

## BÀI TẬP CHƯƠNG 5 – MẠCH TỔ HỢP

### I. Lý thuyết:

1. Kể tên 1 vài IC có khả năng thực hiện 1 số tác vụ sau:

- a) Cộng và trừ 2 số 4-bit
- b) Bộ tính toán số học và luận lý (ALU) giữa 2 số 4-bit
- c) Mạch giải mã (decoder) 2x4, 3x8
- d) Mạch mã hóa ưu tiên (priority encoder) 8x3

2. Nêu 1 vài ứng dụng của mạch giải mã (decoder), mã hóa (encoder), multiplexer, demultiplexer

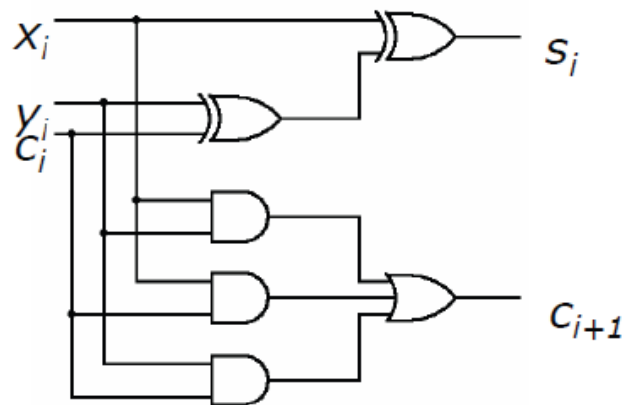
### II. Bài tập:

1. Cho biết mã Gray là một loại mã mà 2 từ mã liên tiếp chỉ khác nhau 1 bit như bảng sau:

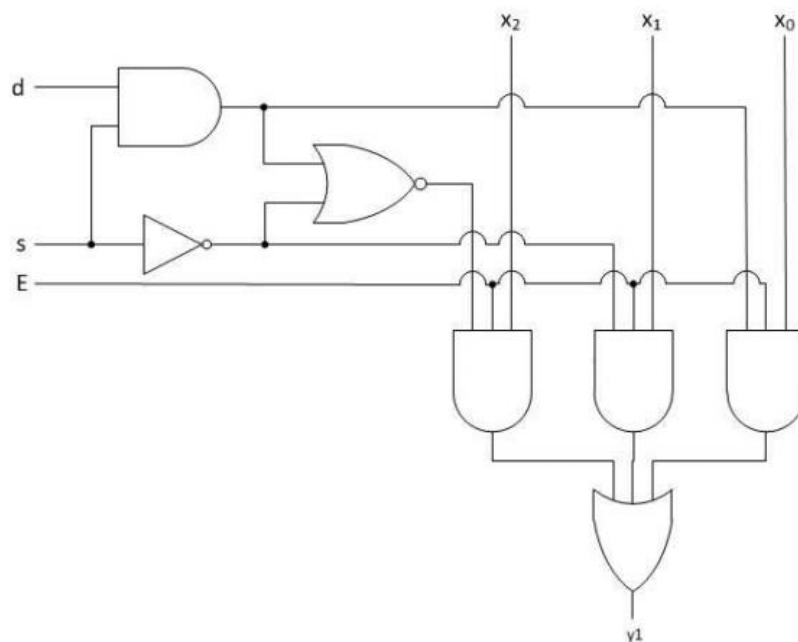
Decimal	Gray code
0	0000
1	0001
2	0011
3	0010
4	0110
5	0111
6	0101
7	0100
8	1100
9	1101
10	1111
11	1110
12	1010
13	1011
14	1001
15	1000

Hãy thiết kế mạch tổ hợp giúp chuyển đổi mã BCD sang mã Gray tương ứng.

2. Thiết kế một mạch tổ hợp có 3 ngõ nhập A, B, C và một ngõ xuất Y. Ngõ xuất  $Y = 1$  (HIGH) khi và chỉ khi giá trị thập phân tương đương của ngõ nhập (ABC) nhỏ hơn 4 hoặc lớn hơn 6 (với A là MSB, C là LSB).
3. Cho hàm Boolean  $F(a, b, c) = \sum m(4, 6, 7)$ . Hãy trình bày thiết kế tối ưu nhất về tài nguyên cho hàm F theo từng cách sau
- Decoder 3x8 và một vài cổng luận lý (logic gate) cơ bản
  - Decoder 2x4 và một vài cổng luận lý (logic gate) cơ bản
  - Decoder 2x4 và Decoder 1x2 và một vài cổng luận lý (logic gate) cơ bản
  - Chỉ sử dụng multiplier 8:1
  - Chỉ sử dụng multiplier 4:1
4. Chuyển mạch FA dưới đây sang mạch với các cổng NAND



5. Cho sơ đồ sau:



- Viết hàm boolean cho sơ đồ
- Lập bảng chân trị cho hàm **y1**

6. Trong các máy vi tính hoặc các máy tính, thường có một mạch để xác định overflow. Có vài cách để làm việc này. Một trong những cách có thể được sử dụng cho mạch cộng số bù 2 được thực hiện theo các bước như sau:

B1. Kiểm tra bit dấu của 2 số được cộng

B2. Kiểm tra bit dấu kết quả

B3. Có overflow khi các số được cộng cùng là số dương và bit dấu của kết quả là 1 hoặc khi các số được cộng là số âm và bit dấu của kết quả là 0.

Phương pháp này có thể được kiểm chứng bằng các ví dụ.

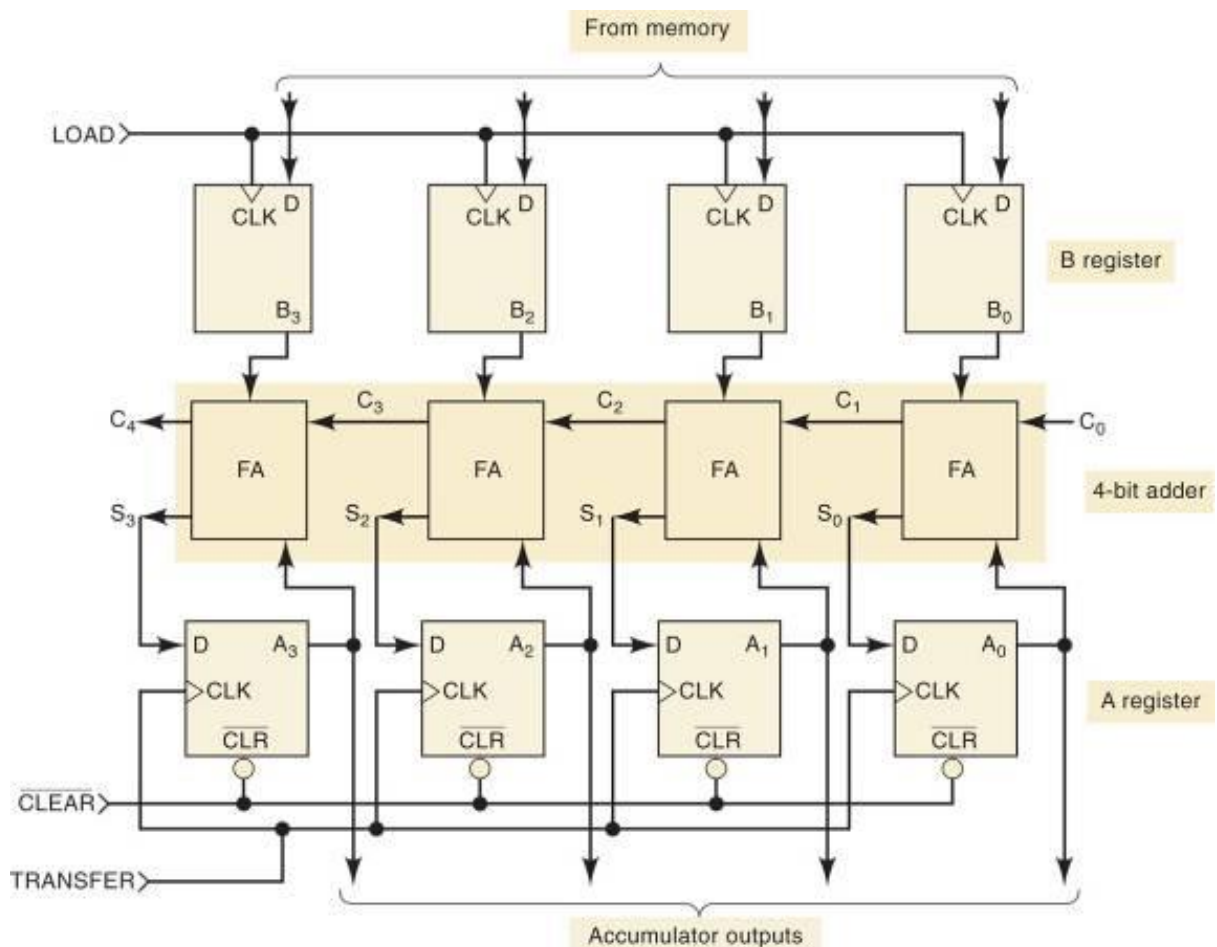
a.  $5 + 4$

b.  $-4 + (-6)$

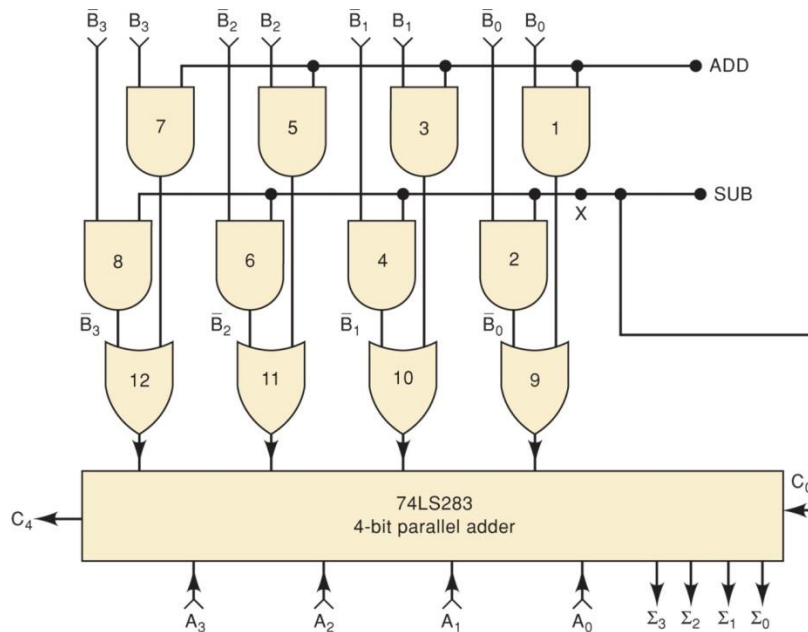
c.  $3 + 2$

Trường hợp a, b sẽ có overflow còn trường hợp c thì không.

Vì vậy, bằng cách kiểm tra bit dấu, một mạch logic có thể được thiết kế để cho ra 1 khi nào overflow xuất hiện. Hãy thiết kế mạch kiểm tra overflow cho mạch cộng bên dưới.



7. Cho mạch bên dưới, xác định giá trị ngõ ra trong các trường hợp sau:

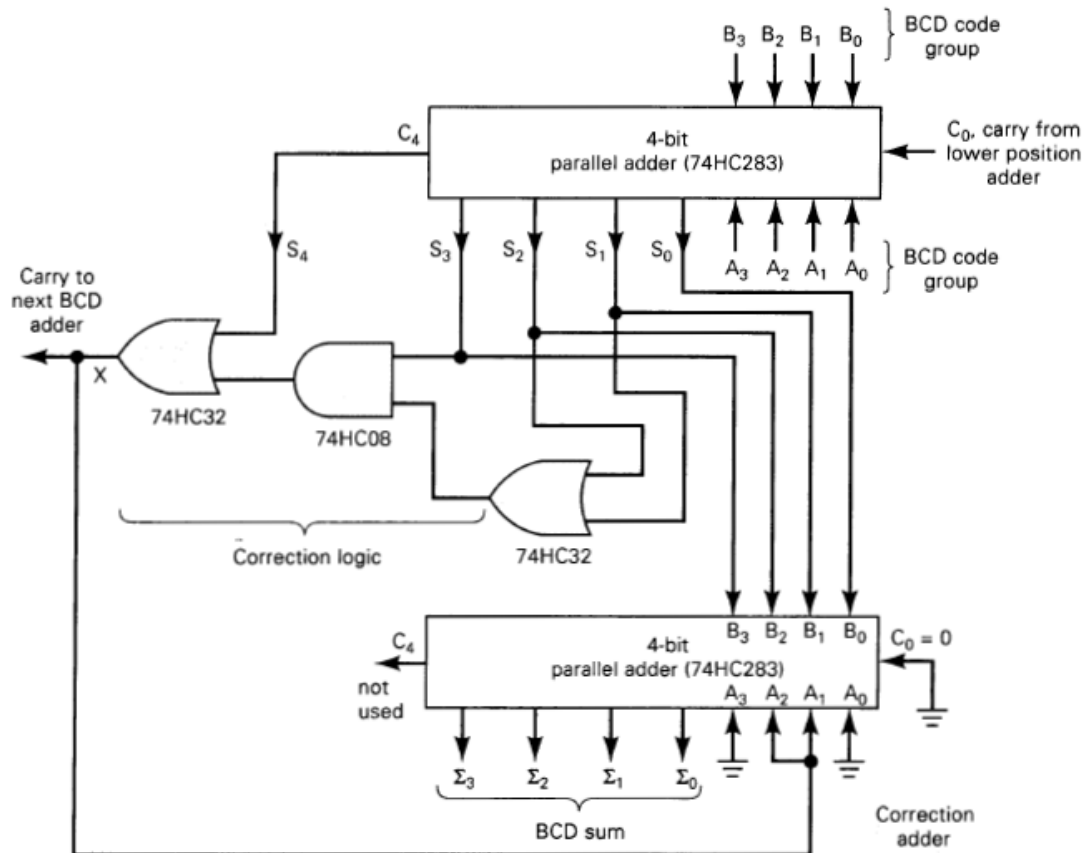


- Thanh ghi A = 0101 (+5), thanh ghi B = 0010 (2); SUB = 1, ADD = 0
- Thanh ghi A = 1100 (-4), thanh ghi B = 1110 (-2); SUB = 0, ADD = 1

8. Sửa lại mạch ở bài 3 với chỉ 1 ngõ vào X thay cho 2 ngõ vào ADD và SUB. Mạch mới này sẽ thực hiện nhiệm vụ mạch cộng khi X = 0, và mạch trừ khi X = 1. Sau đó rút gọn các cổng.

(Gợi ý: bây giờ mỗi tập cổng hoạt động như là inverter điều khiển)

9. Đối với mạch cộng BCD bên dưới, có hay không có sự khác biệt gì nếu C0 của mạch cộng phía trên được giữ ở mức thấp trong khi C0 của mạch cộng dưới được sử dụng như là số nhớ đầu vào (carry in)? Giải thích.



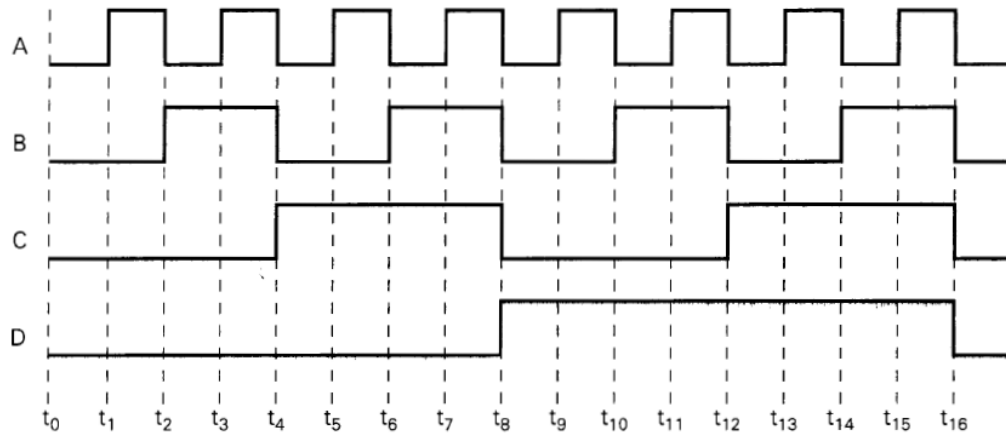
10. Mỗi phát biểu dưới đây liên hệ đến 1 decoder hay 1 encoder?

- Có nhiều input hơn output
- Được sử dụng để chuyển đổi 1 phím được bấm sang mã nhị phân
- Chỉ 1 output được tích cực tại một thời điểm
- Có thể được sử dụng để giao tiếp 1 input dạng BCD với 1 bộ hiển thị LED

11. Xem xét dạng sóng dưới đây. Đưa tín hiệu này vào IC 74LS138 như sau:

$A \Rightarrow A$ ;  $B \Rightarrow B$ ;  $C \Rightarrow C$ ;  $D \Rightarrow G1$

Giả sử  $G2A\_L$  và  $G2B\_L$  được giữ ở mức thấp. Hãy vẽ dạng sóng cho ngõ ra  $Y0\_L$ ,  $Y3\_L$ ,  $Y6\_L$ ,  $Y7\_L$



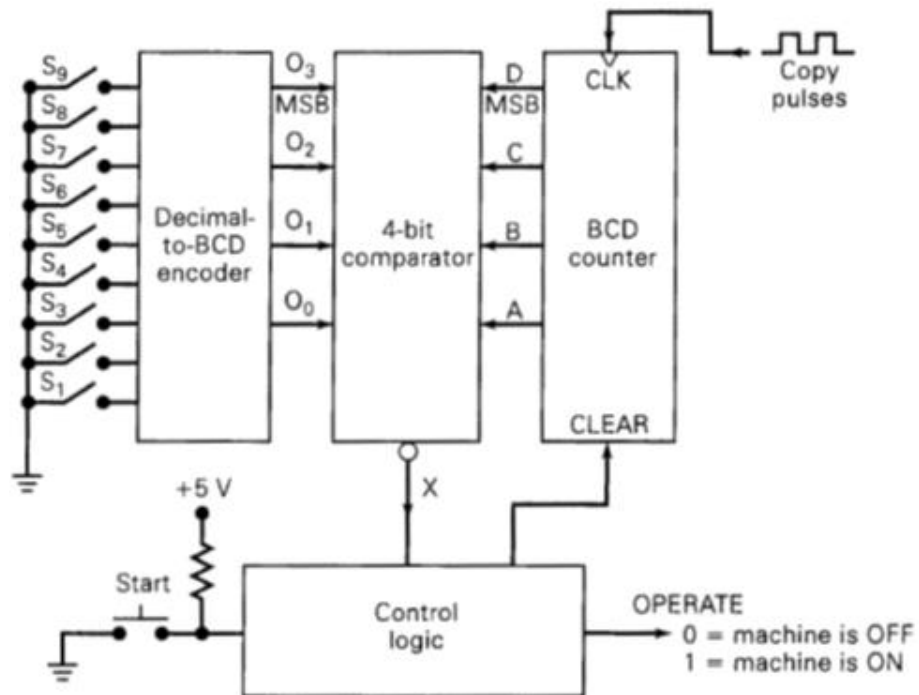
Inputs						Outputs							
G1	G2A_L	G2B_L	C	B	A	Y7_L	Y6_L	Y5_L	Y4_L	Y3_L	Y2_L	Y1_L	Y0_L
0	x	x	x	x	x	1	1	1	1	1	1	1	1
x	1	x	x	x	x	1	1	1	1	1	1	1	1
x	x	1	x	x	x	1	1	1	1	1	1	1	1
1	0	0	0	0	0	1	1	1	1	1	1	1	0
1	0	0	0	0	1	1	1	1	1	1	1	0	1
1	0	0	0	1	0	1	1	1	1	1	0	1	1
1	0	0	0	1	1	1	1	1	1	0	1	1	1
1	0	0	1	0	0	1	1	1	0	1	1	1	1
1	0	0	1	0	1	1	1	0	1	1	1	1	1
1	0	0	1	1	0	1	0	1	1	1	1	1	1
1	0	0	1	1	1	0	1	1	1	1	1	1	1

Bảng sự thật của chip 74LS138

12. Bên dưới là sơ đồ khối của mạch điều khiển số lượng bản copy của 1 máy photocopy. Người thao tác trên máy sẽ thiết lập một số lượng bản copy mong muốn bằng cách đóng một trong các switch S1 đến S9. Số này được mã hóa BCD bởi mạch mã hóa và được đưa vào mạch comparator. Người thao tác máy sẽ nhấn nút START, để xóa counter và khởi tạo ngõ ra OPERATE ở mức cao để làm tín hiệu cho máy thực hiện việc copy.

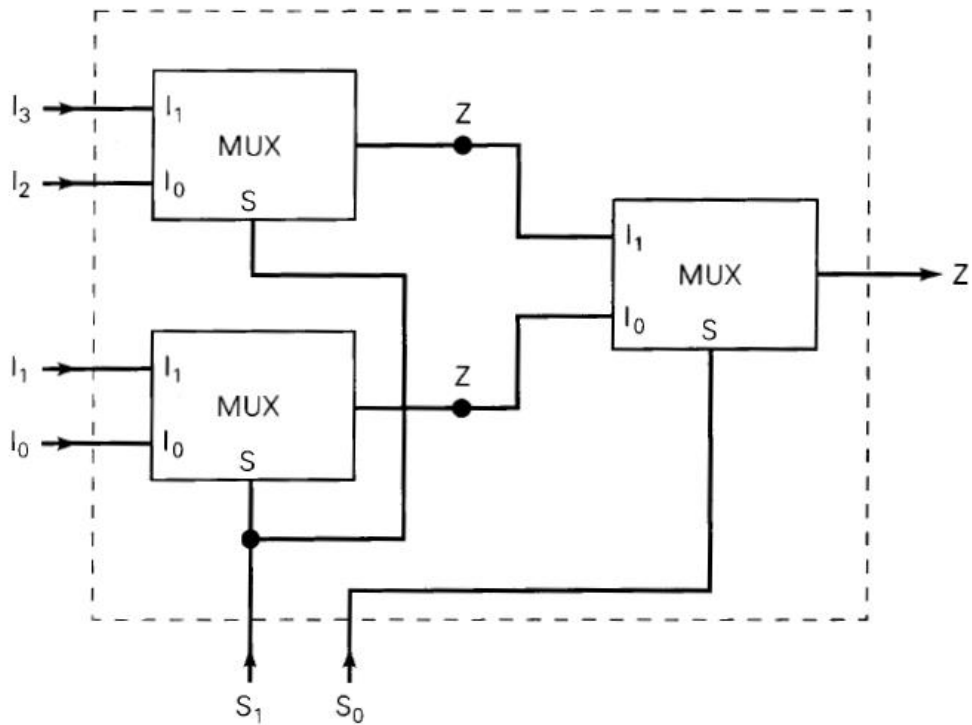
Khi máy hoạt động, một xung copy được tạo ra và đưa vào BCD counter. Ngõ ra của counter liên tục được so sánh (bởi mạch comparator) với mã hóa của các switch. Khi 2 số BCD này giống nhau, cho biết đã copy đủ số lượng, ngõ ra X của comparator sẽ xuống mức THẤP, làm cho tín hiệu OPERATE trở về mức THẤP và ngưng máy photocopy không cho copy thêm bản nào nữa. Kích hoạt nút START sẽ lặp lại quá trình này.

Hãy thiết kế một mạch comparator hoàn chỉnh và mạch điều khiển của hệ thống này.



**Lưu ý:** Khi tín hiệu Start và X cùng tích cực, thì tín hiệu Start có độ ưu tiên cao hơn

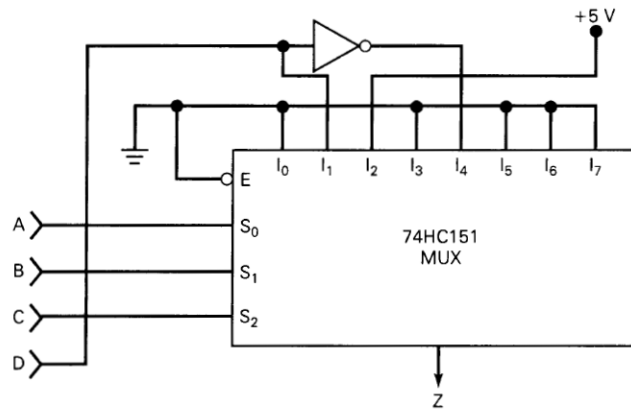
**13.** Mạch ở dưới sử dụng 3 multiplexer 2-1. Xác định hàm được thực hiện bởi mạch này.



14. Mạch bên dưới biểu diễn cách 1 multiplexer 8 inputs được sử dụng để tạo ra một hàm logic 4 biến ngay cả khi MUX chỉ có 3 tín hiệu SELECT đầu vào. Ba biến A, B, và C được nối vào SELECT. Biến thứ tư D và đảo của nó  $\bar{D}$  được nối vào một vài chân dữ liệu của MUX có chọn lựa như yêu cầu của hàm logic mong muốn. Những chân dữ liệu khác của MUX được giữ ở mức THẤP hoặc CAO qui định bởi hàm logic.

- Lập bảng sự thật cho mạch
- Viết biểu thức của Z dưới dạng SOP và tối ưu hóa Z để kiểm tra rằng  

$$Z = \bar{C}B\bar{A} + D\bar{C}\bar{B}A + \bar{D}C\bar{B}\bar{A}$$



15. Mô tả cách sử dụng IC 74151 (multiplexer 8 inputs) để tạo ra hàm logic  
 $z = AB + BC + AC$ .

16. Chứng minh theo Shannon biểu thức sau:

- $AB + B'CD + ACD = AB + B'CD$
- $(A+B+C)(A'+D)(B+C+D) = (A+B+C)(A'+D)$

17. Một Full subtractor (x-y) có 3 bits vào: x, y, borrow-in  $b_{in}$  và 2 đầu ra: hiệu số d (defference), bit mượn borrow-out  $b_{out}$

- Xây dựng bảng sự thật
- Viết hàm dạng SOP
- Viết hàm dạng POS
- Vẽ sơ đồ mạch của Full Sub cho câu b hoặc c

18. Cho các hàm sau:

- $F1(A,B,C,D) = \Sigma(1,2,4,7) + d(3)$ ,
- $F2(A,B,C,D) = \Sigma(0,3,14) + d(15)$ ,
- $F3(A,B,C,D) = \Sigma(12,15)$

- Hiện thực các hàm sử dụng 4-16 Decoders và cổng OR
- Hiện thực các hàm sử dụng 3-8 Decoders và cổng OR
- Hiện thực các hàm sử dụng 2-4 Decoders và cổng OR



**19.** Cho hàm  $F(A,B,C) = \Sigma(0,3,5,7) + d(6)$

- a. Hiện thực các hàm sử dụng 8-1 MUX
- b. Hiện thực các hàm sử dụng 4-1 MUX
- c. Hiện thực các hàm sử dụng 2-1 MUX

**20.** Cho hàm sau:  $F(A,B,C) = \Sigma(0,2,4,6,7) + d(1)$

- a. Hiện thực hàm F sử dụng ít nhất các mạch 2-4 decoders và cổng OR
- b. Hiện thực hàm F sử dụng ít nhất các mạch 4-1 MUX
- c. Hiện thực hàm F sử dụng ít nhất các mạch 2-1 MUX

**21.** Mỗi phát biểu dưới đây liên hệ đến 1 decoder, 1 encoder, 1 MUX, hay 1 DEMUX?

- a. Có nhiều inputs hơn outputs
- b. Sử dụng chân input SELECT
- c. Có thể được sử dụng để chuyển từ song song sang tuần tự (parallel-to-serial conversion)
- d. Tạo ra 1 mã nhị phân ở outputs
- e. Chỉ một trong những outputs được tích cực tại một thời điểm
- f. Có thể được sử dụng để hướng 1 tín hiệu input tới 1 hoặc một vài outputs
- g. Có thể được sử dụng để sinh ra hàm logic bất kỳ