**MỤC LỤC** Trang

**MỤC LỤC** 1

**LỜI NÓI ĐẦU** 6

**PHẦN I: LÝ THUYẾT KỸ THUẬT SỐ**

**CHƯƠNG 1: HỆ THỐNG ĐẾM VÀ MÃ** 7

1.1. CÁC KHÁI NIỆM CƠ BẢN 7

1.1.1. Tín hiệu tương tự và tín hiệu số 7

1.1.1.1. Khái niệm 7

1.1.1.2. Ưu điểm của mạch số 7

1.1.1.3. Nhược điểm của mạch số 8

1.1.2. Trạng thái nhị phân và mức logic 8

1.1.3. Bit, byte, word 8

1.2. CÁC HỆ THỐNG SỐ 8

1.2.1. Hệ cơ số 10(thập phân – Decimal system) 8

1.2.2. Hệ cơ số 2(nhị phân – Binary system) 9

1.2.3. Hệ cơ số 8(bát phân – Octal system) 9

1.2.4. Hệ cơ số 16(thập lục phân – Hexadecimal system) 9

1.3. CHUYỂN ĐỔI GIỮA CÁC HỆ THỐNG SỐ 10

1.3.1. Đối một số từ hệ N sang hệ 10(N = 2, 8, 16) 11

1.3.2. Đổi một số từ hệ 10 sang hệ N 11

1.4. CÁC PHÉP TÍNH TRONG HỆ NHỊ PHÂN 11

1.4.1. Phép cộng 11

­1.4.2. Số nhị phân có dấu 12

1.4.3. Phép trừ 13

1.4.4. Phép nhân 14

1.4.5. Phép chia 14

BÀI TẬP 15

**CHƯƠNG 2: ĐẠI SỐ LOGIC** 16

2.1. CÁC KHÁI NIỆM 16

2.2. TÍNH CHẤT CỦA CÁC HÀM LOGIC CƠ BẢN 16

2.2.1. Tính chất cơ bản 16

2.2.2. Tính song đối 17

2.2.3. Định lý De Morgan 17

2.3. PHƯƠNG PHÁP BIỂU DIỄN HÀM LOGIC 18

2.3.1 Biểu diễn bằng bảng chân lý 18

2.3.2. Các phương pháp biểu diễn hàm logic 19

2.3.2.1 Phương pháp biểu diễn hàm logic bằng dạng tổng của các tích 19

2.3.2.2 Phương pháp biểu diễn hàm logic bằng dạng tích của các tổng 20

2.3.3. Đơn giản hóa hàm logic 20

2.3.3.1. Đơn giản hóa hàm logic bằng phương pháp đại số 20

2.3.3.2. Đơn giản hóa hàm logic bằng phương pháp bản đồ Karnaugh 22

BÀI TẬP 26

**CHƯƠNG 3: CỔNG LOGIC** 27

3.1. CÁC CỔNG LOGIC CƠ BẢN 27

3.1.1. Cổng AND 27

3.1.2. Cổng OR 28

3.1.3. Cổng NOT 28

3.1.4. Cổng NAND 29

3.1.5. Cổng NOR 30

3.1.6. Cổng XOR 31

3.1.7. Cổng X-NOR 32

3.2. THÔNG SỐ KỸ THUẬT CỦA IC SỐ 32

3.2.1. Các đại lượng điện đặc trưng 33

3.2.2. Công suất tiêu tán (Power requirement) 33

3.3.3. Fan-Out 33

3.2.4. Thời trễ truyền (Propagation delays) 34

3.2.5. Tích số công suất-vận tốc (speed- power product) 34

3.2.6. Tính miễn nhiễu (Noise immunity) 34

3.3. THÔNG SỐ CÁC HỌ LOGIC 35

3.3.1. Họ TTL (Transistor - Transistor Logic) 35

3.3.2. Họ MOS 36

BÀI TẬP 37

**CHƯƠNG 4: MẠCH TỔ HỢP** 38

4.1 MẠCH MÃ HÓA VÀ MẠCH GIẢI MÃ 38

4.1.1 Mạch mã hóa 38

4.1.1.1. Bộ mã hóa bát phân thành nhị phân 39

4.1.1.2. Bộ mã hóa ưu tiên thập phân thành BCD 40

4.1.1.3. Bộ mã hóa nhị phân 4 bit sang mã bù 2 41

4.1.2 Mạch giải mã 42

4.1.2.1. Bộ giải mã 3 sang 8 42

4.1.2.2. Bộ giải mã BCD sang thập phân 44

4.2 HỆ CHUYỂN ĐỔI MÃ 46

4.2.1 Chuyển đổi mã BCD sang led 7 đoạn 46

4.2.1.1. Led 7 đoạn 46

4.2.1.2 Mạch giải mã BCD sang Led 7 đoạn 46

4.2.1.3 Hiển thị 7 đoạn bằng tinh thể lỏng(LCD - Liquid Crystal Display) 47

4.2.2 Chuyển đổi mã hệ 2 sang mã Gray 48

4.3 MẠCH CHỌN KÊNH VÀ PHÂN KÊNH 49

4.3.1 Mạch chọn kênh(MUX) 49

4.3.1.1. Bộ chọn kênh 2 đầu vào(MUX 2 : 1) 50

4.3.1.2. Bộ chọn kênh 4 đầu vào(MUX 4 : 1) 50

4.1.1.3. Bộ chọn kênh 8 đầu vào(MUX 8 : 1) 51

4.3.2. Mạch phân kênh(DMUX – Demultiplexer) 53

4.3.2.1. Bộ phân kênh 2 đầu ra(DMUX 1:1) 53

4.3.2.2. Bộ phân kênh 8 đầu ra(DMUX 1 : 8) 53

BÀI TẬP 55

**CHƯƠNG 5: MẠCH LOGIC DÃY** 57

5.1. CÁC TRIGƠ SỐ 58

5.1.1. Định nghĩa và phân loại 58

5.1.2. Hoạt động của mạch flip – flop 58

5.1.3. Các flip – flop cơ bản 60

5.1.3.1. Flip – flop RS 60

5.1.3.2. Flip – flop JK 62

5.1.3.3. Flip – flop T 63

5.1.3.4. Flip – flop D 65

5.2. CHUYỂN ĐỔI GIỮA CÁC FLIP - FLOP 66

5.3. CÁC BỘ ĐẾM 68

5.3.1 Định nghĩa 68

5.3.2. Phân loại 68

5.3.3. Các bước thiết kế bộ đếm 69

5.4. CÁC BỘ GHI DỊCH 73

5.4.1. Khái niệm 73

5.4.2. Bộ ghi dịch nối tiếp 73

5.4.3. Bộ ghi dịch song song 74

5.4.4. Bộ ghi dịch trái – phải 75

BÀI TẬP 77

**CHƯƠNG VI: BỘ NHỚ BÁN DẪN** 78

6.1 CÁC KHÁI NIỆM 78

6.1.1. Chức năng của bộ nhớ 78

6.1.2. Các thông số cơ bản của bộ nhớ 78

6.1.3. Phân loại 79

6.2 BỘ NHỚ TRUY CẬP NGẪU NHIÊN RAM 79

6.2.1. Khái niệm 79

6.2.2. Phân loại 79

6.2.2.1. RAM tĩnh(SRAM - Static RAM) 80

6.2.2.2. RAM động(DRAM - Dynamic RAM) 81

6.3 BỘ NHỚ CHỈ ĐỌC ROM 82

6.3.1. Khái niệm 82

6.3.2. Phân loại 82

**CHƯƠNG 7: CHUYỂN ĐỔI TÍN HIỆU** 83

7.1. MẠCH CHUYỂN ĐỔI SỐ - TƯƠNG TỰ 83

7.1.1. Khái niệm 83

7.1.2. Cấu trúc 83

7.1.3 Bộ khuếch đại thuật toán(OP - AMP) 84

7.1.4. Bộ biến đổi D/A cơ bản 85

7.2 MẠCH CHUYỂN ĐỔI TƯƠNG TỰ - SỐ 86

7.2.1. Khái niệm 86

7.2.2 Mạch lấy mẫu và giữ mẫu 86

7.2.3. Bộ biến đổi A/D cơ bản 86

**PHẦN II: THỰC HÀNH KỸ THUẬT SỐ**

GIỚI THIỆU CÁC MODUN TRÊN PANEL ĐIỆN TỬ SỐ 89

BÀI THỰC HÀNH SỐ 1 92

BÀI THỰC HÀNH SỐ 2 93

BÀI THỰC HÀNH SỐ 3 97

BÀI THỰC HÀNH SỐ 4 101

BÀI THỰC HÀNH SỐ 5 103

BÀI THỰC HÀNH SỐ 6 106

BÀI THỰC HÀNH SỐ 7 109

BÀI THỰC HÀNH SỐ 8 112

BÀI THỰC HÀNH SỐ 9 115

BÀI THỰC HÀNH SỐ 10 119

BÀI THỰC HÀNH SỐ 11 122

BÀI THỰC HÀNH SỐ 12 125

BÀI THỰC HÀNH SỐ 13 128

**TÀI LIỆU THAM KHẢO** 131

**LỜI NÓI ĐẦU**

Trong sự phát triển của khoa học kỹ thuật, nhu cầu làm việc và tiếp xúc của con người về lĩnh vực điện tử số ngày càng trở nên quan trọng. Với mong muốn trang bị cho sinh viên ngành điện nói riêng và những người yêu thích về điện tử số nói chung những kiến thức cơ bản cô đọng nhất, cuốn sách “Kỹ thuật số thực hành” đã ra đời nhằm giải quyết nhu cầu nói trên.

Với kinh nghiệm từ thực tế, cuốn sách **CHƯƠNG 1: HỆ THỐNG ĐẾM VÀ MÃ**

**1.1. CÁC KHÁI NIỆM CƠ BẢN**

**1.1.1. Tín hiệu tương tự và tín hiệu số.**

**1.1.1.1. Khái niệm**

*- Tín hiệu tương tự*: Tín hiệu tương tự là tín hiệu có biên độ biến thiên liên tục theo thời gian. Trong thực tế, tín hiệu tương tự là các đại lượng vật lý như vận tốc, nhiệt độ môi trường, biên độ tín hiệu đầu ra của loa trong máy thu vô tuyến…vv.



*Hình 1.1: Biểu diễn tín hiệu tương tự*

Tín hiệu tương tự có thể được chuyển đổi thành tín hiệu số bởi mạch chuyển đổi tương tự sang số(ADC – Analog to Digital Converter)

*- Tín hiệu số:* Tín hiệu số là tín hiệu có biên độ gián đoạn không liên tục theo thời gian. Biên độ chỉ có mức: Mức 1tương ứng với giá trị 5V đặc trưng cho giá trị biên độ cao, Mức 0 tương ứng với giá trị 0V đặc trưng cho giá trị biên độ thấp.



*Hình 1.2: Biểu diễn tín hiệu số*

Tín hiệu tương tự có thể được chuyển đổi thành tín hiệu số bởi mạch chuyển đổi số sang tương tự (DAC – Digital to Analog Converter)

Thời gian biến thiên giữa hai mức gọi là thời gian chuyển tiếp, thời gian này càng ngắn, càng tốt.

**1.1.1.2. Ưu điểm của mạch số.**

+ Khả năng chống nhiễu và sự méo dạng cao

+ Lưu trữ và truy cập thông tin dễ dàng và nhanh chóng.

+ Tính chính xác và độ tin cậy cao hơn

+ Thuận tiện cho công nghệ tích hợp

+ Dễ thiết kế, lắp ráp và sửa chữa.

**1.1.1.3. Nhược điểm của mạch số.**

+ Hầu hết các đại lượng vật lý đều mang bản chất tương tự

+ Trong một vài trường hợp mạch số tốn kém hơn.

Tuy nhiên trong hầu hết trường hợp mạch số mới có khả năng về tốc độ, độ chính xác, mức độ phức tạp cần thiết.

**1.1.2. Trạng thái nhị phân và mức logic.**

Trong hệ thống kỹ thuật số, thông tin được xử lý đều biểu diễn dưới dạng nhị phân. Bất kỳ thiết bị nào chỉ có hai trạng thái hoạt động đều có thể biểu diễn được các đại lượng dưới dạng nhị phân.

*- Trạng thái nhị phân:*Trạng thái nhị phân là trạng thái được thể hiện qua hai con số nhị phân 0 và 1.

**Thí dụ:**

+ Số nhị phân 3 bit: 000, 001, 010…

+ Số nhị phân 4 bit: 0000, 0011, 0111…

+ Số nhị phân 8 bit: 00011010, 11101101…

*- Mức logic:* Mức logic được thể hiện qua hai mức: mức cao tương ứng với giá trị 5V và mức thấp tương ứng với giá trị 0V hoặc biểu thị qua hai mức 0 hoặc 1.

**1.1.3. Bit, byte, word**

*- Bit(viết tắt của Binary digit):* Là một trong hai số 0 hoặc 1 dùng trong các thiết bị số để biểu thị các số, các ký tự. Mỗi con số trong số nhị phân được gọi là một bit Bit là đơn vị nhỏ nhất của thông tin

**Thí dụ:**  Cho số nhị phân sau

10110101

Bit đầu tiên(hàng tận cùng bên trái) có giá trị cao nhất(1x27) và được gọi là MSB (Most Significant Bit – bit có nghĩa cao nhất), bit cuối cùng (hàng tận cùng bên phải) có giá trị thấp nhất(1x20) và được gọi là LSB(Least Significant Bit – bit có nghĩa ít nhất).

*- Byte:* Hầu hết thông tin được xử lý, lưu trữ dưới dạng nhóm 8 bit, vì vậy chuỗi 8 bit được gọi là 1 byte

**Thí dụ:** 1 byte **=** 10110101

*- Word:* Thông tin dữ liệu được tạo thành từ một đơn vị cơ bản gọi là từ (word). Tùy theo từng thông tin mà dữ liệu có thể là 8 bit, 16 bit, 32 bit… Như vậy một word tương ứng với hai byte.

**1.2. CÁC HỆ THỐNG SỐ**

**1.2.1. Hệ cơ số 10(thập phân – Decimal system).**

Hệ thập phân là hệ thống số thông dụng sử dụng 10 chữ số là các con số trong tập hợp:

S10 =(0, 1, 2, 3, 4, 5, 6, 7, 8, 9)

**Thí dụ:**

N = 201010 = 2000 + 10 = 2x103 + 0x102 + 1x101 + 0x100.

N = 3,1410 = 3 + 0,1 + 0,04 = 3x101 + 1x10-1 + 4x10-2.

**1.2.2. Hệ cơ số 2(nhị phân – Binary system).**

Hệ nhị phân là hệ thống số sử dụng hai chữ số là các con số trong tập hợp:

S2 =(0, 1)

Số N trong hệ nhị phân:

N = (bnbn-1bn-2......b0 , b-1b-2 . . .b-m)2

Có giá trị là:

N = bn 2n + bn-12n-1 +. . . + b020 + b-1 2-1 + b-22-2 +. . . + b-m2-m

bn là bit có trọng số lớn nhất được gọi là MSB(Most Significant Bit) và b-m là bit có trọng số nhỏ nhất gọi là LSB(Least Significant Bit).

**Thí dụ:**

N = 1001,12 = 1x23+ 0x22 + 0x21 + 1x20 + 1x2-1 = 9,510

N = 10112 **=** 1x23+ 0x22 + 1x21 + 1x20 = 1110

**1.2.3. Hệ cơ số 8(bát phân – Octal system).**

Hệ bát phân là hệ sử dụng tám chữ số là các con số trong tập hợp

S8 =(0, 1, 2, 3, 4, 5, 6, 7)

Số N trong hệ bát phân:

N = (bnbn-1bn-2......b0 , b-1b-2 . . .b-m)8

Có giá trị là:

N = bn 8n + bn-18n-1 +. . . + b080 + b-1 8-1 + b-28-2 +. . . + b-m8-m

**Thí dụ:**

N = 113,28 = 1x82 + 1x81 + 3x80 + 2x8-1 = 75,2510

N = 238 = 2x81 + 3x80 = 1910

**1.2.4. Hệ cơ số 16(thập lục phân – Hexadecimal system).**

Hệ thập lục phân được dùng rất thuận tiện để con người giao tiếp với máy tính, hệ này gồm mười sáu số trong tập hợp

S16 =(0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F)

(A tương đương với 1010, B = 1110, …F = 1510)

Số N trong hệ bát phân:

N = (bnbn-1bn-2......b0 , b-1b-2 . . .b-m)16

Có giá trị là:

N = bn 16n + bn-116n-1 +. . . + b0160 + b-1 16-1 + b-216-2 +. . . + b-m16-m

**Thí dụ:**

N = 13CA,616 = 1x163 + 3x162 + 12x161 + 10x160 + 6x16-1 = 5066,37510.

N = 4DA16 = 4x162 + 13x161 + 10x160 = 124210.

**1.3. CHUYỂN ĐỔI GIỮA CÁC HỆ THỐNG SỐ**

**1.3.1. Đối một số từ hệ N sang hệ 10(N = 2, 8, 16).**

Biểu diễn số A bất kỳ trong hệ cơ số N bằng các chữ số

bnbn-1bn-2......b0 , b-1b-2 . . .b-m

Có giá trị là:

A = bn Nn + bn-1Nn-1 +. . . + b0N0 + b-1 N-1 + b-2N-2 +. . . + b-mN-m

Với N: là cơ số hệ đếm

bi các chữ số dùng trong hệ đếm N.

**Thí dụ:**

10110112 **=** 1x26+ 0x25 + 1x24 + 1x23 + 0x22+ 1x21 + 1x20 = 9110

124**8 =** 1x82 + 2x81 + 4x80 = 8410

5D16 = 5x161 + 13x160 = 9310

**1.3.2. Đổi một số từ hệ 10 sang hệ N.**

Khi đổi một số từ hệ 10 sang hệ N thì ta lấy số của hệ 10 chia cho cơ số của hệ cần đổi và lấy số dư, số dư đầu tiêncó trọng số nhỏ nhất và số dư cuối cùng có trọng số lớn nhất.

Tổng quát một số A cho ở hệ 10 đổi sang hệ N có dạng

AN = (bnbn-1…b1b0)

**\* Đổi một số từ hệ 10 sang hệ 2**

*+ Cách 1:* Lấy số từ hệ 10 chia cho 2 và lấy số dư

**Thí dụ 1:** A10 = 47 → A2 = ?.

47 : 2 = 23 dư 1(LSB)  b0 = 1

23 : 2 = 11 dư 1  b1 = 1

11 : 2 = 5 dư 1  b2 = 1

5 : 2 = 2 dư 1  b3 = 1

2 : 2 = 1 dư 0  b4 = 0

1 : 2 = 0 dư 1(MSB)  b5 = 1

Kết quả: A2 = 101111

*+ Cách 2:* Đối với số thập phân lớn, cách gọn hơn là tìm hiệu số liên tiếp của số thập phân với lũy thừa của 2 có giá trị thấp hơn nhưng gần số thập phân nhất

**Thí dụ 2:** A10 = 725 → A2 = ?.

Hiệu số: 725 213 85 19 3 1

Số trừ : 512 128 64 16 2 1

(29) (27) (26) (24) (21) (20)

Kết quả: 7252 = 29 + 27 + 26 + 24 + 21 + 20

= 1011010011

**\* Đổi một số từ hệ 10 sang hệ 8**

**Thí dụ:** A10 = 145 → A8 = ?.

145 : 8 = 17 dư 9  b0 = 9

17 : 8 = 2 dư 1  b1 = 1

2 : 8 = 0 dư 2  b2 = 2

Kết quả: A8 = 219

**\* Đổi một số từ hệ 10 sang hệ 16**

**Thí dụ:** A10 = 167 → A16 = ?.

167 : 16 = 10 dư 7  b0 = 7

10 : 16 = 0 dư 10  b1 = A

Kết quả: A16 = A7

**1.4. CÁC PHÉP TÍNH TRONG HỆ NHỊ PHÂN**

Các phép tính trong hệ nhị phân được thực hiện tương tự như trong hệ thập phân, tuy nhiên để thực hiện phép tính thì trong hệ nhị phân cũng có quy tắc riêng được thể hiện như sau:

**1.4.1. Phép cộng**

Bảng thực hiện phép cộng

|  |  |  |  |
| --- | --- | --- | --- |
| x | y | x + y | Nhớ |
| 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 |

*Nhận xét: -* Nếu số bit 1 là chẵn sẽ cho kết quả là 0

**-**  Nếu số bit 1 là lẻ sẽ cho kết quả là 1

**-** Hai số 1 cộng lại với nhau sẽ nhớ một lần

**Thí dụ 1:** Thực hiện phép cộng hai số nhị phân 4 bit sau

5 = 0101

**+ 7 =** 0111

12 = 1100

**Thí dụ 2:** Thực hiện phép cộng hai số nhị phân 8 bit sau

47 + 35 = ?

Thực hiện phép biến đổi

47 : 16 = 2 dư 15  b0 = F

2 : 16 = 0 dư 2  b1 = 2

Suy ra 4710 = 2F16.

Và 35 : 16 = 2 dư 3  b0 = 3

2 : 16 = 0 dư 2  b0 = 2

Suy ra 3510 = 2316.

Vì vậy: 4710 = 2F16 = 001011112

+3510 = 2316 = 001000112

8210 = 5216 = 010100102

**­1.4.2. Số nhị phân có dấu.**

Thực tế, các số nhị phân là số không dấu và được hiểu là số dương. Tuy nhiên trong tính toán số học người ta thêm vào một bit gọi là bit dấu, thông thường bit 0 là bit dấu biểu thị số dương và bit 1 là bit dấu biểu thị số âm. Bit dấu này được thêm vào ở vị trí ngoài cùng bên trái.

- *Số bù 1*: là số nhị phân mà ta lấy đảo, có nghĩa thay bit 0 thành bit 1 và bit 1 thành bit 0.

**Thí dụ:**

Số nhị phân ban đầu: 1010011

Số bù 1 dãy số nhị phân trên là: 0101100

*- Số bù 2:* Bù hai của một số nhị phân có được bằng cách lấy bù 1 của số đó cộng thêm một đơn vị.

**Thí dụ:**

Số nhị phân ban đầu: 10101

Số bù 1: 01010

Cộng 1 + 1

Số bù 2 của số nhị phân ban đầu: 01011

* **Quy tắc tìm số bù 2.**

+ Nếu bit có ý nghĩa ít nhất(LSB) là 0 thì giữ nguyên các bit từ LSB đến bit 1 cuối cùng, các bit còn lại thực hiện đảo bit

Bit 1 cuối cùng

**Thí dụ:** LSB

Số ban đầu: 101100 Số bù 2: 010100

Đảo bit Giữ nguyên

+ Nếu LSB là 1 thì giữ nguyên LSB, các bit còn lại thực hiện đảo bit

**Thí dụ:** LSB

Số ban đầu: 100101Số bù 2: 011011

Đảo bit Giữ nguyên

* **Biểu diễn số có dấu trong hệ bù 2**

+ Nếu là số dương, trị tuyệt đối được biểu diễn theo dạng số nhị phân thực sự và bit dấu 0 được đặt trước MSB

+ Nếu là số âm, trị tuyệt đối được biểu diễn ở dạng bù 2 và bit dấu 1được đặt trước MSB.

**Thí dụ:** 0110100 Biểu diễn giá trị (+52)10

Bit dấu Số nhị phân thực sự

1001100 Biểu diễn giá trị (-52)10

Bit dấu Dạng bù hai của tuyệt đối

Khi chuyển sang dạng bù hai ta thực hiện đối với cả bit dấu thì số bù 2 của một số biểu diễn số âm của số đó.

**Thí dụ:**

(+52)10 = 0110100 bù 2 1001100 = (-52)10

(-52)10 = 1001100 bù 2 0110100 = (+52)10

**1.4.3. Phép trừ**

Bảng thực hiện phép trừ

|  |  |  |  |
| --- | --- | --- | --- |
| x | y | x - y | Mượn |
| 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 0 |

*Nhận xét: -* Nếu số bit 1 là chẵn sẽ cho kết quả là 0

**-** Nếu số bit 1 là lẻ sẽ cho kết quả là 1

**-** Nếu bit 0 trừ cho bit 1 thì phải mượn 1 cho bit nhỏ hơn

**Thí dụ 1:** Tính 1011 - 0101

1011

* 0101

0110

**Thí dụ 2:** Thực hiện phép trừ 1510 – 1310 biểu diễn trong hệ nhị phân 8 bit

*+ Cách 1*: Dựa vào bảng thực hiện phép trừ

1510 = 000011112

* 1310 = 000011012

210  = 000000102

*+ Cách 2*: Với phép trừ: 1510 – 1310 ta lấy bù 2 của (+13)10 để được (- 13)10 và thực hiện phép cộng: 1510 + (– 1310)

Tìm số bù 2 của số 13 bằng số nhị phân 8 bit.

1310 = 000011012

Số bù 1(13) = 11110010

Cộng 1 + 1

11110011 = -13

Thực hiện phép cộng: 1510 = 000011112

+ (-1310) = 111100112

Kết quả là: (+2)10 =100000010

Bit dấu(bỏ) **1.4.4. Phép nhân**

Bảng thực hiện phép nhân

|  |  |  |
| --- | --- | --- |
| x | y | x \* y |
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

*Nhận xét:*

+ Quá trình thực hiện phép nhân thực chất là việc thực hiện phép cộng và phép dịch trái bit.

+ Khi nhân hai giá trị đầu vào có giá trị là 1 thì đầu ra mới ở mức tích cực

+ Quá trình thực hiện phép cộng tương tự bảng đã cho ở trên.

**Thí dụ:** Thực hiện phép nhân( 1210) x (1410)

1210 = 11002

x 1410 = 11102

0000

1100

1100

1100

16810 = 101010002

**1.4.5. Phép chia**

- Phép chia trong hệ nhị phân tương tự như phép chia trong hệ thập phân. Quá trình thực hiện phép chia thực chất là thực hiện phép trừ.

**Thí dụ:** Thực hiện phép chia sau

(4510) : (710)

101101 0111

- 0111 110 = thương số

01000

- 0111

00011 = số dư

**BÀI TẬP**

**Bài 1:** Chuyển đổi các hệ đếm sau

a. 10110111012 → hệ 10

b. 1478 → hệ 10

c. 3AEC16 → hệ 10

d. 4DA16 → hệ 2

e. 6710 → hệ 2

f. 15810 → hệ 8

g. 29710 → hệ 16

**Bài 2:** Thực hiện các phép tính sau trong hệ bù 2

a.(8)10 + (20)10 ; (22)10 + (-26)10 với số bit quy định cho trị tuyệt đối là 5

b. (15)10 - (29)10 ; (-13)10 - (-32)10 với số bit quy định cho trị tuyệt đối là 5

c. (25)10 x (7)10 ; (-11)10 x (-15)10 ; (35)10 x (-16)10

d. (25)10 : (4)10 ; (55)10 : (11)10 ; (67)10 : (8)10

**Bài 3:** Chuyển đổi hệ 2 sang hệ 8 và hệ 16

a. 100110110 → hệ 8

b. 01101001→ hệ 8

c. 110010110110 → hệ 16

d. 1101001010 → hệ 16

**CHƯƠNG 2: ĐẠI SỐ LOGIC**

Năm 1854 **Georges Boole**, một triết gia đồng thời là nhà toán học người Anh cho xuất bản một tác phẩm về lý luận logic, nội dung của tác phẩm đặt ra những mệnh đề mà để trả lời người ta chỉ phải dùng một trong hai từ đúng (có, yes) hoặc sai (không, no).

Tập hợp các thuật toán dùng cho các mệnh đề này hình thành môn Đại số Boole. Đây là môn toán học dùng hệ thống số nhị phân mà ứng dụng của nó trong kỹ thuật chính là các mạch logic, nền tảng của kỹ thuật số.

Chương này không có tham vọng trình bày lý thuyết Đại số Boole mà chỉ giới hạn trong việc giới thiệu các **hàm logic cơ bản** và các **tính chất cần thiết** để giúp sinh viên hiểu vận hành của một hệ thống logic.

**2.1. CÁC KHÁI NIỆM**

**- Trạng thái logic**: là trạng thái của một thực thể. Xét về mặt logic, thực thể chỉ tồn tại ở một trong hai trạng thái.

Thí dụ: Đối với bóng đèn ta chỉ quan tâm tới bóng đèn ở trạng thái sáng hay tắt. Vậy sáng hay tắt là hai trạng thái logic của một thực thể

- **Biến logic**: là đặc trưng cho các trạng thái logic của các thực thể. Người ta biểu diễn biến logic bởi một ký hiệu(chữ hay dấu) và nó chỉ nhận một trong hai giá trị: 0 hoặc 1.

Thí dụ: Trạng thái logic của một công tắc là đóng hoặc mở mà ta có thể đặc trưng bởi trị số 1 hoặc 0.

- **Hàm logic:** diễn tả bởi một nhóm biến logic liên hệ với nhau bởi các phép toán logic. Cũng như biến logic, hàm logic chỉ nhận một trong hai giá trị:0 hoặc 1 tùy theo điều kiện liên quan đến các biến.

Thí dụ: Một mạch điện có nguồn hiệu điện thế cấp cho bóng đèn qua hai công tắc mắc nối tiếp với nhau, bóng đèn chỉ sáng khi hai công tắc đều ở trạng thái đóng. Trạng thái của bóng đèn là một hàm theo hai biến là trạng thái của hai công tắc

|  |  |  |
| --- | --- | --- |
| A | B | Y=f(A,B) |
| 0(hở) | 0(hở) | 0(tắt) |
| 0(hở) | 1(đóng) | 0(tắt) |
| 1(đóng) | 0(hở) | 0(tắt) |
| 1(đóng) | 1(đóng) | 1(sáng) |

**2.2.** **TÍNH CHẤT CỦA CÁC HÀM LOGIC CƠ BẢN**

**2.2.1. Tính chất cơ bản**

* Có một phần tử trung tính cho mỗi toán tử cộng và nhân

A + 0 = A ; 0 là phần tử trung tính của hàm OR

A.1 = A ; 1 là phần tử trung tính của hàm AND

* Tính giao hoán

A + B = B + A

A.B = B.A

* Tính kết hợp

(A +B) + C = A + (B + C)

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

* Tính phân phối

A.(B + C) = A.B + A.C

A + B.C = (A + B)(A + C)

* Không có phép tính lũy thừa và thừa số

A + A +……+ A = A

A.A……A = A

* Tính bù

; ; 

**2.2.2. Tính song đối**

Biểu thức logic vẫn đúng khi [thay phép cộng(+) bằng phép nhân(.) và 0 bởi 1] hay ngược lại. Điều này có thể dễ dàng chứng minh cho các biểu thức ở trên.

  

  

  

**2.2.3. Định lý De Morgan**





**Thí dụ 1**: Chứng minh rằng nếu A, B, C là các biến logic thì



Biến đổi vế trái của biểu thức ta có





**** (vì  và )



**Thí dụ 2**: Chứng minh rằng



Biến đổi vế trái của biểu thức ta có









(vì và )



**2.3. PHƯƠNG PHÁP BIỂU DIỄN HÀM LOGIC**

**2.3.1 Biểu diễn bằng bảng chân lý**

- Biểu diễn hàm logic n biến thì bảng chân lý sẽ có (n + 1)cột trong đó n cột biểu diễn giá trị của biến, cột còn lại biểu diễn giá trị của hàm

- Bảng này liệt kê tất cả các giá trị tổ hợp của biến đầu vào, với mỗi tập giá trị đầu vào thì sẽ tương ứng với một giá trị đầu ra

**Thí dụ 1**: Cho sơ đồ mạch điện sau



Hãy biểu diễn mối quan hệ của các công tắc X1, X2, X3 với bóng đèn Y?

Lập bảng chân lý

|  |  |  |  |
| --- | --- | --- | --- |
| X1 | X2 | X3 | Y |
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 0 |
| 0 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 |
| 1 | 0 | 1 | 1 |
| 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 |

Từ bảng chân lý rút ra được công thức: Y = X1.X2 + X3

**Thí dụ 2**: Lập bảng chân lý thực hiện việc kiểm tra tính chẵn lẻ từ mã nhị phân 4 bit(tức là kiểm tra số bit 1 ở trong từ mã là chẵn hay lẻ)

Gọi Y là hàm logic xác định tính chẵn lẻ

Y = 0 là từ mã chẵn

Y = 1 là từ mã lẻ

Dựa vào yêu cầu kiểm tra tính chẵn lẻ của từ mã nhị phân mà ta xây dựng được bảng chân lý sau:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | X1 | X2 | X3 | X4 | Y |
| 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 1 | 1 |
| 2 | 0 | 0 | 1 | 0 | 1 |
| 3 | 0 | 0 | 1 | 1 | 0 |
| 4 | 0 | 1 | 0 | 0 | 1 |
| 5 | 0 | 1 | 0 | 1 | 0 |
| 6 | 0 | 1 | 1 | 0 | 0 |
| 7 | 0 | 1 | 1 | 1 | 1 |
| 8 | 1 | 0 | 0 | 0 | 1 |
| 9 | 1 | 0 | 0 | 1 | 0 |
| 10 | 1 | 0 | 1 | 0 | 0 |
| 11 | 1 | 0 | 1 | 1 | 1 |
| 12 | 1 | 1 | 0 | 0 | 0 |
| 13 | 1 | 1 | 0 | 1 | 1 |
| 14 | 1 | 1 | 1 | 0 | 1 |
| 15 | 1 | 1 | 1 | 1 | 0 |

Từ bảng chân lý, biểu diễn bằng số thứ tự của tập đầu vào mà đầu ra có giá trị bằng 1 trong bảng chân lý: y = (1, 2, 4, 7, 8, 11, 13, 14)

**2.3.2. CÁC PHƯƠNG PHÁP BIỂU DIỄN HÀM LOGIC**

**2.3.2.1 Phương pháp biểu diễn hàm logic bằng dạng tổng của các tích**

Công thức tổng quát:



trong đó: *km*= x1.x2…xn với m = 0÷ 2n - 1

ym là giá trị đầu ra ứng với tổ hợp giá trị đầu vào tại giá trị thứ m

*\* Quy tắc viết hàm logic dạng tổng của các tích.*

- Hàm logic dạng tổng của các tích chỉ quan tâm đến các tổ hợp biến mà đầu ra có giá trị bằng 1

- Trong mỗi tổ hợp biến mà đầu ra có giá trị bằng 1, nếu biến *xi* = 0 thì viết  còn nếu *xi*= 1 thì viết *xi* .Khi đó hàm y sẽ là tổng của các tổ hợp biến đầu vào có giá trị đầu ra bằng 1

**Thí dụ** : Cho hàm 4 biến sau

y = (1, 2, 4, 7, 8, 11, 13, 14)

Biểu diễn hàm logic bằng dạng tổng của các tích?

Dựa vào bảng chân lý từ ví dụ 2 ta được





**2.3.2.2 Phương pháp biểu diễn hàm logic bằng dạng tích của các tổng**

Công thức tổng quát:



Trong đó: *hm =* (x1 + x2 …+xn)

*\* Quy tắc viết hàm logic dạng tích của các tổng.*

- Hàm logic dạng tích của các tổng chỉ quan tâm đến các tổ hợp biến mà đầu ra có giá trị bằng 0

- Trong mỗi tổ hợp biến mà đầu ra có giá trị bằng 0, nếu biến *xi* = 0 thì viết  còn nếu *xi*= 1 thì viết  .Khi đó hàm y sẽ là tích của các tổ hợp biến đầu vào có giá trị đầu ra bằng 0

**Thí dụ** **1**: Cho hàm 4 biến sau

y = (1, 2, 4, 7, 8, 11, 13, 14)

Biểu diễn hàm logic bằng dạng tích của các tổng?

Dựa vào bảng chân lý từ ví dụ 2 ta được:





**2.3.3. Đơn giản hóa hàm logic**

Để thực hiện một hàm logic bằng mạch điện tử, người ta luôn luôn nghĩ đến việc sử dụng lượng linh kiện ít nhất. Muốn vậy, hàm logic phải ở dạng tối giản, nên vấn đề rút gọn hàm logic là **bước đầu tiên phải thực hiện** trong quá trình thiết kế. Có 2 phương pháp rút gọn hàm logic:

- Phương pháp đại số

- Phương pháp dùng bảng Karnaugh

**2.3.3.1. Đơn giản hóa hàm logic bằng phương pháp đại số.**

Phương pháp này bao gồm việc áp dụng các tính chất của hàm logic cơ bản. Một số đẳng thức thường sử dụng được nhóm lại như sau:

(1) và  (1’)

(2)  và  (2’)

(3)  và  (3’)

Chứng minh các đẳng thức (1), (2), (3)

(1)

(2) 

(3) 

Các đẳng thức (1’), (2’), (3’) là song đối của (1), (2), (3).

**Các quy tắc rút gọn:**

**\* Quy tắc 1**: Dựa vào các đẳng thức trên nhóm các số hạng lại với nhau.

**Thí dụ**: Rút gọn biểu thức 

Theo (1): 

Vậy 

Theo (3): 

Vậy kết quả cuối cùng là: 

**\* Quy tắc 2**: Ta có thể thêm một số hạng đã có trong biểu thức logic vào biểu thức mà không làm thay đổi tính chất của biểu thức.

**Thí dụ**: Rút gọn biểu thức 

Thêm giá trị *ABC* vào biểu thức: 

Theo (1) các nhóm trong dấu ngoặc được rút gọn thành 

Vậy 

**\* Quy tắc 3**: Có thể bỏ số hạng chứa các biến đã có trong số hạng khác

**Thí dụ 1:** Rút gọn biểu thức 

Biểu thức không đổi nếu ta nhân một số hạng trong biểu thức với 1, ví dụ :



Triển khai số hạng cuối cùng của vế phải, ta được:



Thừa số chung: 

Tóm lại: 

Trong bài tóan này ta đã đơn giản được số hạng AC

**Thí dụ 2:** Rút gọn biểu thức 

Biểu thức không đổi nếu ta thêm vào một thừa số có trị =0, ví dụ 





Theo (2’) và 

Vậy 

Trong bài tóan này ta đã bỏ số hạng A+C

**\* Quy tắc 4**: Có thể đơn giản bằng cách sử dụng phương pháp biểu diễn hàm logic có chứa số hạng ít nhất.

**Thí dụ:** Hàm f(A, B, C) = 

với trọng số tương ứng là A = 4, B = 2, C = 1.

f(A, B, C) = = 

Vậy f(A, B, C) = 

**2.3.3.2. Đơn giản hóa hàm logic bằng phương pháp bản đồ Karnaugh.**

Trong thực tế khi giải các bài toán kỹ thuật có một vấn đề cần phải quan tâm là làm thế nào để có thể dễ dàng thực hiện các sơ đồ mạch điện nhờ các phần tử logic sơ cấp. Phương pháp rút gọn hàm logic bằng phương pháp đại số có nhược điểm chỉ áp dụng với những bài toán có số lượng ít biến đầu vào. Vì vậy phương pháp bản đồ Karnaugh cho phép rút gọn dễ dàng hàm logic chứa từ 3 đến 6 biến.

**a. Nguyên tắc**

Xét hai tổ hợp biến *AB* và , hai tổ hợp này chỉ khác nhau ở một bit, ta gọi chúng là hai tổ hợp kề nhau.

Ta có:  biến B đã được đơn giản

Phương pháp bản đồ Karnaugh dựa vào việc nhóm các tổ hợp kề nhau trên bảng để đơn giản biến có giá trị khác nhau trong các tổ hợp này.

*\* Quy tắc rút gọn hàm logic được thực hiện theo bốn bước sau:*

+ Vẽ bảng Karnaugh theo số biến của hàm

+ Chuyển hàm cần đơn giản vào bảng Karnaugh

+ Khoanh vòng các ô chứa các tổ hợp kề nhau hoặc đối xứng nhau lại thành các nhóm sao cho có thể rút gọn hàm tới mức tối giản

+ Viết kết quả hàm rút gọn từ các nhóm đã khoanh vòng.

**b. Vẽ bảng Karnaugh**

**-** Bảng Karnaugh thực chất là một dạng khác của bảng sự thật, trong đó mỗi ô của bảng tương đương với một hàng trong bảng sự thật.

**-** Với hàm logic n biến, bảng Karnaugh gồm 2n ô, mỗi ô tương ứng với tổ hợp biến này. Các ô trong bảng được sắp đặt sao cho hai ô kề nhau chỉ khác nhau ở một bit

**-** Với hai biến AB, sự sắp đặt sẽ theo thứ tự AB = 00, 01, 11, 10 nhưng để cho dễ dàng dùng các chữ số để đọc thứ tự này: 0, 1, 3, 2

\* *Với hàm hai biến f(A, B)*:



\* *Với hàm ba biến f(A, B, C)*:

****

\* *Với hàm ba biến f(A, B, C, D)*



**c. Chuyển hàm logic vào bảng Karnaugh.**

Trong mỗi ô của bảng đưa vào giá trị của hàm tương ứng với tổ hợp biến , để đơn giản ta chỉ ghi các trị 1 mà bỏ qua trị 0 của hàm. Ta có các trường hợp sau:

***Trường hợp 1.* Các tổ hợp cho giá trị hàm xác định**

Các tổ hợp có giá trị đầu ra của hàm đã cho có giá trị bằng 1, giá trị không cho có giá trị bằng 0.

**Thí dụ 1**: f(A,B,C) = 



**Thí dụ 2:** f(A,B,C,D) = 

****

***Trường hợp 2.* Một số tổ hợp có giá trị hàm không xác định**

Các tổ hợp này hàm có thể có giá trị 1 hoặc 0, khi đó ghi dấu X vào các ô tương ứng với các tổ hợp này, lúc gom nhóm giá trị không xác định X như giá trị 1 hay 0 một cách tùy ý sao cho có được kết quả rút gọn nhất.

**Thí dụ:** f(A,B,C,D) = Σ(3,4,5,6,7) với các tổ hợp từ 10 dến 15 cho hàm có trị bất kỳ (không xác định).



**d. Quy tắc khoanh vòng.**

Các tổ hợp biến có trong hàm logic hiện diện trong bảng Karnaugh dưới dạng các số 1 trong các ô, vì vậy việc gom nhóm các tổ hợp lại với nhau được thực hiện theo quy tắc:

* Khoanh vòng số 1 không thể kết hợp với ô nào khác
* Khoanh vòng các số 1 kề nhau hoặc đối xứng nhau thành từng nhóm sao cho số nhóm là ít nhất.
* Khoanh vòng các số 1 lại thành nhóm sao cho là bội của 2k(mỗi nhóm có thể có 1, 2, 4, 8… số 1). Cứ mỗi nhóm chứa 2k số 1 thì tổ hợp biến tương ứng với nhóm đó giảm đi k biến đầu vào.

**Thí dụ:** Cho hàm logic có bốn đầu vào

f(A,B,C,D) = Σ(0,1,3,4,5,6,9,11, 13,15)

Quy tắc khoanh vòng có dạng như sau

|  |  |
| --- | --- |
|  |  |

**e. Kết quả rút gọn.**

* Kết quả cuối cùng là tổng các tích của các nhóm đã khoanh vòng.
* Mỗi số hạng của tổng tương ứng với một nhóm các số 1 đã khoanh vòng và số hạng này là tích của các biến.

**Thí dụ**: f(A,B,C,D) = Σ(0,1,3,4,5,6,9,11, 13,15)

Rút gọn hàm logic trên?

****

Dựa vào hình trên thu được kết quả như sau:

**; ; ; **

Suyra : ****

**BÀI TẬP**

**Bài 1:** Chứng minh bằng đại số các biểu thức sau

a) 

b) 

c) 

d) 

e) 

f) 

g) 

**Bài 2:** Rút gọn hàm dưới đây bằng phương pháp đại số

a) 

b) 

c) 

d) 

e) 

f) 

**Bài 3:** Viết hàm logic dưới dạng tổng của các tích của các hàm sau:

a/ f(A,B,C) = 1 nếu số nhị phân (ABC)2 là số chẵn

b/ f(A,B,C) = 1 nếu có ít nhất 2 biến số = 1

c/ f(A,B,C) = 1 nếu số nhị phân (ABC)2 >5

d/ f(A,B,C) = 1 nếu số biến số 1 là số chẵn

e/ f(A,B,C) = 1 nếu có 1 và chỉ 1 biến số =1

**Bài 4:** Viết hàm logic dưới dạng tích của các tổng của các hàm trên

**Bài 5:** Dùng bảng Karnaugh rút gọn các hàm sau

a/ f(A,B,C) = Σ(1,3,4)

b/ f(A,B,C) = Σ(1,3,7)

c/ f(A,B,C) = Σ(0,3,4,6,7)

d/ f(A,B,C) = Σ(1,3,4) . Các tổ hợp biến 6,7 cho hàm không xác định

e/ 

f/ f(A,B,C,D) = Σ(0,1,2,4,5,7,8,10,12,14)

g/ f(A,B,C,D) = Σ(0,2,3,5,6,8,10,11,13,15)

h/ f(A,B,C,D) = Σ(0,1,2,3,4,6,8,10,12,14)

i/ f(A,B,C,D) = Σ(0,2,3,5,7,9,11,12,14,15)

j/ f(A,B,C,D) = Π(0,1,3,5,7,8,10,11,13,15)

**CHƯƠNG 3: CỔNG LOGIC**

Cổng logic là tên gọi chung của các mạch điện tử có chức năng thực hiện các hàm logic. Cổng logic có thể được chế tạo bằng các công nghệ khác nhau (lưỡng cực, MOS), có thể được tổ hợp bằng các linh kiện rời nhưng thường được chế tạo bởi công nghệ tích hợp IC (Integrated circuit).

Chương này giới thiệu các loại cổng cơ bản, các họ IC số, các tính năng kỹ thuật và sự giao tiếp giữa chúng.

**3.1. CÁC CỔNG LOGIC CƠ BẢN.**

**3.1.1. Cổng AND**

**-** *Định nghĩa:*Là cổng có từ hai đầu vào trở lên và một đầu ra bằng tổ hợp AND các biến đầu vào.

- Ký hiệu cổng AND có hai ngõ vào cho hai biến

X1

X2

Y

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Bảng chân lý   |  |  |  | | --- | --- | --- | | X1 | X2 | Y | | 0  0  1  1 | 0  1  0  1 | 0  0  0  1 | | Mạch điện thực hiện quan hệ logic AND |

*Nhận xét: +*Ngõ ra cổng AND bằng 0 khi có vào ít nhất một ngõ vào bằng 0.

+ Ngõ ra cổng AND bằng 1 khi tất cả các ngõ vào bằng 1.

- Giản đồ thời gian:



- Biểu thức: Y = X1.X2

­- Trong thực tế, các loại IC chứa cổng AND là: IC7408, IC7409, IC 7411

**3.1.2. Cổng OR**

**-** *Định nghĩa:*Là cổng có từ hai đầu vào trở lên và một đầu ra bằng tổ hợp OR các biến đầu vào.

- Ký hiệu cổng OR có hai ngõ vào cho hai biến

Y

X1

X2

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Bảng chân lý   |  |  |  | | --- | --- | --- | | X1 | X2 | Y | | 0  0  1  1 | 0  1  0  1 | 0  1  1  1 | | Mạch điện thực hiện quan hệ logic OR |

*Nhận xét: +*Ngõ ra cổng OR bằng 1 khi có vào ít nhất một ngõ vào bằng 1.

+ Ngõ ra cổng OR bằng 0 khi tất cả các ngõ vào bằng 0.

- Giản đồ thời gian:



- Biểu thức: Y = X1 + X2

­- Loại IC chứa cổng OR là: IC7432

**3.1.3. Cổng NOT**

**-** *Định nghĩa:*Là cổng duy nhất một đầu vào trạng thái logic và đầu ra luôn ngược với đầu vào

****- Ký hiệu cổng NOT

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Bảng chân lý   |  |  | | --- | --- | | X | Y | | 0  1 | 1  0 | | Mạch điện thực hiện quan hệ logic NOT |

*Ghi chú: Nút nhấn X là loại thường đóng*

*Nhận xét:* Trạng thái ngõ vào và ra của cổng NOT luôn đối nhau.

- Giản đồ thời gian:



- Biểu thức: 

­- Loại IC chứa cổng NOT là: IC7404, IC7405

**3.1.4. Cổng NAND**

**-** *Định nghĩa:*Là mạch có từ hai đầu vào trở lên và một đầu ra bằng tổ hợp NAND các biến đầu vào. Kết hợp cổng AND với cổng NOT ta sẽ được cổng NAND.

- Ký hiệu cổng NAND có hai ngõ vào cho hai biến

X1

X2

Y

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Bảng chân lý   |  |  |  | | --- | --- | --- | | X1 | X2 | Y | | 0  0  1  1 | 0  1  0  1 | 1  1  1  0 | | Mạch điện thực hiện quan hệ logic NAND |

*Nhận xét: +* Trạng thái ngõ ra của cổng NAND là đảo của ngõ ra cổng AND.

*+*Ngõ ra cổng NAND bằng 1 khi có vào ít nhất một ngõ vào bằng 0.

+ Ngõ ra cổng NAND bằng 0 khi tất cả các ngõ vào bằng 1.

- Giản đồ thời gian:



- Biểu thức: 

­- Các loại IC chứa cổng NAND là: IC7400, IC7401, IC 7412, IC 7413

**3.1.5. Cổng NOR**

**-** *Định nghĩa:*Là mạch có từ hai đầu vào trở lên và một đầu ra bằng tổ hợp NOR các biến đầu vào. Kết hợp cổng OR với cổng NOT ta sẽ được cổng NOR.

- Ký hiệu cổng NOR có hai ngõ vào cho hai biến

Y

X1

X2

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Bảng chân lý   |  |  |  | | --- | --- | --- | | X1 | X2 | Y | | 0  0  1  1 | 0  1  0  1 | 1  0  0  0 | | Mạch điện thực hiện quan hệ logic NOR |

*Nhận xét: +* Trạng thái ngõ ra của cổng NOR là đảo của ngõ ra cổng OR

*+*Ngõ ra cổng NOR bằng 0 khi có vào ít nhất một ngõ vào bằng 1.

+ Ngõ ra cổng NOR bằng 1 khi tất cả các ngõ vào bằng 0.

- Giản đồ thời gian:



- Biểu thức: 

­- Loại IC chứa cổng OR là: IC7402, IC7427

**3.1.6. Cổng XOR**

**-** Hàm chứa cổng XOR gọi là hàm hoặc loại trừ hay hàm hoặc tuyệt đối.

Ký hiệu cổng XOR hai ngõ vào Bảng chân lý

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Y  X1  X2 | |  |  |  | | --- | --- | --- | | X1 | X2 | Y | | 0  0  1  1 | 0  1  0  1 | 0  1  1  0 | |

*Nhận xét: +*Ngõ ra cổng XOR bằng 0 khi các ngõ vào cùng trạng thái

+ Ngõ ra cổng XOR bằng 1 khi các ngõ vào khác trạng thái.

- Giản đồ thời gian:



- Biểu thức: 

­- Loại IC chứa cổng XOR là: IC7486

**3.1.7. Cổng X-NOR**

**-** Hàm chứa cổng X-NOR gọi là hàm không hoặc loại trừ hay hàm không hoặc tuyệt đối.

Ký hiệu cổng X-NOR hai ngõ vào Bảng chân lý

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Y  X1  X2 | |  |  |  | | --- | --- | --- | | X1 | X2 | Y | | 0  0  1  1 | 0  1  0  1 | 1  0  0  1 | |

*Nhận xét: +*Ngõ ra cổng X-NOR bằng 1 khi các ngõ vào cùng trạng thái

+ Ngõ ra cổng X-NOR bằng 0 khi các ngõ vào khác trạng thái.

- Giản đồ thời gian:



- Biểu thức: 

­- Loại IC chứa cổng X-NOR là: IC74LS266

**3.2 THÔNG SỐ KỸ THUẬT CỦA IC SỐ**

Để sử dụng các IC số có hiệu quả, ngoài sơ đồ chân và bảng sự thật cần thiết chúng ta nên biết một số thuật ngữ chỉ các thông số kỹ thuật và đặc tính của IC.

**3.2.1 Các đại lượng điện đặc trưng.**

- **VCC**: Điện thế nguồn (power supply): khoảng điện thế cho phép cấp cho IC để hoạt động tốt. Thí dụ với IC số họ TTL, **VCC** =5±0,5 V , họ CMOS **VDD**=3-15V (Người ta thường dùng ký hiệu **VDD** và **VSS** để chỉ nguồn và mass của IC họ MOS)

- **VIH** (min): Điện thế ngõ vào mức cao (High level input voltage): Điện thế ngõ vào nhỏ nhất tương ứng với mức logic 1.

- **VIL** (max): Điện thế ngõ vào mức thấp (Low level input voltage): Điện thế ngõ vào lớn nhất tương ứng với mức logic 0.

- **VOH** (min): Điện thế ngõ ra mức cao (High level output voltage): Điện thế nhỏ nhất của ngõ ra khi ở mức cao.

- **VOL**(max): Điện thế ngõ ra mức thấp (Low level output voltage): Điện thế lớn nhất của ngõ ra khi ở mức thấp.

- **IIH**: Dòng điện ngõ vào mức cao (High level input current): Dòng điện ngõ vào IC lớn nhất khi ngõ vào ở mức cao.

- **IIL**: Dòng điện ngõ vào mức thấp (Low level input current) : Dòng điện ngõ vào IC lớn nhất khi ngõ vào ở mức thấp

- **IOH**: Dòng điện ngõ ra mức cao (High level output current): Dòng điện lớn nhất ngõ ra có thể cấp cho tải khi ngõ ra ở mức cao.

- **IOL**: Dòng điện ngã ra mức thấp (Low level output current): Dòng điện lớn nhất ngõ ra có thể nhận khi ở mức thấp.

- **ICCH**, **ICCL** dòng điện chạy qua IC khi ngõ ra lần lượt ở mức cao và thấp.

*Ngoài ra còn một số thông số khác được nêu ra dưới đây*

**3.2.2 Công suất tiêu tán (Power requirement)**

Khi hoạt động dòng qua IC thường xuyên thay đổi giữa hai trạng thái cao và thấp vì vậy mỗi linh kiện sẽ tiêu thụ một công suất nhất định từ nguồn cung cấp VCC (hay VDD ) . Công suất tiêu tán này xác định bởi điện thế nguồn và dòng điện qua IC và công suất tính được là công suất tiêu tán trung bình.

PD(avg) = ICC(avg).VCC

Với 

Đối với các cổng logic họ TTL, công suất tiêu tán ở hàng mW và với cổng logic họ MOS thì ở hàng nW.

**3.3.3 Fan-Out**

Ngõ ra của một mạch logic đòi hỏi phải cấp dòng cho một số ngõ vào các mạch logic khác. Fan Out là số ngõ vào lớn nhất có thể nối với ngõ ra của một IC cùng loại mà vẫn bảo đảm mạch hoạt động bình thường. Nói cách khác, Fan Out chỉ khả năng chịu tải đầu ra của một cổng logic.

**Phân loại:** Fan-Out có hai loại tương ứng với 2 trạng thái logic của ngõ ra:





Thông thường hai giá trị Fan-Out này có giá trị khác nhau vì vậy khi sử dụng để đảm bảo an toàn cho IC sử dụng trị nhỏ nhất trong hai trị này.

Fan-Out được tính theo đơn vị Unit Load UL (tải đơn vị).

**3.2.4 Thời trễ truyền (Propagation delays)**

Tín hiệu logic khi truyền qua một cổng luôn có một khoảng thời gian trễ.

**Phân loại:***Có hai loại thời trễ truyền:*

- Thời trễ truyền từ thấp lên cao tPLH

- Thời trễ truyền từ cao xuống thấp tPHL.

Hai giá trị này thường khác nhau. Sự thay đổi trạng thái được xác định ở tín hiệu ra. Thí dụ tín hiệu qua một cổng đảo, thời trễ truyền được xác định như ở (H 3.14)

Tùy theo họ IC, thời trễ truyền thay đổi tử vài ns đến vài trăm ns. Thời trễ truyền càng lớn thì tốc độ làm việc của IC càng nhỏ và ngược lại.

**3.2.5 Tích số công suất-vận tốc (speed- power product)**

Quá trình đánh giá chất lượng IC dựa vào đại lượng tích số công suất-vận tốc đó là tích số công suất tiêu tán và thời trễ truyền.

Thí dụ họ IC có thời trễ truyền là 10 ns và công suất tiêu tán trung bình là 50 mW thì tích số công suất-vận tốc là:

10 ns x 5 mW =10.10-9x5.10-3 = 50x10-12 watt-sec = 50 picojoules (pj)

Trong quá trình chế tạo IC, người ta luôn muốn đạt được các IC có công suất tiêu tán và thời trễ truyền càng nhỏ càng tốt. Như vậy một IC có chất lượng càng tốt khi tích số công suất-vận tốc càng nhỏ. Tuy nhiên trên thực tế hai giá trị này thay đổi theo chiều ngược với nhau do đó rất khó đạt được các giá trị theo ý muốn, dù sao trong quá trình phát triển của công nghệ chế tạo linh kiện điện tử trị số này luôn được cải thiện.

**3.2.6 Tính miễn nhiễu (Noise immunity)**

Các tín hiệu nhiễu như tia lửa điện, cảm ứng từ, môi trường làm việc có thể làm thay đổi trạng thái logic của tín hiệu do đó ảnh hưởng đến kết quả hoạt động của mạch.

Tính miễn nhiễu của một mạch logic tùy thuộc khả năng dung nạp hiệu thế nhiễu của mạch và được xác định bởi lề nhiễu. Lề nhiễu có được do sự chênh lệch của các điện thế giới hạn (còn được gọi là ngưỡng logic) của mức cao và thấp giữa ngõ ra và ngõ vào của các cổng (H 3.15).

- Lề nhiễu mức cao: VNH = VOH(min) - VIH(max).

- Lề nhiễu mức thấp: VNL = VIL(max) - VOL(min).

Khi tín hiệu ra ở mức cao đưa vào ngõ vào, bất cứ tín hiệu nhiễu nào có giá trị âm và biên độ >VNH đều làm cho điện thế ngõ vào rơi vào vùng bất định và mạch không nhận ra được tín hiệu thuộc mức logic nào. Tương tự cho trường hợp ngõ ra ở mức thấp tín hiệu nhiễu có giá trị dương và biên độ >VNL sẽ đưa mạch vào trạng thái bất định.

**3.3 THÔNG SỐ CÁC HỌ LOGIC**

Trong quá trình phát triển của công nghệ chế tạo mạch số gồm có các họ: RTL (Resistor – Transistor Logic), DCTL (Direct Couple – Transistor Logic), RCTL (Resistor – Capacitor - Transistor Logic), DTL (Diod – Transistor Logic), ECL (Emitter – Couple Logic) v.v.... Hiện tại đang tồn tại hai họ logic có nhiều tính năng kỹ thuật cao như thời trễ truyền nhỏ, tiêu hao công suất ít, đó là họ TTL (Transistor – Transistor Logic) dùng công nghệ chế tạo BJT và họ MOS (Công nghệ chế tạo MOS)

Dưới đây, lần lượt khảo sát các cổng logic của hai họ TTL và MOS

**3.3.1 Họ TTL (Transistor - Transistor Logic)**

Các IC số họ TTL được sản xuất lần đầu tiên vào năm 1964 bởi hãng Texas Instrument Corporation của Mỹ, lấy số hiệu là 74XXXX và 54XXXX. Sự khác biệt giữa 2 họ 74XXXX và 54 XXXX chỉ ở hai điểm:

74: Vcc=5 ± 0,5 V và khoảng nhiệt độ hoạt động từ 0o C đến 70o C

54: Vcc=5 ± 0,25 V và khoảng nhiệt độ hoạt động từ -55o C đến 125o C

Trước số 74 thường có thêm ký hiệu để chỉ hãng sản xuất. Thí dụ SN của hãng Texas, DM của National Semiconductor, S của Signetics

Ngoài ra trong quá trình phát triển, các thông số kỹ thuật (nhất là tích số công suất vận tốc) luôn được cải tiến và ta có các loạt khác nhau: 74 chuẩn, 74L (Low power), 74 H (High speed), 74S (Schottky), 74LS (Low power Schottky), 74AS (Advance Schottky), 74ALS (Advance Low power Schottky), 74F (Fast, Fair Child).

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Thông số kỹ thuật | 74 | 74L | 74H | 74S | 74LS | 74AS | 74ALS | 74F |
| Thời trễ truyền(ns)  Công suất tiêu tán(mW)  Tích số công suất vận tốc (pJ)  Tần số xung Ck max(MHz)  Fan out (cùng loại)  **Điện thế**  VOH(min)  VOL(max)  VIH(min)  VIL(max) | 9  10  90  35  10  2,4  0,4  2,0  0,8 | 33  1  33  3  20  2,4  0,4  2,0  0,7 | 6  23  138  50  10  2,4  0,4  2,0  0,8 | 3  20  60  125  20  2,7  0,5  2,0  0,8 | 9,5  2  19  45  20  2,7  0,5  2,0  0,8 | 1,7  8  13,6  200  40  2,5  0,5  2,0  0,8 | 4  1,2  4,8  70  20  2,5  0,4  2,0  0,8 | 3  6  18  100  33  2,5  0,5  2,0  0,8 |

- Loạt 74S: Các transistor trong mạch được mắc thêm một Diod Schottky giữa hai cực CB với mục đích giảm thời gian chuyển trạng thái của transistor do đó làm giảm thời trễ truyền.

- Loạt 74AS và 74ALS được cải tiến từ 74S để làm giảm hơn nữa giá trị tích số Công suất - Vận tốc.

- Loạt 74F: Dùng trong kỹ thuật đặc biệt là làm giảm diện dung ký sinh vì vậy cải thiện thời trễ truyền của cổng.

**3.3.2 Họ MOS.**

Gồm các IC số dùng công nghệ chế tạo của transistor MOSFET loại tăng, kênh N và kênh P .

Với transistor kênh N là loại NMOS, transistor kênh P là loại PMOS và kết hợp cả hai loại transistor kênh P và N là loại CMOS. Tính năng kỹ thuật của loại NMOS và PMOS gần như giống nhau ngoại trừ nguồn cấp điện ngược chiều nhau vì vậy ở đây chỉ xét loại NMOS và CMOS

Các transistor MOS dùng trong IC số cũng chỉ hoạt động ở một trong 2 trạng thái: dẫn hoặc ngưng.

- Khi dẫn, tùy theo nồng độ pha của chất bán dẫn mà transistor có nội trở rất nhỏ (từ vài chục Ω đến hàng trăm KΩ) tương đương với một khóa đóng.

- Khi ngưng, transistor có nội trở rất lớn (hàng 1010Ω), tương đương với một khóa hở.

**BÀI TẬP**

**Bài 1**: Vẽ mạch sử dụng cổng AND, OR, NOT sau

a. 

b. 

c. 

**Bài 2:** Thực hiện các hàm logic sau

**** ;  ****

a. Bằng cổng NOR hai đầu vào

b. Bằng cổng NOR có số đầu vào tùy ý

**Bài 3:** Thực hiện các hàm logic sau

**** ;  ****

a. Bằng cổng NAND hai đầu vào

b. Bằng cổng NAND có số đầu vào tùy ý

**Bài 4:** Vẽ sơ đồ mạch logic thực hiện hàm logic sau:

****

a. Chỉ dùng mạch NAND 7400

b. Chỉ dùng mạch NOR 7402

**Bài 5:** Cho mạch hình a

|  |  |
| --- | --- |
|  |  |

*Hình a Hình b*

a. Viết biểu thức hàm Y theo các biến A, B, C

b. Rút gọn hàm logic này

c.Thay thế mạch trên bằng một mạch chỉ gồm cổng NAND hai ngõ vào

**Bài 6:** Cho mạch hình b

a. Viết biểu thức hàm Y theo các biến A, B, C

b. Rút gọn hàm logic này

c.Thay thế mạch trên bằng một mạch chỉ gồm cổng NAND hai ngõ vào

**CHƯƠNG 4: MẠCH TỔ HỢP**

Các mạch số được chia ra làm hai loại: **Mạch tổ hợp** và **Mạch tuần tự**.

- Mạch tổ hợp: Trạng thái ngõ ra chỉ phụ thuộc vào tổ hợp các biến ngõ vào ổn định. Ngõ ra Q của mạch tổ hợp là hàm logic của các biến ngõ vào A, B, C . . ..

Q = f(A,B,C . . .)

- Mạch tuần tự : Trạng thái ngõ ra không những phụ thuộc vào tổ hợp các biến ngõ vào mà còn phụ thuộc trạng thái ngõ ra trước đó. Ta nói mạch tuần tự có tính nhớ. Ngõ ra Q+ của mạch tuần tự là hàm logic của các biến ngõ vào A, B, C . . . . và ngõ ra Q trước đó.

Q+ = f(Q,A,B,C . . .)

Chương này nghiên cứu một số mạch tổ hợp thông dụng thông qua việc thiết kế một số mạch đơn giản và khảo sát một số IC trên thực tế. Với một mạch logic tổ hợp bất kỳ nếu cho trước chức năng ta đều có thể thiết kế và thực hiện được. Quá trình thiết kế bao gồm các bước tiến hành sau:

1. Từ yêu cầu chức năng, lập bảng chân lý cho hàm.
2. Xây dựng phương trình logic từ bảng chân lý
3. Đơn giản hóa hàm logic.
4. Thiết kế mạch logic bằng các phần từ logic cơ bản.

**4.1 MẠCH MÃ HÓA VÀ MẠCH GIẢI MÃ**

**4.1.1 Mạch mã hóa**

Bộ mã hóa có M = 2n đầu vào và chỉ một trong số đó được kích hoạt tại thời điểm xác định tạo ra số tổ hợp mã đầu ra N = n bit tùy thuộc đầu vào nào được kích hoạt

Sơ đồ tổng quát của bộ mã hóa có M đầu vào và N đầu ra tích cực ở mức cao như hình 4.1 sau

Bộ

mã

hóa

Đầu vào

có M = 2n

Đầu ra

N = n bit

*Hình 4.1: Sơ đồ tổng quát bộ mã hóa*

- Mô hình mạch có ngõ vào và ra tác động ở mức cao (hình 4.2.a): Khi các ngõ vào đều ở mức thấp mạch chưa hoạt động, các ngõ ra đều ở mức thấp. Khi có một ngõ vào được tác động bằng cách ấn khóa K tương ứng để đưa ngõ vào đó lên mức cao, các ngõ ra sẽ cho số nhị phân tương ứng.

- Mô hình mạch có ngõ vào và ra tác động ở mức thấp (hình 4.2.b): Hoạt động tương tự như mạch trên nhưng có mức tác động ngược lại

Mã

hóa

2n n

Mã hóa

2n n

*Hình 4.2a Hình 4.2b*

Dĩ nhiên, người ta cũng có thể thiết kế theo kiểu ngõ vào tác động thấp và ngõ ra tác động cao hay ngược lại. Trên thực tế, có thể bất cứ loại ngõ vào hay ra tác động theo kiểu mức cao hay thấp.

Ngoài ra, để tránh trường hợp mạch cho ra một mã sai khi người sử dụng vô tình (hay cố ý) tác động đồng thời vào hai hay nhiều ngõ vào, người ta thiết kế các mạch mã hóa ưu tiên: là mạch chỉ cho ra một mã duy nhất có tính ưu tiên khi có nhiều ngõ vào cùng được tác động.

**4.1.1.1. Bộ mã hóa bát phân thành nhị phân**

Bộ mã hóa có 8 tín hiệu đầu vào(I0 ÷ I7) tương ứng với 8 ký số trong hệ bát phân và tạo mã đầu ra 3 bit (O2 ÷ O0) tương ứng với các đầu vào được kích hoạt.

- Bảng chân lý

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| I0 | I1 | I2 | I3 | I4 | I5 | I6 | I7 | O2 | O1 | O0 |
| 1  0  0  0  0  0  0  0 | 0  1  0  0  0  0  0  0 | 0  0  1  0  0  0  0  0 | 0  0  0  1  0  0  0  0 | 0  0  0  0  1  0  0  0 | 0  0  0  0  0  1  0  0 | 0  0  0  0  0  0  1  0 | 0  0  0  0  0  0  0  1 | 0  0  0  0  1  1  1  1 | 0  0  1  1  0  0  1  1 | 0  1  0  1  0  1  0  1 |

- Phương trình logic:

O2 = I4 + I5 + I6 + I7

O1 = I2 + I3 + I6 + I7

O0 = I1 + I3 + I5 + I7

- Thiết kế sơ đồ mạch logic như hình 4.3



*Hình 4.3: Sơ đồ mạch mã hóa bát phân thành nhị phân*

Từ sơ đồ trên nhận thấy, tại mỗi thời điểm chỉ được phép kích hoạt mỗi đầu vào được một lần, nếu tại mỗi thời điểm kích hoạt từ hai đầu vào trở lên thì đầu ra sẽ nhận một giá trị bất kỳ nào đó không xác định được trước.

**Thí dụ**: Kích hoạt I3 và I5 cùng một lúc thì đầu ra sẽ có giá trị là 111. Rõ ràng đây không phải là mã cho cả hai đầu vào được kích hoạt

*Để khắc phục nhược điểm này người ta dùng bộ mã hóa ưu tiên*

**4.1.1.2. Bộ mã hóa ưu tiên thập phân thành BCD**

Bộ mã hóa ưu tiên được xác định theo thứ tự ưu tiên từ cao xuống thấp. Nếu có nhiều tín hiệu đồng thời xuất hiện ở đầu vào thì chỉ có tín hiệu nào có mức ưu tiên cao nhất trong các tín hiệu đó mới được mã hóa.

Bộ mã hóa thập phân gồm 10 đầu vào(I0 ÷ I9) tương ứng với 10 chữ số thập phân và 4 đầu ra(O3 ÷ O0) tạo thành mã nhị phân 4 bit.

- Bảng chân lý

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| I9 | I8 | I7 | I6 | I5 | I4 | I3 | I2 | I1 | I0 | ­O3 | O2 | O1 | O0 |
| 0  0  0  0  0  0  0  0  0  1 | 0  0  0  0  0  0  0  0  1  x | 0  0  0  0  0  0  0  1  x  x | 0  0  0  0  0  0  1  x  x  x | 0  0  0  0  0  1  x  x  x  x | 0  0  0  0  1  x  x  x  x  x | 0  0  0  1  x  x  x  x  x  x | 0  0  1  x  x  x  x  x  x  x | 0  1  x  x  x  x  x  x  x  x | 1  x  x  x  x  x  x  x  x  x | 0  0  0  0  0  0  0  0  1  1 | 0  0  0  0  1  1  1  1  0  0 | 0  0  1  1  0  0  1  1  0  0 | 0  1  0  1  0  1  0  1  0  1 |

- Phương trình logic















- Thiết kế sơ đồ mạch logic: Căn cứ vào kết quả thu được từ phương trình logic, bạn đọc sẽ tự thiết kế sơ đồ mạch logic trên

**4.1.1.3. Bộ mã hóa nhị phân 4 bit sang mã bù 2**

Trong kỹ thuật số, sử dụng số bù 2 để biểu diễn số âm nhờ vậy mà người ta dễ dàng thực hiện phép tính trừ bằng cách cộng số bị trừ với số bù 2 của số trừ. Dựa vào quy tắc tìm số bù 2thành lập được bảng chân lý của bộ biến đổi mã nhị phân 4 bit có các đầu vào là A0 ÷ A3 thành mã bù 2 có đầu ra là B0 ÷ B3.

- Bảng chân lý

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Số thập phân | Mã nhị phân | | | | Mã bù 2 | | | |
| A3 | A2 | A1 | A0 | B3 | B2 | B1 | B0 |
| 0  1  2  3  4  5  6  7  8  9  10  11  12  13  14  15 | 0  0  0  0  0  0  0  0  1  1  1  1  1  1  1  1 | 0  0  0  0  1  1  1  1  0  0  0  0  1  1  1  1 | 0  0  1  1  0  0  1  1  0  0  1  1  0  0  1  1 | 0  1  0  1  0  1  0  1  0  1  0  1  0  1  0  1 | 0  1  1  1  1  1  1  1  1  0  0  0  0  0  0  0 | 0  1  1  1  1  0  0  0  0  1  1  1  1  0  0  0 | 0  1  1  0  0  1  1  0  0  1  1  0  0  1  1  0 | 0  1  0  1  0  1  0  1  0  1  0  1  0  1  0  1 |

- Phương trình logic









- Thiết kế sơ đồ mạch logic

****

*Hình 4.4: Sơ đồ mạch logic mã hóa nhị phân 4 bit sang mã bù 2*

**4.1.2 Mạch giải mã**

Quá trình ngược với mạch mã hóa gọi là mạch giải mã. Nghĩa là từ một tổ hợp N = n biến giá trị đầu vào thì sẽ xuất hiện M = 2n giá trị đầu ra tương ứng của bộ giải mã.

Sơ đồ tổng quát của mạch giải mã

Bộ

giải

mã

Đầu vào có N = n

Đầu ra

có M = 2n

*Hình 4.5: Sơ đồ tổng quát của mạch giải mã*

**4.1.2.1. Bộ giải mã 3 sang 8**

Bộ giải mã có 3 tín hiệu đầu vào(A0 ÷ A2) thành 8 tín hiệu đầu ra (F0 ÷ F7) tương ứng. Tại mỗi thời điểm chỉ có một đầu ra có giá trị logic bằng 1 ứng với tổ hợp biến đầu vào nào đó, các đầu ra khác có giá trị logic bằng 0.

- Bảng chân lý

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Đầu vào | | | Đầu ra | | | | | | | |
| A2 | A1 | A0 | F0 | F1 | F2 | F3 | F4 | F5 | F6 | F7 |
| 0  0  0  0  1  1  1  1 | 0  0  1  1  0  0  1  1 | 0  1  0  1  0  1  0  1 | 1  0  0  0  0  0  0  0 | 0  1  0  0  0  0  0  0 | 0  0  1  0  0  0  0  0 | 0  0  0  1  0  0  0  0 | 0  0  0  0  1  0  0  0 | 0  0  0  0  0  1  0  0 | 0  0  0  0  0  0  1  0 | 0  0  0  0  0  0  0  1 |

- Phương trình logic

; ; ; 

; ; ; 

- Thiết kế sơ đồ mạch logic như hình 4.6



*Hình 4.6: Sơ đồ mạch logic của bộ giải mã 3 sang 8*

Trong thực tế, bộ giải mã 1 trong 8 được tích hợp dưới dạng IC 74LS138. Ký hiệu logic như hình 4.7a và sơ đồ logic của IC 74LS138 như hình 4.7b



*Hình 4.7a Hình 4.7b*

**4.1.2.2. Bộ giải mã BCD sang thập phân**

Bộ giải mã BCD sang thập phân có các lối vào S0 ÷ S3 và các lối ra F0 ÷ F9 biểu diễn các số thập phân từ 0 ÷ 9.

- Bảng chân lý

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Đầu vào | | | | Đầu ra | | | | | | | | | |
| S3 | S2 | S1 | S0 | F0 | F1 | F2 | F3 | F4 | F5 | F6 | F7 | F8 | F9 |
| 0  0  0  0  0  0  0  0  1  1 | 0  0  0  0  1  1  1  1  0  0 | 0  0  1  1  0  0  1  1  0  0 | 0  1  0  1  0  1  0  1  0  1 | 1  0  0  0  0  0  0  0  0  0 | 0  1  0  0  0  0  0  0  0  0 | 0  0  1  0  0  0  0  0  0  0 | 0  0  0  1  0  0  0  0  0  0 | 0  0  0  0  1  0  0  0  0  0 | 0  0  0  0  0  1  0  0  0  0 | 0  0  0  0  0  0  1  0  0  0 | 0  0  0  0  0  0  0  1  0  0 | 0  0  0  0  0  0  0  0  1  0 | 0  0  0  0  0  0  0  0  0  1 |

- Phương trình logic

; ; ; ; 

; ; ; ; 

- Thiết kế sơ đồ mạch logic: Từ hàm logic trên, ta có thể xây dựng sơ đồ giải mã này khi dùng 4 mạch NOT và 10 mạch NAND có 4 lối vào.



*Hình 4.8: Sơ đồ mạch logic của bộ giải mã BCD sang thập phân*

Các vi mạch giải mã có trong thực tế như 7442, 74L42, 74LS42, 7445, 74145, các vi mạch này có 16 chân có 4 lối vào S3 ÷ S0 và 10 chân lối ra tác động thấp 0, 1, 2, …9. Các IC này tuy có ký hiệu khác nhau nhưng đều có cùng sơ đồ logic và ký hiệu logic giống nhau và có dạng như hình 4.9a và 4.9b.



*Hình 4.9a Hình 4.9b*

**4.2 HỆ CHUYỂN ĐỔI MÃ**

**4.2.1 Chuyển đổi mã BCD sang led 7 đoạn**

**4.2.1.1. Led 7 đoạn**

Led 7 đoạn dùng để hiển thị các số thập phân từ 0 đến 9. Led gồm 7 đoạn a, b, c, d, e, f, g, mỗi đoạn là một led hoặc một nhóm led mắc song song với nhau. Qui ước các đoạn được cho bởi hình 4.10



*Hình 4.10: Quy ước led 7 đoạn*

*Có hai loại Led 7 đoạn*

- Loại Anode chung: tất cả Anode của các đoạn được nối chung với nhau và nối với Vcc, Cathode của mỗi đoạn được nối qua các điện trở giới hạn dòng tới đầu ra phù hợp của bộ giải mã. Bộ giải mã có đầu ra tích cực ở mức thấp.

- Loại Cathode chung: tất cả Cathode của các đoạn được nối chung với nhau và nối với GND , Anode của mỗi đoạn được nối qua các điện trở giới hạn dòng tới đầu ra phù hợp của bộ giải mã. Bộ giải mã có đầu ra tích cực ở mức cao.



*Hình 4.11a Hình 4.11b*

- Với màn hình tinh thể lỏng LCD, bộ giải mã có đầu ra tích cực ở mức cao

**4.2.1.2 Mạch giải mã BCD sang Led 7 đoạn**

Mạch có 4 ngõ vào cho số BCD và 7 ngõ ra thích ứng với các ngõ vào a, b, c, d, e, f, g của led 7 đọan, sao cho các thanh led sáng tạo được số thập phân đúng với mã BCD ở ngõ vào.

- Bảng chân lý của bộ giải mã BCD sang led 7 đoạn với đầu ra tác động ở mức cao

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Số thập phân | Mã BCD ở đầu vào | | | | Đầu ra Led 7 đoạn | | | | | | |
| D | C | B | A | a | b | c | d | e | f | g |
| 0  1  2  3  4  5  6  7  8  9 | 0  0  0  0  0  0  0  0  1  1 | 0  0  0  0  1  1  1  1  0  0 | 0  0  1  1  0  0  1  1  0  0 | 0  1  0  1  0  1  0  1  0  1 | 1  0  1  1  0  1  1  1  1  1 | 1  1  1  1  1  0  0  1  1  1 | 1  1  0  1  1  1  1  1  1  1 | 1  0  1  1  0  1  1  0  1  1 | 1  0  1  0  0  0  1  0  1  0 | 1  0  0  0  1  1  1  0  1  1 | 0  0  1  1  1  1  1  0  1  1 |

Dùng Bảng Karnaugh hoặc có thể đơn giản trực tiếp với các hàm chứa ít tổ hợp sẽ cho kết quả như sau:

****; ****; ****;****

****; ****; ****

Từ các kết quả trên ta có thể thiết kế mạch giải mã 7 đoạn dùng các cổng logic cơ bản. Trong thực tế nhà sản xuất đã chế tạo các vi mạch có sẵn để giải mã nhị phân sang Led 7 đoạn

Các vi mạch 7448, 74LS48, 7449, 74LS49 là các IC giải mã 7 đoạn có ngõ ra tác động ở mức cao. Các vi mạch này về cơ bản giống như bảng chân lý trên, chỉ khác đôi chút là số 6 không dùng thanh a và số 9 không dùng thanh d.

Các vi mạch 7447A, 74L47, 74S47 là các IC 16 chân có ngõ ra tác động ở mức thấp tương ứng led 7 đoạn loại anode chung

**4.2.1.3 Hiển thị 7 đoạn bằng tinh thể lỏng(LCD - Liquid Crystal Display)**

LCD gồm 7 đoạn như led thường và có chung một cực nền (backplane). Khi có tín hiệu xoay chiều với biên độ khoảng 3 - 15 VRMS và tần số trong khoảng 25 - 60 Hz áp giữa một đoạn và cực nền thì đoạn đó được tác động và sáng lên.

Sử dụng IC 4511(Giải mã BCD sang 7 đoạn, tác động cao) kết hợp với các cổng EX-OR để tác động tới đầu vào của LCD (H 4.15). Khi một ngõ ra mạch giải mã lên cao, ngõ ra cổng EX-OR cho một tín hiệu đảo pha với tín hiệu ở nền, đoạn tương ứng xem như nhận được tín hiệu có biên độ gấp đôi và sẽ sáng lên. Với các ngõ ra mạch giải mã ở mức thấp, ngõ ra cổng EX-OR cho một tín hiệu cùng pha với tín hiệu ở nền nên đoạn tương ứng không sáng.

**Thí dụ:** Sơ đồ mắc IC giải mã 7 đoạn có đầu ra tích cực ở mức thấp với đèn LED chỉ thị như trên hình dưới đây

****

7446 hoặc 7447

Bộ giải mã BCD thành 7 đoạn

*Hình 4.12: Bộ giải mã BCD thành 7 đoạn*

**4.2.2 Chuyển đổi mã hệ 2 sang mã Gray**

Đặc điểm: + Mã Gray là mã nhị phân không có trọng số

+ Các từ mã kề cận nhau chỉ khác nhau ở một bit

*Quy tắc:*

- Gọi từ mã nhị phân n bit có dạng: Bn-1Bn-2..B1B0

- Dịch mã nhị phân sang phải 1 bit: Bn-1Bn-2..B1

- Thực hiện phép cộng không có nhớ

- Mã Gray có kết quả tương ứng là: Gn-1Gn-2..G1G0

**Thí dụ 1:** Cho mã nhị phân 0111. Xác định mã Gray tương ứng

0111

+ 0011

0100

**Thí dụ 2:** Thiết kế bộ chuyển đổi mã nhị phân sang mã Gray 3 bit

- Bảng chân lý

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STT | Mã nhị phân | | | Mã Gray | | |
| B3 | B2 | B1 | G3 | G2 | G1 |
| 0  1  2  3  4  5  6  7 | 0  0  0  0  1  1  1  1 | 0  0  1  1  0  0  1  1 | 0  1  0  1  0  1  0  1 | 0  0  0  0  1  1  1  1 | 0  0  1  1  1  1  0  0 | 0  1  1  0  0  1  1  0 |

- Lập bảng Karnaugh và rút gọn hàm logic sẽ cho kết quả như sau:

****; ****; ****

- Thiết kế mạch logic

****

*Hình 4.13: Sơ đồ mạch logic chuyển đổi mã nhị phân sang Gray 3 bit*

**4.3 MẠCH CHỌN KÊNH VÀ PHÂN KÊNH**

**4.3.1 Mạch chọn kênh(MUX)**

Mạch chọn kênh (digital multiplexer) là mạch logic chấp nhận nhiều đầu vào dữ liệu số và chọn ra một trong số chúng tại mỗi thời điểm xác định để chuyển đến đầu ra. Quá trình truyền dữ liệu từ đầu vào đến đầu ra do đầu vào SELECT(đầu vào địa chỉ) chi phối.

Sơ đồ chức năng tổng quát của bộ hợp kênh số có dạng như hình 4.14 sau.



*Hình 4.14: Sơ đồ tổng quát bộ hợp kênh*

Bộ chọn kênh có 2n đầu vào dữ liệu có giá trị từ D0 → Dn-1 và một đầu ra F. Bộ chọn kênh hoạt động như mạch chuyển vị trong đó mã dạng số áp đầu vào SELECT sẽ cho phép đầu vào dữ liệu nào được chuyển đến đầu ra. Khi đó cần phải có n đầu vào địa chỉ cho 2n đầu vào dữ liệu.

**4.3.1.1. Bộ chọn kênh 2 đầu vào(MUX 2 : 1)**

Bộ chọn kênh 2 đầu vào sử dụng hai đầu vào dữ liệu D0, D1, một đầu vào địa chỉ SELECT và một đầu ra F.

 Bảng chân lý Ký hiệu

|  |  |
| --- | --- |
| S | F |
| 0 | D0 |
| 1 | D1 |

- Phương trình logic: 

- Thiết kế mạch logic



**- Ứng dụng:**

MUX 2 đầu vào được ứng dụng trong hệ thống máy vi tính sử dụng hai tín hiệu Master Clock khác nhau: xung nhịp tốc độ cao đối với một số chương trình, xung nhịp tốc độ thấp cho số khác. Hai xung nhịp tốc độ cao được đưa vào hai đầu vào dữ liệu D0 và D1. Tín hiệu từ phần logic điều khiển của máy tính sẽ kích thích đầu vào S, đầu vào địa chỉ sẽ quyết định tín hiệu xung tại đầu ra F.

**4.3.1.2. Bộ chọn kênh 4 đầu vào(MUX 4 : 1)**

Bộ chọn kênh sẽ bao gồm 4 đầu vào dữ liệu D0→ D3 và hai đầu vào địa chỉ S0, S1. Hai đầu vào địa chỉ sẽ tạo ra bốn tổ hợp khả dĩ, mỗi đầu vào dữ liệu bị chi phối với một tổ hợp khác nhau

Bảng chân lý Ký hiệu

|  |  |  |
| --- | --- | --- |
| S0 | S1 | F |
| 0 | 0 | D0 |
| 0 | 1 | D1 |
| 1 | 0 | D2 |
| 1 | 1 | D3 |



- Phương trình logic:

- Thiết kế sơ đồ mạch logic:



*Hình 4.15: Sơ đồ mạch bộ chọn kênh 4 đầu vào*

Có thể dùng MUX 2:1 để tạo thành MUX 4:1 như hình dưới 4.16

****

*Hình 4.16*

**4.1.1.3. Bộ chọn kênh 8 đầu vào(MUX 8 : 1)**

Xét sơ đồ logic của bộ chọn kênh 8 đầu vào(74LS151/HC151) được cho trên hình 4.27. Đầu vào cho phép (), khi = 0 thì MUX sẽ chọn 1 trong 8 đầu vào dữ liệu đưa tới đầu ra tùy theo đầu vào địa chỉ(S2 S1 S0). Khi =1 thì F = 0.

- Ký hiệu logic MUX 8:1



*Hình 4.17: Ký hiệu logic MUX 8:1*

- Bảng chân lý

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | S2 | S1 | S0 | F |
| 1 | x | x | x | 1 |
| 0 | 0 | 0 | 0 | D0 |
| 0 | 0 | 0 | 1 | D1 |
| 0 | 0 | 1 | 0 | D2 |
| 0 | 0 | 1 | 1 | D3 |
| 0 | 1 | 0 | 0 | D4 |
| 0 | 1 | 0 | 1 | D5 |
| 0 | 1 | 1 | 0 | D6 |
| 0 | 1 | 1 | 1 | D7 |

- Phương trình logic:

- Thiết kế sơ đồ mạch logic.



*Hình 4.18: Sơ đồ mạch bộ chọn kênh 8 đầu vào*

**4.3.2. Mạch phân kênh(DMUX – Demultiplexer)**

DMUX hoạt động ngược lại với MUX có nghĩa là mạch logic có một đầu vào dữ liệu và phân phối dữ liệu cho nhiều đầu ra.

Sơ đồ khối của bộ phân kênh được cho trên hình



*Hình 4.19: Sơ đồ khối của bộ phân kênh*

Mã đầu vào SELECT quyết định đến quá trình truyền đầu vào dữ liệu D đến đầu ra F0 ÷ Fn-1. Nói cách khác, bộ phân kênh lấy một nguồn dữ liệu vào và phân phối có chọn lọc đến một trong số n kênh ra.

**4.3.2.1. Bộ phân kênh 2 đầu ra(DMUX 1:1)**

Bộ phân kênh 2 đầu ra có một đầu vào dữ liệu D, hai đầu ra F0, F1 và một đầu vào địa chỉ S quyết định đến dữ liệu từ đầu vào đến đầu ra.

Bảng chân lý Sơ đồ khối

|  |  |  |
| --- | --- | --- |
| S | F0 | F1 |
| 0 | D | 0 |
| 1 | 0 | D |

  
- Phương trình logic: ; 

- Sơ đồ logic của mạch



*Hình 4.18: Sơ đồ mạch bộ phân kênh 2 đầu ra*

**4.3.2.2. Bộ phân kênh 8 đầu ra(DMUX 1 : 8)**

Bộ phân kênh 8 đầu ra có lối vào dữ liệu D, các lối ra F0 ÷ F7, cần 3 đầu vào địa chỉ S2, S1, S0.

- Bảng chân lý

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| S0 | S1 | S2 | F0 | F1 | F2 | F3 | F4 | F5 | F6 | F7 |
| 0  0  0  0  1  1  1  1 | 0  0  1  1  0  0  1  1 | 0  1  0  1  0  1  0  1 | D  0  0  0  0  0  0  0 | 0  D  0  0  0  0  0  0 | 0  0  D  0  0  0  0  0 | 0  0  0  D  0  0  0  0 | 0  0  0  0  D  0  0  0 | 0  0  0  0  0  D  0  0 | 0  0  0  0  0  0  D  0 | 0  0  0  0  0  0  0  D |

- Phương trình logic:

; ; ; ; ;

; 

- Thiết kế sơ đồ mạch logic

Nhận thấy rằng có thể sử dụng bộ phân kênh như bộ giải mã với đầu vào dữ liệu là đầu vào cho phép và ngược lại có thể sử dụng bộ giải mã như bộ phân kênh với đầu vào cho phép là đầu vào dữ liệu. Vì vậy, hãng chế tạo IC thường gọi đây là bộ phân kênh giải mã

IC 74LS138 (mạch giải mã 3 sang 8 có ký hiệu như hình 4.19) được dùng như bộ phân kênh với đầu vào được chọn làm đầu vào dữ liệu D và hai đầu vào cho phép còn lại là duy trì ở trạng thái tích cực. Mã địa chỉ là các đầu vào I0 I1I2.

Đầu vào dữ liệu

SELECT



Bộ giải mã/phân kênh

*Hình 4.19a* *Hình 4.19b*

**BÀI TẬP**

**Bài 1**: Thiết kế mạch mã hóa 32 đường sang 5 đường dùng IC 74148 và cổng logic.

**Bài 2**: Thiết kế mạch giải mã 4 đường sang 16 đường từ mạch giải mã 2 đường sang 4 đường có ngã vào cho phép.

**Bài 3**: Thiết kế mạch biến đổi mã nhị phân 4 bit sang mã Gray dùng IC 74LS86 và thiết kế mạch biến đổi mã Gray sang mã nhị phân 4 bit.

**Bài 4**: Vẽ sơ đồ logic của bộ giải mã từ BCD sang thập phân dùng các mạch logic NOT và NOR

**Bài 5**: Vẽ sơ đồ logic bộ giải mã 7 đoạn điều khiển sự hiển thị của các đèn chỉ thị số(đầu ra tích cực ở mức cao)

**Bài 6**: Dùng MUX 2:1 và MUX 4:1 để tạo thành MUX 8:1

**Bài 7**: Sắp xếp nhiều bộ dồn kênh 8:1 để tạo thành bộ dồn kênh 16:1sử dụng IC IC74151

* Dùng 2 IC74151 kết hợp thành bộ dồn kênh 16 đầu vào như hình dưới



Dữ liệu vào

* Bộ dồn kênh chập 4 hai đầu vào(74157/LS157/HC157) được cho như hình dưới.



* Bảng chân lý của IC 74157

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| E | S | Fa | Fb | Fc | Fd |
| 1 | X | 0 | 0 | 0 | 0 |
| 0 | 0 | D0a | D0b | D0c | D0d |
| 0 | 1 | D1a | D1b | D1c | D1d |

* Sơ đồ logic của bộ hợp kênh 74157



**Bài 8**: Trình bày cách sử dụng bộ giải mã 7442 làm bộ phân kênh 1 : 8

**Bài 9**: MUX 4:1 có lối ra là hàm logic 3 biến:  Hai lối vào điều khiển là A và B, tìm các lối vào dữ liệu.

**Bài 10**: Dùng MUX 4:1 để thực hiện hàm logic có 3 biến A,B,C sau

1.  với các biến điều khiển là B, C
2.  với các biến điều khiển là A, C

**Bài 11**: Thực hiện hàm logic 4 biến 

1. Dùng MUX 4:1 và mạch NAND
2. Dùng MUX 8:1

**Bài 12**: Dùng MUX 4:1 xây dựng sơ đồ bộ giải mã BCD sang 7 đoạn

**CHƯƠNG 5: MẠCH LOGIC DÃY**

Trong chương trước, chúng ta đã khảo sát các loại mạch tổ hợp, đó là các hệ thống logic mà tại mỗi thời điểm nhất định ngõ ra của nó chỉ phụ thuộc vào các biến ở ngõ vào mà không phụ thuộc vào trạng thái đó của mạch. Ngược lại trong kỹ thuật số tồn tại các hệ logic được gọi là logic dãy. Đó là các hệ thống logic mà tại thời điểm nhất định ngõ ra không chỉ phụ thuộc vào các biến ở ngõ vào(biến hệ 2) mà còn phụ thuộc vào các biến ở thời điểm trước đó, các biến này được nhớ lại trong bộ nhớ của hệ. Vì vậy người ta còn gọi hệ logic dãy là hệ thống logic có nhớ hay hệ logic phụ thuộc vào trạng thái trong của hệ.

Như vậy sự khác nhau cơ bản giữa hệ logic dãy và hệ logic tổ hợp là hệ dãy có số trạng thái trong lớn hơn một.

Một cách tổng quát có thể biểu diễn cấu trúc của một hệ logic dãy như hình sau

****

Hệ logic tổ hợp

Các trigơ

Flip – Flop

(bộ nhớ)

Xung nhịp

Đầu ra

Đầu vào

*Hình 5.1: Cấu trúc của hệ logic dãy*

Hệ logic dãy bao gồm:

- *Khối logic tổ hợp*: có nhiệm vụ điều khiển quá trình thu nhận và đưa ra các thông tin vào/ra cũng như điều hành mọi hoạt động của các flip – flop.

- *Khối bộ nhớ(flip – flop)*: chứa các bộ nhớ của hệ để nhớ các thông tin trung gian.

Như vậy, tổ hợp các trạng thái của các flip – flop ở một thời điểm nào đó chính là trạng thái trong của hệ. Rõ ràng là khi số flip – flop của hệ tăng lên thì số trạng thái trong của hệ cũng tăng lên. Vì vậy khi thiết kế các hệ logic dãy, cần phải chú ý vòng hồi tiếp giữa phần hệ tổ hợp điều khiển và các flip – flop hoặc giữa các flip – flop. Khi đưa một dãy xung nhịp có độ ổn định cao vào hệ để đồng bộ các flip – flop có một ý nghĩa rất lớn để hình thành hai loại kết cấu thuộc hệ logic dãy là đồng bộ và không đồng bộ.

**5.1. CÁC TRIGƠ SỐ**

**5.1.1. Định nghĩa và phân loại**

*a. Định nghĩa*

Trigơ (hay Flip – Flop viết tắt FF) là một phần tử nhớ có hai trạng thái cân bằng ổn định tương ứng với hai mức logic 0 hoặc 1. Dưới tác động của tín hiệu điều khiển ở đầu vào, trigơ có thể chuyển về một trong hai trạng thái cân bằng và giữ nguyên trạng thái cân bằng đó chừng nào chưa có tín hiệu điều khiển làm thay đổi trạng thái của nó. Trạng thái tiếp theo của trigơ phụ thuộc không những vào tín hiệu ở đầu vào mà còn phụ thuộc vào cả trạng thái hiện hành của nó.



Đầu vào

Đầu ra

*Hình 5.2: Mạch flip - flop*

Một Flip - Flop thường có:

- Một hoặc hai ngõ vào dữ liệu, một ngõ vào xung CK và có thể có các ngõ vào với các chức năng khác.

- Hai ngõ ra  (ngõ ra chính) và  (ngõ ra phụ) luôn đối nhau. Người ta thường dùng trạng thái của ngõ ra để chỉ trạng thái của FF. Nếu hai ngõ ra có trạng thái giống nhau ta nói FF ở **trạng thái cấm.**

Flip - flop có thể được tạo nên từ mạch chốt (Latch)

*b. Phân loại*

- Theo chức năng làm việc của các đầu vào điều khiển: Hiện nay thường sử dụng loại FF một đầu vào là FF – T và FF – D, loại FF hai đầu vào là FF – RS và FF – JK. Ngoài ra đôi khi còn có thể gặp loại trigơ nhiều đầu vào.

- Theo phương pháp ghi thông tin vào flip – flop: Loại flip – flop đồng bộ và flip – flop không đồng bộ.

**5.1.2. Hoạt động của mạch flip – flop.**

Quá trình hoạt động của các Flip – flop có thể khác nhau nhưng về cơ bản thì mỗi flip - flop có thể coi gồm haiphần chính sau đây:

+ Phần flip – flop cơ bản.

+ Phần điều khiển.

**a. Phần flip – flop cơ bản**

Phần flip – flop cơ bản thường gồm hai mạch điện tử chứa các phần tử logic giống nhau. Các phần tử logic này có đặc tính của hàm NOT hay có liên quan đến hàm này. Mỗi mạch điện tử có một (hoặc nhiều hơn) đầu vào và một đầu ra, các mạch được nối với nhau theo một vòng kín và được biểu diễn như hình dưới

****

Phần tử logic

Vào 1

Ra 2

Vào 2

Ra 1

Phần tử logic

*Hình 5.3: Biểu diễn phần flip – flop cơ bản*

Giả sử đầu ra 2 của mạch ở trang thái cao(Q = 1) sẽ kéo theo đầu vào của mạch 2 cũng ở thái cao, khi đó đầu ra 1 phải ở trạng thái thấp(). Trạng thái này thỏa mãn tính chính xác của đầu vào 1 ở trạng thái thấp. Như vậy cả hai đầu vào đều được thỏa mãn, mạch làm việc ở trạng thái ổn định. Bằng cách lập luân tương tự ta thấy rằng trạng thái Q = 0(đầu ra 2 của mạch ở trạng thái thấp) cũng là một trạng thái ổn định với .

**b. Phần điều khiển**

Mỗi mạch FF để hoạt động được ở trạng thái tốt nhất thì chúng ta phải kiểm soát được trạng thái của mạch và làm mạch thay đổi theo ý muốn chủ quan của con người. Vì vậy ta cần phải thêm vào mỗi mạch phần điều khiển.

Phần điều khiển FF có hai loại chính là điều khiển trực tiếp và điều khiển đồng bộ được biểu diễn như hình dưới:



Mạch điều khiển đồng bộ

Điều khiển trực tiếp

Phần tử logic

Phần tử logic

Điều khiển trực tiếp

Đầu vào điều khiển đồng bộ

Đầu vào xung nhịp (Clk)

*Hình 5.4: Biểu diễn dạng tín hiệu điều khiển flip – flop*

- Đầu vào điều khiển trực tiếp thường được đưa trực tiếp vào hai mạch điện thành phần chứa các phần tử logic của FF. Chúng được sử dụng để xác định trực tiếp trạng thái của Q hoặc bắt Q phải ở một trong hai trạng thái 0 hay 1. Khi một hoặc các đầu vào điều khiển trực tiếp đạng hoạt động thì Q không tuân theo trạng thái của các đầu vào đồng bộ. Hai đầu vào này thường có tên là PRESET(đặt trước) và CLEAR(xóa). Đầu vào PRESET khi được kích thích luôn luôn đưa Q đến trạng thái 1 còn đầu vào CLEAR thì đưa Q về trạng thái 0.

- Đầu vào điều khiển đồng bộ điều khiển FF cơ bản trung gian của một mạch điều khiển đồng bộ và dưới sự điều khiển của xung nhịp(Clk) được đưa vào theo một đầu riêng. Các đầu ra Q và  chịu sự điều khiển của trạng thái logic của các đầu vào khi có xung nhịp. Khi không có xung nhịp, các đầu vào có thể thay đổi trạng thái mà không gây ảnh hưởng gì đến Q và 

**5.1.3. Các flip – flop cơ bản**

**5.1.3.1. Flip – flop RS**

Flip – flop RS là loại FF đơn giản nhất chỉ có hai đầu vào điều khiển trực tiếp R(RESET) và S(SET). Mạch được điều khiển bởi đầu vào đồng bộ và một đầu vào xung nhịp Clock(Clk) điều khiển chung cho cả hai lối vào. Chỉ khi nào có tác động của xung nhịp này Flip – flop mới chuyển trạng thái theo tác động của R và S . Ký hiệu và sơ đồ sử dụng cổng NAND của FF – RS được biểu diễn trên hình 5.5 và hình 5.6

|  |  |
| --- | --- |
|  |  |

*Hình 5.5: Ký hiệu FF – RS cơ bản Hình 5.6: Sơ đồ FF – RS sử dụng cổng NAND*

- Bảng chân lý của FF – RS như sau

Quy ước: Q: trạng thái ban đầu

Qm: trạng thái mới

|  |  |  |  |
| --- | --- | --- | --- |
| R | S | Q | Qm |
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 1 |
| 0 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | x |
| 1 | 1 | 1 | x |

|  |  |  |
| --- | --- | --- |
| R | S | Qm |
| 0 | 0 | Q |
| 0 | 1 | 1 |
| 1 | 0 | 0 |
| 1 | 1 | Cấm |

Từ bảng chân lý rút ra nhận xét như sau:

+ Khi R = S = 0(hai ngõ vào đều không tác động), ngõ ra không thay đổi trạng thái.

+ Khi R = 0,S = 1(ngõ vào S tác động), chốt được Set tức là Qm = 1

+ Khi R = 1,S = 0(ngõ vào R tác động), chốt được Reset tức là Qm = 0

+ Khi R = S = 1(hai ngõ vào đều tác động), ngõ ra rơi vào trạng thái cấm.

- Từ bảng chân lý FF-RS cơ bản, ta rút gọn thành biểu thức như sau:



Biểu thức: 

Biểu thức trên cho thấy: đầu ra không những là hàm số của đầu vào mà còn phụ thuộc vào trạng thái trước đó của đầu ra

**5.1.3.2. Flip – flop JK**

Flip – flop JK là loại FF đa năng có rất nhiều ứng dụng trong kỹ thuật số. Trong FF này có hai hoặc nhiều đầu vào điều khiển đồng bộ J, K và đầu vào xung nhịp (Clk). Tương tự như FF-RS thì FF-JK cũng có hai đầu vào kích thích là Preset (đặt trước) và Clear(xóa) và được biểu diễn như hình sau.

|  |  |
| --- | --- |
| *Hình 5.7: Ký hiệu FF-JK cơ bản* | *Hình 5.8: Sơ đồ FF-JK sử dụng cổng NAND* |

Về cấu tạo, FF-JK phức tạp hơn FF-RS nhưng bù lại chúng có khả năng ứng dụng rộng rãi hơn với những lý do sau:

+ Đầu ra Q phụ thuộc đầu vào J, K

+ Khi J = K = 1 thì mạch hoạt động như một FF – T nghĩa là Q vẫn được xác định để trở thành trạng thái đảo. Người ta lợi dụng trạng thái này để thiết kế mạch đếm

- Bảng chân lý của FF – JK như sau

|  |  |  |  |
| --- | --- | --- | --- |
| J | K | Q | Qm |
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 0 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 0 | 1 |
| 1 | 0 | 1 | 1 |
| 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 0 |

|  |  |  |
| --- | --- | --- |
| J | K | Qm |
| 0 | 0 | Q |
| 0 | 1 | 0 |
| 1 | 0 | 1 |
| 1 | 1 |  |

Từ bảng chân lý rút ra nhận xét như sau:

+ Khi J = K = 0(hai ngõ vào đều không tác động), ngõ ra không thay đổi trạng thái.

+ Khi J = 0,K = 1(ngõ vào K tác động), đầu ra Qm = 0

+ Khi J = 1,K = 0(ngõ vào J tác động), đầu ra Qm = 1

+ Khi J = K = 1(hai ngõ vào đều tác động), ngõ ra đảo trạng thái.

Phương trình đặc trưng của FF-JK là: 

Để FF-JK hoạt động được ở chế độ đồng bộ, hai lối vào bất đồng bộ phải để ở mức điện áp tương ứng, nếu trên ký hiệu của FF các lối vào bất đồng bộ có vòng tròn nhỏ hoặc dấu gạch ngang ở trên chữ ()thì các chân này phải để ở mức điện áp cao.

**5.1.3.3. Flip – flop T**

Flip – flop T là loại FF có một đầu vào dữ liệu và hai đầu ra Q và . Nối chung hai ngõ vào J và K của FF-JK ta được FF-T. Ký hiệu và sơ đồ của FF-T được thể hiện như hình 5.11 và hình 5.12

|  |  |
| --- | --- |
|  |  |

*Hình 5.9: Ký hiệu FF-T*



*Hình 5.10: FF-T tạo thành từ FF-JK sử dụng cổng NAND*

- Bảng chân lý của FF – T như sau:

|  |  |  |
| --- | --- | --- |
| T | Q | Qm |
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

|  |  |
| --- | --- |
| T | Qm |
| 0 | Q |
| 1 |  |

Từ bảng chân lý rút ra nhận xét như sau:

+ Khi T=0, **FF không đổi trạng thái** dù có tác động của xung Clock.

+ Khi T=1, **FF đổi trạng thái** saumỗi lần có xung Clock tác động.

Phương trình đặc trưng của FF-T là: 

Thực tế, mạch thường không có các đầu vào đồng bộ mà chỉ còn và . FF này tuần tự thay đổi trạng thái của Q mỗi lần có xung kích thích. Như vậy với kích thích liên tục ở đầu vào Clk thì Q và  cũng liên tục thay đổi trạng thái.

**5.1.3.4. Flip – flop D**

Flip – flop D là loại FF có một đầu vào dữ liệu hoạt động đồng bộ với xung Clk và được dùng nhiều trong các bộ ghi lưu trữ các bit thông tin nhị phân. Thông tin ở đầu vào D được chuyển đến đầu ra Q của FF-D sau một xung nhịp, tức là thông tin được chuyển đến đầu ra chậm mất một khoảng thời gian bằng độ rộng của xung nhịp. Ký hiệu và sơ đồ của FF-D được thể hiện như hình 5.14 và hình 5.15.

|  |  |  |
| --- | --- | --- |
|  |  |  |

*Hình 5.11: Ký hiệu FF-T*



*Hình 5.12: FF-D sử dụng cổng NAND*

- Bảng chân lý của FF – D như sau:

|  |  |  |
| --- | --- | --- |
| D | Q | Qm |
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |

|  |  |
| --- | --- |
| D | Qm |
| 0 | 0 |
| 1 | 1 |

Từ bảng chân lý rút ra nhận xét như sau:

+ Khi D=0, đầu ra Qm**=**0dù có tác động của xung Clock.

+ Khi D=1, đầu ra Qm**=**1 saumỗi lần có xung Clock tác động.

Phương trình đặc trưng của FF-D là: 

**5.2. CHUYỂN ĐỔI GIỮA CÁC FLIP - FLOP**

Trong thực tế, dựa vào nhu cầu sử dụng mà chúng ta phải thực hiện quá trình chuyển đổi giữa các FF, với bốn loại FF(FF-RS, FF-JK, FF-D, FF-T) có được 12 khả năng chuyển đổi. Sơ đồ khối của bộ chuyển đổi có dạng như sau:



Mạch logic

FF loại i

*Hình 5.13: Sơ đồ chuyển đổi giữa các FF*

Sơ đồ chuyển đổi giữa các FF bao gồm hai khối:

*+ Khối flip-flop loại i*: Khối này sử dụng FF loại nào trong bốn loại FF cần được chuyển đổi.

*+ Khối mạch logic:* Dựa vào kết quả của hàm logic mà khối này chứa các cổng logic cơ bản(hoặc không).

Ngoài ra trong sơ đồ trên, các đầu vào j là các đầu vào của FF loại j cần thiết kế, đầu ra của mạch logic là các đầu vào của FF loại i cho trước.

*Để thực hiện chuyển đổi từ FF loại i sang FF loại j cần thực hiện các bước sau:*

**+ Xác định hệ hàm i = f(j,Q) từ bảng các đầu vào điều khiển.**

**+ Cực tiểu hóa hàm logic**

**+ Xây dựng sơ đồ chuyển đổi từ kết quả rút gọn ở trên.**

**Thí dụ:** Chuyển đổi FF-RS sang FF-JK

FF-RS là FF cho trước và FF-JK là FF cần được chuyển đổi dựa vào sơ đồ chuyển đổi giữa các FF ta có đầu vào của mạch logic là J và K và đầu ra là R và S. vì vậy chúng ta phải xác định mối tương quan hàm R = f(J,K,Q) và S = f(J,K,Q).

- Bảng các đầu vào điều khiển.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| J | K | Q | Qm | R | S |
| 0 | x | 0 | 0 | x | 0 |
| 1 | x | 0 | 1 | 0 | 1 |
| x | 1 | 1 | 0 | 1 | 0 |
| x | 0 | 1 | 1 | 0 | x |

- Cực tiểu hàm logic

+ Tìm R= f(J,K,Q)



Kết quả: 

+ Tìm S= f(J,K,Q)



Kết quả: 

- Xây dựng sơ đồ chuyển đổi có dạng như hình 5.14



*Hình 5.14: Sơ đồ chuyển đổi FF-RS sang FF-JK*

**5.3. CÁC BỘ ĐẾM**

**5.3.1 Định nghĩa**

Bộ đếm là mạch dãy tuần hoàn có một đầu vào cho xung đếm và một đầu ra với số trạng thái trong bằng hệ số đếm. Dưới tác động của tín hiệu đầu vào là xung đếm (Clk) mạch sẽ chuyển từ trạng thái trong này sang trạng thái trong khác theo một thứ tự nhất định, cứ sau Kđ tín hiệu vào thì mạch sẽ chuyển về trạng thái ban đầu. Ký hiệu bộ đếm có dạng như hình 5.15

**

Bộ đếm

Kđ

*Hình 5.15: Ký hiệu bộ đếm*

Trong đó: Kđ : Hệ số đếm

Xđ: Đầu vào đếm

Y: Đầu ra

- Đồ hình trạng thái tổng quát của bộ đếm hình 5.16.



*Hình 5.16: Đồ hình tổng quát của bộ đếm*

**5.3.2. Phân loại**

Có nhiều phương pháp kết hợp các FF cho nên có rất nhiều loại bộ đếm. Tuy nhiên chúng ta có thể sắp xếp thành ba loại chính sau

**+ Phân loại theo hệ số đếm**

* Bộ đếm có hệ số đếm Kđ = 2n
* Bộ đếm có hệ số đếm Kđ ≠ 2n

**+ Phân loại theo chức năng của mạch đếm**

* Các mạch đếm lên(Up Counter): còn gọi mạch đếm cộng, mạch đếm thuận
* Các mạch đếm xuống(Down Counter): còn gọi mạch đếm trừ, mạch đếm nghịch

**+ Phân loại theo chế độ làm việc của bộ đếm**

* Phương pháp đồng bộ: Trong phương pháp này xung nhịp đưa đến các FF cùng một lúc
* Phương pháp không đồng bộ: Trong phương pháp này xung nhịp chỉ đưa đến một FF, các FF còn lại tự kích thích lẫn nhau.

**5.3.3. Các bước thiết kế bộ đếm**

+ ***Bước 1***: Vẽ đồ hình trạng thái của bộ đếm

Căn cứ vào yêu cầu của bộ đếm cần thiết như: hệ số đếm(Kđ) và một số yêu cầu khác để xây dựng đồ hình trạng thái mô tả hoạt động của bộ đếm.

+ ***Bước 2***: Xác định số FF của bộ đếm, mã hóa các trạng thái trong của bộ đếm

Dựa vào hệ số đếm (Kđ) là chẵn hay lẻ theo trọng số của bộ đếm mà xác định được n số FF cần sử dụng cho bộ đếm.

* Đối với mã nhị phân và mã Gray: n ≥ log2Kđ → 2n ≥ Kđ
* Đối với mã vòng : n = Kđ
* Đối với mã Johnson: : n = (1/2) Kđ

+ ***Bước 3***: Xác định linh kiện cần sử dụng cho bộ đếm.

+ ***Bước 4***: Lập bảng chuyển đổi trạng thái của bộ đếm

Căn cứ vào bảng trạng thái của FF-RS, FF-JK, FF-D, FF-T để xây dựng được bảng chuyển đổi trạng thái của bộ đếm.

+ ***Bước 5*:** Cực tiểu hóa hàm logic từ bảng chuyển đổi trạng thái của bộ đếm

* Dựa vào bảng chuyển đổi trạng thái để xác định phương trình đầu vào điều khiển cho các FF và phương trình đầu ra
* Dựa trực tiếp vào đồ hình chuyển đổi trạng thái để xác định phương trình đầu vào điều khiển cho các FF và phương trình đầu ra

+ ***Bước 6*:** Xây dựng sơ đồ mạch điện của bộ đếm

**Thí dụ 1**: Xây dựng bộ đếm nhị phân có Kđ = 8, thuận, đồng bộ.

- Đồ hình trạng thái của bộ đếm có dạng như hình 5.17



*Hình 5.17: Đồ hình trạng thái với Kđ =8*

- Từ giả thiết, gọi n là số FF cần thiết kế: 2n ≥ Kđ tương đương 2n ≥ 8 = 23 hay n = 3 . Như vậy phải dùng 3 FF để mã hóa trạng thái trong của bộ đếm(S0 ….S7) theo mã nhị phân (000…111)

- Sử dụng linh kiện cần thiết kế là FF-JK

- Lập bảng chuyển đổi trạng thái của bộ đếm.

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Trạng thái hiện tại | | | Trạng thái tiếp theo | | | FF-JK1 | | FF-JK2 | | FF-JK3 | | **Y** |
| **Q1** | **Q2** | **Q3** | **Qm1** | **Qm2** | **Qm3** | **J1** | **K1** | **J2** | **K2** | **J3** | **K3** |
| 0  0  0  0  1  1  1  1 | 0  0  1  1  0  0  1  1 | 0  1  0  1  0  1  0  1 | 0  0  0  1  1  1  1  0 | 0  1  1  0  0  1  1  0 | 1  0  1  0  1  0  1  0 | 0  0  0  1  x  x  x  x | x  x  x  x  0  0  0  1 | 0  1  x  x  0  1  x  x | x  x  0  1  x  x  0  1 | 1  x  1  x  1  x  1  x | x  1  x  1  x  1  x  1 | 0  0  0  0  0  0  0  1 |

- Rút gọn hàm logic

+ Tìm J1:



Kết quả: J1 = Q2Q3

Tương tự, rút gọn K1, J2, K2, J3, K3 ta được kết quả như sau

K1 = Q2Q3 ; J2 = K2 = Q3; J3 = K3 = 1

Sau khi bộ đếm thực hiện mã hóa trạng thái trong đến trạng thái cuối cùng thì quay trở về trạng thái ban đầu vì vậy đầu ra Y = Q1Q2Q3.

- Sơ đồ mạch điện của bộ đếm như hình 5.18



*Hình 5.18: Sơ đồ mạch điện Kđ = 8*

**Thí dụ 2**: Xây dựng bộ đếm nhị phân có Kđ = 6, thuận, đồng bộ.

- Đồ hình trạng thái của bộ đếm có dạng như hình 5.97



*Hình 5.19: Đồ hình trạng thái với Kđ = 6*

- Từ giả thiết, số FF cần thiết kế là: 2n ≥ 6 suy ra chọn n = 3. Như vậy phải dùng 3 FF để mã hóa trạng thái trong của bộ đếm(S0 ….S5) theo mã nhị phân (000…101).

- Sử dụng linh kiện cần thiết kế là FF-D

- Lập bảng chuyển đổi trạng thái của bộ đếm

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Q1** | **Q2** | **Q3** | **Qm1** | **Qm2** | **Qm3** | **D1** | **D2** | **D3** | **Y** |
| 0  0  0  0  1  1  1  1 | 0  0  1  1  0  0  1  1 | 0  1  0  1  0  1  0  1 | 0  0  0  1  1  0  0  0 | 0  1  1  0  0  0  0  0 | 1  0  1  0  1  0  0  0 | 0  0  0  1  1  0  0  0 | 0  1  1  0  0  0  0  0 | 1  0  1  0  1  0  0  0 | 0  0  0  0  0  1  0  0 |

- Rút gọn hàm logic

+ Tìm D1:



Kết quả: 

Tương tự, rút gọn D2, D3 ta được kết quả như sau:

; 

Trường hợp này, bộ đếm thực hiện việc đếm lần thứ 6 sẽ là trạng thái cuối cùng rồi quay trở về trạng thái ban đầu vì vậy đầu ra 

- Sơ đồ mạch điện của bộ đếm như hình 5.20



*Hình 5.20: Sơ đồ mạch điện với Kđ = 6*

**5.4. CÁC BỘ GHI DỊCH**

**5.4.1. Khái niệm**

Bộ ghi dịch hay còn gọi là thanh ghi dịch là phần tử logic có khả năng ghi và dịch thông tin. Cấu trúc của bộ ghi dịch có dạng như hình 5.21



Đầu vào

*Hình 5.21: Cấu trúc của bộ ghi dịch*

Bộ ghi dịch gồm một dãy các phần tử nhớ được mắc liên tiếp với nhau và đóng trong cùng một vỏ. Các phần tử nhớ thường dùng là loại FF-D hoặc các phần tử khác nhưng mắc theo kiểu phần tử FF-D. Để ghi n bit thông tin, người ta sử dụng n phần tử nhớ, đầu ra của phần tử nhớ này là đầu vào của phần tử nhớ kia. Thông tin được nạp vào bộ ghi dịch từng bit một và đồng bộ với xung Clk

Có hai phương pháp đưa dữ liệu vào mạch là nối tiếp(serial) và song song(parallel) tạo thành bộ ghi dịch nối tiếp và bộ ghi dịch song song.

**5.4.2. Bộ ghi dịch nối tiếp**

Trên hình 5.22 là sơ đồ của một mạch ghi dịch 3 bit dùng FF-JK. Nó làm nhiệm vụ thu nhận và chuyển tín hiệu theo phương pháp nối tiếp. Mỗi FF được điều khiển bởi đầu ra (Q và Q) của FF trước đó. Xung kích thích Clk được đưa đồng thời đến các FF.

Đầu ra song song



Đầu ra nối tiếp

Đầu vào

nối tiffzếp

*Hình 5.22: Bộ ghi dịch nối tiếp*

Trước khi mô tả hoạt động của mạch ghi này, giả sử các FF ở trạng thái như sau:

Đối với FF-A: QA = 1 JB = 1 thì FF-B sẵn sàng lên mức 1

 KB = 0

Đối với FF-B: QB = 0 JC = 0 thì FF-C sẵn sàng xuống mức 0

 KC = 1

Như vậy khi có xung nhịp Clk, FFA sẽ nhận trạng thái mới từ đầu vào nối tiếp còn QB = 1 và QC = 0. Khi đó bit 1 từ QA đã chuyển sang QB và bit 0 từ QB đã chuyển sang QC. Mặc dù FFA cũng đổi trạng thái cùng một lúc với FFB nhưng vì các FF khi hoạt động bao giờ cũng có tính chậm trễ cho nên khi có xung nhịp Clk đến thì thông tin đã có thể chạy vào FFB theo sự điều khiển của JB trước khi QA kịp thời thay đổi trạng thái.

Nếu tại thời điểm ban đầu, cả 3 FF đều có đầu ra Q = 0 và dữ liệu 3 bit là I1, I2, I3 thì

+ Sau xung nhịp thứ nhất: QA = I1; QB = 0; QC = 0

+ Sau xung nhịp thứ hai : QA = I2; QB = I1; QC = 0

+ Sau xung nhịp thứ hai : QA = I3 QB = I2; QC = I1

Vì vậy, sau 3 xung nhịp cả 3 bit dữ liệu nằm hoàn toàn trong mạch ghi dịch nối tiếp. Do đó rút ra các kết luận sau.

+ Một dữ liệu dài n bit được chuyển hoàn toàn vào trong một mạch ghi dịch nối tiếp sau n xung nhịp

+ Một mạch ghi dịch chứa m FF chỉ có thể chứa được m bit

Khi dữ liệu nằm hoàn toàn trong các phần tử FF, ta có thể lấy các bit ra cùng một lúc qua các đầu ra song song(đầu ra A, B, C)

+ Cho đầu vào nối tiếp I = 0

+ Cho xung nhịp vào mạch

Khi đó:

+ Sau xung nhịp thứ nhất: QA = 0; QB = I2; QC = I1

+ Sau xung nhịp thứ hai : QA = 0; QB = 0; QC = I1

+ Sau xung nhịp thứ hai : QA = 0; QB = 0; QC = 0

Như thế, dữ liệu từng bit 1 đã được đưa ra khỏi mạch qua chỉ một đầu ra QC(đầu ra nối tiếp). Sau 3 xung nhịp, tất cả các bit ra khỏi mạch và các FF lại trở về trạng thái 0.

**5.4.3. Bộ ghi dịch song song**

Bộ ghi dịch này sẽ đưa thông tin của cùng một dữ liệu vào FF cùng một lúc. Phương pháp này thường phải dùng các đầu vào điều khiển trực tiếp  và của các FF qua trung gian của một số cổng logic như sơ đồ trên hình 5.23.

Đầu ra song song



Đầu ra nối tiếp

Nhận vào

Đầu vào nối tiếp

Đầu vào song song

*Hình 5.23: Bộ ghi dịch song song*

Ở đầu vào song song, các bit của dữ liệu được đặt tại các đường vào A, B, C.

+ Khi đường điều khiển “*nhận vào*” ở mức logic 0, tín hiệu qua cổng logic sẽ đưa đến đầu vào làm đầu ra Q tương ứng xuống mức 0.

+ Khi đường điều khiển “*nhận vào*” ở mức logic 1, các cổng NAND mở cho bit tương ứng đi qua. Khi đó tín hiệu qua cổng logic sẽ đưa đến đầu vào làm đầu ra Q tương ứng xuống mức 1.

Sau đó tác động vào Clk để đưa xung nhịp vào mạch dịch chuyển các bit tùy ý từ trái sang phải. Trong một số trường hợp, mạch ghi dịch rất hữu ích vì trong một thiết bị điện tử số có thể xảy ra dữ liệu này là song song còn dữ liệu kia là nối tiếp, do đó mạch ghi dịch này đóng vai trò trung gian thu nhận và cung cấp dữ liệu thích hợp cho cả phần nối tiếp và song song

**5.4.4. Bộ ghi dịch trái – phải.**

Trong kỹ thuật đôi khi sử dụng một mạch ghi dịch có thể dịch chuyển dữ liệu từ trái sang phải hoặc ngược lại theo tín hiệu điều khiển. Trên hình 5.24 thể hiện bộ ghi dịch trái – phải. Bộ ghi dịch cần phải sử dụng một số cổng logic để thay đổi mối liên hệ giữa các FF.

****

*Hình 5.24: Bộ ghi dịch trái – phải*

Trong đó sử dụng một số ký hiệu với nội dung sau:

A: tín hiệu vào nối tiếp từ trái sang phải

B: dịch chuyển tín hiệu

C: mức logic

D: tín hiệu vào nối tiếp từ phải sang trái

*Nguyên lý hoạt động của mạch:*

+ Nếu A = 1, B = 1, C = 1 thi FF1 sẵn sàng lên mức 1, kích cho các FF từng xung đầu vào thì dữ liệu sẽ được đưa vào từng bộ nhớ từ trái sang phải.

+ Nếu A = 1, B = 1, C = 0 thì FF3 sẵn sàng xuống mức 0, kích cho các FF từng xung đầu vào thì dữ liệu sẽ ra khỏi từng bộ nhớ từ phải sang trái.

+ Nếu A = 1, B = 0, D = 1 thi FF3 sẵn sàng lên mức 1, kích cho các FF từng xung đầu vào thì dữ liệu sẽ được đưa vào từng bộ nhớ từ phải sang trái.

+ Nếu A = 1, B = 0, D = 0 thì FF1 sẵn sàng xuống mức 0, kích cho các FF từng xung đầu vào thì dữ liệu sẽ ra khỏi từng bộ nhớ từ trái sang phải.

**BÀI TẬP**

**Bài 1:** Thực hiện chuyển đổi sau

a. Từ FF – RS sang FF – D, FF – T

b. Từ FF – JK sang FF – RS, FF – D, FF – T

c. Từ FF – D sang FF – RS, FF – JK, FF – T

d. Từ FF – T sang FF – RS, FF – JK, FF – D

**Bài 2:** Thiết kế bộ đếm với Kđ = 8, thuận, đồng bộ sử dụng

a. FF – RS

b. FF – T

c. FF – D

**Bài 3:** Thiết kế bộ đếm với Kđ = 7, thuận, đồng bộ sử dụng

a. FF – RS

b. FF - JK

c. FF – T

d. FF – D

**CHƯƠNG VI: BỘ NHỚ BÁN DẪN**

**6.1 CÁC KHÁI NIỆM**

**6.1.1. Chức năng của bộ nhớ**

Tính ưu việt chủ yếu của các hệ thống số so với hệ thống tương tự là khả năng lưu trữ một lượng lớn thông tin số và dữ liệu trong những khoảng thời gian nhất định. Vì vậy đối với thiết bị số, khả năng chứa đựng được dữ liệu là một yêu cầu quan trọng và là thành phần không thể thiếu được của các thiết bị số. Chẳng hạn trong một máy tính số, bộ nhớ trong chứa các con số cần thiết để thực hiện những lệnh mà theo đó máy tính có thể hoàn tất công việc của mình với sự tham gia ít nhất của con người.

Bộ nhớ bán dẫn được sử dụng làm **bộ nhớ chính** trong các máy tính nhờ vào khả năng thỏa mãn tốc độ truy xuất dữ liệu của bộ xử lý trung tâm (CPU).

Phần tử có tính nhớ là các Flip - flop . Khi liên kết các FF theo các nhóm tạo thành thanh ghi để lưu trữ và dịch chuyển thông tin. Các FF chính là các phần tử nhớ tốc độ cao được dùng rất nhiều trong việc điều hành bên trong máy tính, nơi mà dữ liệu dịch chuyển liên tục từ nơi này đến nơi khác.

Tiến bộ trong công nghệ chế tạo LSI và VLSI cho phép kết hợp một lượng lớn FF trong một chip tạo thành các bộ nhớ với các dạng khác nhau. Những bộ nhớ bán dẫn với công nghệ chế tạo transistor lưỡng cực (BJT) và MOS là những bộ nhớ nhanh nhất và giá thành của nó liên tục giảm khi các công nghệ LSI và VLSI ngày càng được cải tiến.

Bộ nhớ bán dẫn được dùng như là **bộ nhớ trong** chính của máy tính, nơi mà việc vận hành nhanh được xem như ưu tiên hàng đầu và cũng là nơi mà tất cả dữ liệu của chương trình lưu chuyển liên tục trong quá trình thực hiện một tác vụ do CPU yêu cầu.

Mặc dù bộ nhớ bán dẫn có tốc độ làm việc cao, rất phù hợp cho bộ nhớ trong, nhưng giá thành tính trên mỗi bit lưu trữ cao khiến cho nó không thể là loại thiết bị có tính chất lưu trữ khối (mass storage), là loại thiết bị có khả năng lưu trữ hàng tỷ bit mà không cần cung cấp năng lượng và được dùng như là **bộ nhớ ngoài** (đĩa từ , băng từ , CD ROM . . .). Tốc độ xử lý dữ liệu ở bộ nhớ ngoài tương đối chậm nên khi máy tính làm việc thì dữ liệu từ bộ nhớ ngoài được chuyển vào bộ nhớ trong.

**6.1.2. Các thông số cơ bản của bộ nhớ**

Khi lựa chọn hay so sánh với các bộ nhớ, người ta thường quan tâm đến các đại lượng sau:

+ *Từ nhớ*: là một nhóm các bit trong bộ nhớ dùng biểu diễn các lệnh hay dữ liệu dưới dạng dãy số nhị phân. Thí dụ một thanh ghi 8 FF là một phần tử nhớ lưu trữ từ 8 bit. Kích thước của từ nhớ trong máy tính hiện đại có chiều dài từ 4 đến 64 bit

+ *Dung lượng bộ nhớ(capacity)*: khối lượng thông tin hay dữ liệu có thể lưu trữ trong bộ nhớ. Đại lượng đặc trưng cho dung lượng bộ nhớ là số bit (hoặc *Kilobit*, *Megabit*)

**Thí dụ**:

1 byte = 8 bit

1KB = 210 bit = 1024 bit

1MB = 1KBx1KB = 210x210 bit= 1048576 bit

+ *Thời gian truy cập*(access time): là thời gian bộ nhớ bắt đầu nhận một địa chỉ mới cho tới khi dữ liệu khả dụng ở ngõ ra bộ nhớ. Đây chính là thời gian cần thiết để hoàn tất một tác vụ đọc

**6.1.3. Phân loại**

Bộ nhớ bán dẫn được chia ra làm ba loại:

+ *Bộ nhớ truy cập ngẫu nhiên RAM(Randon Access Memory)*: là bộ nhớ bán dẫn tác động nhanh có thể ghi và đọc dữ liệu ra tại cùng một thời điểm.

+ *Bộ nhớ chỉ đọc ROM(Read Only Memory)*: là bộ nhớ chỉ có khả năng đọc dữ liệu ra từ bộ nhớ.

+ *Thiết bị logic lập trình được (Programmable Logic Devices, PLD)* có thể nói điểm khác biệt giữa PLD với ROM và RAM là qui mô tích hợp của PLD thường không lớn như ROM và RAM và các tác vụ của PLD thì có phần hạn chế.

**6.2 BỘ NHỚ TRUY CẬP NGẪU NHIÊN RAM**

**6.2.1. Khái niệm**

Bộ nhớ RAM thường được sử dụng trong các bộ nhớ vừa có khả năng ghi và đọc dữ liệu vào. Sau khi nhập thông tin dữ liệu vào, RAM sẽ nhớ trong thời gian dài và thông tin lưu trữ trong RAM có thể được lấy ra bất kỳ lúc nào trước khi không còn nguồn điện.

**6.2.2. Phân loại**

Tùy thuộc vào tính chất và đặc điểm của từng loại RAM mà có thể chia ra hai loại sau:

+ *Bộ nhớ RAM dùng transistor lưỡng cực*: sử dụng FF làm đơn vị nhớ cơ bản nên có vận tốc rất cao, thời gian truy cập vào khoảng vài chục nanosec và công suất tiêu thụ khoảng 1nanowat/bit.

+ *Bộ nhớ RAM dùng MOSFET*: Khi ngắt nguồn nuôi các dữ liệu đã nhớ bị xóa. Loại RAM này có hai loại.

* RAM tĩnh(Static): cấu tạo từ các trigơ lưỡng cực, MOS hoặc CMOS. Dữ liệu sẽ còn nguyên vẹn khi duy trì nguồn điện.
* RAM động(Dynamic): cấu tạo từ các transistor, MOS hoặc CMOS có thêm các tụ điện để lưu trữ số liệu

**6.2.2.1. RAM tĩnh(SRAM - Static RAM)**

Cấu trúc tổng quát của RAM tĩnh là được tổ chức dưới dạng ma trận các ô nhớ. Mỗi ô nhớ là phần tử FF – D hoặc FF – RS có kích thích bởi xung đầu vào. Để có thể ghi và đọc thông tin một cách dễ dàng, các ô nhớ được sắp xếp và định vị theo hàng và cột. Việc truy cập vào từng ô nhớ để đọc hoặc ghi thông tin được giải quyết nhờ khối giải mã địa chỉ hàng và cột

*Đặc tính*

+ Bộ nhớ này là tĩnh bởi vì dữ liệu được lưu trữ vĩnh viễn trong mạch chừng nào còn cung cấp nguồn điện cho nó.

+ Tốc độ truy cập thông tin cao

+ Phương pháp chế tạo đơn giản và kích thước của một đơn vị nhớ nhỏ nên có thể cho phép chế tạo bộ nhớ với dung lượng lớn.

+ Thời gian thâm nhập lớn hơn loại lưỡng cực vì có điện dung ký sinh ở cực cổng của MOSFET

+ Có thể chế tạo theo phương pháp kênh P, kênh N hay CMOS

IC RAM tĩnh là IC7489 lưu trữ được 64 bit. Sơ đồ chân của IC này được thể hiện trên hình 6.1



*Hình 6.1: Sơ đồ chân của IC 7489*

Trong đó:

+ : ngõ vào cho phép được sử dụng để chọn RAM trong lúc đọc hay ghi.

+ : ngõ vào cho phép ghi(tích cực ở mức thấp)

+ D1 ÷ D4 : ngõ vào dữ liệu

+ A0 ÷ A3 : ngõ vào địa chỉ

+ : ngõ ra dữ liệu

**Thí dụ**:

Một mạch ghi thông tin 0110 vào vị trí từ 3 có địa chỉ A3A2A1A0 = 0110. Từ nhớ được đặt trong bộ nhớ bằng cách đặt số nhị phân 0011 đến ngõ vào địa chỉ của RAM. Để nhập 0110 vào địa chỉ trên, đặt 0 ở D1, 1 ở D2, 1 ở D3, 0 ở D4 và , dữ liệu được ghi vào bộ nhớ

**6.2.2.2. RAM động(DRAM - Dynamic RAM)**

RAM động lợi dụng các điện dung ký sinh giữa các cực của transistor MOS, trạng thái tích điện hay không của tụ tương ứng với hai bit 1 và 0. Do RAM động có mật độ tích hợp cao, dung lượng bộ nhớ thường rất lớn nên để định vị các phần tử nhớ người ta dùng phương pháp đa hợp địa chỉ, mỗi từ nhớ được chọn khi có đủ hai địa chỉ hàng và cột được lần lượt tác động.

*Đặc tính*

*+* Khi không đọc – viết, mạch hầu như không tiêu thụ công suất. Vì vậy loại RAM này tiêu thụ rất ít công suất.

+ Thời gian thâm nhập bé hơn các loại RAM tĩnh vì cấu trúc của mạch đơn giản hơn.

+ Tốc độ truy cập chậm hơn RAM tĩnh, vì có dòng điện dò của tụ nên chỉ trong một thời gian ngắn là thông tin trong DRAM bị mất do vậy DRAM phải được nạp lại thông tin một lần gọi là chế độ làm tươi của DRAM

Trên hình 6.2 biểu diễn một bộ nhớ có dung lượng 16K x 8 bit dùng 8 vi mạch loại TMS4116. Lưu ý tất cả các đường địa chỉ và đường điều khiển của các vi mạch đều phải nối song song với nhau. Lúc đó nếu thao tác đọc (hoặc viết) thì tất cả các vi mạch được đặt ở chế độ đọc (hoặc viết). Các đường dữ liệu tạo thành một hệ thống nhớ có cỡ 8 bit



*Hình 6.2: Bộ nhớ có dung lượng 16K x 8bit dùng 8 vi mạch loại TMS 4116*

*Trong đó:*

****(Read/Write - Đọc/Viết): Cho phép **đọc** dữ liệu ra khi ở mức cao và **ghi** dữ liệu vào khi ở mức thấp

****(Chip select - Chọn chip): Khi chân này xuống mức thấp thì IC được chọn

****(Row Address Strobe): Chốt địa chỉ hàng.

**(**Column Address Strobe): Chốt địa chỉ cột

Trong trường hợp chip nhớ có dung lượng lớn, để giảm kích thước của mạch giải mã địa chỉ bên trong IC, người ta chia số chân ra làm 2: địa chỉ hàng và địa chỉ cột. Như vậy phải dùng 2 mạch giải mã địa chỉ nhưng mỗi mạch nhỏ hơn rất nhiều. Thí dụ với 10 chân địa chỉ, thay vì dùng 1 mạch giải mã 10 đường sang 1024 đường, người ta dùng 2 mạch giải mã 5 đường sang 32 đường, hai mạch này rất đơn giản so với một mạch kia. Một vị trí nhớ có 2 địa chỉ : hàng và cột, dĩ nhiên muốn truy xuất một vị trí nhớ phải có đủ 2 địa chỉ nhờ 2 tín hiệu **** và ****.

**6.3 BỘ NHỚ CHỈ ĐỌC ROM**

**6.3.1. Khái niệm**

ROM (Read Only Memory) là bộ nhớ có khả năng đọc mà không viết ngay dữ liệu vào được. Dữ liệu trong bộ nhớ được ghi vào từ trước và thường được lưu trữ một cách vĩnh viễn.

Các ROM được sắp xếp như bộ nhớ không xác định vì nó không làm mất dữ liệu khi mất nguồn điện. Dữ liệu được nạp vào trong ROM do nhà chế tạo thực hiện khi sản xuất có thể là hằng số vật lý, toán học như số **,** số e hay công thức toán học. Ngoài ra dữ liệu cũng có thể là các lệnh điều khiển khởi động máy tính, các chương trình con điều khiển sự hoạt động của máy tính hay các thiết bị điều khiển tự động

**6.3.2. Phân loại**

- ***PROM (Program ROM)***: là bộ nhớ chỉ đọc có khả năng lập trình. Khi đã lập trình được thì nội dung ghi trong PROM không thể sửa hoặc xóa được. Các vi mạch loại PROM thường được sử dụng là: 54/74LS188(32 x 8bit), 54/74S287 (256 x 4bit), 54/74S472(512 x 8bit).

- ***EPROM (Erase Program ROM)***: là bộ nhớ có thể lập trình hóa, nội dung có thể xóa hoặc nạp lại được khi thay đổi chương trình. EPROM có hai loại:

*+* *UV – EPROM (Ultra Violet)*: Bộ nhớ lập trình hóa có thể xóa bằng tia cực tím. Ta có thể dễ dàng nhận ra hình dạng bên ngoài của nó khi được đóng trong cùng một vỏ nhựa có cửa sổ tròn trong suốt. Nhờ cửa sổ này có thể xóa nội dung thông tin ghi trong EPROM bằng cách dùng đèn tử ngoại chiếu tia cực tím vào chip. Bộ nhớ sau khi xóa có thể lập trình lại. Khi dùng loại EPROM này cần che cửa sổ để tránh ánh sáng mặt trời dọi vào làm mất thông tin ghi trong mạch.

Các vi mạch UV – EPROM của hãng National Semiconductor có các ký hiệu sau: M2708 (1028 x 8 bit), MM2704(512 x 8 bit).

*+ E – EPROM (Electrically Program and Erasable ROM):* Bộ nhớ lập trình xóa bằng xung điện 20V.

**CHƯƠNG 7: CHUYỂN ĐỔI TÍN HIỆU**

**7.1 MẠCH CHUYỂN ĐỔI SÔ – TƯƠNG TỰ**

**7.1.1. Khái niệm**

Mạch chuyển đổi số - tương tự (Digital to Analog Converter - DAC) là một mạch dùng để chuyển tín hiệu đầu vào là các con số thành tín hiệu tương tự ở đầu ra là giá trị điện áp hoặc dòng điện.

**7.1.2. Cấu trúc**

Xét bộ chuyển đổi DAC với tín hiệu đầu vào số 4 bit và đầu ra là giá trị điện áp trong khoảng 0 ÷ 3V được cho như bảng chân lý sau:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **STT** | **Ngõ vào số** | | | | **Ngõ ra tương tự**  **(Vi)** |
| **D** | **C** | **B** | **A** |
| Hàng 1  Hàng 2  Hàng 3  Hàng 4  Hàng 5  Hàng 6  Hàng 7  Hàng 8  Hàng 9  Hàng 10  Hàng 11  Hàng 12  Hàng 13  Hàng 14  Hàng 15  Hàng 16 | 0  0  0  0  0  0  0  0  1  1  1  1  1  1  1  1 | 0  0  0  0  1  1  1  1  0  0  0  0  1  1  1  1 | 0  0  1  1  0  0  1  1  0  0  1  1  0  0  1  1 | 0  1  0  1  0  1  0  1  0  1  0  1  0  1  0  1 | 0  0,2  0,4  0,6  0,8  1,0  1,2  1,4  1,6  1,8  2,0  2,2  2,4  2,6  2,8  3,0 |

Theo bảng chân lý ta nhận thấy:

+ Nếu số nhị phân 0000 xuất hiện ở ngõ vào của bộ biến đổi D/A thì ngõ ra tương tự sẽ là 0V

+ Nếu số nhị phân 0001 xuất hiện ở ngõ vào thì ngõ ra tương ứng là 0,2V

+ Nếu số nhị phân 1111 xuất hiện ở ngõ vào thì ngõ ra tương ứng là 3,0V

Như vậy, nếu ta tác động lần lượt tín hiệu đầu vào theo bảng chân lý từ hàng 1 đến hàng 16 thì điện áp đầu ra Analog của bộ chuyển đổi lần lượt tăng mỗi lần 0,2V.

Sơ đồ khối cơ bản của bộ chuyển đổi DAC có dạng như hình 7.1

Ngõ ra Analog



Bộ khuếch đại cộng

Hệ thống điện trở

Ngõ vào Digital

*Hình 7.1: Sơ đồ khối cơ bản của DAC*

Hệ thống gồm 2 phần: hệ thống điện trở và bộ khuếch đại cộng, ngõ ra được đo bằng voltkế.

+ *Hệ thống điện trở:* được sắp xếp theo dạng bậc thang điện trở theo yêu cầu mức 1 ở ngõ vào trước gấp hai lần mức 1 ở ngõ vào sau.

+ *Bộ khuếch đại cộng*: lấy điện áp ngõ ra từ hệ thống điện trở sau đó khuếch đại để lấy điện áp đầu ra đạt được các giá trị mong muốn.

Bộ khuếch đại cộng điển hình là bộ khuếch đại thuật toán dùng IC chuyên dụng.

**7.1.3 Bộ khuếch đại thuật toán(OP - AMP)**

Bộ khuếch đại thuật toán có đặc điểm là trở kháng ngõ vào cao, trở kháng ngõ ra thấp và độ lợi áp có thể thay đổi khi ta điều chỉnh các giá trị điện trở bên ngoài.

- Ký hiệu

Ngõ vào không đảo



Ngõ ra

Ngõ vào đảo

*Hình 7.2: Bộ khuếch đại thuật toán*

Sơ đồ bộ khuếch đại không đảo có dạng như hình 7.3



*Hình 7.3: Bộ khuếch đại thuật toán không đảo*

Theo định luật Kiechhop ta có: suy ra 

Độ lợi áp của bộ khuếch đại: .

**Thí dụ**: Cho bộ khuếch đại không đảo như hình 7.3 với Rin = 10K,

Rht = 10K, điện áp ngõ vào 5v. Xác định điện áp ngõ ra?

Độ lợi áp của bộ khuếch đại là: 

Vậy Ur = Av.Uv = 1 x 5 = 5V

**7.1.4. Bộ biến đổi D/A cơ bản**

Sơ đồ mạch điện của bộ biến đổi D/A cơ bản là:



*Hình 7.4: Sơ đồ bộ biến đổi DAC có bản*

Với R1 = 150K, R2 = 75K, R3 = 37.5K, R4 = 18.75K và Rht = 10 K.

- Giả sử hệ thống chuyển mạch vị trí A

Khi đó tương ứng với dãy số nhị phân ở đầu vào là DCBA = 0001

Ta có: 

Suy ra: Ur = Av.Uv = 0.066 x 5V = 0.2V

- Giả sử hệ thống chuyển mạch vị trí D

Khi đó tương ứng với dãy số nhị phân ở đầu vào là DCBA = 1000

Ta có: 

Suy ra: Ur = Av.Uv = 0.535 x 5V = 1.6V

- Nếu công tắc A và B chuyển mạch thì dãy số nhị phân là: DCBA = 0011 tương ứng với hai điện trở R1//R2. Khi đó Rtđ = 50K và điện áp đầu ra Ur = 0.6v.

*Chú ý*: Sử dụng điện áp ngõ vào sao cho điện áp này không vượt quá điện áp giới hạn của bộ khuếch đại thuật toán. Giả sử ở sơ đồ trên là và có thể thêm nhiều đầu vào vị trí (trọng số) bằng cách thêm các chuyển mạch.

- Trong thực tế, sử dụng IC 0808 làm bộ chuyển đổi số sang tương tự.

**7.2 MẠCH CHUYỂN ĐỔI TƯƠNG TỰ - SỐ**

**7.2.1. Khái niệm**

Mạch chuyển đổi tương tự - số (Analog to Digital Converter - ADC) là mạch biến đổi tín hiệu tương tự thành tín hiệu số có mã số tỷ lệ với giá trị biên độ tín hiệu tương tự ở đầu vào.

**7.2.2 Mạch lấy mẫu và giữ mẫu**

Khi biến đổi một tín hiệu tương tự sang tín hiệu số, người ta không thể biến đổi mọi giá trị của tín hiệu tương tự mà chỉ có thể biến đổi một số giá trị cụ thể bằng cách **lấy mẫu** tín hiệu đó theo một chu kỳ xác định nhờ một tín hiệu có dạng xung. Ngoài ra, mạch biến đổi cần một khoảng thời gian cụ thể (khoảng 1μs - 1ms) do đó **cần giữ mức tín hiệu** biến đổi trong khoảng thời gian này để mạch có thể biến đổi chính xác. Vì vậy nhiệm vụ của mạch là lấy mẫu ở những thời điểm xác định và duy trì giá trị đó cho đến thời điểm lấy mẫu tiếp theo.

Hình 7.5 là dạng mạch lấy mẫu và giữ mẫu cơ bản: Điện thế tương tự cần biến đổi được lấy mẫu trong thời gian rất ngắn do tụ nạp điện nhanh qua tổng trở ra thấp của OP-AMP khi các transistor dẫn và giữ giá trị này trong khoảng thời gian transistor ngưng (tụ phóng rất chậm qua tổng trở vào rất lớn của OP-AMP)

|  |  |
| --- | --- |
|  |  |

*Hình 7.5: Mạch lấy mẫu và giữ mẫu cơ bản*

**7.2.3. Bộ biến đổi A/D cơ bản**

Sơ đồ khối của bộ biến đổi A/D có dạng như hình 7.6



Bộ biến đổi D/A

Bộ

so sánh điện áp

Mạch đếm CLK

Ngõ vào Analog

Ngõ ra số

Xung nhịp

*Hình 7.6: Bộ biến đổi ADC*

Bộ biến đổi A/D bao gồm một bộ so sánh điện áp, cổng AND, mạch đếm nhị phân và bộ biến đổi D/A. Bộ so sánh điện áp sẽ kiểm tra điện áp hồi tiếp từ bộ biến đổi D/A với điện áp ngõ vào. Nếu điện áp ngõ vào analog ở điểm A lớn hơn điện áp ở điểm B của bộ so sánh, xung nhịp được đưa vào trong mạch đếm 4 bit. Thực hiện đếm trong mạch đếm vẫn tiếp tục cho đến khi điện áp hồi tiếp từ bộ biến đổi D/A lớn hơn điện áp ngõ vào analog thì bộ so sánh sẽ làm dừng phép đếm trong mạch đếm.

|  |  |
| --- | --- |
| Trong thực tế, ADC0804 là bộ chuyển đổi tương tự sang số thuộc họ ADC800 của hãng National Semiconductor. Chip này cũng được nhiều hãng khác sản xuất. Chip có điện áp nuôi +5V và độ phân giải 8 bit.  Ngoài độ phân giải thì thời gian chuyển đổi cũng là một tham số quan trọng khi đánh giá bộ ADC. Thời gian chuyển đổi được định nghĩa là thời gian mà bộ ADC cần để chuyển một đầu vào tương tự thành một số nhị phân. Đối với ADC0804 thì thời gian chuyển đổi phụ thuộc vào tần số đồng hồ được cấp tới chân CLK R và CLK IN và không bé hơn 110μs. | *adc0804smlHình 7.7: Bộ chuyển đổi ADC0804* |

Các chân khác của ADC0804 có chức năng như sau:

**+ CS (Chip select):** (chân số 1) là chân chọn Chip, đầu vào tích cực mức thấp được sử dụng để kích hoạt chip ADC0804

**+ RD (Read):** (chân số 2) là một tín hiệu vào, tích cực ở mức thấp. RD được sử dụng để có dữ liệu đã được chuyển đổi tới đầu ra của ADC0804. Khi CS = 0 nếu có một xung cao xuống thấp áp đến chân RD thì dữ liệu ra dạng số 8 bit được đưa tới các chân dữ liệu (DB0 – DB7).

**+ WR (Write):** (chân số 3) là chân vào tích cực mức thấp được dùng để báo cho ADC biết bắt đầu quá trình chuyển đổi. Nếu CS = 0 khi WR tạo ra xung cao xuống thấp thì bộ ADC0804 bắt đầu quá trình chuyển đổi giá trị đầu vào tương tự Vin về số nhị phân 8 bit. Khi việc chuyển đổi hoàn tất thì chân INTR được ADC hạ xuống mức thấp

**+ CLK IN và CLK R:** (chân số 4 và chân 19)

* CLK IN là chân nối tới đồng hồ ngoài được sử dụng để tạo thời gian.
* Tuy nhiên ADC0804 cũng có một bộ tạo xung đồng hồ riêng. Để dùng đồng hồ riêng thì các chân CLK IN và CLK R được nối với một tụ điện và một điện trở như hình vẽ. Khi ấy tần số được xác định bằng biểu thức:



* Với R=10 kΩ, C=150pF và tần số f=606 kHz và thời gian chuyển đổi là 110μs.

**+ Ngắt INTR (Interupt):** (chân số 5) là chân ra tích cực mức thấp. Bình thường chân này ở trạng thái cao,. Khi INTR xuống thấp, cần đặt CS = 0V và gửi một xung cao xuống thấp tới chân RD để lấy dữ liệu.

+ **Vin (+) và Vin (-):** (chân số 6) và (chân số 7) đây là 2 đầu vào tương tự vi sai, trong đó Vin = Vin (+) – Vin (–). Thông thường Vin (–) được nối đất và Vin (+) được dùng làm đầu vào tương tự và sẽ được chuyển đổi về dạng số.

+ **Vcc:** (chân số 20) là chân nguồn nuôi +5V. Chân này còn được dùng làm điện áp tham chiếu khi đầu vào Vref/2 để hở.

+ **Vref/2:** (chân số 9) là chân điện áp đầu vào được dùng làm điện áp tham chiếu. Nếu để hở thì đầu vào tương tự có điện áp tham chiếu 0→5V. Khi ứng dụng có đầu vào tương tự cần tham chiếu với dải điện áp khác thì chân Vref/2 được dùng để thực hiện các điện áp tham chiếu tùy chọn 0 → Vref.

**THỰC HÀNH KỸ THUẬT SỐ**

**A. GIỚI THIỆU CÁC MÔ ĐUN TRÊN PANEN ĐIỆN TỬ SỐ:**

**1. Phần nguồn gồm:**

|  |  |
| --- | --- |
| Nguồn một chiều ổn áp: +5V bố trí như sau: |  |

**2. Nguồn xung:**

|  |  |
| --- | --- |
| Nguồn xung tần số thấp có đèn LED chỉ thị và có thể thay đổi được tần số bằng cách thay đổi biến trở VR. |  |

**3. Các chuyển mạch đầu vào:**

Có 8 công tắc gạt cung cấp lối ra 8 bit.

**4. Các đèn chỉ thị logic được sử dụng cho đầu ra:**

LED chỉ thị trạng thái logic cao (sáng), thấp ( tắt).

 

Các chuyển mạch đầu vào

Các đèn chỉ thị đầu ra

**5. Cổng AND :**

Có 4 cổng AND hai đầu vào và 2 cổng AND 3 đầu vào



**7. Cổng NAND :**

Có 4 cổng NAND hai đầu vào và 2 cổng NAND 3 đầu vào



**8. Cổng OR :**

Có 4 cổng OR hai đầu vào và 2 cổng OR 3 đầu vào



**9. Cổng NOR :**

Có 4 cổng NOR hai đầu vào và 2 cổng NOR 3 đầu vào



**10. Cổng X – OR :**

Có 4 cổng X-OR hai đầu vào.



**11. Cổng NOT:**

Có 5 cổng NOT.



**B. MỘT SỐ BÀI THỰC HÀNH TRÊN PANEN THÍ NGHIỆM CÁC MẠCH LÔ GÍC:**

**Bài thực hành số 1**

**MỨC LOGIC**

**I. MỤC ĐÍCH THÍ NGHIỆM:**

* Tìm hiểu về bản chất mức logic và sự tồn tại vật lý của chúng.

**II. TÓM TẮT KIẾN THỨC CÓ LIÊN QUAN:**

* Trong kỹ thuật số, các giá trị đúng, sai của một biến logic thường được ký hiệu là 1 và 0. Cần lưu ý là: 1 và 0 ở đây là các ký hiệu mà không phải là các chữ số hệ hai.
* Biện pháp kỹ thuật để thực hiện các giá trị logic phụ thuộc vào việc chọn các trị số vật lý để biểu diễn. Chẳng hạn, đối với logic dương, với các vi mạch thuộc họ TTL thì mức +5V biểu diễn 1 còn mức 0V biểu diễn 0, nhưng với các vi mạch CMOS thì 1 thể hiện mức từ +3V đến +18V và 0 thể hiện mức 0V. Đối với logic âm thì ngược lại.

**III. CÁC BƯỚC TIẾN HÀNH THÍ NGHIỆM:**

1) Nối các dây dẫn theo sơ đồ nguyên lý hình 1-1.



*Hình 1-1*

2) Sử dụng các công tắc cho các đầu vào để tạo mức logic, sử dụng đèn LED cho việc biểu thị mức logic.

3) Bật công tắc theo các vị trí ký hiệu “1” và “0”. Điền vào bảng kết quả thu được khi tiến hành thí nghiệm.

**IV. BÁO CÁO THÍ NGHIỆM:**

1) Số thứ tự, tên bài thực hành.

2) Mục đích của bài thực hành.

3) Trình tự đã tiến hành thí nghiệm và kết quả thu được.

|  |  |  |
| --- | --- | --- |
| Điện áp | Mức logic | Đèn LED |
| +5V | 1 |  |
| 0V | 0 |  |

**V. CÂU HỎI KIỂM TRA:**

Đối với logic âm thì đèn LED sẽ chỉ thị như thế nào?

**Bài thực hành số 2**

**CỔNG AND**

**I. MỤC ĐÍCH THÍ NGHIỆM:**

* Tìm hiểu hoạt động và ký hiệu của cổng AND.
* Xây dựng bảng chân lý cho cổng AND.

**II. TÓM TẮT KIẾN THỨC CÓ LIÊN QUAN:**

*1) Cổng AND 2 đầu vào:*

Cổng AND hai đầu vào có thể được mô tả như hình 2-1a. Trong đó, đèn Y sáng khi cả hai công tắc X1, X2 đóng lại.

X1

X2

Y

- +

*Hình 2-1*

|  |  |  |
| --- | --- | --- |
| Khoá X1 | Khoá X2 | Đèn Y |
| Mở | Mở | Tắt |
| Mở | Đóng | Tắt |
| Đóng | Mở | Tắt |
| Đóng | Đóng | Sáng |

*Bảng 2-1*

- Như trên bảng 2-1, có 4 sự kết hợp các trường hợp đóng và mở của hai công tắc X1 và X2. Bảng này được gọi là “Bảng chân lý”. Từ hình 2-1 và bảng 2-1 thấy rằng, đầu ra, tức đèn Y chỉ sáng khi cả hai đầu vào, tức là 2 khoá X1 và X2 đều đóng.

- Ký hiệu logic của cổng AND có hình dạng như hình 2-2a. Các đầu vào được biểu diễn bởi X1 và X2, đầu ra là Y. Bảng chân lý của cổng AND hai đầu vào được trình bày ở bảng 2-2b và các đầu vào được biểu diễn bởi các số “0” và “1”. Mức logic 0 tương đương với điện áp là 0V và mức logic 1 tương đương với điện áp +5V.



*Hình 2-2a: Ký hiệu của cổng AND 2 đầu vào*

|  |  |  |
| --- | --- | --- |
| Đầu vào | | Đầu ra |
| X1 | X2 | Y |
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

*Hình* 2-2b: *Bảng chân lý của cổng AND*

* Phép toán logic:

Ký hiệu phép toán logic của cổng AND là dấu “.” và có thể viết là: Y = X1.X2

*2) Cổng AND có 3 đầu vào:*

Như hình 2-3a, cổng AND có 3 đầu vào X1, X2, X3 được gọi là cổng AND 3 đầu vào



*Hình 2-3a Cổng AND 3 đầu vào*

* Bảng chân lý: Bảng 2-3b là bảng chân lý của cổng AND 3 đầu vào

|  |  |  |  |
| --- | --- | --- | --- |
| **Đầu vào** | | | **Đầu ra** |
| X1 | X2 | X3 | Y |
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 0 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 0 | 0 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 0 |
| 1 | 1 | 1 | 1 |

*Bảng 2-3b Bảng chân lý của cổng AND 3 đầu vào*

Phép toán logic đối với cổng AND 3 đầu vào X1, X2, X3được viết là: Y = X1.X2.X3

**III. CÁC BƯỚC TIẾN HÀNH THÍ NGHIỆM:**

1) Nối các dây dẫn theo sơ đồ nguyên lý hình 2-4.



*Hình 2-4*

2) Sử dụng các công tắc cho các đầu vào X1 và X2, sử dụng đèn LED cho đầu ra Y.

3) Điền vào bảng chân lý kết quả thu được khi tiến hành thí nghiệm.

|  |  |  |
| --- | --- | --- |
| **Đầu vào** | | **Đầu ra** |
| X1 | X2 | Y |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

4) Nối các dây dẫn theo sơ đồ nguyên lý hình 2-5.



*Hình 2-5*

2) Sử dụng các công tắc cho các đầu vào X1, X2, X3 và sử dụng đèn LED cho đầu ra Y.

3) Điền vào bảng kết quả thu được khi tiến hành thí nghiệm.

|  |  |  |  |
| --- | --- | --- | --- |
| **Đầu vào** | | | **Đầu ra** |
| X1 | X2 | X3 | Y |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**IV. BÁO CÁO THÍ NGHIỆM:**

1) Số thứ tự, tên bài thực hành.

2) Mục đích của bài thực hành.

3) Trình tự đã tiến hành thí nghiệm và kết quả thu được.

**V. CÂU HỎI KIỂM TRA:**

Khi thay đổi vị trí của X1, X2, X3 trong bảng kết quả thì đầu ra Y có thay đổi không? Tại sao?

**Bài thực hành số 3**

**CỔNG OR**

**I. MỤC ĐÍCH THÍ NGHIỆM:**

* Tìm hiểu hoạt động và ký hiệu của cổng OR.
* Xây dựng bảng chân lý cho cổng OR.

**II. TÓM TẮT KIẾN THỨC CÓ LIÊN QUAN:**

*1) Cổng OR hai đầu vào:*

Cổng OR có thể được mô tả như hình 3-1a. Trong đó, đèn Y sáng khi một trong hai công tắc X1, X2 đóng lại.

## X1

|  |  |  |
| --- | --- | --- |
| **Đầu vào** | | **Đầu ra** |
| Khoá X1 | Khoá X2 | Đèn Y |
| Mở | Mở | Tắt |
| Mở | Đóng | Sáng |
| Đóng | Mở | Sáng |
| Đóng | Đóng | Sáng |

## X2

## Y

- +

*Hình 3-1a*

*Bảng 3-1b*

Như ở bảng 3-1b, có 4 sự kết hợp các trường hợp đóng và mở của hai công tắc X1 và X2.Từ hình 3-1a và bảng 3-1b chúng ta thấy rằng, đầu ra, tức đèn Y sáng khi một trong hai đầu vào, tức là một trong hai khoá X1 và X2 đóng.

Cổng OR hai đầu vào có hình dạng như hình 3-2a. Các đầu vào được biểu diễn bởi X1 và X2, đầu ra là Y. Nếu có hai đầu vào, chúng ta gọi là cổng OR hai đầu vào. Bảng chân lý của cổng OR hai đầu vào được trình bày trên bảng 3-2b



*Hình 3-2a Cổng OR hai đầu vào*

|  |  |  |
| --- | --- | --- |
| Đầu vào | | Đầu ra |
| X1 | X2 | Y |
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1  *Bảng 3-2b Bảng chân lý của cổng OR* | 1 | 1 |

Ký hiệu logic của cổng OR là dấu “+” và đầu ra Y có thể viết là Y = X1 + X2

*2) Cổng OR có 3 đầu vào:*

Cổng OR có 3 đầu vào X1, X2, X3 được gọi là cổng OR 3 đầu vào. Xem hình 3-3a



*Hình 3-3a Cổng OR 3 đầu vào*

Bảng chân lý: Bảng 3-3b là bảng chân lý của cổng OR 3 đầu vào

|  |  |  |  |
| --- | --- | --- | --- |
| **Đầu vào** | | | **Đầu ra** |
| X1 | X2 | X3 | Y |
| 0 | 0 | 0 | 0 |
| 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 |

Bảng 3-3b *Bảng chân lý của cổng AND 3 đầu vào*

Phép toán logic đối cổng OR 3 đầu vào được viết là: Y = X1 + X2 + X3

**III. CÁC BƯỚC TIẾN HÀNH THÍ NGHIỆM:**

1) Nối các dây dẫn theo sơ đồ nguyên lý hình 3-4.



*Hình 3-4*

2) Sử dụng các công tắc cho các đầu vào X1 và X2, sử dụng đèn LED cho đầu ra Y.

3) Điền vào bảng chân lý kết quả thu được khi tiến hành thí nghiệm

|  |  |  |
| --- | --- | --- |
| Đầu vào | | Đầu ra |
| X1 | X2 | Y |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

4) Nối các dây dẫn theo sơ đồ nguyên lý hình 3-5.



*Hình 3-5*

5) Sử dụng các công tắc cho các đầu vào X1, X2 và X3 sử dụng đèn LED cho đầu ra Y.

6) Điền vào bảng chân lý kết quả thu được khi tiến hành thí nghiệm:

|  |  |  |  |
| --- | --- | --- | --- |
| **Đầu vào** | | | **Đầu ra** |
| X1 | X2 | X3 | Y |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**IV. BÁO CÁO THÍ NGHIỆM:**

1) Số thứ tự, tên bài thực hành.

2) Mục đích của bài thực hành.

3) Trình tự đã tiến hành thí nghiệm và kết quả thu được.

**VI. CÂU HỎI KIỂM TRA:**

So sánh hai bảng kết quả thí nghiệm của cổng AND và cổng OR có thể rút ra nhận xét gì?

**Bài thực hành số 4**

**CỔNG NOT**

**I. MỤC ĐÍCH THÍ NGHIỆM:**

* Tìm hiểu hoạt động và ký hiệu của cổng NOT.
* Xây dựng bảng chân lý cho cổng NOT.

**II. TÓM TẮT KIẾN THỨC CÓ LIÊN QUAN:**

*1) Cổng NOT :*

* Cổng NOT có tên khác là “bộ đảo” và nó chỉ có một đầu vào và một đầu ra. Hoạt động của nó là: trạng thái ở đầu ra là tín hiệu đảo của tín hiệu ở đầu vào.
* Cổng NOT có ký hiệu logic được mô tả như hình 4-1a. Trong đó, trạng thái của đầu vào luôn ngược với đầu ra. Hình 4-1b là bảng chân lý của cổng NOT.

|  |  |
| --- | --- |
| **Đầu vào** | **Đầu ra** |
| 0 | 1 |
| 1 | 0 |



*Hình 4-1a*

*Bảng 4-1b*

Ký hiệu phép toán logic của cổng OR là dấu “­”: 

Nếu sử dụng hai cổng đảo thì đầu ra: 

**III. CÁC BƯỚC TIẾN HÀNH THÍ NGHIỆM:**

1) Nối các dây dẫn theo sơ đồ nguyên lý hình 4-3.



*Hình 4-3*

2) Sử dụng các công tắc cho đầu vào X, sử dụng đèn LED cho đầu ra Y.

3) Điền vào bảng chân lý kết quả thu được khi tiến hành thí nghiệm

|  |  |
| --- | --- |
| Đầu vào | Đầu ra |
| X | Y |
| 0 |  |
| 1 |  |

4) Nối các dây dẫn theo sơ đồ nguyên lý hình 4-4.



*Hình 4-4*

5) Sử dụng các công tắc cho đầu vào X, sử dụng đèn LED cho các đầu ra Y1, Y2.

6) Điền vào bảng chân lý kết quả thu được khi tiến hành thí nghiệm:

|  |  |  |
| --- | --- | --- |
| **Đầu vào** | **Đầu ra** | |
| X | Y1 | Y2 |
| 0 |  |  |
| 1 |  |  |

7) Có thể tăng số cổng NOT lên tới 5 cổng khi thí nghiệm để rút ra nhận xét.

**IV. BÁO CÁO THÍ NGHIỆM:**

1) Số thứ tự, tên bài thực hành.

2) Mục đích của bài thực hành.

3) Trình tự đã tiến hành thí nghiệm và kết quả thu được.

**VI. CÂU HỎI KIỂM TRA:**

Có nhận xét gì khi tiến hành thí nghiệm với số cổng đảo là chẵn và lẻ?

**Bài thực hành số 5**

**CỔNG NAND**

**I. MỤC ĐÍCH THÍ NGHIỆM:**

* Tìm hiểu hoạt động và ký hiệu của cổng NAND.
* Xây dựng bảng chân lý cho cổng NAND.

**II. TÓM TẮT KIẾN THỨC CÓ LIÊN QUAN:**

*1) Cổng NAND 2 đầu vào:*

* Ký hiệu của cổng NAND 2 đầu vào được minh họa ở hình 5-1a



*Hình 5-1a Kí hiệu của cổng NAND 2 đầu vào*

* Bảng chân lý của cổng NAND 2 đầu vào:

|  |  |  |
| --- | --- | --- |
| Đầu vào | | Đầu ra |
| X1 | X2 | Y |
| 0 | 0 | 1 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

Bảng 5-2b *Bảng chân lý của cổng NAND 2 đầu vào*

Ký hiệu phép toán logic của cổng NAND: Y = 

*2) Cổng NAND có 3 đầu vào:*

* Như hình 5-3a, cổng NAND có 3 đầu vào X1, X2, X3 được gọi là cổng NAND 3 đầu vào.



*Hình 5-3a Cổng NAND 3 đầu vào*

* Bảng chân lý: Bảng 5-3b là bảng chân lý của cổng NAND 3 đầu vào

|  |  |  |  |
| --- | --- | --- | --- |
| **Đầu vào** | | | **Đầu ra** |
| X1 | X2 | X3 | Y |
| 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 | 0 |

Bảng 5-3b *Bảng chân lý của cổng NAND 3 đầu vào*

Phép toán logic đối cổng NAND 3 đầu vào được viết là: Y = 

**III. CÁC BƯỚC TIẾN HÀNH THÍ NGHIỆM:**

1) Nối các dây dẫn theo sơ đồ nguyên lý hình 5-4.



###### Hình 5-4

2) Sử dụng các công tắc cho các đầu vào X1 và X2, sử dụng đèn LED cho đầu ra Y.

3) Điền vào bảng chân lý kết quả thu được khi tiến hành thí nghiệm.

|  |  |  |
| --- | --- | --- |
| **Đầu vào** | | **Đầu ra** |
| X1 | X2 | Y |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

Ghi chú: Nối tắt 2 đầu vào cổng NAND, loại bỏ công tắc X2 sau đó thực hiện lại thí nghiệm trên và rút ra nhận xét.

4) Nối các dây dẫn theo sơ đồ nguyên lý hình 5-5.



##### Hình 5-5

2) Sử dụng các công tắc cho các đầu vào X1, X2, X3 và sử dụng đèn LED cho đầu ra Y.

3) Điền vào bảng kết quả thu được khi tiến hành thí nghiệm.

|  |  |  |  |
| --- | --- | --- | --- |
| **Đầu vào** | | | **Đầu ra** |
| X1 | X2 | X3 | Y |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**IV. BÁO CÁO THÍ NGHIỆM:**

1) Số thứ tự, tên bài thực hành.

2) Mục đích của bài thực hành.

3) Trình tự đã tiến hành thí nghiệm và kết quả thu được.

**VI. CÂU HỎI KIỂM TRA:**

So sánh kết quả thí nghiệm khi thí nghiệm cổng AND và cổng NAND có thể rút ra kết luận gì?

Nếu nối tắt 2 đầu và của một cổng NAND ta được một phần tử lô gíc tương đương với phần tử lôgíc cơ bản nào?

**Bài thực hành số 6**

**CỔNG NOR**

**I. MỤC ĐÍCH THÍ NGHIỆM:**

* Tìm hiểu hoạt động và ký hiệu của cổng NOR.

**II. TÓM TẮT KIẾN THỨC CÓ LIÊN QUAN:**

*1) Cổng NOR 2 đầu vào:*

* Ký hiệu của cổng NOR 2 đầu vào được minh họa ở hình 6-1a



*Hình 6-1a Kí hiệu của cổng NAND 2 đầu vào*

* Bảng chân lý của cổng NOR 2 đầu vào:

|  |  |  |
| --- | --- | --- |
| Đầu vào | | Đầu ra |
| X1 | X2 | Y |
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 0 |

Hình 5-2b *Bảng chân lý của cổng NOR 2 đầu vào*

Ký hiệu phép toán logic của cổng NOR: Y = 

*2) Cổng NOR có 3 đầu vào:*

* Như hình 6-3a, cổng NOR có 3 đầu vào X1, X2, X3 được gọi là cổng NOR 3 đầu vào.



*Hình 6-3a Cổng NAND 3 đầu vào*

* Ký hiệu phép toán logic của cổng NOR 3 đầu vào: Y = 
* Bảng chân lý: Bảng 6-3b là bảng chân lý của cổng NOR 3 đầu vào

|  |  |  |  |
| --- | --- | --- | --- |
| **Đầu vào** | | | **Đầu ra** |
| X1 | X2 | X3 | Y |
| 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 0 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 0 | 0 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 0 |
| 1 | 1 | 1 | 0 |

Bảng 6-3b *Bảng chân lý của cổng NOR 3 đầu vào*

* Phép toán logic đối cổng NOR 3 đầu vào được viết là: Y = 

**III. CÁC BƯỚC TIẾN HÀNH THÍ NGHIỆM:**

1) Nối các dây dẫn theo sơ đồ nguyên lý hình 6-4.



*Hình 6-4*

2) Sử dụng các công tắc cho các đầu vào X1 và X2, sử dụng đèn LED cho đầu ra Y.

3) Điền vào bảng chân lý kết quả thu được khi tiến hành thí nghiệm.

|  |  |  |
| --- | --- | --- |
| **Đầu vào** | | **Đầu ra** |
| X1 | X2 | Y |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

Ghi chú: Nối tắt 2 đầu vào cổng NOR, loại bỏ công tắc X2 sau đó thực hiện lại thí nghiệm trên và rút ra nhận xét.

4) Nối các dây dẫn theo sơ đồ nguyên lý hình 6-5.



*Hình 6-5*

2) Sử dụng các công tắc cho các đầu vào X1, X2, X3 và sử dụng đèn LED cho đầu ra Y.

3) Điền vào bảng kết quả thu được khi tiến hành thí nghiệm.

|  |  |  |  |
| --- | --- | --- | --- |
| **Đầu vào** | | | **Đầu ra** |
| X1 | X2 | X3 | Y |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**IV. BÁO CÁO THÍ NGHIỆM:**

1) Số thứ tự, tên bài thực hành.

2) Mục đích của bài thực hành.

3) Trình tự đã tiến hành thí nghiệm và kết quả thu được.

**VI. CÂU HỎI KIỂM TRA:**

So sánh kết quả thí nghiệm khi thí nghiệm cổng OR và cổng NOR có thể rút ra kết luận gì?

Nếu nối tắt 2 đầu và của một cổng NOR ta được một phần tử lô gíc tương đương với phần tử lôgíc cơ bản nào?

**Bài thực hành số 7**

**CỔNG X-OR**

**I. MỤC ĐÍCH THÍ NGHIỆM:**

* Tìm hiểu hoạt động, bảng chân lý và ký hiệu của cổng X-OR.

**II. TÓM TẮT KIẾN THỨC CÓ LIÊN QUAN:**

* X-OR là cổng logic hoặc – loại trừ. Bảng chân lý của cổng X-OR như ở bảng 7-1a. Qua bảng này, có thể thấy: khi số các số “1” ở đầu vào là lẻ thì đầu ra là “1”, khi không có số “1” ở đầu vào hoặc số các số “1” là chẵn thì đầu ra là “0”. Do đó, có thể gọi cổng X-OR là cổng kiểm tra số lẻ bit “1”.

|  |  |  |
| --- | --- | --- |
| **Đầu vào** | | **Đầu ra** |
| X1 | X2 | Y |
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

*Bảng 7-1 Bảng chân lý của cổng X-OR*

* Cổng OR có hình dạng như hình 7-2a. Các đầu vào được biểu diễn bởi X1 và X2, đầu ra là Y. Nếu có hai đầu vào, chúng ta gọi nó là cổng X-OR hai đầu vào.



*Hình 7-2a Cổng X-OR hai đầu* *vào*

Đầu ra Y được viết là Y = 

**III. CÁC BƯỚC TIẾN HÀNH THÍ NGHIỆM:**

1) Nối các dây dẫn theo sơ đồ nguyên lý hình 7-4.



*Hình 7-4*

2) Sử dụng các công tắc cho các đầu vào X1 và X2, sử dụng đèn LED cho đầu ra Y.

3) Điền vào bảng chân lý kết quả thu được khi tiến hành thí nghiệm

|  |  |  |
| --- | --- | --- |
| Đầu vào | | Đầu ra |
| X1 | X2 | Y |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**IV. BÁO CÁO THÍ NGHIỆM:**

1) Số thứ tự, tên bài thực hành.

2) Mục đích của bài thực hành.

3) Trình tự đã tiến hành thí nghiệm và kết quả thu được.

**VI. CÂU HỎI KIỂM TRA:**

Cổng X-OR có thể coi là sự kết hợp của những phép logic cơ bản nào?

**Bài thực hành số 8**

**XÂY DỰNG CỔNG NAND TỪ CỔNG AND VÀ CỔNG NOT**

**I. MỤC ĐÍCH THÍ NGHIỆM:**

* Tìm hiểu ứng dụng của cổng AND và cổng NOT để xây dựng cổng NAND.

**II. TÓM TẮT KIẾN THỨC CÓ LIÊN QUAN:**

* Từ các bài thí nghiệm về cổng AND và cổng NAND, có thể rút ra nhận xét sau: bảng chân lý của cổng AND và cổng NAND là đảo của nhau. Từ đó, có thể xây dựng mạch logic gồm hai cổng: cổng AND và cổng NOT có chức năng tương đương với cổng NAND. Có thể chứng minh điều này từ sơ đồ nguyên lý hình 8-1. Thật vậy: .



*Hình 8-1*

* Tương tự, cũng có thể xây dựng cổng NAND 3 đầu vào từ cổng AND 3 đầu vào và cổng NOT: 



*Hình 8 -2*

**III. CÁC BƯỚC TIẾN HÀNH THÍ NGHIỆM:**

1) Nối các dây dẫn theo sơ đồ nguyên lý hình 8-3.



*Hình 8-3*

2) Sử dụng các công tắc cho các đầu vào X1 và X2, sử dụng đèn LED cho các đầu ra Y1 và đầu ra Y.

3) Điền vào bảng kết quả thu được khi tiến hành thí nghiệm:

|  |  |  |  |
| --- | --- | --- | --- |
| Đầu vào | | Đầu ra | |
| X1 | X2 | Y1 | Y |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

4) Nối các dây dẫn theo sơ đồ nguyên lý hình 8-4.



*Hình 8-4*

5) Sử dụng các công tắc cho các đầu vào X1 và X2, sử dụng đèn LED cho các đầu ra Y1 và đầu ra Y.

6) Điền vào bảng kết quả thu được khi tiến hành thí nghiệm:

|  |  |  |  |
| --- | --- | --- | --- |
| **Đầu vào** | | | **Đầu ra** |
| X1 | X2 | X3 | Y |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**IV. BÁO CÁO THÍ NGHIỆM:**

1) Số thứ tự, tên bài thực hành.

2) Mục đích của bài thực hành.

3) Trình tự đã tiến hành thí nghiệm và kết quả thu được.

**VI. CÂU HỎI KIỂM TRA:**

Nếu dùng cổng NOT sau đầu ra Y thì thu được kết quả gì?

**Bài thực hành số 9**

**XÂY DỰNG CỔNG AND TỪ CỔNG NAND VÀ NOT**

**I. MỤC ĐÍCH THÍ NGHIỆM:**

* Tìm hiểu ứng dụng của cổng NAND và cổng NOT để xây dựng cổng AND.

**II. TÓM TẮT KIẾN THỨC CÓ LIÊN QUAN:**

* Cũng có thể xây dựng mạch logic gồm hai cổng: cổng NAND và cổng NOT có chức năng tương đương với cổng AND. Thật vậy: 



*Hình 9-1 Cổng AND được cấu tạo từ cổng NAND và cổng NOT*

* Cũng có thể xây dựng mạch logic có chức năng tương đương với cổng AND 3 đầu vào từ cổng NAND 3 đầu vào và cổng NOT. Sơ đồ nguyên lý như hình 8-2. Tương tự, cũng có thể chứng minh như sau: 



*Hình 9 -2 Cổng AND được cấu tạo từ cổng NAND và cổng NOT*

**III. CÁC BƯỚC TIẾN HÀNH THÍ NGHIỆM:**

1) Nối các dây dẫn theo sơ đồ nguyên lý hình 9-3.



*Hình 9-3*

2) Sử dụng các công tắc cho các đầu vào X1, X2 sử dụng đèn LED cho các đầu ra Y1 và đầu ra Y.

3) Điền vào bảng chân lý kết quả thu được khi tiến hành thí nghiệm

|  |  |  |  |
| --- | --- | --- | --- |
| Đầu vào | | Đầu ra | |
| X1 | X2 | Y1 | Y |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

4) Nối các dây dẫn theo sơ đồ nguyên lý hình 9-4.



*Hình 9-4*

5) Sử dụng các công tắc cho các đầu vào X1, X2 và X3, sử dụng đèn LED cho các đầu ra Y1 và đầu ra Y.

6) Điền vào bảng chân lý kết quả thu được khi tiến hành thí nghiệm

|  |  |  |  |
| --- | --- | --- | --- |
| **Đầu vào** | | | **Đầu ra** |
| X1 | X2 | X3 | Y |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**IV. BÁO CÁO THÍ NGHIỆM:**

1) Số thứ tự, tên bài thực hành.

2) Mục đích của bài thực hành.

3) Trình tự đã tiến hành thí nghiệm và kết quả thu được.

**VI. CÂU HỎI KIỂM TRA:**

Nếu dùng cổng NOT sau đầu ra Y thì thu được kết quả gì?

**Bài thực hành số 10**

**XÂY DỰNG CỔNG NOR CỔNG OR VÀ CỔNG NOT**

**I. MỤC ĐÍCH THÍ NGHIỆM:**

* Tìm hiểu ứng dụng của cổng OR và cổng NOT để xây dựng cổng NOR.

**II. TÓM TẮT KIẾN THỨC CÓ LIÊN QUAN:**

* Từ các bài thí nghiệm về cổng OR và cổng NOT, có thể rút ra nhận xét sau: bảng chân lý của cổng OR và cổng NOR là đảo của nhau. Từ đó, cũng có thể xây dựng mạch logic gồm hai cổng: cổng OR và cổng NOT có chức năng tương đương với cổng NOR. Có thể dễ dàng chứng minh điều này từ sơ đồ nguyên lý hình 10-1. Thật vậy: 



*Hình 10-1*

* Tương tự, cũng có thể xây dựng cổng NOR 3 đầu vào từ cổng OR 3 đầu vào và cổng NOT: 



*Hình 10 -2*

**III. CÁC BƯỚC TIẾN HÀNH THÍ NGHIỆM:**

1) Nối các dây dẫn theo sơ đồ nguyên lý hình 10-3.



*Hình 10-3*

2) Sử dụng các công tắc cho các đầu vào X1 và X2, sử dụng đèn LED cho các đầu ra Y1 và đầu ra Y.

3) Điền vào bảng kết quả thu được khi tiến hành thí nghiệm:

|  |  |  |  |
| --- | --- | --- | --- |
| Đầu vào | | Đầu ra | |
| X1 | X2 | Y1 | Y |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**4**) Nối các dây dẫn theo sơ đồ nguyên lý hình 10-4.



*Hình 10-4*

5) Sử dụng các công tắc cho các đầu vào X1, X2 và X3, sử dụng đèn LED cho các đầu ra Y1 và đầu ra Y.

6) Điền vào bảng chân lý kết quả thu được khi tiến hành thí nghiệm

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Đầu vào | | | Đầu ra | |
| X1 | X2 | X3 | Y1 | Y |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**IV. BÁO CÁO THÍ NGHIỆM:**

1) Số thứ tự, tên bài thực hành.

2) Mục đích của bài thực hành.

3) Trình tự đã tiến hành thí nghiệm và kết quả thu được.

**VI. CÂU HỎI KIỂM TRA:**

Nếu dùng cổng NOT sau đầu ra Y thì thu được kết quả gì?

**Bài thực hành số 11**

**XÂY DỰNG CỔNG OR TỪ CỔNG NOR VÀ CỔNG NOT**

**I. MỤC ĐÍCH THÍ NGHIỆM:**

* Tìm hiểu ứng dụng của cổng NOR và cổng NOT để xây dựng cổng OR.

**II. TÓM TẮT KIẾN THỨC CÓ LIÊN QUAN:**

* Có thể xây dựng mạch logic gồm hai cổng: cổng NOR và cổng NOT có chức năng tương đương với cổng OR. Có thể chứng minh điều này từ sơ đồ nguyên lý hình 11-1. Thật vậy: 



*Hình 11-1*

Tương tự, cũng có thể xây dựng cổng OR 3 đầu vào từ cổng NOR 3 đầu vào và cổng NOT (như ở hình 11-2): 



*Hình 11 -2*

**III. CÁC BƯỚC TIẾN HÀNH THÍ NGHIỆM:**

1) Nối các dây dẫn theo sơ đồ nguyên lý hình 11-3.



*Hình 11-3*

2) Sử dụng các công tắc cho các đầu vào X1 và X2, sử dụng đèn LED cho các đầu ra Y1 và đầu ra Y.

3) Điền vào bảng kết quả thu được khi tiến hành thí nghiệm:

|  |  |  |  |
| --- | --- | --- | --- |
| Đầu vào | | Đầu ra | |
| X1 | X2 | Y1 | Y |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

4) Nối các dây dẫn theo sơ đồ nguyên lý hình 11-4.



*Hình 11-4*

5) Sử dụng các công tắc cho các đầu vào X1, X2 và X3, sử dụng đèn LED cho các đầu ra Y1 và đầu ra Y.

6) Điền vào bảng chân lý kết quả thu được khi tiến hành thí nghiệm

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Đầu vào | | | Đầu ra | |
| X1 | X2 | X3 | Y1 | Y |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**IV. BÁO CÁO THÍ NGHIỆM:**

1) Số thứ tự, tên bài thực hành.

2) Mục đích của bài thực hành.

3) Trình tự đã tiến hành thí nghiệm và kết quả thu được.

**VI. CÂU HỎI KIỂM TRA:**

Nếu dùng cổng NOT sau đầu ra Y thì thu được kết quả gì?

**Bài thực hành số 12**

**XÂY DỰNG CỔNG NAND TỪ CỔNG OR VÀ CỔNG NOT**

**I. MỤC ĐÍCH THÍ NGHIỆM:**

* Tìm hiểu ứng dụng của cổng OR và cổng NOT

**II. TÓM TẮT KIẾN THỨC CÓ LIÊN QUAN:**

* Có thể xây dựng mạch logic gồm hai cổng: cổng NOR và cổng NOT có chức năng tương đương với cổng NAND như hình 12-1. Có thể chứng minh như sau: Từ phép toán của cổng NAND 2 đầu vào:  có thể suy ra được: . Như vậy, cổng NAND có thể được xây dựng từ cổng OR và cổng NOT như sau:



*Hình 12-1*

* Tương tự, cũng có thể xây dựng cổng NAND 3 đầu vào từ cổng OR 3 đầu vào và cổng NOT như hình 12-2: 



*Hình 12 -2*

**III. CÁC BƯỚC TIẾN HÀNH THÍ NGHIỆM:**

1) Nối các dây dẫn theo sơ đồ nguyên lý hình 12-3.



Hình 12-3

2) Sử dụng các công tắc cho các đầu vào X1 và X2, sử dụng đèn LED cho các đầu ra Y1 và đầu ra Y.

3) Điền vào bảng kết quả thu được khi tiến hành thí nghiệm:

|  |  |  |  |
| --- | --- | --- | --- |
| Đầu vào | | Đầu ra | |
| X1 | X2 | Y1 | Y |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

4) Nối các dây dẫn theo sơ đồ nguyên lý hình 12-4.



*Hình 12-4*

5) Sử dụng các công tắc cho các đầu vào X1, X2 và X3, sử dụng đèn LED cho đầu ra Y.

6) Điền vào bảng chân lý kết quả thu được khi tiến hành thí nghiệm.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Đầu vào | | | Đầu ra | |
| X1 | X2 | X3 | Y1 | Y |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**IV. BÁO CÁO THÍ NGHIỆM:**

1) Số thứ tự, tên bài thực hành.

2) Mục đích của bài thực hành.

3) Trình tự đã tiến hành thí nghiệm và kết quả thu được.

**VI. CÂU HỎI KIỂM TRA:**

Nếu dùng cổng NOT sau đầu ra Y thì thu được kết quả gì?

**Bài thực hành số 13**

**XÂY DỰNG CỔNG NOR TỪ CỔNG AND VÀ CỔNG NOT**

**I. MỤC ĐÍCH THÍ NGHIỆM:**

* Tìm hiểu ứng dụng của cổng AND và cổng NOT

**II. TÓM TẮT KIẾN THỨC CÓ LIÊN QUAN:**

* Có thể xây dựng mạch logic gồm hai cổng: cổng AND và cổng NOT có chức năng tương đương với cổng OR. Có thể chứng minh như sau: Từ phép toán logic của cổng NOR 2 đầu vào:  có thể suy ra được: . Như vậy, cổng NOR có thể được xây dựng từ cổng AND và cổng NOT như sau:



*Hình 13-1*

Tương tự, cũng có thể xây dựng cổng NOR 3 đầu vào từ cổng AND 3 đầu vào và cổng NOT: 



*Hình 13 -2*

**III. CÁC BƯỚC TIẾN HÀNH THÍ NGHIỆM:**

1) Nối các dây dẫn theo sơ đồ nguyên lý hình 13-3.



*Hình 13-3*

2) Sử dụng các công tắc cho các đầu vào X1 và X2, sử dụng đèn LED cho các đầu ra Y1 và đầu ra Y.

3) Điền vào bảng kết quả thu được khi tiến hành thí nghiệm:

|  |  |  |
| --- | --- | --- |
| Đầu vào | | Đầu ra |
| X1 | X2 | Y |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**4**) Nối các dây dẫn theo sơ đồ nguyên lý hình 13-4.



*Hình 13-4*

5) Sử dụng các công tắc cho các đầu vào X1, X2 và X3, sử dụng đèn LED cho các đầu ra Y1 và đầu ra Y.

6) Điền vào bảng chân lý kết quả thu được khi tiến hành thí nghiệm

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Đầu vào | | | Đầu ra | |
| X1 | X2 | X3 | Y1 | Y |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**IV. BÁO CÁO THÍ NGHIỆM:**

1) Số thứ tự, tên bài thực hành.

2) Mục đích của bài thực hành.

3) Trình tự đã tiến hành thí nghiệm và kết quả thu được.

**VI. CÂU HỎI KIỂM TRA:**

Nếu dùng cổng NOT sau đầu ra Y thì thu được kết quả gì?

**TÀI LIỆU THAM KHẢO**

1. KS.Chu Khắc Huy , *Thực hành kỹ thuật số*, nhà xuất bản Hà nội(2007)
2. Đoàn Thị Thanh Thảo, Phạm Văn Ngọc, *Kỹ thuật xung số*, vụ giáo dục chuyên nghiệp
3. Nguyễn Trung Lập, *Kỹ thuật số*, Tài liệu lưu hành nội bộ