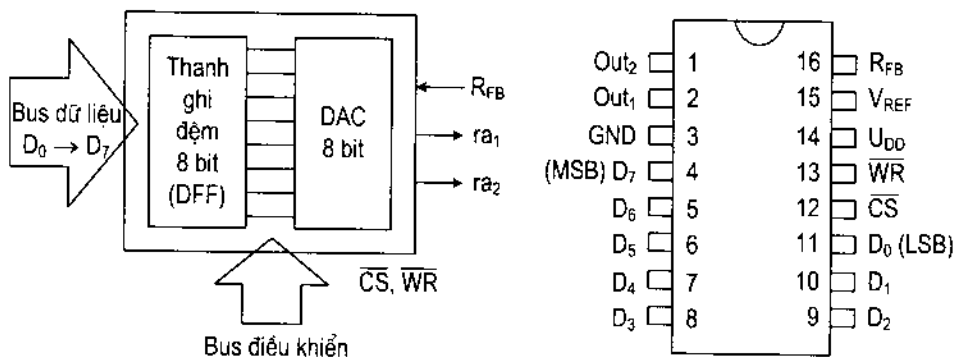
Hình 11.19. Bài tập 5

6. Cho số nhị phân 8 bit ví dụ $Z = 1001\ 0100$ tác động vào mạch DAC mạng điện trở (tương tự hình 11.7).

a) Hãy vẽ mạch điện cấu tạo và nêu các điều kiện làm việc của mạch.

b) Tính sai số do nội trở $R_{nguồn} = 0,01\Omega$ của U_0 gây ra cho mạch (phần của U_0 bị mất trên $R_{nguồn}$ không cấp được cho mạch) khi $R = 10k\Omega$ với $U_0 = 12,8V$. Tính dòng điện ra I_Z .

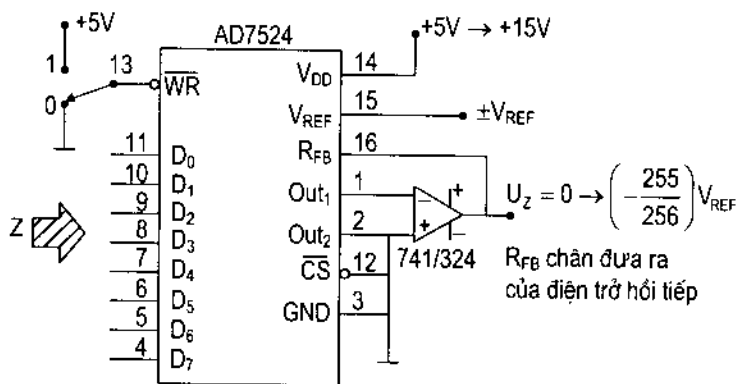
- c) Nếu các số lối vào ở dạng BCD8421 (2 decac) thì mạch cần có gì thay đổi để phù hợp với mã vào?
7. Hình 11.20 cho cấu trúc khối và bố trí chân của vi điện tử DAC loại AD7524 và hình 11.21 cho cách nối các chân của IC trên để tạo thành DAC 8 bit.



Hình 11.20. IC AD7524

IC AD7524 thuộc nhóm DAC 8 bit loại mạng điện trở R – 2R công nghệ CMOS với giá trị $R = 10k\Omega$; $R_{in} = 10k\Omega$ (hình 11.7) chân R_{FB} là lối ra của R_{in} .

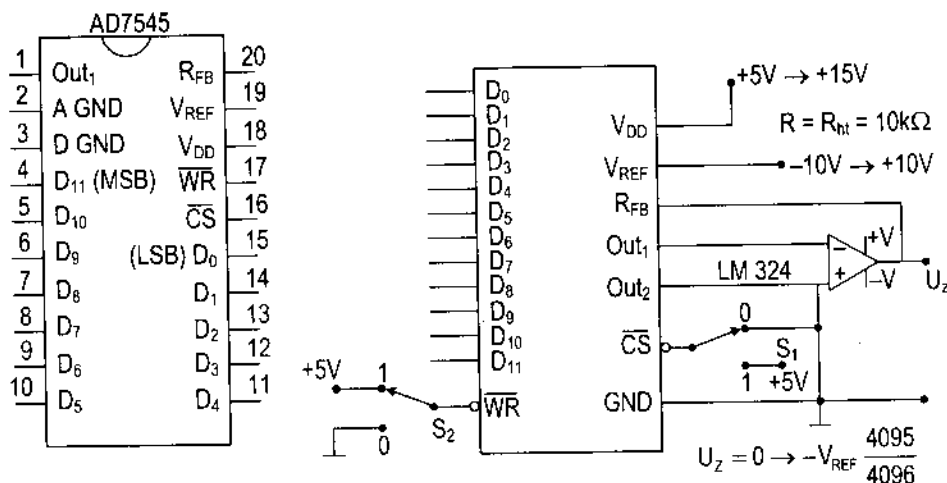
- a) Hãy nêu nhiệm vụ chức năng các chân của IC và mô tả quy trình làm việc theo cấu trúc hình 11.20.
- b) Với dữ liệu $D_0 \rightarrow D_7$ áp vào (ví dụ 1000 0111) hãy xác định điện áp tại lối ra trong hai trường hợp: giá trị của tín hiệu $\overline{WR}$ tích cực ở mức 0 ($\overline{WR} = 0$) và không tích cực (ở mức 1: $\overline{WR} = 1$) như thể hiện trên hình 11.21.



Hình 11.21. Cách nối DAC 8 bit dùng IC 7524

c) Với dữ liệu vào cần biến đổi chỉ có 6 bit $D_0 \rightarrow D_5$ hay 4 bit $D_0 \rightarrow D_3$ có dùng được mạch hình 11.21 không? Nếu được hãy nêu cách nối lại mạch đã cho.

8. Cho IC DAC (loại 12 bit dữ liệu vào) AD 7545 có sơ đồ bố trí chân và cách nối mạch trên hình 11.22.



Hình 11.22. Bố trí chân và cách nối IC AD7545

a) Hãy nêu nhiệm vụ và chức năng các chân của IC, mô tả quy trình làm việc thực hiện DAC khi dùng IC này theo cách nối đã thể hiện.

b) Phân tích chế độ của mạch tương ứng với 4 trạng thái khác nhau của hai chuyển mạch S₁ và S₂ trên hình vẽ và lập thành bảng chức năng điều khiển của IC với các lối vào và lối ra tương ứng.

c) Tìm hệ thức tính điện áp ra U_Z và tính U_Z khi Z = 1100 0101 1010 khi đặt U_{ref} = -10V với WR = 0, CS = 1 và với WR = 0, CS = 0.

d) Với các dữ liệu vào chỉ có 8 bit hay 4 bit có dùng được IC7545 không? Nếu được hãy chỉ ra cách đấu lại mạch hình 11.22.

Các tham số của 7545: Dòng tiêu thụ lớn nhất 2mA;

Thời gian biến đổi 2μs;

R_v (của V_{REF}) ~ 11kΩ;

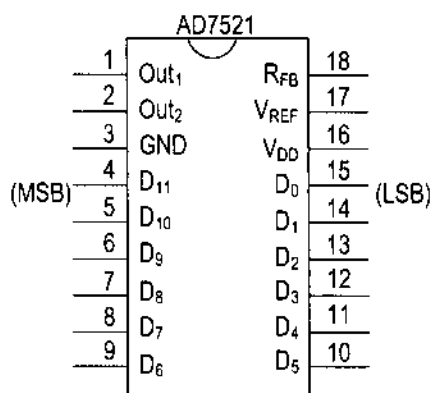
Điện áp nuôi V_{DD} +5V đến +15V;

Độ phân giải 12bit vào.

9. Cho IC 7521 có sơ đồ chân trên hình 11.23.

a) Hãy kết hợp với một IC tuyến tính (741 hay 324) vẽ mạch điện đấu chân thực hiện DAC 12bit.

b) Khi cần chức năng như AD7545 (bài tập 8) kiểm soát dữ liệu đầu vào trước khi thực hiện biến đổi DA cần bổ sung



Hình 11.23. Sơ đồ bố trí chân của IC AD7521

phần tử IC nào tại các lối vào dữ liệu phù hợp với lệnh $\overline{WR} = 0$ (là lệnh nạp dữ liệu chuẩn bị biến đổi).

c) Có thể bổ sung lệnh $\overline{CS}$ bằng cách nào đơn giản nhất?

10. Cho các mạch ứng dụng DAC0830 loại 8 bit dùng mạng điện trở R-2R trên các hình 11.24 và 11.25.

a) Hãy phân tích hoạt động của các mạch DAC đã cho và qua đó vẽ sơ đồ bố trí chân của IC (tương tự như hình 11.23 hay 11.20).

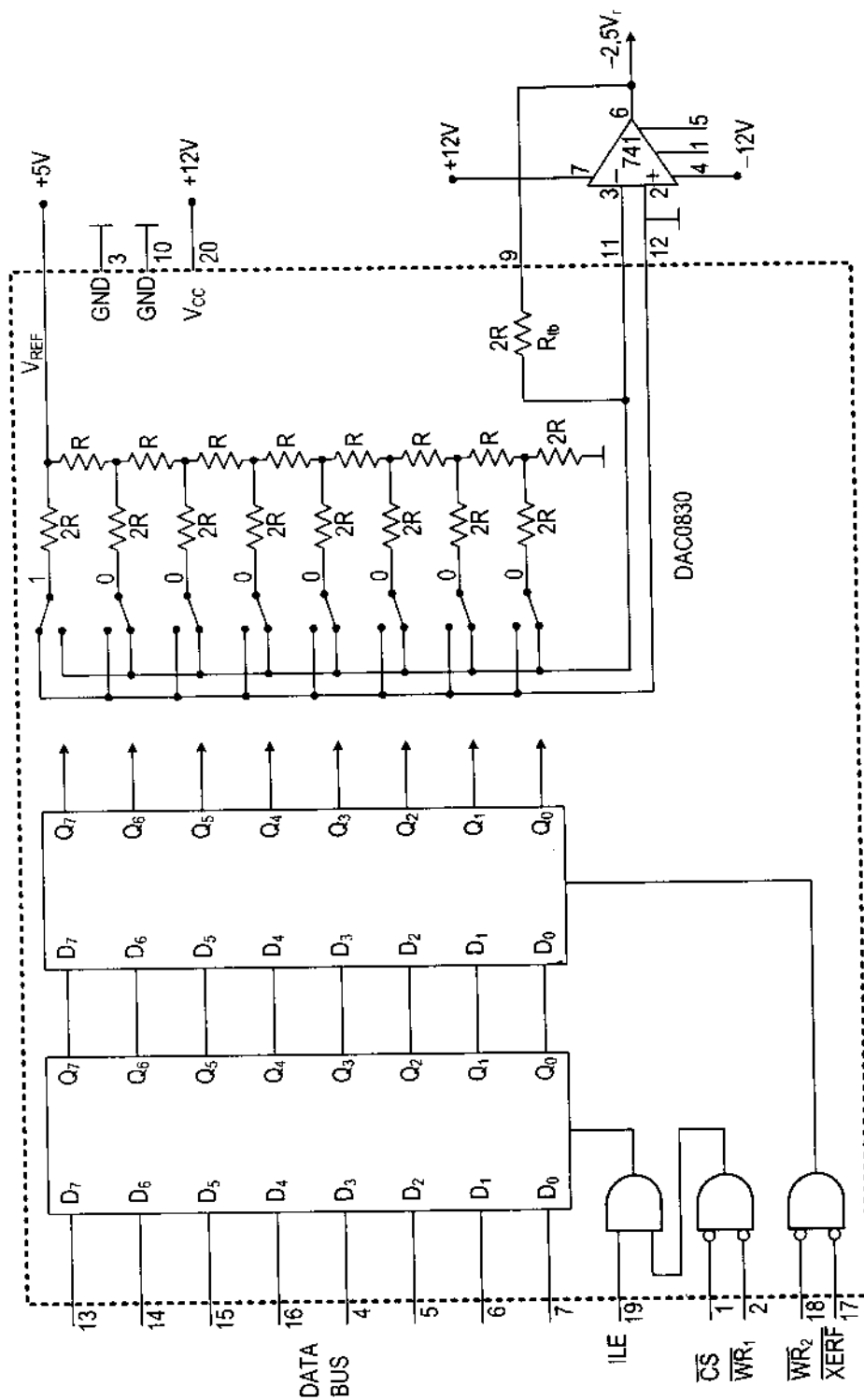
b) Xác định giá trị điện áp lượng tử U_{LSB} và các hệ thức tính U_r theo giá trị các biến Z lối vào cùng mức V_{REF} trong hai trường hợp đã cho. Tính U_r khi có $Z_1 = 1001\ 0110$ và $Z_2 = 0110\ 1001$ tác động vào mạch hình 11.24 hoặc vào mạch hình 11.25 nhận xét kết quả ra có liên quan gì tới giá trị vào (chú ý giả thiết $Z_2 = \overline{Z_1}$).

11 a) Hãy tìm hiểu bản chất của hai tín hiệu điều khiển đưa tới các bộ đệm nối tiếp nhau ở đầu vào của IC DAC 0830 và nêu lý do (ý nghĩa) việc cấu trúc kiểu này ở mạch vào dữ liệu.

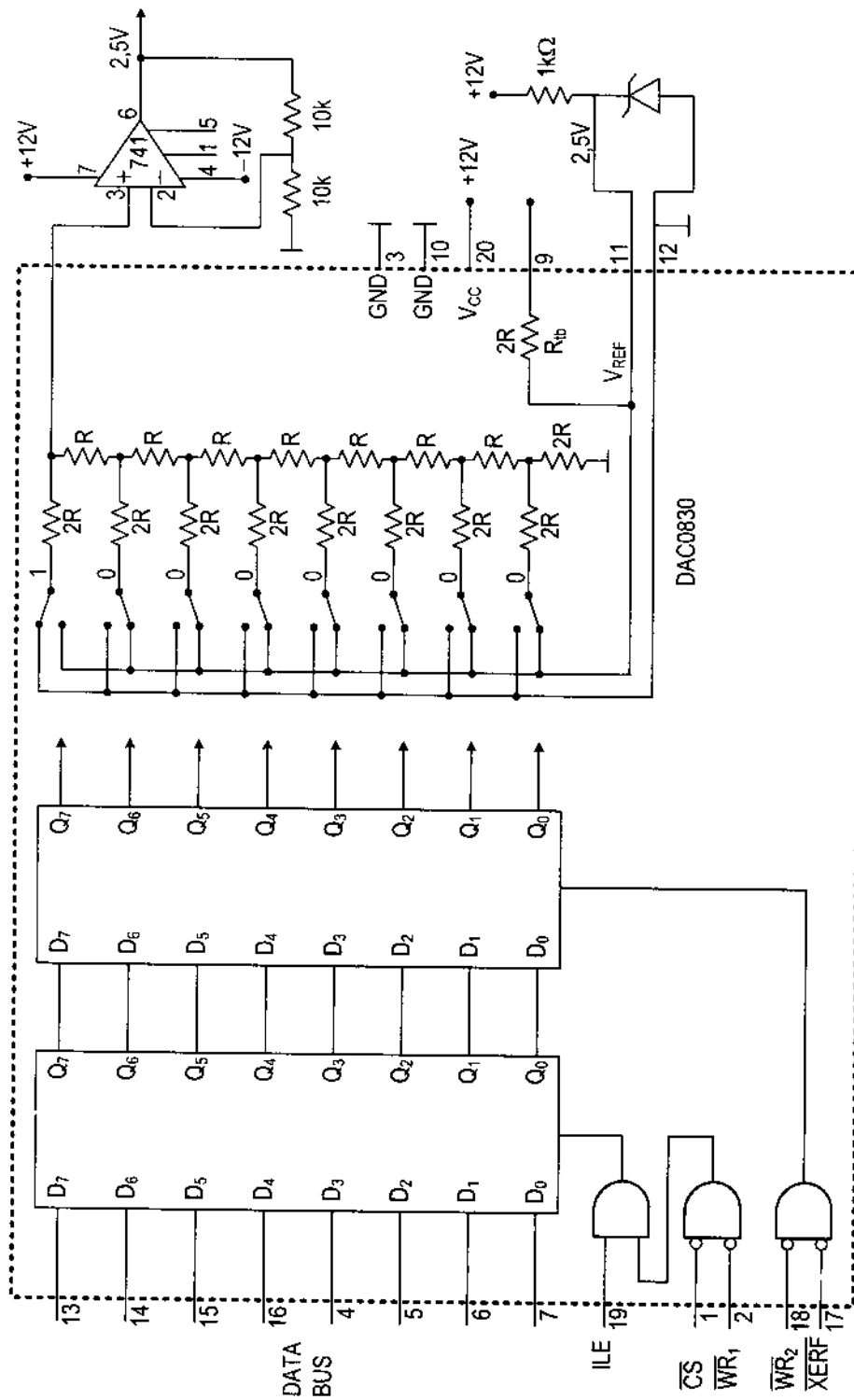
b) So sánh hai cách nối bộ khuếch đại đệm tại lối ra về hai mặt:

- Việc phối ghép giữa mạng R – 2R với tải tiếp nhận điện áp ra.
- Sai số trong việc chia V_{REF} của mạng điện trở R – 2R khi đưa trực tiếp V_{REF} tới các nút trên của mạng (mạch hình 11.24) và đưa V_{REF} (có giá trị thấp hơn một nửa) qua các chuyển mạch điện từ S_i tới mạng R – 2R từ phía điện trở dọc (mạch hình 11.25).

(Hãy chú ý tới điện thế 0V của hai trường hợp tại các chuyển mạch S_i).



Hình 11.24. Cấu tạo IC DAC 0830 với đầu ra dùng bộ khuếch đại đảo dấu (cho bài tập 10)



Hình 11.25. Cấu tạo IC DAC 0830 với đầu ra dùng bộ khuếch đại không đảo (cho bài tập 10)

BÀI 12. MẠCH CHUYỂN ĐỔI TƯƠNG TỰ – SỐ (ADC)

12.1. KHÁI NIỆM CHUNG

12.1.1. Nhiệm vụ cơ bản của ADC

ADC có nhiệm vụ biến đổi một đại lượng tương tự tác động tới đầu vào của nó thành một đại lượng số (biểu diễn giá trị đại lượng vào) tại đầu ra. Để thực hiện được nhiệm vụ trên, mỗi ADC cần tiến hành liên tiếp 3 quá trình cơ bản là:

1. Thực hiện việc lấy mẫu đại lượng vào: chọn và giữ các giá trị U_x tại lối vào ở những thời điểm nhất định và lặp lại với chu kỳ là $T_{\text{lấy mẫu}}$. Đây là quá trình rời rạc hóa tín hiệu analog và cần tuân theo định lý lấy mẫu là tần số lấy mẫu phải đủ nhanh: $f_{\text{lấy mẫu}} \geq 2f_{x\text{max}}$ với $f_{x\text{max}}$ là tần số biến thiên nhanh nhất của U_x ở lối vào. Ví dụ, trong thông tin điện thoại $f_{x\text{max}} = 3400\text{Hz}$ nên chọn $f_{\text{lấy mẫu}} = 8000\text{Hz}$.

2. Tiến hành làm tròn các giá trị mẫu đã lấy và giữ gọi là quá trình lượng tử hóa. Các giá trị lấy mẫu là $U_k = U_x(k.T_{\text{lấy mẫu}})$ với k là số nguyên dương bất kỳ, khi đó cần chọn một giá trị có nghĩa nhỏ nhất gọi là đơn vị làm tròn (hay đơn vị lượng tử, ký hiệu là U_{LSB}) từ việc chia nhỏ giá trị của một nguồn điện áp tham chiếu U_{REF} và đánh giá U_k được bao nhiêu lần phần nguyên của đơn vị làm tròn :

$$U_k \approx k_x \cdot U_{\text{LSB}}$$

3. Biểu diễn số các giá trị k_x vừa tìm được trong hệ đếm mong muốn gọi là quá trình mã hóa: k_x trong hệ đếm 10 thành Z_x , ví dụ trong hệ đếm 2 hay hệ 2 – 10.

12.1.2. Các sai số quan trọng khi thực hiện ADC

Các sai số quan trọng nhất có thể gặp phải trong khi thực hiện ADC là:

– Sai số khe ΔU_k phân bố xung quanh U_k do $T_{\text{mẫu}}$ không chính xác (xuất hiện với dung sai $\Delta T_{\text{mẫu}}$). Sai số loại này do độ chính xác của $T_{\text{mẫu}}$ và phụ thuộc vào tốc độ biến thiên của U_x .

– Sai số lượng tử khi làm tròn U_k do đã bỏ qua phần lẻ của U_k so với U_{LSB} , chỉ lấy phần nguyên k_x . Theo quy tắc làm tròn nếu bỏ qua phần lẻ sau dấu phẩy thập phân, sai số này $< U_{LSB}$. Còn khi bỏ qua phần lẻ dưới $0,5U_{LSB}$ và lấy thêm một đơn vị (là $k_x + 1$) nếu phần lẻ $\geq 0,5U_{LSB}$ thì sai số lượng tử lớn nhất là $\pm 0,5U_{LSB}$.

– Các tham số cơ bản để đánh giá một bộ ADC là:

1. Độ phân giải (quyết định đến độ chính xác của phép biến đổi).

Độ phân giải do giá trị U_{LSB} quyết định $U_{LSB} = \frac{U_{ref}}{2^n}$ với n số bit biểu diễn U_x . Như vậy, U_{ref} càng nhỏ và càng chính xác, n càng lớn thì độ phân giải càng cao và do đó trị Z_x biểu diễn U_x càng gần với trị thực.

2. Tốc độ chuyển đổi được đánh giá theo số lượng các lần chuyển đổi trong 1 giây hay thời gian cần thiết để biến đổi xong một giá trị số của U_x (hoặc U_k).

3. Dải biến đổi là khoảng U_{xmin} đến U_{xmax} tại lối vào. Với yêu cầu nhất định về độ chính xác, dải biến đổi càng rộng thì số bit biểu diễn càng nhiều và do đó tốc độ biến đổi nói chung bị giảm.

12.1.3. Các phương pháp thực hiện ADC

Có hai phương pháp cơ bản: biến đổi trực tiếp từ U_x sang Z_x bằng phương pháp so sánh sau một nhịp hay sau nhiều nhịp biến đổi. Phương pháp thứ hai là biến đổi từ U_x qua một đại lượng trung gian là thời gian hay tần số tỷ lệ và do thời gian hay tần số sẽ suy ra giá trị tương ứng của U_x .

12.2. ADC THEO PHƯƠNG PHÁP SONG SONG (so sánh đồng thời)

ADC dùng phương pháp so sánh đồng thời nhận được kết quả biến đổi $U_x \rightarrow Z_x$ chỉ sau một nhịp làm việc nên có tốc độ nhanh nhất trong các ADC. Muốn vậy, cần thực hiện các bước sau:

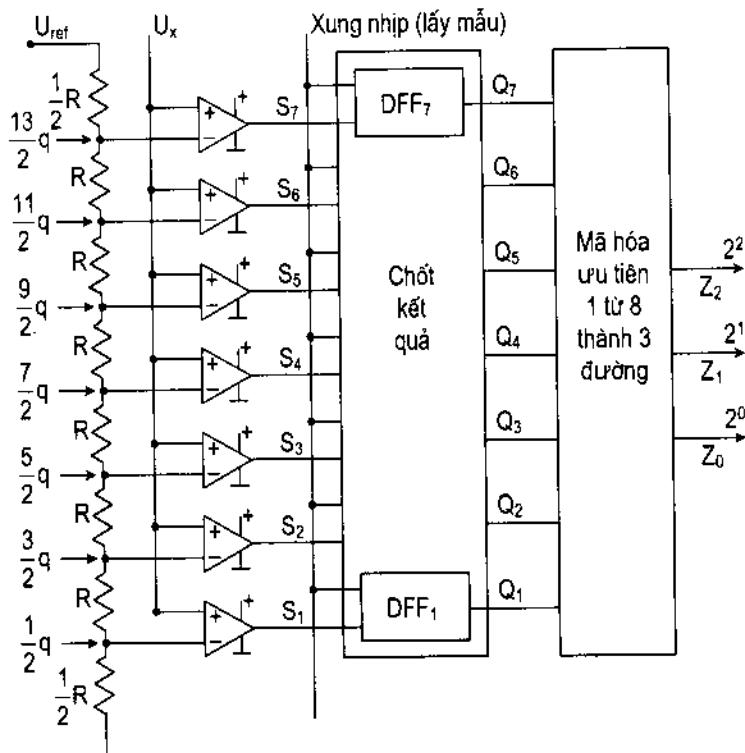
1. Chia nguồn điện áp mẫu U_{ref} thành $(2^n - 1)$ mức ngưỡng đều nhau. Mức có giá trị thấp nhất bằng $\frac{1}{2}U_{LSB}$ và hai mức ngưỡng liên tiếp nhau sẽ hơn kém nhau một đơn vị làm tròn (một bước lượng tử).

$$U_{i+1} - U_i = U_{LSB} \equiv q \quad (12.1)$$

2. Sử dụng $2^n - 1$ phần tử so sánh (ví dụ loại so sánh thuận) để thực hiện việc so sánh trực tiếp đại lượng vào U_x đồng thời với tất cả các mức ngưỡng U_i vừa tạo ra. Mục đích của bước này là tìm ra trạng thái $U_i \leq U_x < U_{i+1}$. Khi đó, vì $U_i \leq U_x$ nên tại các lối ra của nhóm bộ so sánh này có kết quả S_i ở mức cao ($S_i = S_{i-1} = S_{i-2} = \dots = 1$). Còn đối với nhóm trên, vì $U_x < U_{i+1}$ nên kết quả so sánh là các lối ra ở mức thấp ($S_{i+1} = S_{i+2} = \dots = 0$).

3. Thực hiện việc mã hóa $(i)_{10}$ thành $(Z_x)_2$ nhờ bộ mã hóa 1 từ i thành mã nhị phân 8421. Tuy nhiên vì đồng thời với $S_i = 1$ còn có $S_{i-1} = S_{i-2} = \dots = 1$ nên đây là loại mã hóa ưu tiên (chỉ số i) các chỉ số thấp hơn i dù ở tích cực cũng bị bỏ qua không được bộ mã hóa chấp nhận.

Mạch ADC kiểu song song với kết quả ra $Z_x = Z_2 Z_1 Z_0$ (3 bit) có cấu trúc khối được cho trên hình 12.1 và việc mô tả kết quả hoạt động của mạch cho trên bảng 12.1. Mọi bit kết quả Z_i được xác định giá trị (bằng 1 hay bằng 0) đồng thời sau một nhịp biến đổi. Khi số bit kết quả nhiều (độ phân giải cao hay U_x có giá trị lớn) số lượng linh kiện cần dùng tăng nhanh.



Hình 12.1. Cấu trúc khối của ADC song song 3 bit ra ($q \equiv U_{LSB}$ là đơn vị làm tròn)

Bảng 12.1. Mô tả hoạt động của ADC song song (3 bit)

Giá trị vào so sánh $U_i \leq U_x < U_{i+1}$	Trạng thái ra so sánh								Đầu ra mã hóa (ưu tiên)		
	S ₇	S ₆	S ₅	S ₄	S ₃	S ₂	S ₁	S ₀	Z ₂	Z ₁	Z ₀
$0 \leq U_x < \frac{q}{2}$	0	0	0	0	0	0	0	0	0	0	0
$\frac{q}{2} \leq U_x < \frac{3q}{2}$	0	0	0	0	0	0	0	1	0	0	1
$\frac{3q}{2} \leq U_x < \frac{5q}{2}$	0	0	0	0	0	0	1	1	0	1	0
$\frac{5q}{2} \leq U_x < \frac{7q}{2}$	0	0	0	0	0	1	1	1	0	1	1
$\frac{7q}{2} \leq U_x < \frac{9q}{2}$	0	0	0	0	1	1	1	1	1	0	0
$\frac{9q}{2} \leq U_x < \frac{11q}{2}$	0	0	0	1	1	1	1	1	1	0	1
$\frac{11q}{2} \leq U_x < \frac{13q}{2}$	0	0	1	1	1	1	1	1	1	1	0
$\frac{13q}{2} \leq U_x$	0	1	1	1	1	1	1	1	1	1	1

12.3. ADC DÙNG PHƯƠNG PHÁP XẤP XỈ NHIỀU NHỊP

12.3.1. ADC loại hai nhịp: thực hiện hai nhịp biến đổi chính

Nhịp 1: Thực hiện đánh giá đồng thời (song song) một nhóm bit cao (đánh giá thô) trước (ví dụ với $Z_x = Z_7 \dots Z_4 Z_3 \dots Z_0$ thì xác định $Z_7 Z_6 Z_5 Z_4$ trước).

Nhịp 2: Đánh giá tiếp đồng thời các bit còn lại ($Z_3 Z_2 Z_1 Z_0$), tức là đánh giá tinh.

Khi thực hiện phương pháp này (sơ đồ khối thực hiện cho trên hình 12.2) cần các lưu ý quan trọng sau:

1. Trước khi thực hiện nhịp đánh giá tinh, cần xác định phần dư còn lại trong U_x sau khi đã đánh giá thô, tức là phải tạo ra $U_{Z \text{ (thô)}}$ nhờ một DAC cùng số bit với nhóm bit thô và mang $U_{Z \text{ (thô)}}$ so sánh kiểu vi sai với U_x để lập phần dư $U_x - U_{Z \text{ (thô)}}$.

2. Tốc độ xử lý của DAC phải đủ nhanh và nguồn mẫu của DAC với ADC thô là giống nhau (cùng bằng U_{ref}).

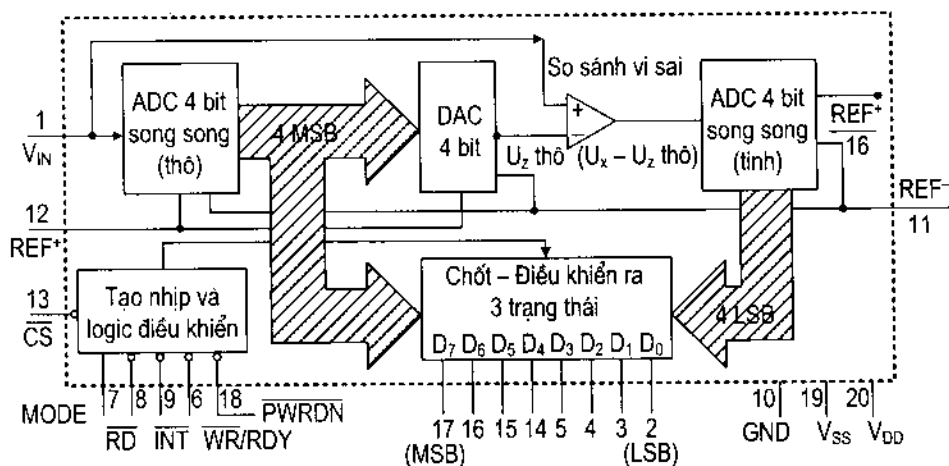
3. Nguồn mẫu cho ADC tính phải giảm đi một số cấp nhị phân đúng bằng số lượng bit thô đã đánh giá (ví dụ $U_{ref}/2^4$).

4. Cần có bộ chốt dữ liệu và điều khiển mạch ra 3 trạng thái để phối hợp hai nhóm kết quả.

12.3.2. ADC loại xấp xỉ dần từng bit (loại n nhị)

Nội dung phương pháp xấp xỉ dần từng bit là đánh giá từng bit kết quả, bắt đầu từ bit cao nhất (Z_{MSB}) và kết thúc ở bit thấp nhất (Z_{LSB}). Như vậy, với ADC n bit cần n nhị đánh giá từ thô nhất đến tinh nhất. Trong mỗi nhị cần thực hiện việc so sánh hơn kém để đánh giá giá trị của bit thử nếu kết quả là 1 ($U_x \geq U_z$) thì không cần lập phần dư. Như vậy, các nhị kế tiếp liên kế luôn phụ thuộc vào kết quả đánh giá của nhị ngay trước nó là hơn hay kém mẫu (bit) thử. Ngày nay, phần lớn các IC ADC đều là loại song song và loại xấp xỉ từng bit vừa nêu.

IC MAX 153 là loại ADC 8 bit 2 nhị có cấu trúc khối cho trên hình 12.2. Bố trí chân và chức năng các chân cho trên bảng 12.2.



Hình 12.2. Cấu trúc khối IC ADC MAX 153 loại 8 bit 2 nhị và vị trí các chân

Các thông số kỹ thuật của IC:

Thời gian biến đổi 660ns.

Thời gian chuyển chế độ tích cực/tiết kiệm: 200ns. Chuyển chế độ trích mẫu 160ns.

Mạch trích mẫu bên trong IC; Thông lượng tốc độ số hóa 1M mẫu/s.

Tiêu thụ năng lượng: chế độ bình thường 20mW, chế độ tiết kiệm 5μW.

Dải tần: 1MHz, không cần xung nhịp ngoài.

Đầu vào đơn cực và lưỡng cực.

Nguồn cấp +5V hoặc ±5V.

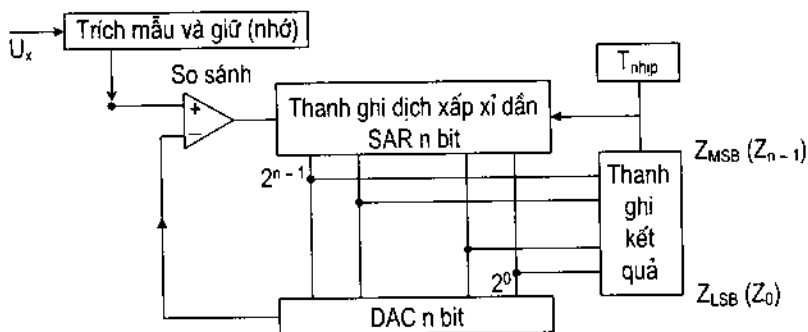
IC công nghệ CMOS AD775 có 24 chân làm việc theo phương pháp tương tự ADC MAX 153 với hai nhóm bit ra 4MSBs (chân số 10, 9, 8, 7) và chân 4LSBs (chân 6, 5, 4, 3) với đầu ra mạch 3 trạng thái $\overline{OE} = 0$ cho phép, ra $\overline{OE} = 1$ trạng thái Hi - z (chân số 1); đầu vào analog (chân 19) và nguồn cấp các chân 11 và 13 cho phần digital. chân 14, 15, 18 cho phần analog 5V.

Bảng 12.2. Chức năng và vị trí các chân của IC ADC MAX 153

Số chân	Ký hiệu	Chức năng logic
1	V_{IN}	Đầu vào tương tự, dải điện áp $V_{REF-} \leq V_{IN} \leq V_{REF+}$
2 → 5	$D_0 \rightarrow D_3$	Dữ liệu ra 3 trạng thái (nhóm 4LSBs)
6	$\overline{WR}/RDY$	Đầu vào điều khiển ghi/Đầu ra báo trạng thái sẵn sàng
7	MODE	Chọn chế độ đọc (MODE = 0) hay ghi - đọc (MODE = 1)
8	$\overline{RD}$	Điều khiển để đọc dữ liệu khi $\overline{RD} = 0$ / Nếu $\overline{WR} = \overline{RD} = 0$ (*)
9	INT	Đầu ra báo quá trình biến đổi đã hoàn thành
10	GND	Đất 0V
11	V_{REF-}	Điện áp chuẩn giới hạn dưới $V_{SS} \leq V_{REF-} < V_{REF+}$
12	V_{REF+}	Điện áp chuẩn giới hạn trên $V_{REF-} \leq V_{REF+} < V_{DD}$
13	$\overline{CS}$	Điều khiển chọn vỏ (cho IC hoạt động) khi $\overline{CS} = 0$
14 → 17	$D_4 \rightarrow D_7$	Dữ liệu ra 3 trạng thái (nhóm 4MSBs)
18	$\overline{PWRDN}$	Đầu vào đặt chế độ tiết kiệm năng lượng (5μW)
19	V_{SS}	Nguồn cấp điện áp âm đơn cực: $V_{SS} = 0$ hay lưỡng cực $V_{SS} = -5V$
20	V_{DD}	Nguồn cấp điện áp dương +5V

(*) Báo hiệu quá trình biến đổi mới, đồng thời đọc kết quả lần biến đổi trước khi $\overline{CS} = 0$.

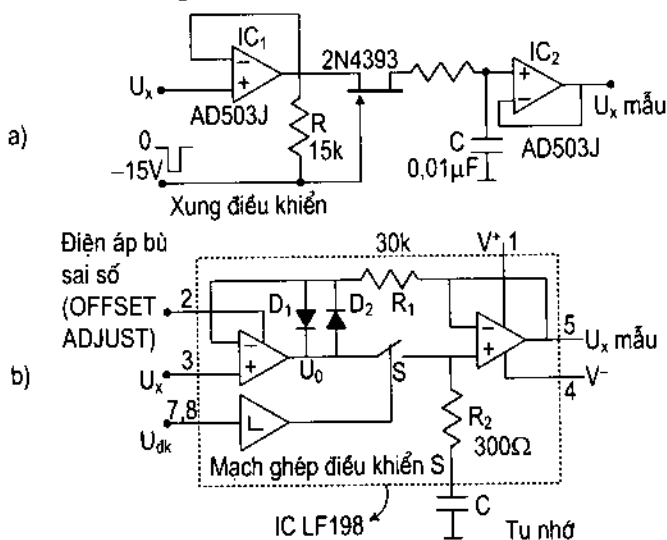
Hình 12.3 là cấu trúc khối của ADC loại xấp xỉ từng bit sử dụng một thanh ghi dịch xấp xỉ dần (SAR – Successive Approximation Register) có dạng tương tự như thanh ghi hình 8.27.



Hình 12.3. Cấu trúc khối ADC xấp xỉ từng bit cho số kết quả có n bit nhị phân

Phương pháp xấp xỉ từng bit có tốc độ chậm hơn phương pháp song song nên cần trích mẫu và lưu giữ (nhớ) kết quả đã trích để trong cả quá trình thực hiện gồm n nhịp, các giá trị Z_{n-1} đến Z_0 đều có được chỉ từ một trị lưu của U_x , tránh sai số do U_x thay đổi khi đang làm. Như vậy, chu kỳ lấy mẫu phải lớn hơn $n \cdot T_{nhịp}$. Tất nhiên, khi U_x là đại lượng một chiều thì không cần khối này.

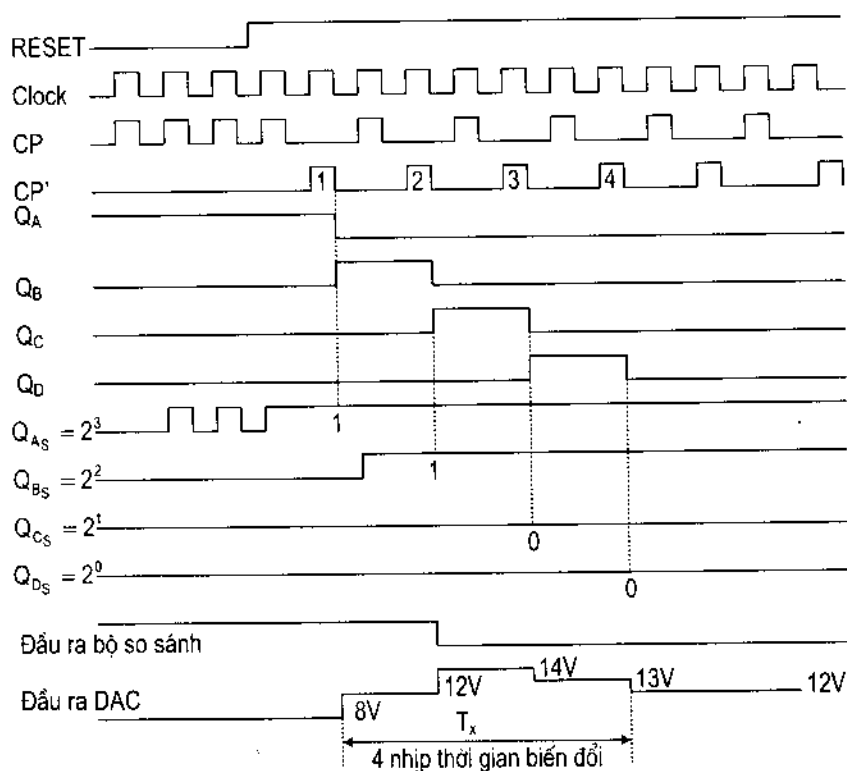
Mạch trích mẫu và nhớ điển hình cho trên hình 12.4. Trên hình 12.4a) phần tử khóa analog là tranzito JFET kênh N loại 2N4393, ở chế độ nổi mạch khi $U_{dk} = 0$, sơ đồ đang thực hiện lấy mẫu. Trị lấy mẫu được nạp trên tụ nhớ. Khi $U_{dk} = -15V$, tranzito chuyển sang chế độ ngắt mạch (dòng $I_D = 0$), giá trị tín hiệu được lưu lại trên C trong thời gian thực hiện biến đổi. Các IC_1 và IC_2 là bộ đệm để chống suy hao do trôi điện áp nhớ trên C (IC_2) và làm tăng tốc độ lúc lấy mẫu khác (IC_1). Điện trở R để giảm ảnh hưởng trễ do tụ C_{GS} gây ra lúc lấy mẫu.



Hình 12.4. Các dạng mạch trích mẫu – giữ mẫu



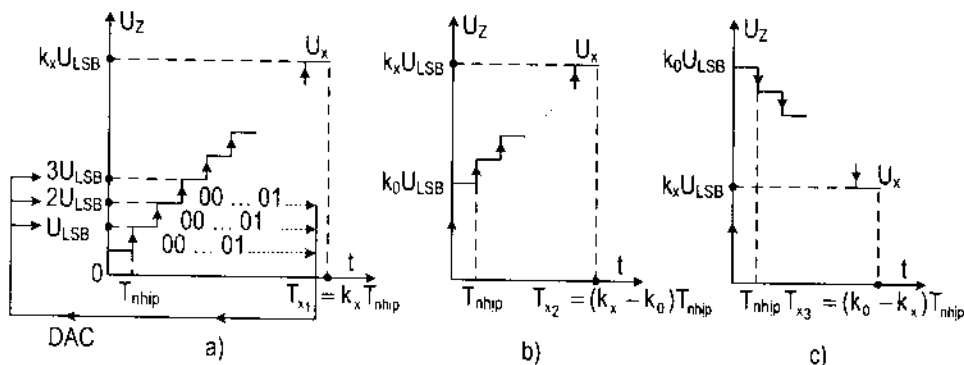
Trên mạch hình 12.5a, nhóm 4JKFF phần trên (ABCD) là thanh ghi xấp xỉ dần SAR, nhóm 4JKFF phía dưới được sử dụng như một thanh ghi 4 bit chốt các kết quả tại lối ra của ABCD ($Q_A Q_B Q_C Q_D$) vào các FF tương ứng $Q_A = Q_{A_S}$; $Q_B = Q_{B_S}$; $Q_C = Q_{C_S}$; $Q_D = Q_{D_S}$. Việc điều khiển hoạt động cho thanh ghi dịch xấp xỉ từng bit SAR thực hiện nhờ hai cổng AND: CP và CP'. Các kết quả bit sau mỗi nhịp được ghi lại tại các FF $A_S B_S C_S D_S$, hiển thị nhờ LED như trên hình vẽ 12.5a. Nhánh bit có LED sáng thể hiện kết quả $Q = Z_i = 1$ còn khi LED tối thể hiện kết quả $Z_i = 0$ (khi $Q = 1$ thì $\bar{Q} = 0$ và LED được kích thích sáng). Đồ thị dạng sóng trên các lối ra $Q_A Q_B Q_C Q_D$ và $Q_{A_S} Q_{B_S} Q_{C_S} Q_{D_S}$ và dạng sóng tại đầu ra DAC, đầu ra so sánh được vẽ trên hình 12.6.



Hình 12.6. Đồ thị dạng sóng của mạch hình 12.5 a) giải thích chu trình xấp xỉ từng bit (cho ví dụ lối vào có $U_x = 12,5V$, với $U_{LSB} = \frac{U_{ref}}{2^4} = 1V$)

12.4. ADC DÙNG PHƯƠNG PHÁP ĐẾM (loại $2^n - 1$ nhị phân hay xấp xỉ từng trạng thái)

ADC theo phương pháp đếm dùng cách xấp xỉ nhờ các bậc thang đều U_z tại lối ra của một DAC, thực hiện bám đến U_x từ giá trị ban đầu bằng 0 hay bằng một hằng số nào đó như được mô tả trên đồ thị hình 12.7.

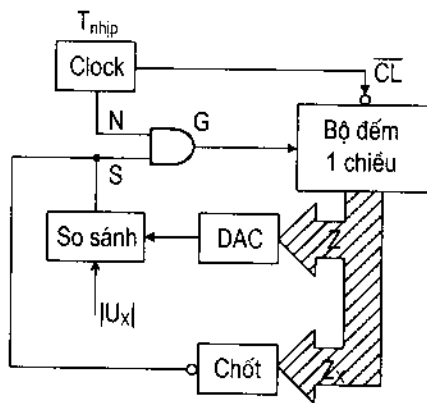


Hình 12.7. Đồ thị thời gian mô tả quá trình ADC đếm:

a) Bám đến U_x từ 0 dùng bộ đếm thuận (một hướng);
 b), c) Bám đến U_x từ hai phía dùng bộ đếm hai hướng có xác lập trước

Hình 12.8 đưa ra một cấu trúc khối thực hiện ADC đếm từ 0 đến U_x .

Ban đầu, xuất hiện xung $\overline{CL}$ ở mức thấp, đưa bộ đếm về $Z = 0$, lúc này, tại lối ra DAC có $U_z = 0$, kết quả là $S = 1$ vì $|U_x| > U_z$, S ở mức cao mở G cho phép xung nhịp (xung đếm N) có chu kỳ T_{nhp} được vào bộ đếm. Trạng thái bộ đếm tăng dần ($Z \uparrow$) làm U_z tăng theo cho tới khi $U_z \geq |U_x|$, lối ra so sánh lúc này về mức thấp ($S = 0$) không cho phép N qua G vào đếm nữa. Mức $S = 0$ là tích cực với mạch chốt nên trạng thái cuối cùng của bộ đếm là Z_x được giữ lại, đây là kết quả của quá trình biến đổi.



Hình 12.8. Cấu trúc khối ADC bám từ 0 đến U_x bằng các bậc thang đều hình 12.7a)

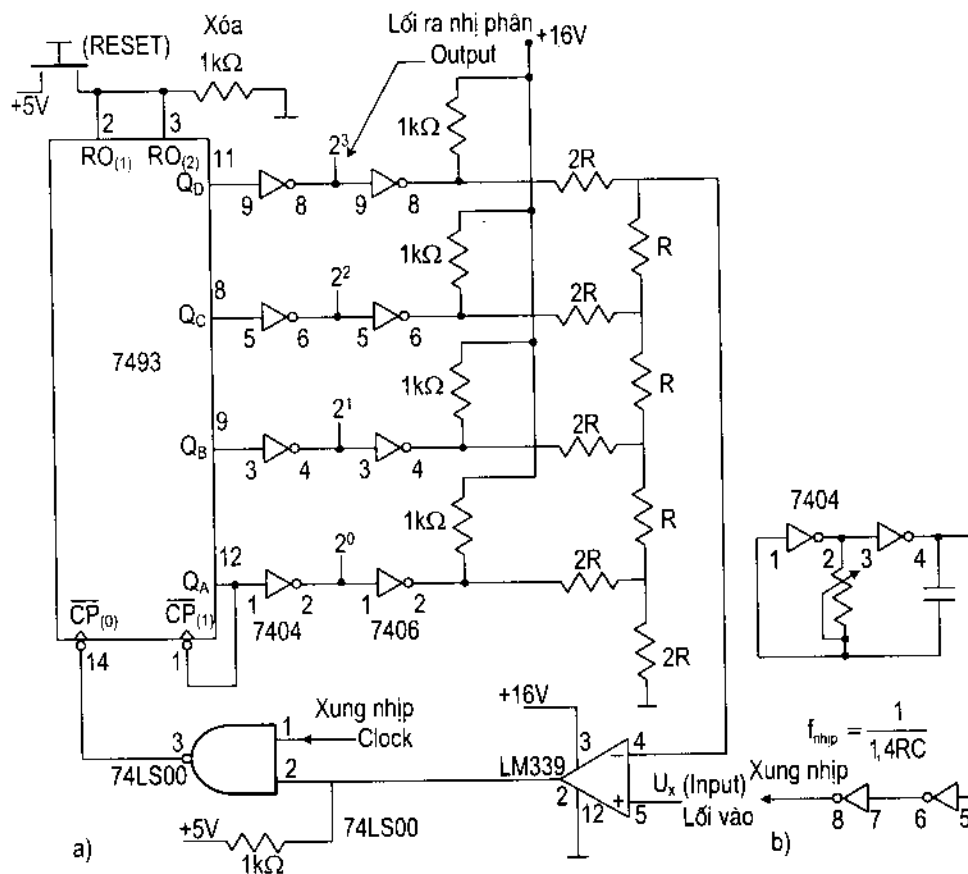
Theo đồ thị hình 12.7 a) có $k_x U_{LSB} = U_x$ (U_{LSB} là đơn vị lượng tử để làm tròn). Từ đó, $k_x = \frac{U_x}{U_{LSB}}$ (12.2)

Hay $(k_x)_{10} = (Z_x)_2$ chỉ thị tại bộ đếm (mã hóa thập phân sang nhị phân 8421 hay nhị phân BCD 8421).

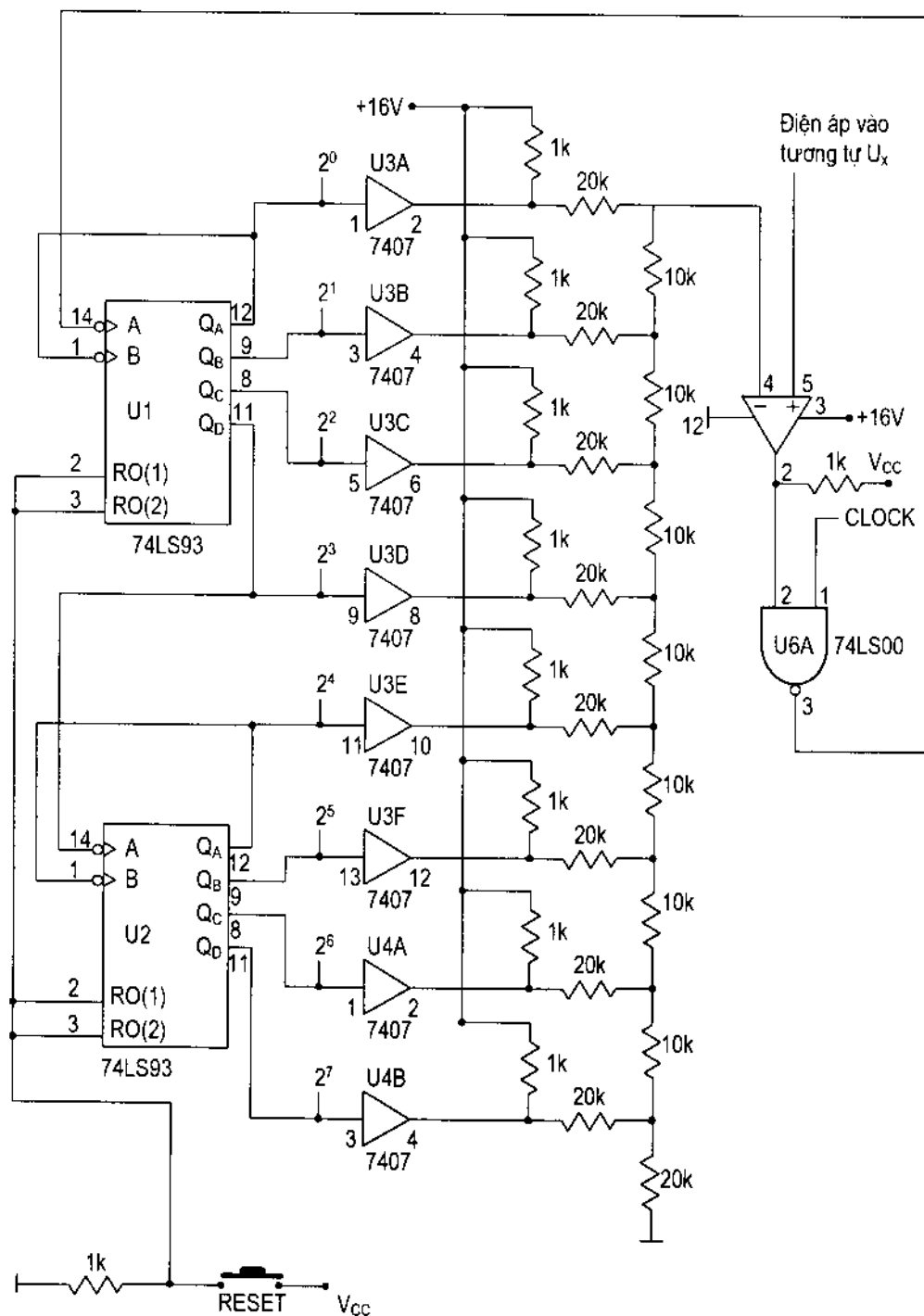
$$\text{Thời gian biến đổi } T_x = k_x \cdot T_{nhịp} \quad (12.3)$$

Để giảm bớt thời gian biến đổi T_x (có giá trị cực đại $T_{x_{\max}} = (2^n - 1)T_{nhịp}$) có thể dùng bộ đếm 2 hướng có khả năng đặt trước trạng thái đầu như mô tả trên đồ thị hình 12.7 b và c. Khi đó:

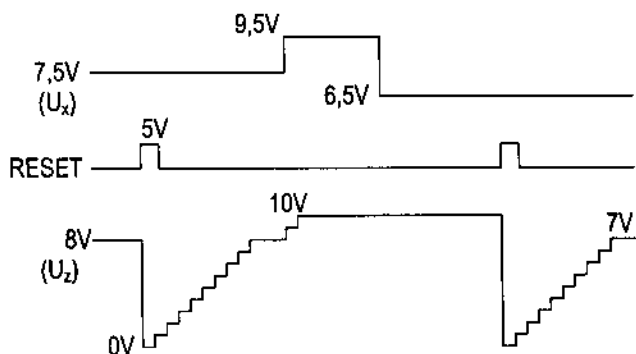
$$T_{x2} = (k_x - k_o)T_{nhịp} < T_{x1} = k_x \cdot T_{nhịp} \text{ hay } T_{x3} = (k_o - k_x) \cdot T_{nhịp} < T_{x1} \quad (12.4)$$



Hình 12.9. ADC 4 bit dùng bộ đếm một chiều tổ chức theo
a) cấu trúc khối; b) khối tạo nhịp



Hình 12.10 a) ADC 8 bit dùng phương pháp đếm
(mở rộng mạch ADC 4 bit hình 12.9)



Hình 12.10 b) Đồ thị điện áp tại các lối vào chân 5 (U_x) và chân 4 (U_z) của IC LM 339 hình 12.9 hay 12.10a)

Hình 12.9 là một mạch ADC dùng bộ đếm một chiều 7493 xây dựng theo cấu trúc khối hình 12.8 (trong mạch hình 12.9 chưa vẽ bộ chốt kết quả). Khối DAC sử dụng loại mạng điện trở $R - 2R$, bộ so sánh sử dụng IC LM 339. Các chân xóa R_0 (1) và R_0 (2) của bộ đếm 7493 tích cực ở mức cao qua một cổng NAND bên trong IC (hình 8.24) sẽ đưa tới các chân $\overline{CLR}$ của 4 JKFF: chế độ đếm nhị phân đầu vào nhịp C_{P1} của FF_B được nối tới lối ra Q_A của FF_A theo cách tổ chức của bộ đếm nối tiếp (hình 8.15). Dữ liệu kết quả ra lấy song song tại các đầu ra $Q_D Q_C Q_B Q_A$ của các JKFF có thể dùng LED hiển thị từng bit.

Để có dữ liệu ra là kết quả 8 bit $Z_x = Z_7 Z_6 Z_5 Z_4 Z_3 Z_2 Z_1 Z_0$, sử dụng mạch hình 12.10 biểu diễn một ADC 8 bit theo phương pháp đếm. Mạch hình 12.8 là sự mở rộng mạch hình 12.9 khi ghép liên tiếp hai bộ đếm 7493 (ở đây dùng 2 IC 74LS93 có tốc độ nhanh hơn). Giao tiếp giữa DAC mạng điện trở $R - 2R$ 8 bit với các cổng TTL dùng các cổng lặp (YES) loại colectơ để hở 7407 (sử dụng 1 $\frac{2}{6}$ IC 7407). Điện áp vào U_x tương tự như mạch 12.9, được đưa tới một đầu vào (chân số 5) của IC so sánh LM339. Tốc độ biến đổi của phương pháp đếm là chậm nhất so với phương pháp xấp xỉ từng bit hay phương pháp song song. Với kết quả ra có n bit thì thời gian biến đổi tối đa là:

$$T_{x_{\max}} = (2^n - 1)T_{\text{nhịp}} \quad (12.5)$$

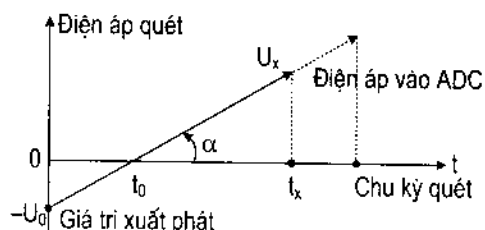
– Thời gian thực hiện biến đổi để đến kết quả cuối cùng tùy theo giá trị của U_x sẽ thay đổi từ 0 đến $(2^n - 1)T_{\text{nhịp}}$ ví dụ với mạch hình 12.9 có $T_{x_{\max}} = 15 T_{\text{nhịp}}$, và với hình 12.10 có $T_{x_{\max}} = 255 T_{\text{nhịp}}$.

12.5. ADC DÙNG PHƯƠNG PHÁP TÍCH PHÂN (phương pháp quét)

– Phương pháp quét sử dụng một điện áp biến thiên tuyến tính theo thời gian như thể hiện trên hình 12.11. Với tham số góc nghiêng α đã biết thì

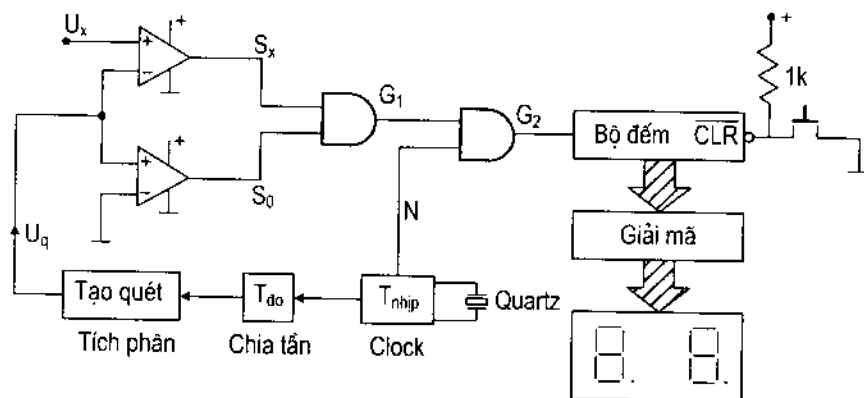
$$\operatorname{tg} \alpha = \frac{U_x}{t_x - t_0} \text{ hay } U_x = (\operatorname{tg} \alpha)(t_x - t_0) \quad (12.9)$$

Thực hiện xác định t_0 , t_x và qua đó đo được $(t_x - t_0)$ thì sẽ xác định được giá trị của đại lượng vào U_x . Vậy ADC theo phương pháp quét thực hiện quá trình biến đổi trung gian điện áp (U_x) thành khoảng thời gian $(t_x - t_0)$ và đo U_x gián tiếp qua việc đo $(t_x - t_0)$.



Hình 12.11. Đồ thị mô tả nội dung phương pháp quét một lần

Hình 12.12 là mạch điện cấu trúc khối của một ADC quét (tích phân 1 nhịp) và đồ thị thời gian minh họa hoạt động của các khối cho trên hình 12.13.



Hình 12.12. Cấu trúc khối ADC quét một nhịp

Khối tích phân tạo ra điện áp quét có dạng

$$U_q(t) = -U_0 + \frac{U_{\text{ref}}}{RC} t \quad (12.10)$$

Ở đây, U_{ref} là điện áp tham chiếu của mạch tích phân và $RC = \tau_{\text{tích phân}}$.

Bộ so sánh S_0 thực hiện so sánh $U_q(t)$ với 0V để tìm t_0 :

Khi $t < t_0$ thì $S_0 = 0$, khi $t \geq t_0$ thì $S_0 = 1$.

tại $t = t_0$ có

$$U_q(t_0) = -U_0 + \frac{U_{ref}}{RC} t_0 = 0 \quad (12.11)$$

Bộ so sánh S_x thực hiện so sánh $U_q(t)$ với U_x để tìm t_x :

Khi $t < t_x$ thì $S_x = 1$ (dạng so sánh đảo).

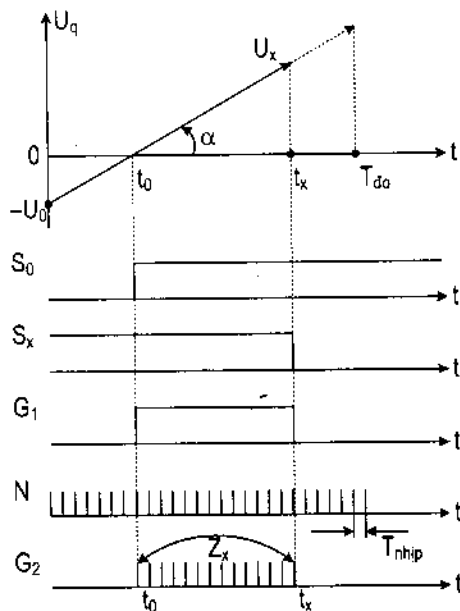
Khi $t \geq t_x$ thì $S_x = 0$

lúc $t = t_x$ ta có:

$$U_q(t_x) = -U_0 + \frac{U_{ref}}{RC} t_x = U_x \quad (12.12)$$

Nếu thực hiện trừ hệ thức (12.12) cho (12.11) có:

$$\frac{U_{ref}}{RC} (t_x - t_0) = U_x \quad (12.13)$$



Hình 12.13. Đồ thị điện thế minh họa hoạt động của mạch hình 12.12

Từ hình 12.13, với chu kỳ bộ phát nhịp là $T_{nhịp}$

với $G_1 = S_0 \cdot S_x$ và $G_2 = N \cdot G_1$, ta có:

$$(t_x - t_0) = Z_x \cdot T_{nhịp} \quad (12.14)$$

Ở đây, Z_x là số xung bộ đếm, đếm được trong khoảng $t_0 \rightarrow t_x$.

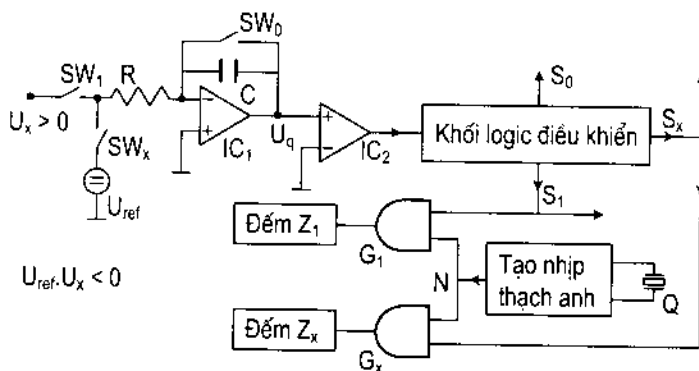
Thay (12.14) vào (12.13) có liên hệ giữa U_x và Z_x là hệ thức cần tìm:

$$\left(\frac{U_{ref}}{\tau_{tích\ phân}} T_{nhịp} \right) Z_x = U_x \quad (12.15)$$

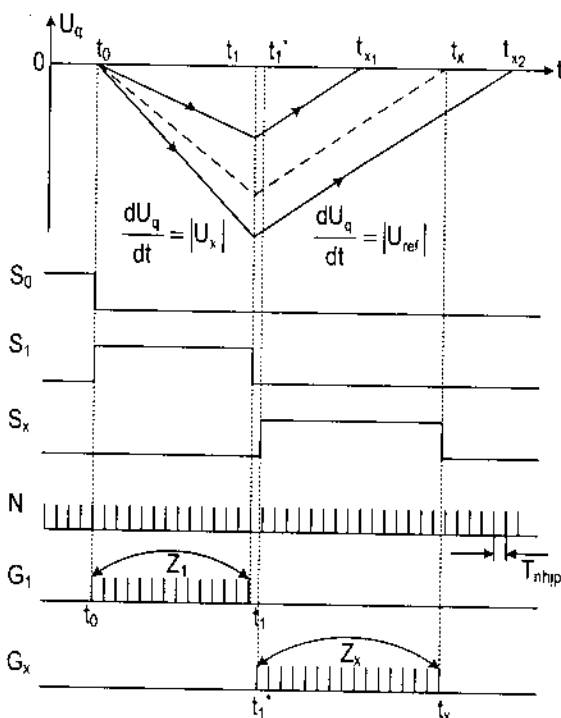
Độ chính xác của phép biến đổi này phụ thuộc nhiều vào các tham số $T_{nhịp}$, U_{ref} và đặc biệt là $\tau_{tích\ phân} = RC$ thường sai số gặp phải không nhỏ do ảnh hưởng của $\tau_{tích\ phân}$ ($T_{nhịp}$ được ổn định nhờ dùng tinh thể Quartz, U_{ref} có độ chính xác cao).

Hình 12.14 là cấu trúc khối của ADC quét loại 2 nhịp có độ chính xác cao hơn so với mạch hình 12.12. Đồ thị điện thế minh họa hoạt động được thể hiện trên hình 12.15.

Mạch hình 2.14 có hai nhịp hoạt động chính và 2 nhịp phụ mang tính chất chuẩn bị cho nhịp chính hoạt động. Các xung điều khiển S_0 , S_1 , S_x do khối logic điều khiển tạo ra điều hành các nhịp làm việc của ADC bằng cách điều khiển khóa điện tử cùng tên SW_0 , SW_1 và



Hình 12.14. Cấu trúc ADC quét 2 nhịp:
 SW_0 , SW_1 , SW_2 là các chuyển mạch (khóa) điện tử



Hình 12.15. Đồ thị điện áp minh họa hoạt động của các khối mạch trên hình 2.14

SW_x . Nhịp phụ thứ nhất xuất hiện xung S_0 nối mạch SW_0 (SW_0^+) trong khoảng thời gian từ 0 đến t_0 , SW_1 và SW_x lúc này ngắt mạch (SW_1^- , SW_x^-). Như vậy, C được nối ngắn mạch để xả hết điện tích dư trên nó và $U_q(t_0) = 0$.

Nhịp chính 1: S_{w0}^- , S_x^- , S_{w1}^+ (do xung S_1 điều khiển nối mạch) đưa đại lượng vào U_x tới mạch tích phân đảo dùng IC₁ trong thời gian t_0 đến t_1 cố định.

Giả thiết U_x có cực tính dương nên $U_q(t)$ là một đường quét hướng đi xuống, có tốc độ quét tỷ lệ với độ lớn của $|U_x|$ nghĩa là khi $|U_{x2}| > |U_{x1}|$ (trị mô đun) thì đường quét ứng với U_{x2} sẽ dốc hơn. Trong khoảng thời gian $t_0 \rightarrow t_1$ của nhịp quét lần 1, xung S_1 đồng thời mở G_1 cho phép xung N vào bộ đếm 1, do đó nếu bộ đếm đếm được Z_1 xung thì:

$$(t_1 - t_0) = Z_1 \cdot T_{nhịp} \quad (12.17)$$

Nhịp phụ thứ hai thực hiện sau t_1 và trong một khoảng thời gian đủ ngắn: $t_1^* - t_1 < T_{nhịp}$. Khi đó yêu cầu cùng ngắt mạch SW_0^- , SW_1^- , SW_x^- , lúc này, cần xác định dấu của U_x và chọn U_{ref} có dấu ngược lại ($U_{ref} \cdot U_x < 0$).

Nhịp quét thứ hai bắt đầu từ lúc t_1^* , yêu cầu SW_0^- , SW_1^- ; xung S_x xuất hiện từ khối logic điều khiển tác động nối mạch SW_x^+ , U_{ref} được đưa tới tích phân. Trong nhịp làm việc lần này $U_q(t)$ có các tính chất:

– Chiều quét thay đổi ngược lại với nhịp quét lần 1 (quét lên).

– Tốc độ quét $\left(\frac{dU_q}{dt} = U_{ref} \right)$ không đổi theo thời gian. Các đường

quét do đó chạy song song nhau.

– Thời gian quét phụ thuộc thời điểm t_x lúc đó $U_q(t_x) = 0$. IC₂ là một bộ so sánh U_q với 0V có nhiệm vụ xác định t_x và do đó quyết định độ rộng xung S_x .

Trong khoảng thời gian quét lần 2, xung S_x mở cổng AND G_x cho phép xung đếm N vào bộ đếm 2, đếm được Z_x xung, do đó:

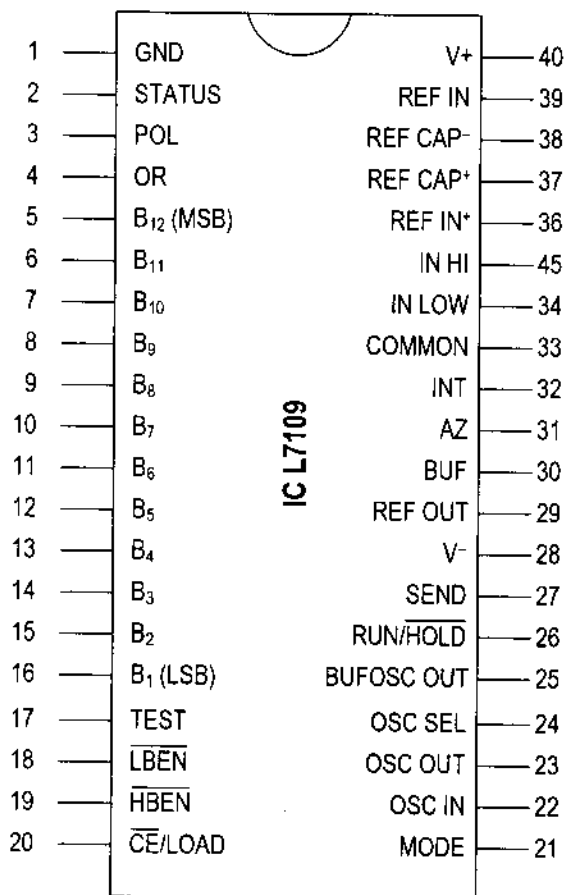
$$t_x - t_1^* \approx t_x - t_1 = Z_x \cdot T_{nhịp} \quad (12.18)$$

Từ các hệ thức đã có trên, dễ dàng suy ra:

$$Z_x = \frac{Z_1}{U_{ref}} \cdot U_x \quad (12.19)$$

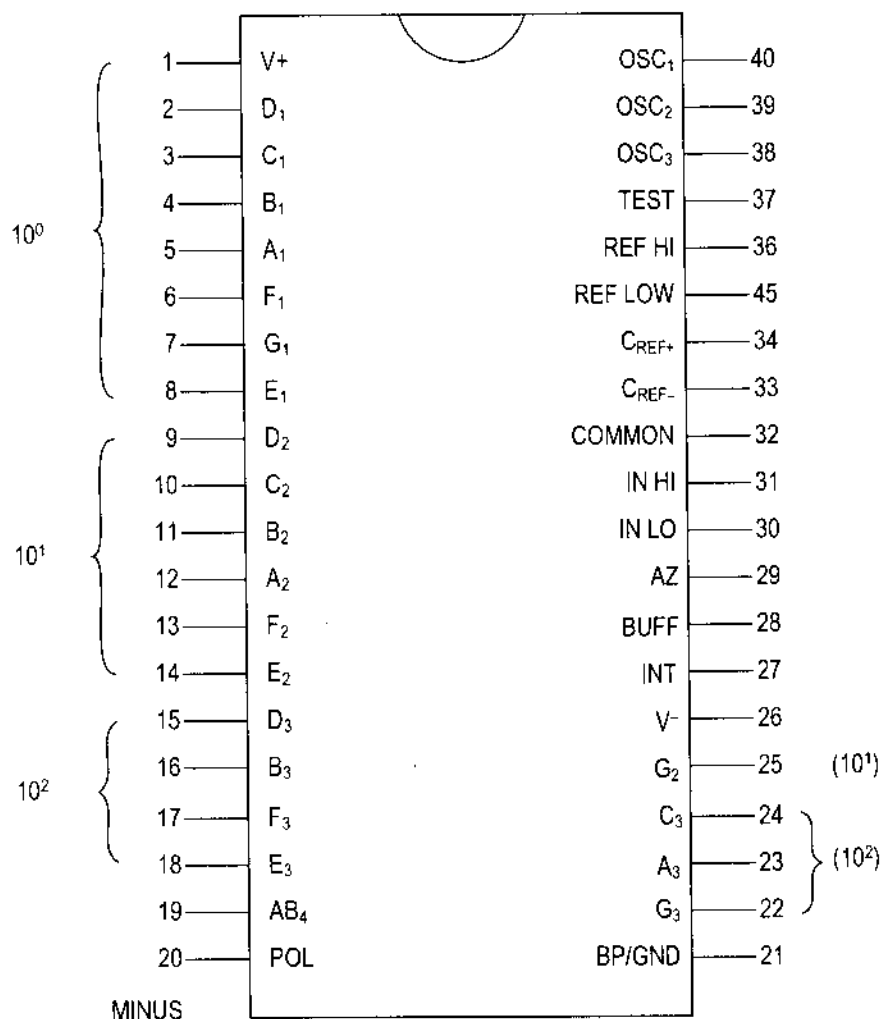
Kết quả thu được ở hệ thức 12.19 không phụ thuộc $\tau_{\text{tích phân}} = RC$, không phụ thuộc vào $T_{\text{nhịp}}$ do đó ADC loại hai nhịp có độ chính xác cao hơn loại trước.

IC ADC IC L7109 làm việc theo nguyên lý 2 nhịp tích phân như vừa khảo sát trên. Cấu trúc, sơ đồ logic cho trên hình 12.16, chức năng tóm tắt các chân cho trên bảng 12.2. Dữ liệu ra $B_{12} \rightarrow B_1$ dưới dạng mã nhị phân tự nhiên (8421).



Hình 12.16. Sơ đồ logic bố trí chân của IC L7109

Vì điện tử IC L7107 làm việc theo nguyên lý tích phân hai nhịp có sơ đồ bố trí chân cho trên hình 12.17 có lối ra ở dạng mã BCD 8421 biểu diễn $3\frac{1}{2}$ digit với số hiển thị tối đa là 1999. Đầu ra có bộ dồn kênh và giải mã, do vậy, có thể nối trực tiếp với LED 7 vạch.



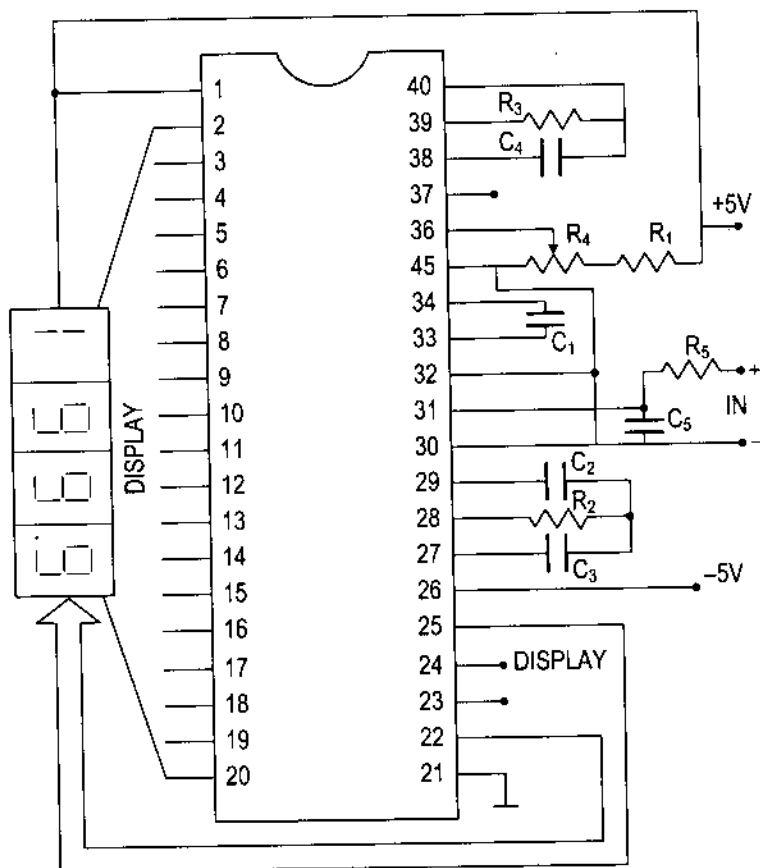
Hình 12.17. Bố trí chân và ký hiệu logic của IC L7107

(chức năng các chân điều khiển xem bảng 12.3)

Mạch điện hình 12.18 đưa ra cách đấu nối các chân của IC L7107 cho hoạt động ADC với tín hiệu vào toàn thang là 200mV. Trong trường hợp cần thang 2,000V có thể lựa chọn các linh kiện theo bảng 12.4 hoặc dùng một bộ chia 1/10 tại lối vào analog (chân 30 – 31). Cùng với nhóm IC DAC, các IC ADC được dùng phổ biến trong việc ghép nối với bộ vi xử lý hay với máy tính qua các cổng giao diện số, trong các ứng dụng xử lý số tín hiệu nói chung.

Bảng 12.3. Chức năng các chân của IC L7109

Số chân	Ký hiệu	Chức năng
1	GND	Chân nối đất.
2	STATUS	Lỗi ra, ở mức cao trong cả quá trình biến đổi cho đến khi dữ liệu ra được chốt lại xuống thấp khi tín hiệu được chuyển đổi xong.
3	POL	Bit thông báo cực tính của tín hiệu vào analog. Mức 1 khi tín hiệu vào dương
4	OR	Bit thông báo tràn. Mức 1 nếu bị tràn
5 ↓ ...	B12 ...	Bit MSB ... Tất cả đều có lỗi ra 3 trạng thái
16	B1	
	Bit LSB	
17	TEST	Bình thường ở mức cao. Nổi lên mức cao nếu không dùng. Nếu đặt vào mức thấp thì tất cả các bit lỗi ra lên cao – dùng cho việc kiểm tra.
18	LBEN	Chân cho phép xuất byte thấp. Cùng với MODE = 0 (chân 21) ở mức thấp và chân CE/LOAD (chân 20) ở mức thấp sẽ cho phép xuất bit thấp từ B ₁ → B ₈ .
19	HBEN	Chân cho phép xuất byte cao. Cùng với MODE = 0 và CE/LOAD = 0 sẽ cho phép xuất các bit cao B ₉ → B ₁₂ và bit POL/OR.
20	CE/LOAD	Chân cho phép. Kết hợp với MODE = 0 điều khiển cho phép lỗi ra. Khi CE/LOAD = 1 các chân B ₁ → B ₁₂ , POL/OR bị cấm (cấm 14 bit ra).
21	MODE	Khi MODE = 0, các chân CE/LOAD, LBEN, HBEN điều khiển trực tiếp các bit lỗi ra. Khi được cấp xung thì chuẩn bị hoạt động theo kiểu "handshake". Khi MODE = 1, các chân cho phép CE/LOAD, LBEN, HBEN xem như các lỗi ra và IC hoạt động theo kiểu handshake (thiết lập liên hệ).
22	OSC IN	Lỗi vào của dao động.
23	OSC OUT	Lỗi ra của dao động.
24	OSC SEL	Chọn tần số dao động. Mức cao thì tần số và pha tại OSC IN, OSC OUT bằng 1/58 tần số tại BUFF OSC OUT.
25	BUFF OSC OUT	Lỗi ra dao động đệm.
26	RUN/HOLD	Lỗi vào ở mức cao: Biến đổi được thực hiện trong 8192 xung đồng hồ. Lỗi vào mức thấp: Quá trình biến đổi kết thúc.
27	SEND	Lỗi vào nối +5 nếu không dùng.
28	V-	Nguồn -5V
29	REF OUT	Điện áp ngõ ra chuẩn 2,8V
30	BUFFER	Lỗi ra khuếch đại đệm.
31	AUTO ZERO	Tự động điều chỉnh mức 0.
32	INTERER ATOR	Lỗi ra kết hợp.
33	COMMON	Điểm chung.
34	INPUT LOW/	Đầu vào tương tự
35	INPUT HI	
36	REF IN+	Điện áp chuẩn dương (lỗi vào)
37	REF CAP+	Áp dương chuẩn trên tụ
38	REF/CAD-	Áp âm chuẩn trên tụ.
39	REF IN-	Điện áp chuẩn âm (lỗi vào).
40	V+	Nguồn cấp +.



Hình 12.18. Các đấu nối IC L7107 điển hình ở chế độ hiển thị 200mV toàn thang

Bảng 12.4. Lựa chọn giá trị linh kiện ngoài cho 2 thang đo của IC L7107

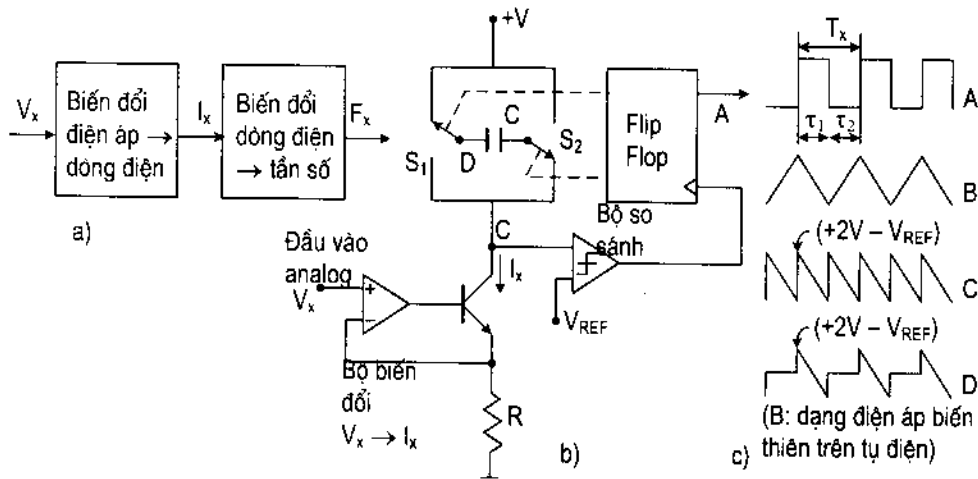
Linh kiện	200,0mV Toàn thang	2,000V Toàn thang
C ₂ max	0,47μF	0,047μF (*)
R ₁	24kΩ	1,5kΩ
R ₂	47kΩ	470kΩ
C ₂ min	0,1μF	0,022μF (*)
R ₁	220kΩ	150kΩ
R ₂	180kΩ	1,8MΩ
R ₄	10kΩ	100kΩ

(*) tự động dò 0

12.6. ADC DÙNG PHƯƠNG PHÁP BIẾN ĐỔI ĐIỆN ÁP – TẦN SỐ (VFC – Voltage to Frequency Converter)

Đại lượng vào analog trong phương pháp VFC trước tiên được biến đổi thành một điện áp ra của VFC có tần số tỷ lệ với giá trị vào, sau đó thực hiện việc đo tần số tại lối ra của VFC sẽ có kết quả cuối cùng.

12.6.1. VFC dùng dao động đa hài điều khiển bằng dòng điện có cấu trúc khối cho trên hình 12.19a), mạch nguyên lý và giản đồ điện áp minh họa cho trên hình 12.19b) và c).



Hình 12.19a) Cấu trúc khối; b) Mạch nguyên lý; c) Đồ thị thời gian của VFC loại dùng đa hài điều khiển bằng dòng điện

Điện áp tương tự V_x qua bộ biến đổi điện áp thành dòng điện $I_x = \frac{V_x}{R}$ điều khiển quá trình nạp cho tụ C làm điện áp tại điểm C thay đổi, so với V_{REF} tại bộ so sánh để điều khiển lật FF (theo sườn đi lên).

Chu kỳ xung ra tại điểm A được tính bởi: $T_x = \frac{2RC \cdot \Delta V}{V_x}$ hay tần số xung ra

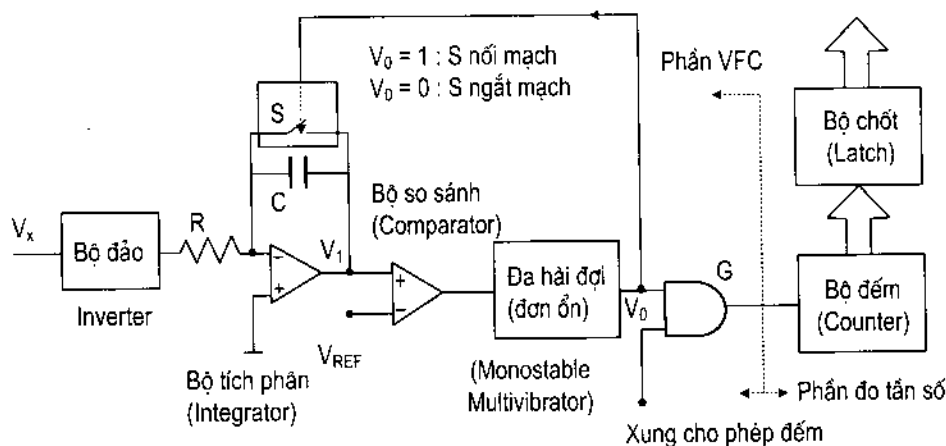
$$f_x = \frac{V_x}{2RC \cdot \Delta V} \text{ với } \Delta V = +V - V_{REF} \quad (12.20)$$

Khi FF lật trạng thái các chuyển mạch điện tử S_1 và S_2 cũng đảo vị trí làm đảo các giá trị vào của FF. Hệ số lấp đầy xung luôn là 50% ($\tau_1 = \tau_2$).

12.6.2. VFC dùng phương pháp cân bằng điện tích

Khối chính (hình 12.20) là một mạch đa hài đợi. Khi V_0 ở lối ra đa hài ở mức thấp (chế độ chờ) thì S hở mạch – V_x được đưa tới tích phân làm V_1 tăng lên tuyến tính cho tới khi $V_1 = V_{REF}$, đa hài lúc này lật trạng thái: V_0 lên mức cao và S nối mạch, tụ C được giải phóng điện tích cho tới khi hết xung V_0 (do thông số riêng của đa hài quyết định) S quay về ngắt mạch thực hiện một chu kỳ mới. Chú ý $\frac{dV_1}{dF} = |V_x|$ do đó tần số lặp lại của V_0 tỷ lệ với V_x xác định theo hệ thức (12.21).

$$f_x = \frac{V_x}{RC.V_{REF}} \quad (12.21)$$

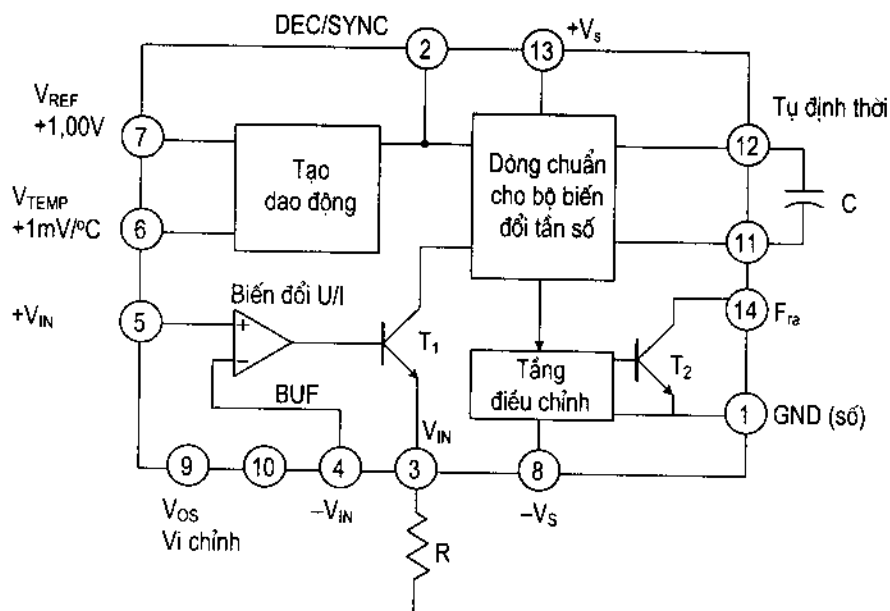


Hình 12.20. Mạch nguyên lý VFC dùng phương pháp cân bằng điện tích và khối đo tần số

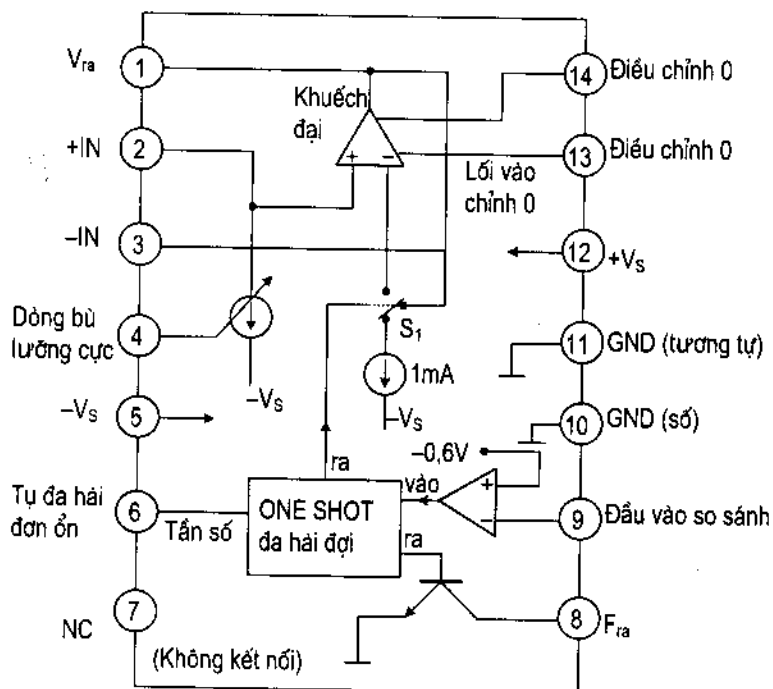
12.6.3. Cấu trúc khối và bố trí chân của IC AD537

Cấu trúc khối và bố trí chân của IC AD537 theo phương pháp điều khiển đa hài (FF) bằng dòng điện cho trên hình 12.21a và IC AD650 theo phương pháp cân bằng điện tích (hình 12.21b). Hệ số biến đổi của VFC: $f_x = kV_x$ hay $k = \frac{f_x}{V_x}$ được tính theo đơn vị Hz/V

được gọi là độ nhạy của VFC (vài trăm Hz đến vài kHz/V).



Hình 12.21a. Cấu trúc khối và bố trí chân của IC VFC AD537
(dùng phương pháp điều khiển FF bằng dòng điện)



Hình 12.21b. Cấu trúc khối và bố trí chân của IC VFC AD650
(phương pháp cân bằng điện tích)

12.7. CÂU HỎI VÀ BÀI TẬP

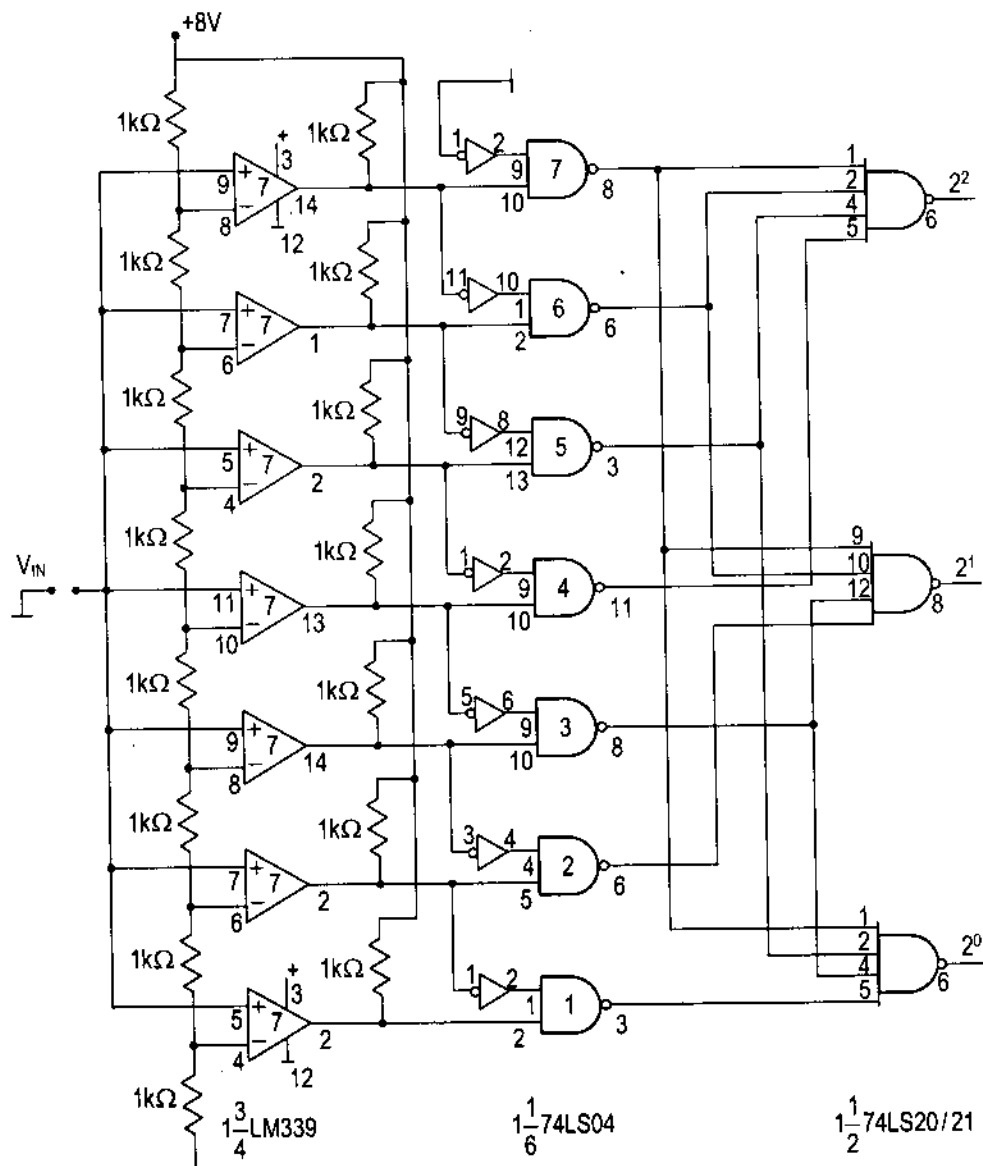
1. Hãy nêu nhiệm vụ của ADC và các bước cơ bản phải thực hiện trong quá trình ADC. Phân biệt sai số khe và sai số lượng tử? Độ phân giải của một ADC là gì? Giảm nhỏ các sai số khe và sai số lượng tử sẽ làm cho độ phân giải của ADC tăng lên có đúng không?
2. Kết luận rằng ADC theo phương pháp song song có thời gian biến đổi nhỏ nhất có đúng không? Vì sao? Hãy vẽ sơ đồ cấu trúc khối và giải thích hoạt động một ADC song song có 4 bit kết quả tại lối ra.
3. Nêu các đặc điểm của ADC loại 2 nhịp và điểm khác biệt so với loại ADC song song 1 nhịp. Hãy vẽ sơ đồ logic bố trí các chân đầu vào điều khiển, đầu vào dữ liệu, các đầu ra dữ liệu và các chân khác của IC ADC 2 nhịp MAX 153 đã cho trên hình 12.2 và bảng 12.2.

Giải thích chức năng logic của nhóm các đầu vào $\overline{CS}$, $\overline{RD}$, $\overline{WR}$ / RDY, INT và MODE.

4. Giải thích nguyên lý vận hành của ADC loại xấp xỉ từng bit. Thanh ghi dịch SAR là gì? Vẽ lại cấu trúc khối của ADC đã cho trên hình 12.5 và giải thích nhiệm vụ các khối vừa vẽ.
5. ADC loại xấp xỉ từng trạng thái (loại đếm) làm việc theo nguyên lý nào? Hãy vẽ đồ thị điện áp vào khối so sánh hình 12.8 để minh họa nguyên lý hoạt động và giải thích đồ thị này. Các hệ thức tính kết quả Z_x và thời gian biến đổi T_x rút ra từ đồ thị này?
6. Hãy giải thích nguyên lý hoạt động của ADC 4 bit trên hình 12.9. Tại sao lại sử dụng IC 7406 mà không dùng IC 7404 kích thích DAC? Hãy nêu nhiệm vụ của IC LM339 và vẽ đồ thị quan hệ điện áp ra (chân 2) với điện áp vào U_x (chân 4) sự có mặt U_x (chân 5) được coi như điện áp ngưỡng.

7. Giải thích hoạt động của mạch hình 12.10a. Tính giá trị các điện áp phân bố tại các nút của mạng $R - 2R$. Giá trị U_{LSB} của mạch là bao nhiêu ?
8. a) Tác động điện áp một chiều $U_{x1} = 9,375V$ và sau đó $U_{x2} = 4,5V$ tới đầu vào mạch ADC hình 12.10a). Biết xung nhịp có $f_{nhịp} = 1kHz$.
 Hãy xác định các bit kết quả tại lối ra trong hai trường hợp đã cho.
 b) Điều gì sẽ xảy ra trên bộ đếm 4 bit cao khi chân số 1 (của U_2) bị hở mạch?
 c) Tính giá trị biên độ U_{xmax} cho phép tác động ở lối vào analog của mạch hình 12.10a).
9. a) Khi muốn có dữ liệu ra ADC 8 bit hình 12.10a) ở dạng mã BCD8421 để đưa tới chỉ thị 2 decac 10^1 cho nhóm bit cao và 10^0 cho nhóm bit thấp thì mạch hình 12.10a) cần có gì thay đổi?
 b) Từ kết quả câu a) nếu mở rộng cho ADC có dữ liệu 12 bit ra (3 decac). Hãy vẽ mô hình cấu trúc mạch thực hiện.
10. Hãy giải thích nguyên lý hoạt động của ADC quét 1 nhịp qua đồ thị điện áp thời gian mô tả hoạt động và cấu trúc khối của ADC.
 Tìm hệ thức quan hệ giữa đại lượng ra số Z_x và đại lượng vào analog U_x .
11. Nguyên lý hoạt động của ADC loại 2 nhịp có gì khác với loại 1 nhịp? Hãy vẽ đồ thị điện áp giải thích hoạt động và mô tả cấu trúc cơ bản của ADC loại này. Tại sao nói ADC loại 2 nhịp có độ chính xác cao hơn loại 1 nhịp?
12. Cho mạch ADC hình 12.22.
 a) Hãy giải thích nguyên lý hoạt động của mạch đã cho sau khi thiết lập mô hình khối thu gọn lại của mạch.
 b) Tính giá trị toàn thang của điện áp vào và giá trị của đơn vị lượng tử U_{LSB} của mạch. Làm thế nào để có thêm lối ra 2^3 tạo thành ADC 4 bit?

c) Cần dùng mấy IC LM339 để thực hiện mạch ADC 4 bit vừa vẽ ở câu b).

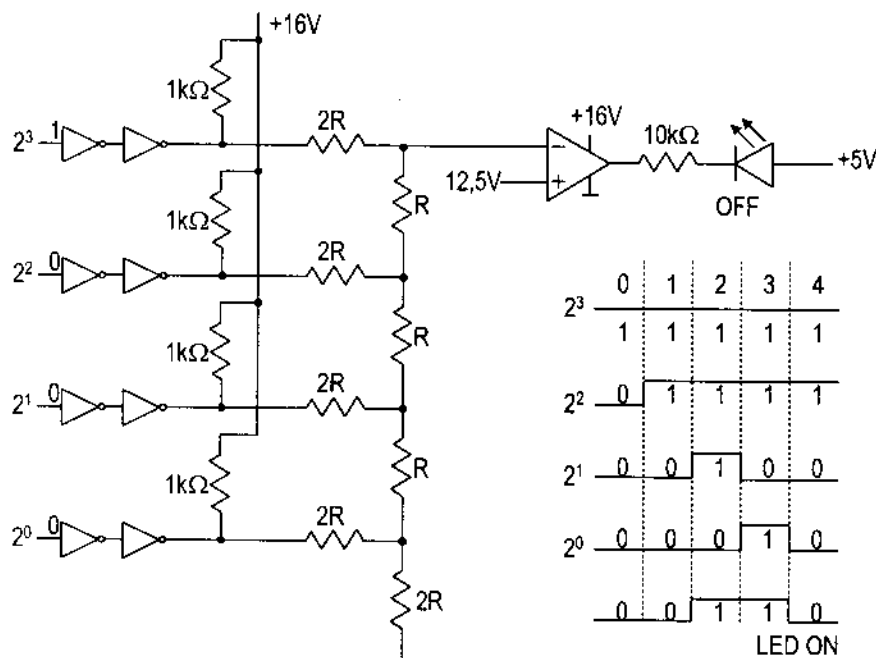


Hình 12.22. Cho bài tập 12

13. Cho mạch hình 12.23:

a) Hãy chỉ ra đầu vào và các đầu ra của mạch đã cho và nêu nguyên lý làm việc của mạch.

- b) Nhiệm vụ của LED trong mạch là gì? Tính giá trị U_{LSB} của mạch đã cho và điện áp ngưỡng trên các nút của ma trận mạng $R - 2R$.
- c) Khi cần mở rộng cho 12 bit có thể ghép liên tiếp 3 khối mạch đã cho được không? Vẽ mạch thực hiện và tính giá trị U_{LSB} trong trường hợp này.

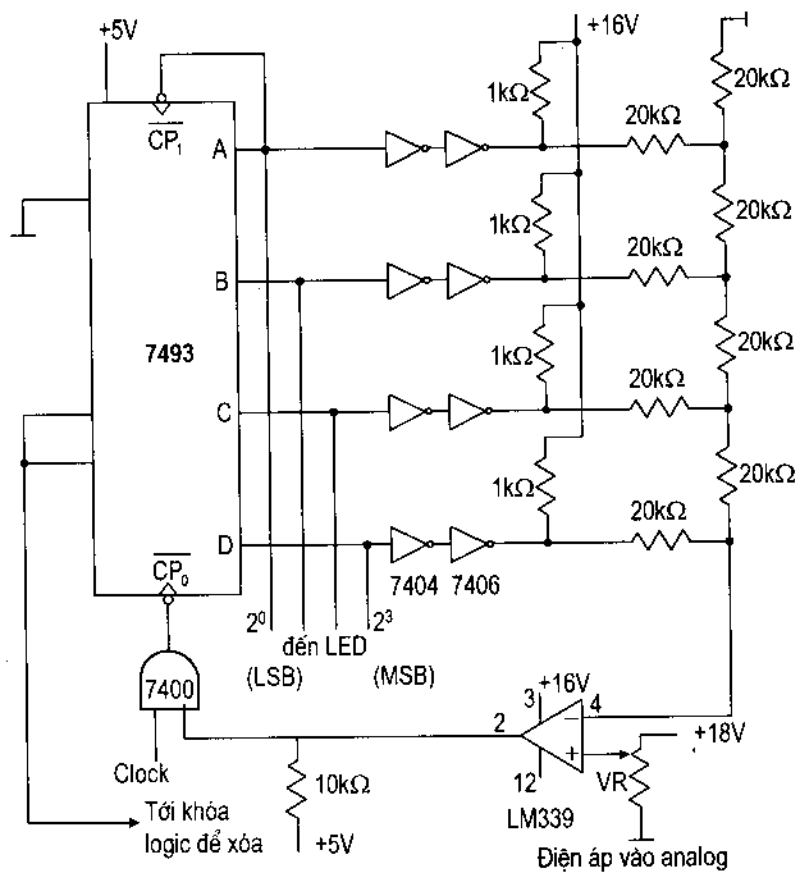


Hình 12.23. Cho bài tập 13

14. Cho mạch điện hình 12.24.

- a) Hãy giải thích hoạt động và nêu nhiệm vụ của các linh kiện sử dụng trong mạch. Điền chỉ số các chân của các IC còn lại 7400, 7404, 7406 và 7493.
- b) Hãy vẽ bổ sung 2 mạch điều khiển hoạt động. Mạch RESET (xóa) cho 7493 và mạch tạo xung đồng hồ Clock có tần số 1Hz. Nếu ngắt chân số 2 của LM339 (hở mạch ra) thì có hiện tượng gì tại lối ra nhị phân?
- c) Hãy chỉ ra trạng thái của các lối ra ABCD khi tại lối vào điều chỉnh VR để có $U_x = 5,5V$.

Hãy tăng giá trị điện áp vào đến $U_{x1} = 10,5V$. Xác định các giá trị bit tại lối ra sau đó chọn $U_{x2} = 6,5V$, xác định lại kết quả.



Hình 12.24. Cho bài tập 14

- 15 a) Muốn mở rộng mạch đã cho trên hình 12.24 với dữ liệu ra MSB là 2^7 và LSB là 2^0 thì phải làm thế nào? Hãy vẽ mạch thực hiện dạng mở rộng này.
- b) Hãy thay đổi kết cấu của mạch đã cho để dữ liệu ra biểu diễn ở mã BCD 8421 (chỉ có các trạng thái 0000 đến 1001 cho một decac). Chú ý đến IC 7493 và V_{REF} cho ma trận DAC. Mở rộng kết quả này cho mạch biểu diễn được kết quả lối ra 8 bit thể hiện hai decac.

BÀI THỰC HÀNH ĐIỂN HÌNH

THỰC HÀNH BÀI 12

Lắp ráp và cân chỉnh mạch ADC

Mục đích củng cố bài học lý thuyết về các phương pháp biến đổi tương tự – số cơ bản nhất. Nâng cao kỹ năng thực hành lắp ráp và cân chỉnh các mạch số phức tạp qua đó rèn luyện tác phong cẩn thận, kiên trì, khoa học.

A. PHẦN CHUẨN BỊ

1. Tài liệu: Phân lý thuyết bài 12 về các mạch ADC.

Các mạch điện hình 12.5a (cho ADC xấp xỉ từng bit) hình 12.9 (ADC đếm) và hình 12.22 (ADC song song).

Cách xác định mức lượng tử trong các mạch thực hành, sơ đồ chân các IC sử dụng.

2. Dụng cụ: Mỏ hàn, kìm cắt và kìm uốn dây, uốn chân linh kiện, panh gấp.

Đồng hồ đo điện vạn năng (VOM); Một bộ nguồn điện một chiều thay đổi được mức ổn áp lối ra +5V đến +18V.

3. Vật liệu: Bảng mạch cắm chân các linh kiện và cắm chân IC loại 14 và 16 chân; Bảng mạch hàn dây, các dây một sợi có vỏ nhựa $\varnothing = 0,2\text{mm}$, thiếc hàn, nhựa thông.

4. Linh kiện: Các chủng loại linh kiện có mặt trong mạch thực hành.

– Với mạch hình 12.5a) gồm: 3 IC 7476, 1 IC 7400; 1 IC 7408; 1 IC 7402; 1 IC 7406; 1 IC LM339; 1 khóa chuyển mạch $4 \times 330 \Omega$; $6 \times 1 \text{ k}\Omega$; $5 \times 20 \text{ k}\Omega$; $4 \times 10 \text{ k}\Omega$; 4 LED.

– Nếu lắp mạch tạo xung clock thì cần thêm 1 IC 7404 hay 7400; $VR = 10 \rightarrow 100 \text{ k}\Omega$; C $1\mu\text{F}$ (hình 8.15 hay hình 8.38).

– Với mạch hình 12.9: 1 IC 7404; 1 IC 7406; 1 IC 7400; 1 IC LM339; 1 IC 7439; $6 \times 1\text{ k}\Omega$; $3 \times 10\text{ k}\Omega$; $5 \times 20\text{ k}\Omega$; 1 công tắc RESET.

Dùng LED và $4 \times 330\ \Omega$ để hiển thị các mức ra (không vẽ trong hình 12.9).

– Với mạch hình 12.22: 2 IC 7420 hoặc 7421; 2 IC 7400; 2 IC 7404; 2 IC LM339; $8 \times 1\text{ k}\Omega$ sai số 1%; $7 \times 1\text{ k}\Omega$ sai số 5%.

3 LED $3 \times 330\ \Omega$ thử mức logic tại các lối ra.

Chú ý các IC trên mạch thực hành chưa vẽ chân nguồn +5V và đất 0V.

IC LM 339 cấp nguồn đơn +16V.

5. Địa điểm thực hành: phòng học lý thuyết và thực hành điện tử.

B. TRÌNH TỰ LẮP RÁP VÀ CÂN CHỈNH MẠCH

Lần lượt thực hành với mạch hình 12.5a, sau đó với mạch 12.9 và cuối cùng với mạch hình 12.22.

Với mỗi mạch thực hành cần thực hiện các bước sau:

Bước 1: Bố trí hợp lý và thực hiện cắm toàn bộ linh kiện trên bảng mạch theo sự hướng dẫn của giáo viên hướng dẫn thực hành theo đúng các yêu cầu kỹ thuật.

Bước 2: Đo khoảng cách chân các linh kiện và thực hiện tuốt dây, uốn và cắm dây nối mạch đúng yêu cầu kỹ thuật.

Bước 3: Kiểm tra chi tiết toàn bộ mạch đã nối theo hình vẽ mạch thực hành và các phần cắm bổ sung: mạch tạo xung nhịp và mạch thử logic dùng LED.

Kiểm tra đường cấp nguồn một chiều nhờ đồng hồ VOM và báo cáo kết quả kiểm tra cho cán bộ hướng dẫn.

Bước 4: Cấp nguồn +5V và +16V cho mạch (sau khi được giáo viên hướng dẫn cho phép).

Chạy thử và kiểm tra hoạt động theo mạch điện nguyên lý.

Điều chỉnh tần số xung clock (VR $100\text{ k}\Omega$) cho thay đổi tốc độ biến đổi AD nếu thấy cần thiết.

Ghi lại các kết quả giá trị điện áp analog đầu vào và các dữ liệu số ở lối ra tương ứng cho mỗi trị đầu vào. Giá trị tần số xung clock và giá trị đơn vị làm tròn.

C. BÁO CÁO CÁC KẾT QUẢ THỰC HÀNH

Tập hợp các số liệu thực hành trong 3 bảng với 3 nội dung thực hành (trên 3 mạch đã cho) theo mẫu.

(*) Chọn 16 giá trị khác nhau của U_v có thể lẻ tới $0,1 U_{LSB}$.

(**) Chọn 8 giá trị khác nhau của U_v có thể lẻ tới $0,1 U_{LSB}$.

ADC xấp xỉ từng bit (hình 12.5a)

U_v (V)	Z_r			
	2^3	2^2	2^1	2^0
(*)				

$U_{LSB} =$
 $f_{nhp} =$
 Thời gian biến đổi
 $T_x =$

ADC đếm (hình 12.9)

U_v (V)	Z_r			
	2^3	2^2	2^1	2^0
(*)				

$U_{LSB} =$
 $f_{nhp} =$
 $T_x =$

ADC song song (hình 12.22)

U_v (V)	Z_r		
	2^2	2^1	2^0
(**)			

$U_{LSB} =$
 $f_{nhp} =$
 $T_x =$

2. Nhận xét và đánh giá các kết quả đã thu được.

So sánh thời gian biến đổi của mỗi loại.

Đánh giá sai số.

TÀI LIỆU THAM KHẢO

1. James Bignell, Robert Donovan, *Digital Electronics*. Thomson Learning, 4th edition, 2003.
2. Intel Corporation, *Microprocessors and IC Families*. Electronic, GmbH, Germany, 1st edition, 1993.
3. Nguyễn Việt Nguyên, *Giáo trình kỹ thuật số*. NXB Giáo dục, 2006.
4. Vũ Đức Thọ (Biên dịch), *Cơ sở kỹ thuật điện tử số*. NXB Giáo dục, 2006.
5. Cát Văn Thành, *Mạch logic – Mạch số*. NXB Thống kê, 2001.
6. Tô Đăng Hải, *Tra cứu vi mạch số CMOS*. NXB Khoa học và Kỹ thuật, 1993.

MỤC LỤC

LỜI NÓI ĐẦU	5
BÀI 1. CƠ SỞ KỸ THUẬT SỐ	5
1.1. Khái niệm tín hiệu tương tự và tín hiệu số	5
1.2. Khái niệm về mã và hệ đếm	7
1.2.1. Hệ đếm thập phân (hệ 10).....	7
1.2.2. Hệ đếm 2.....	8
1.2.3. Hệ đếm 8 (hệ OCTAL).....	8
1.2.4. Hệ đếm 16 (HEXA).....	9
1.2.5. Mã nhị phân BCD.....	10
1.2.6. Các phép biến đổi mã	12
1.2.7. Cách biểu diễn một con số trong một hệ đếm	16
1.3. Các phép tính số học trong hệ nhị phân.....	17
1.3.1. Phép cộng trong hệ 2	17
1.3.2. Phép trừ nhị phân.....	17
1.3.3. Phép nhân hai số nhị phân.....	18
1.3.4. Phép chia hai số nhị phân	19
1.4. Đại số logic (Đại số Boole).....	20
1.4.1. Cơ sở của đại số logic.....	20
1.4.2. Các tính chất quan trọng của tập hợp các biến logic	21
1.4.3. Phương pháp biểu diễn hàm logic	22
1.4.4. Rút gọn (tối thiểu) hàm logic	25
1.4.5. Các hàm logic cơ bản.....	26
1.5. Hệ hàm đầy đủ.....	27
1.6. Câu hỏi và bài tập.....	27
 BÀI 2. CÁC PHẦN TỬ LOGIC CƠ BẢN	 29
2.1. Các thông số quan trọng của mạch điện cổng (phần tử logic cơ bản)	29
2.2. Cổng OR (cổng thực hiện hàm logic $F_1 = A + B$)	30
2.3. Cổng AND (cổng thực hiện hàm logic $F_2 = A.B$).....	31
2.4. Cổng NOT (cổng thực hiện hàm logic $F_0 = \overline{A}$) và cổng YES (hàm $G_0 = A$)....	32
2.5. Cổng NOR (cổng thực hiện hàm logic $F_3 = \overline{A+B}$).....	33
2.6. Cổng NAND (cổng thực hiện hàm logic $F_4 = \overline{A.B}$)	34

2.7. Cổng XOR (cổng thực hiện hàm logic $F_7 = \overline{A}B + A\overline{B} = A \oplus B$)	36
2.8. Câu hỏi và bài tập.....	37
 BÀI 3. PHẦN TỬ LOGIC THÔNG DỤNG	38
3.1. Mạch so sánh	38
3.2. Mạch cổng colectơ để hở (Open Collector Gates – OC)	40
3.3. Mạch cổng logic ba trạng thái (Tri – State – Gates – TS).....	43
3.4. Câu hỏi và bài tập.....	45
 BÀI 4. MẠCH MÃ HÓA (CODE)	48
4.1. Vấn đề chung	48
4.2. Bộ mã hóa nhị phân	48
4.3. Mã hóa BCD8421 (bộ mã hóa một trong 10 đường vào thành 4 đường ra) ...	51
4.4. Bộ mã hóa ưu tiên	51
4.5. Một số loại mã nhị phân thông dụng	54
4.5.1. Các loại mã nhị thập phân (BCD) thông dụng	54
4.5.2. Mã ISO (International Standardization Organization)	56
4.5.3. Mã ASCII (American National Standard Code for Information Interchange) ..	57
4.6. Câu hỏi và bài tập.....	58
 BÀI 5. MẠCH GIẢI MÃ (DECODE)	60
5.1. Khái niệm	60
5.2. Mạch giải mã hai lối vào chọn một trong bốn lối ra	61
5.3. Mạch giải mã 3 lối vào chọn 1 trong 8 lối ra.....	62
5.4. Mạch giải mã 4 đường vào thành 10 đường ra (chọn 1 trong 10).....	64
5.5. Bộ giải mã BCD8421 thành 7 vạch dùng LED/LCD	65
5.6. Câu hỏi và bài tập.....	68
 BÀI 6. MẠCH CHỌN DỮ LIỆU (MUX)	72
6.1. Khái niệm	72
6.2. Mạch dẫn kênh	73
6.3. Bộ dẫn kênh 8 đầu vào dữ liệu.....	73
6.4. Một vài ứng dụng của bộ dẫn kênh.....	77
6.4.1. Định tuyến dữ liệu.....	77
6.4.2. Chuyển đổi một luồng dữ liệu	80
6.4.3. Tạo hàm logic dạng đầy đủ trực tiếp dùng MUX	81
6.5. Câu hỏi và bài tập.....	82

BÀI 7. MẠCH PHÂN KÊNH (DEMUX)	84
7.1. Khái niệm	84
7.2. Cấu trúc khối của mạch phân kênh	84
7.3. Mạch phân kênh 4 lối ra	85
7.4. Mạch phân kênh 8 lối ra	86
7.5. Mạch dồn kênh/phân kênh tương tự (MUX/DEMUX) dùng các cổng IC CMOS.....	87
7.6. Câu hỏi và bài tập.....	91
BÀI 8. MẠCH LOGIC TUẦN TỰ	93
8.1. Khái niệm về mạch trigơ (Flip Flop – FF)	93
8.2. Mạch RSFF	94
8.2.1. Mạch RSFF không đồng bộ	94
8.2.2. RSFF đồng bộ	96
8.2.3. Các đầu vào điều khiển trực tiếp $\overline{PR}$ và $\overline{CLR}$	97
8.3. Trigơ trễ (Delay Flip Flop – DFF) đồng bộ	98
8.4. Mạch trigơ số Chủ – Tớ (Master – Slave Flip Flop – MSFF)	100
8.5. Mạch Trigơ số đa năng (JKFF) và trigơ đếm (TFF).....	101
8.6. Bộ đếm nhị phân và bộ đếm nhị – thập phân (BCD)	104
8.6.1. Bộ đếm nhị phân	104
8.6.2. Bộ đếm nhị thập phân (bộ đếm BCD).....	106
8.7. Bộ đếm có giá trị ban đầu thiết lập trước.....	108
8.8. Bộ đếm hai hướng (đếm thuận/ngược: UP/DOWN counter).....	109
8.9. Thanh ghi dịch	111
8.9.1. Nhiệm vụ cơ bản của thanh ghi dịch	111
8.9.2. Mạch điện thanh ghi dịch	112
8.9.3. Mở rộng chức năng ghi	113
8.9.4. Bộ đếm vòng (thanh ghi dịch từng bit SAR).....	114
8.10. Câu hỏi và bài tập.....	115
BÀI 9. BỘ NHỚ CHỈ ĐỌC (ROM)	120
9.1. Khái niệm chung về bộ nhớ	120
9.1.1. Bộ nhớ	120
9.1.2. Cấu trúc khối tổng quát của một bộ nhớ	121
9.1.3. Nâng cao chỉ tiêu chất lượng của bộ nhớ.....	122
9.2. Các thuộc tính của bộ nhớ chỉ đọc ROM	123
9.3. Cấu trúc ROM.....	123
9.3.1. Khối bộ nhớ	123

9.3.2. Khối giải mã địa chỉ	125
9.3.3. Mạch điện đầu ra của ROM	127
9.3.4. Mạch điều khiển trong ROM có chức năng tạo các tín hiệu điều khiển ..	128
9.4. Các dạng ROM thường gặp	129
9.4.1. Cấu trúc khối của loại ROM lập trình bằng mặt nạ	129
9.4.2. ROM cho phép lập trình (PROM)	131
9.4.3. Chip ROM cho phép xóa và lập trình lại (EPROM – Erasable PROM) ...	133
9.5. Câu hỏi và bài tập	134
 BÀI 10. BỘ NHỚ TRUY CẬP NGẪU NHIÊN	138
10.1. Khái niệm	138
10.2. Mảng ô nhớ của RAM dùng tranzito nhiều emitor	140
10.3. RAM tĩnh (SRAM – Static Random Acces Memory) dùng CMOS	143
10.4. RAM động (DRAM) dùng CMOS	146
10.4.1. Nguyên lý DRAM	146
10.4.2. Hoạt động của một đơn vị nhớ DRAM	146
10.4.3. IC DRAM	148
10.4.4. Sơ đồ khối cấu trúc DRAM	150
10.5. Câu hỏi và bài tập	152
 BÀI 11. MẠCH CHUYỂN ĐỔI SỐ – TƯƠNG TỰ (DAC)	156
11.1. Khái niệm chung	156
11.2. Phương pháp DAC dùng thang điện trở	157
11.2.1. Phương pháp dùng mạch cộng điện áp	157
11.2.2. Phương pháp cộng điện áp dùng thang điện trở	159
11.3. Mạch DAC dùng mạng điện trở	161
11.4. Một vài dạng chuyển mạch 2 vị trí S_1	163
11.5. Mạch DAC dùng phương pháp mã hóa liên tiếp (mã hóa Shannon)	164
11.6. Mạch DAC dùng phương pháp chọn kênh (ma trận $2^n R$)	167
11.7. Một vài ứng dụng của các mạch DAC	168
11.8. Câu hỏi và bài tập	171
 BÀI 12. MẠCH CHUYỂN ĐỔI TƯƠNG TỰ – SỐ (ADC)	178
12.1. Khái niệm chung	178
12.1.1. Nhiệm vụ cơ bản của ADC	178
12.1.2. Các sai số quan trọng khi thực hiện ADC	178
12.1.3. Các phương pháp thực hiện ADC	179

12.2. ADC theo phương pháp song song (so sánh đồng thời)	179
12.3. ADC dùng phương pháp xấp xỉ nhiều nhịp	181
12.3.1. ADC loại hai nhịp	181
12.3.2. ADC loại xấp xỉ dần từng bit (loại n nhịp)	182
12.4. ADC dùng phương pháp đếm	187
12.5. ADC dùng phương pháp tích phân (phương pháp quét)	191
12.6. ADC dùng phương pháp biến đổi điện áp – tần số	199
12.6.1. VFC dùng dao động đa hài điều khiển bằng dòng điện	199
12.6.2. VFC dùng phương pháp cân bằng điện tích	200
12.6.3. Cấu trúc khối và bố trí chân của IC AD537	200
12.7. Câu hỏi và bài tập	202
 BÀI THỰC HÀNH ĐIỂN HÌNH: THỰC HÀNH BÀI 12	207
A. Phần chuẩn bị	207
B. Trình tự lắp ráp và cân chỉnh mạch	208
C. Báo cáo các kết quả thực hành	209
TÀI LIỆU THAM KHẢO	210
MỤC LỤC	211

Chịu trách nhiệm xuất bản:

Chủ tịch HĐQT kiêm Tổng Giám đốc NGÔ TRẦN ÁI

Phó Tổng Giám đốc kiêm Tổng biên tập NGUYỄN QUÝ THAO

Tổ chức bản thảo và chịu trách nhiệm nội dung:

Chủ tịch HĐQT kiêm Giám đốc CTCP Sách ĐH – DN

TRẦN NHẬT TÂN

Biên tập nội dung và sửa bản in:

DƯƠNG VĂN BẰNG

Biên tập mỹ thuật và trình bày bìa:

BÍCH LA

Thiết kế sách và chế bản:

NGUYỄN ĐÌNH PHONG

Kỹ thuật số

Mã số : 6E018Y9 - DAI

In 1.000 bản (QĐ 39), khổ 16 x 24cm tại Công ty cổ phần in Sách giáo khoa tại TP - Hà Nội.

Địa chỉ: Tổ 60 Thị trấn Đông Anh, Hà Nội.

Số ĐKKH xuất bản: 161-2009/CXB/2 - 2008/GD.

In xong và nộp lưu chiểu tháng 7 năm 2009.



CÔNG TY CỔ PHẦN SÁCH ĐẠI HỌC - DẠY NGHỀ
HEVOBCO
25 HÀN THUYỀN - HÀ NỘI
Website : www.hevobco.com.vn



VƯƠNG MIỆN KIM CƯƠNG
CHẤT LƯỢNG QUỐC TẾ

TÌM ĐỌC
SÁCH THAM KHẢO KỸ THUẬT
NHÀ XUẤT BẢN GIÁO DỤC VIỆT NAM
(Bộ giáo trình dùng cho sinh viên hệ Cao đẳng nghề)

1. Kỹ thuật mạch điện tử I
2. Kỹ thuật mạch điện tử II
3. Kỹ thuật mạch điện tử III
4. Kỹ thuật điện tử
5. Kỹ thuật số
6. Sửa chữa máy thu hình

TS. Nguyễn Viết Nguyên (CB)
TS. Nguyễn Viết Nguyên (CB)
TS. Nguyễn Viết Nguyên (CB)
TS. Nguyễn Vũ Sơn (CB)
TS. Nguyễn Viết Nguyên
TS. Đỗ Hoàng Tiến (CB)

Bạn đọc có thể mua sách tại các Công ty Sách - Thiết bị trường học ở các địa phương hoặc các Cửa hàng sách của Nhà xuất bản Giáo dục Việt Nam :

- Tại TP. Hà Nội : 25 Hàn Thuyên ; 187 Giảng Võ ; 232 Tây Sơn ; 23 Tràng Tiền.
- Tại TP. Đà Nẵng : 15 Nguyễn Chí Thanh ; 62 Nguyễn Chí Thanh.
- Tại TP. Hồ Chí Minh : Cửa hàng 451B - 453 ; Hai Bà Trưng - Quận 3 ;
240 Trần Bình Trọng - Quận 5.
- Tại TP. Cần Thơ : 5/5, đường 30/4.

Website : www.nxbgd.com.vn



Giá: 28.500đ