

Chương 2

Số Học và Logic

π Nội dung chương 2

- › Các hệ thống số đếm
- › Các phép toán số học
- › Các phép toán logic
- › Mạch tổ hợp và tuần hoàn

π Các hệ thống số

Hệ thống số đếm	Cơ số	Các ký tự
Nhị phân	2	0,1
Bát phân	8	0, 1, 2, 3, 4, 5, 6, 7
Thập phân	10	0, 1, 2, 3, 4, 5, 6, 7, 8, 9
Thập lục phân	16	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F

- Số N trong hệ cơ số X:

$$N_X = a_n a_{n-1} \dots a_1 a_0 \cdot b_1 b_2 \dots b_{m-1} b_m$$

- Phân bố trọng số:

...	X^3	X^2	X^1	X^0	.	X^{-1}	X^{-2}	X^{-3}	...
-----	-------	-------	-------	-------	---	----------	----------	----------	-----

π Chuyển đổi giữa các hệ cơ số

› Cơ số X sang cơ số 10:

- Số N trong hệ cơ số X:

$$N_X = a_n a_{n-1} \dots a_1 a_0 \cdot b_1 b_2 \dots b_{m-1} b_m$$

- N_X có giá trị là:

$$N_X = a_n X^n + a_{n-1} X^{n-1} + \dots + a_1 X^1 + a_0 X^0 + b_1 X^{-1} + \dots + b_m X^{-m}$$

- Ví dụ:

$$1001_2 = 1 * 2^3 + 0 * 2^2 + 0 * 2^1 + 1 * 2^0 = 9$$

$$35_8 = 3 * 8^1 + 5 * 8^0 = 29$$

$$A2F_{16} = 10 * 16^2 + 2 * 16^1 + 15 * 16^0 = 2607$$

Chuyển đổi giữa các hệ cơ số (tt)

› Cơ số 10 sang cơ số X:

▪ Phần nguyên:

- Chia phần nguyên của N cho X được thương và số dư a_0
- Tiếp tục chia phần thương cho X được thương mới và số dư a_1
- Tiếp tục cho đến khi thương bằng 0 và số dư a_n
- Phần nguyên biểu diễn trong hệ cơ số X là $a_n \dots a_1 a_0$

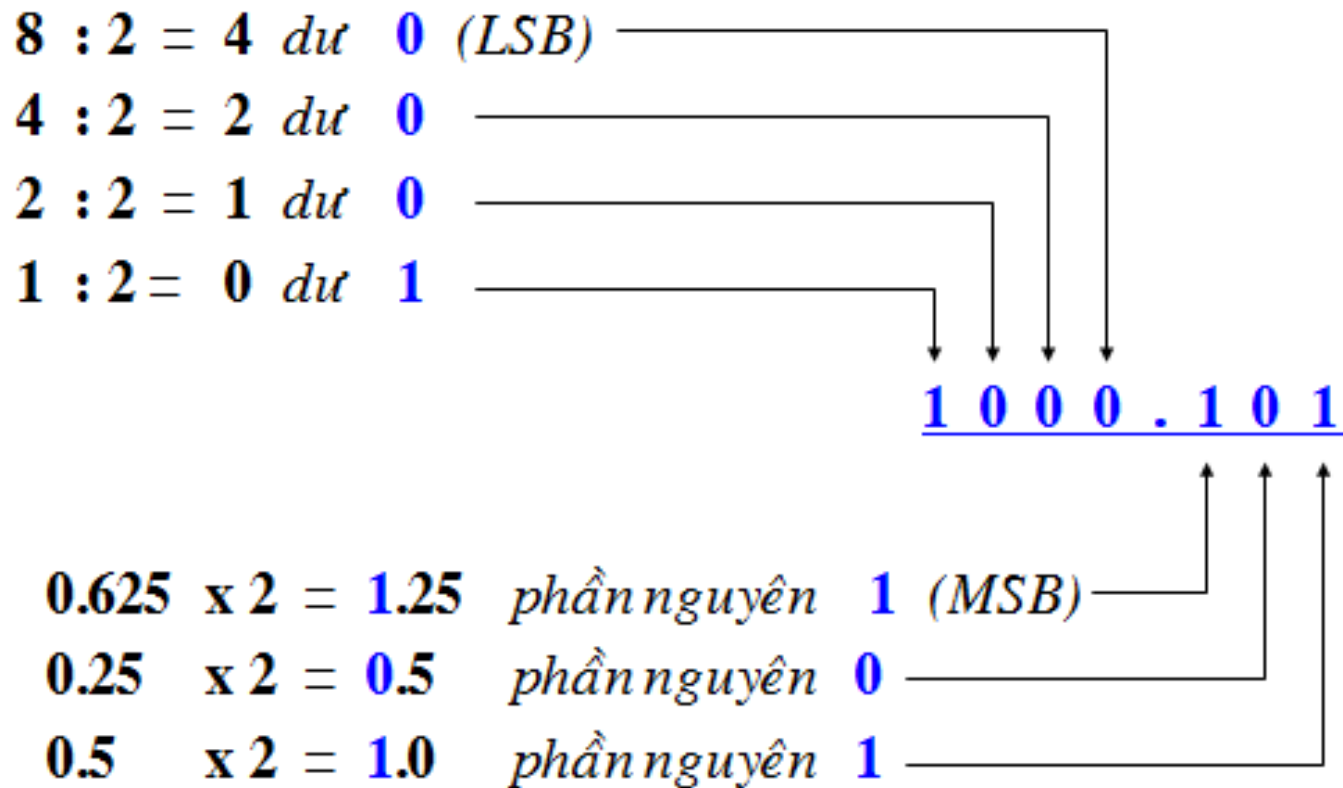
▪ Phần thập phân

- Nhân phần thập phân của N với X được tích có phần nguyên là b_1
- Tiếp tục nhân phần thập phân của tích với X được tích mới có phần nguyên là b_2
- Tiếp tục cho đến khi phần thập phân của tích nhận được bằng 0 hoặc sau một số bước nhất định tùy theo độ chính xác yêu cầu
- Phần thập phân biểu diễn trong hệ cơ số X là $b_1 b_2 \dots b_m$

π Chuyển đổi giữa các hệ cơ số (tt)

› Cơ số 10 sang cơ số X:

- Ví dụ 1: biến đổi 8.625_{10} sang nhị phân



π Chuyển đổi giữa các hệ cơ số (tt)

› Cơ số 10 sang cơ số X:

- Ví dụ 2: biến đổi 1480.4296875_{10} sang thập lục phân

$$1480 : 16 = 92 \text{ dư } 8 \text{ (LSD)}$$

$$92 : 16 = 5 \text{ dư } 12$$

$$5 : 16 = 0 \text{ dư } 5$$

5 C 8 . 6 E

$$0.4296875 \times 16 = 6.875 \text{ phần nguyên } 6 \text{ (MSD)}$$

$$0.875 \times 16 = 14.0 \text{ phần nguyên } 14$$

Chuyển đổi giữa các hệ cơ số (tt)

› Cơ số 8 sang cơ số 2:

- Biến mỗi ký tự trong hệ bát phân thành 3 bit nhị phân tương ứng

Octal	0	1	2	3	4	5	6	7
	↓	↓	↓	↓	↓	↓	↓	↓
Binary	000	001	010	011	100	101	110	111

- Ví dụ: biến đổi 472_8 sang nhị phân

4 7 2
↓ ↓ ↓
100 111 010

100111010₂



Chuyển đổi giữa các hệ cơ số (tt)

› Cơ số 16 sang cơ số 2:

- Biến mỗi ký tự trong hệ thập lục phân thành 4 bit nhị phân tương ứng
- Ví dụ: biến đổi $10AF_{16}$ sang nhị phân

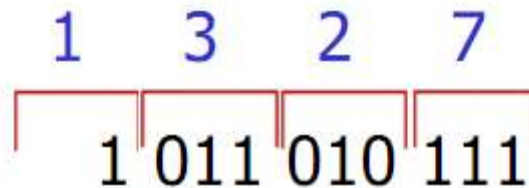
1	0	A	F	
↓	↓	↓	↓	
0001	0000	1010	1111	1000010101111₂

Hexa	Decimal	Binary
0	0	0000
1	1	0001
2	2	0010
3	3	0011
4	4	0100
5	5	0101
6	6	0110
7	7	0111
8	8	1000
9	9	1001
A	10	1010
B	11	1011
C	12	1100
D	13	1101
E	14	1110
F	15	1111

π Chuyển đổi giữa các hệ cơ số (tt)

› Cơ số 2 sang cơ số 8:

- Bắt đầu từ phải sang trái, nhóm các bit nhị phân thành các nhóm 3 bit
- Biến đổi mỗi nhóm 3 bit thành một Octal
- Ví dụ: biến đổi 1011010111_2 sang bát phân

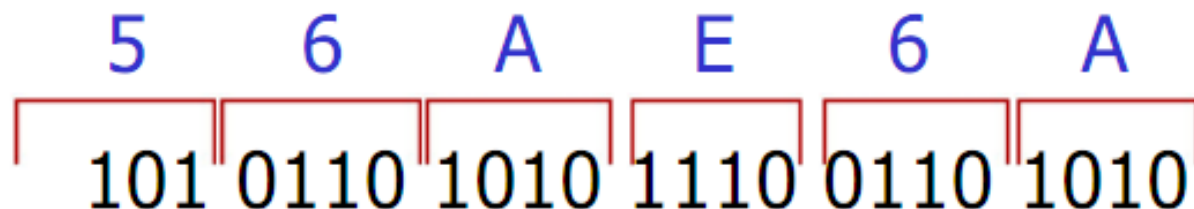


$$1011010111_2 = 1327_8$$

Chuyển đổi giữa các hệ cơ số (tt)

› Cơ số 2 sang cơ số 16:

- Bắt đầu từ phải sang trái, nhóm các bit nhị phân thành các nhóm 4 bit
- Biến đổi mỗi nhóm 4 bit thành một Hexa
- Ví dụ: biến đổi $10101101010111001101010_2$ sang thập lục phân

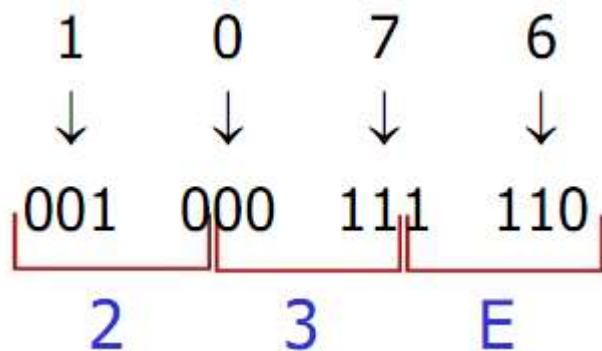


$$10101101010111001101010_2 = 56AE6A_{16}$$

Chuyển đổi giữa các hệ cơ số (tt)

› Cơ số 8 sang cơ số 16:

- Biến đổi số bát phân thành số nhị phân
- Biến đổi số nhị phân thành số thập lục phân
- Ví dụ: biến đổi 1076_8 sang thập lục phân



$$1076_8 = 23E_{16}$$

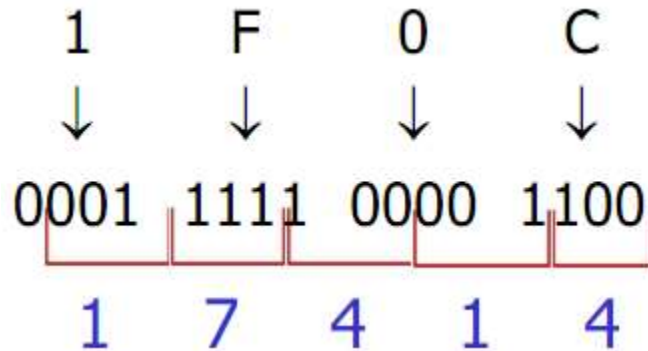
Chuyển đổi giữa các hệ cơ số (tt)

› Cơ số 16 sang cơ số 8:

Biến đổi số thập lục phân thành số nhị phân

Biến đổi số nhị phân thành số bát phân

Ví dụ: biến đổi $1F0C_{16}$ sang bát phân



$$1F0C_{16} = 17414_8$$

π Nội dung chương 2

- › Các hệ thống số đếm
- › Các phép toán số học
- › Các phép toán logic
- › Mạch tổ hợp và tuần hoàn

π Phép cộng nhị phân

› Cộng hai bit nhị phân

A	B	A + B
0	0	0
0	1	1
1	0	1
1	1	0 (nhớ 1)

π Phép cộng nhị phân (tt)

› Cộng hai số nhị phân không dấu

$$\begin{array}{r} \\ \\ + \\ \\ \\ \hline 1 \end{array}$$

π Phép trừ nhị phân

› Trừ hai bit nhị phân

A	B	A - B
0	0	0
0	1	1 (mượn 1)
1	0	1
1	1	0

π Phép nhân nhị phân

› Nhân hai bit nhị phân

A	B	A x B
0	0	0
0	1	0
1	0	0
1	1	1

π Phép chia nhị phân

› Chia hai số nhị phân

$$\begin{array}{r} \begin{array}{r} \mathbf{1\ 0\ 0\ 1\ 0\ 0\ 0\ 1} \\ - \mathbf{1\ 0\ 1\ 1} \\ \hline \mathbf{1\ 1\ 1\ 0} \\ - \mathbf{1\ 0\ 1\ 1} \\ \hline \mathbf{1\ 1\ 0\ 1} \\ - \mathbf{1\ 0\ 1\ 1} \\ \hline \mathbf{1\ 0} \end{array} \quad \Bigg| \begin{array}{r} \mathbf{1\ 0\ 1\ 1} \\ \hline \mathbf{1\ 1\ 0\ 1} \end{array} \end{array}$$

Số nhị phân có dấu – Bù hai

› Số bù hai: đảo bit cộng 1

- Ví dụ: số bù hai (8 bit) của 5

$$5 = 0000101_2$$

$$\text{Đảo bit của 5} = 11111010_2$$

$$\text{Cộng 1} = 11111011_2$$

$$-5 = 11111011_2$$

- Bit có trọng số lớn nhất quy ước dấu
 - Bit dấu bằng 0 xác định số dương
 - Bit dấu bằng 1 xác định số âm
- Tìm giá trị của số âm:
 - Cách 1: Khai triển như số dương nhưng bit có trọng số lớn nhất được nhân thêm với (-1)
 - Cách 2: Lấy bù hai của nó được số dương có cùng biên độ
- Phạm vi biểu diễn của số nhị phân có dấu n bit:
 -2^{n-1} đến $2^{n-1} - 1$

π Cộng trừ số bù hai

- › Thực hiện như số không dấu
- › Thực hiện trên toán hạng có cùng chiều dài và kết quả cũng có cùng số bit
- › Kết quả đúng nếu nằm trong phạm vi biểu diễn số dấu, nếu kết quả sai thì cần mở rộng chiều dài bit

π Cộng trừ số nhị phân có dấu – Bù hai (tt)

› Ví dụ:

$$\begin{array}{r} + \quad -6 : 1010 \\ + \quad +3 : 0011 \\ \hline \quad -3 : 1101 \end{array}$$

$$\begin{array}{r} - \quad -6 : 1010 \\ - \quad -2 : 1110 \\ \hline \quad -4 : 1100 \end{array}$$

$$\begin{array}{r} + \quad -2 : 1110 \\ + \quad -5 : 1011 \\ \hline \quad -7 : 1001 \end{array}$$

$$\begin{array}{r} - \quad +2 : 0010 \\ - \quad -5 : 1011 \\ \hline \quad +7 : 0111 \end{array}$$

π Cộng trừ số nhị phân có dấu – Bù hai (tt)

› Ví dụ:

$$\begin{array}{r} + \\ +4 : 0100 \longrightarrow 00100 \\ +5 : 0101 \longrightarrow 00101 \\ \hline -7 : 1001 \text{ (Kq sai)} \qquad 01001 : +9 \text{ (Kq đúng)} \end{array}$$

$$\begin{array}{r} - \\ -7 : 1001 \longrightarrow 11001 \\ +5 : 0101 \longrightarrow 00101 \\ \hline +4 : 0100 \text{ (Kq sai)} \qquad 10100 : -12 \text{ (Kq đúng)} \end{array}$$

Số nhị phân có dấu – Dấu lượng

› Dấu lượng (sign-and-magnitude)

- Bit có trọng số lớn nhất quy ước dấu
 - Bit dấu bằng 0 xác định số dương
 - Bit dấu bằng 1 xác định số âm
- Các bit còn lại biểu diễn giá trị độ lớn của số
- Ví dụ: biểu diễn 8 bit của 5 và -5

$$5 = 00000101_2$$

$$-5 = 10000101_2$$

- Phạm vi biểu diễn của số nhị phân có dấu n bit: $-2^{n-1} - 1$ đến $2^{n-1} - 1$

π Cộng trừ số dấu lượng

› Cộng hai số nhị phân có dấu:

- Nếu hai số cùng dấu thì thực hiện phép cộng phần biểu diễn giá trị và sử dụng bit dấu cùng dấu với hai số đó
- Nếu hai số khác dấu thì kết quả sẽ nhận dấu của toán tử lớn hơn, và thực hiện phép trừ giữa toán tử có giá trị lớn hơn với toán tử bé hơn

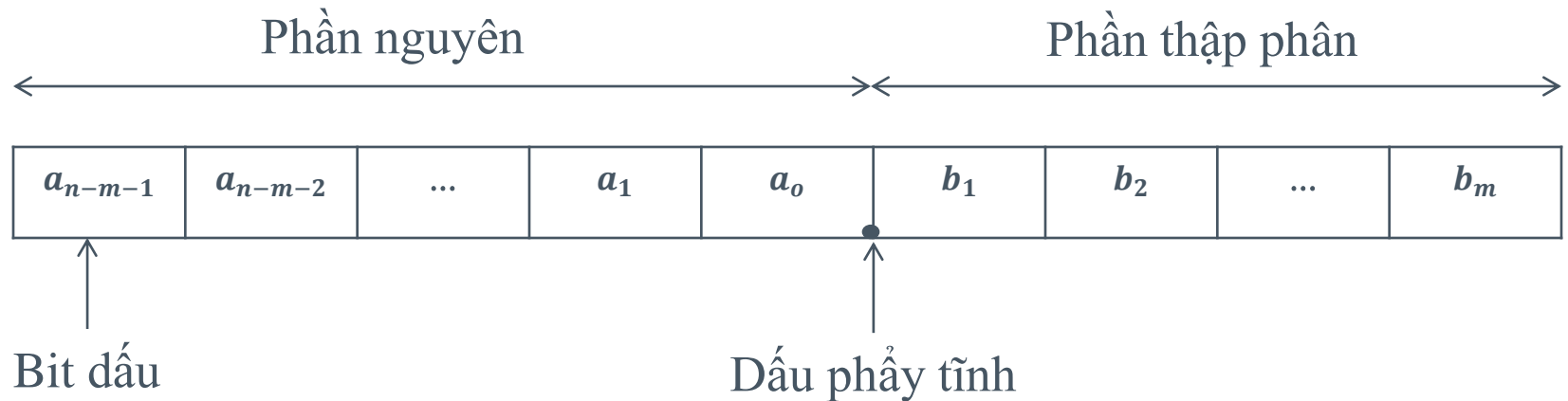
› Trừ hai số nhị phân có dấu:

- Đổi dấu số trừ và thực hiện phép cộng hai số nhị phân có dấu giữa số bị trừ và số trừ sau khi đổi dấu

π

Số dấu phẩy tĩnh (Fixed-point number)

- › Sử dụng dấu chấm ảo để biểu diễn một số thực, phân biệt giữa phần biểu diễn giá trị nguyên và phần lẻ thập phân
- › Định dạng số dấu phẩy tĩnh n bit với m bit dùng cho phần lẻ thập phân:



Số dấu phẩy động (Floating-point number)

- › Số thực N thường được biểu diễn dưới dạng số dấu phẩy động :

$$N = \pm M * X^E$$

Trong đó, M là phần định trị (Mantissa)

X là cơ số (Radix)
Bấm để thêm nội dung

E là phần lũy thừa (Exponent)

π Nội dung chương 2

- › Các hệ thống số đếm
- › Các phép toán số học
- › Các phép toán logic
- › Mạch tổ hợp và tuần hoàn

π Đại số Boole

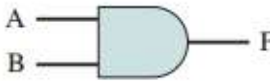
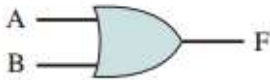
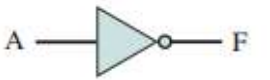

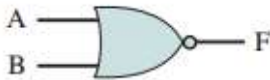
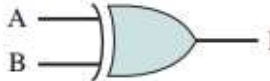
- › Các hằng và biến trong đại số Boole chỉ có hai giá trị 0 và 1
- › Trong đại số Boole không có phân số, số âm, lũy thừa, căn số, ...
- › Đại số Boole chỉ có 3 toán tử: nhân logic (AND), cộng logic (OR) và bù logic (NOT)

x	y	$x \cdot y$
0	0	0
0	1	0
1	0	0
1	1	1

x	y	$x + y$
0	0	0
0	1	1
1	0	1
1	1	1

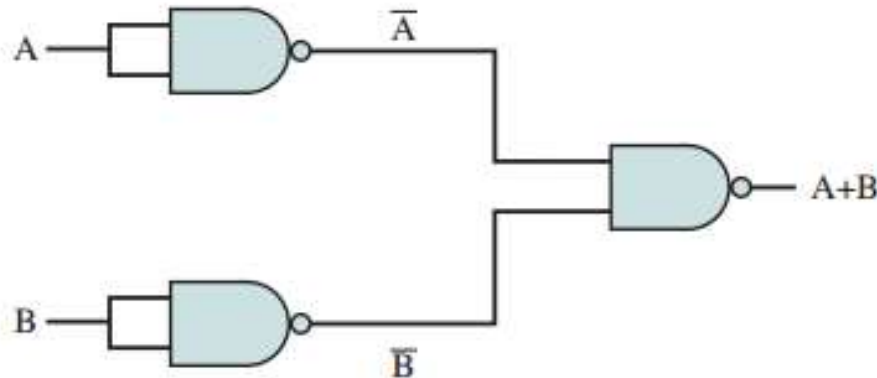
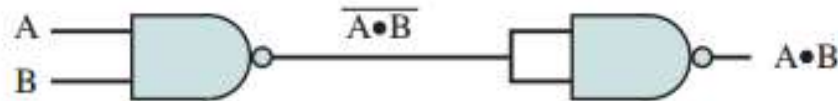
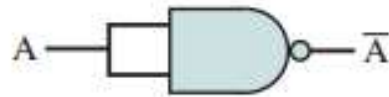
x	\bar{x}
0	1
1	0

π Cổng logic

Name	Graphical Symbol	Algebraic Function	Truth Table															
AND		$F = A \cdot B$ or $F = AB$	<table border="1"> <thead> <tr> <th>A</th> <th>B</th> <th>F</th> </tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>0</td></tr> <tr><td>1</td><td>0</td><td>0</td></tr> <tr><td>1</td><td>1</td><td>1</td></tr> </tbody> </table>	A	B	F	0	0	0	0	1	0	1	0	0	1	1	1
A	B	F																
0	0	0																
0	1	0																
1	0	0																
1	1	1																
OR		$F = A + B$	<table border="1"> <thead> <tr> <th>A</th> <th>B</th> <th>F</th> </tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>0</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>1</td></tr> </tbody> </table>	A	B	F	0	0	0	0	1	1	1	0	1	1	1	1
A	B	F																
0	0	0																
0	1	1																
1	0	1																
1	1	1																
NOT		$F = \bar{A}$ or $F = A'$	<table border="1"> <thead> <tr> <th>A</th> <th>F</th> </tr> </thead> <tbody> <tr><td>0</td><td>1</td></tr> <tr><td>1</td><td>0</td></tr> </tbody> </table>	A	F	0	1	1	0									
A	F																	
0	1																	
1	0																	
NAND		$F = \overline{AB}$	<table border="1"> <thead> <tr> <th>A</th> <th>B</th> <th>F</th> </tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>1</td></tr> <tr><td>0</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>0</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>0</td></tr> </tbody> </table>	A	B	F	0	0	1	0	1	1	1	0	1	1	1	0
A	B	F																
0	0	1																
0	1	1																
1	0	1																
1	1	0																
NOR		$F = \overline{A + B}$	<table border="1"> <thead> <tr> <th>A</th> <th>B</th> <th>F</th> </tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>1</td></tr> <tr><td>0</td><td>1</td><td>0</td></tr> <tr><td>1</td><td>0</td><td>0</td></tr> <tr><td>1</td><td>1</td><td>0</td></tr> </tbody> </table>	A	B	F	0	0	1	0	1	0	1	0	0	1	1	0
A	B	F																
0	0	1																
0	1	0																
1	0	0																
1	1	0																
XOR		$F = A \oplus B$	<table border="1"> <thead> <tr> <th>A</th> <th>B</th> <th>F</th> </tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>0</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>0</td></tr> </tbody> </table>	A	B	F	0	0	0	0	1	1	1	0	1	1	1	0
A	B	F																
0	0	0																
0	1	1																
1	0	1																
1	1	0																

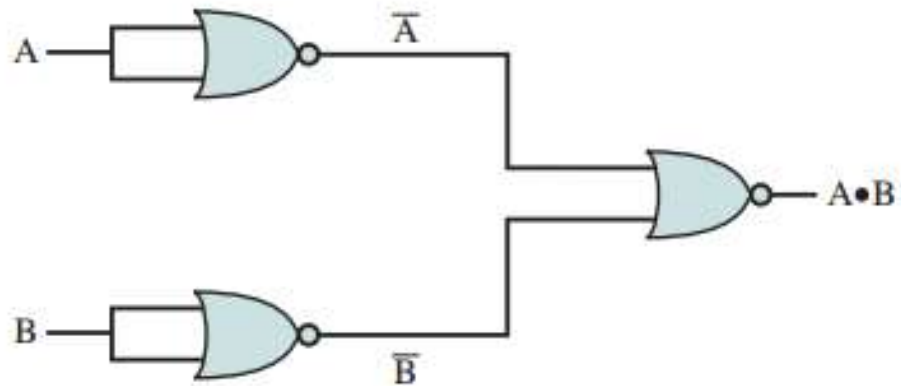
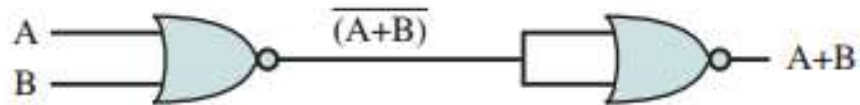
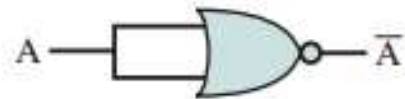
π Cổng logic (tt)

› Dùng cổng NAND thực hiện các cổng khác:



π Cổng logic (tt)

› Dùng cổng NOR thực hiện các cổng khác:



π Mạch tổ hợp

- › Mạch tổ hợp: ngõ ra là các hàm logic theo ngõ vào
- › Các ngõ ra thay đổi trạng thái ngay khi các ngõ vào thay đổi trạng thái (nếu bỏ qua thời gian trễ của các phần tử logic)
- › Tín hiệu ra tại mỗi thời điểm chỉ phụ thuộc vào giá trị các tín hiệu vào tại thời điểm đó
- › Một số mạch tổ hợp:
 - Mạch mã hóa – giải mã
 - Mạch ghép kênh – phân kênh
 - Mạch so sánh
 - Mạch số học

π Mạch giải mã

- › Ứng với mỗi tổ hợp nhị phân của ngõ vào, chỉ có một ngõ ra ở trạng thái tích cực, các ngõ ra còn lại ở trạng thái không tích cực, thường có n ngõ vào và 2^n ngõ ra.
- › Ví dụ mạch giải mã địa chỉ trong bộ nhớ

Click to add text

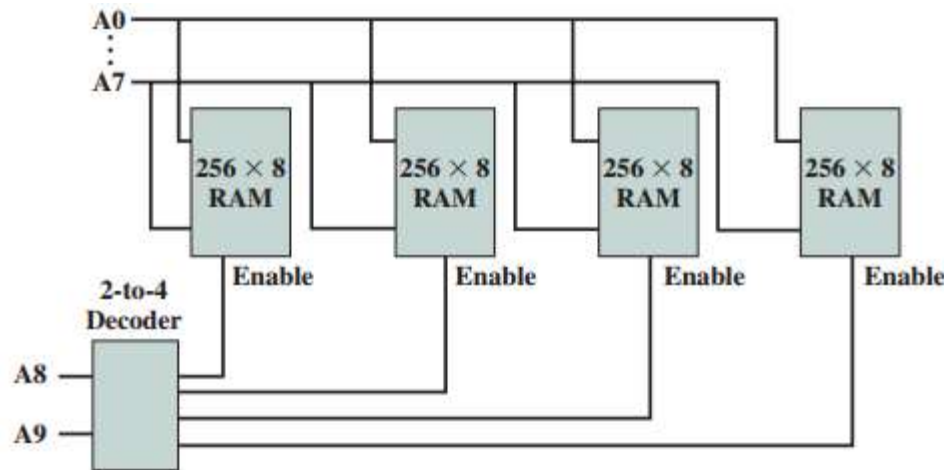
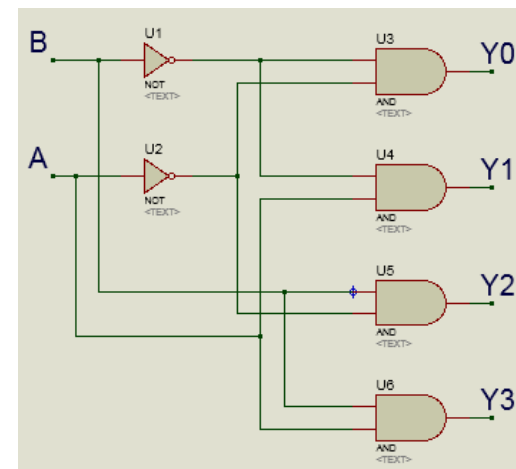


Figure 11.16 Address Decoding

B	A	y ₀	y ₁	y ₂	y ₃
0	0	1	0	0	0
0	1	0	1	0	0
1	0	0	0	1	0
1	1	0	0	0	1



π Mạch cộng toàn phần

(b) Addition with Carry Input				
C_{in}	A	B	Sum	C_{out}
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

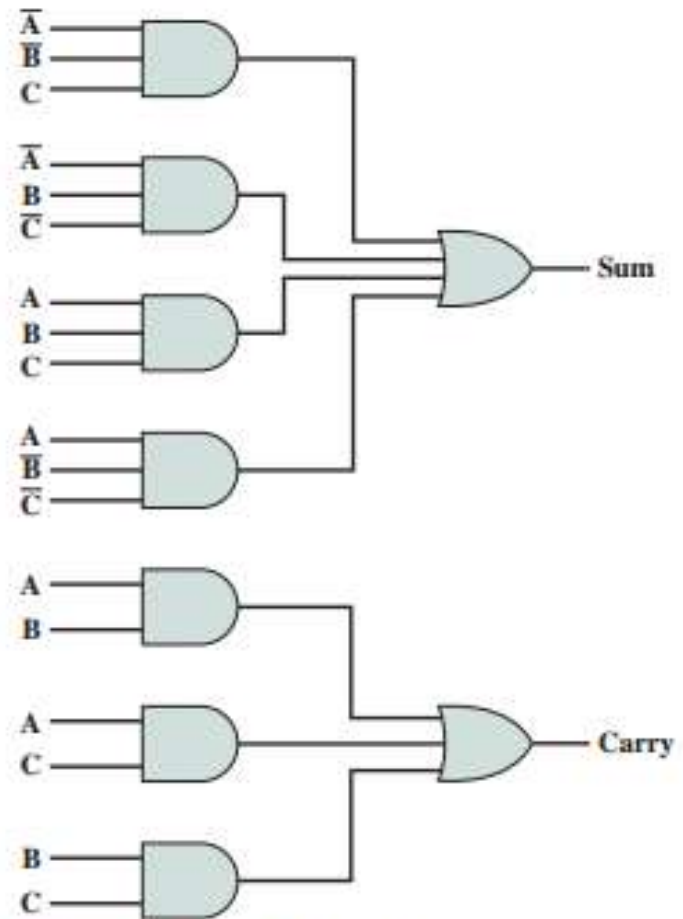


Figure 11.20 Implementation of an Adder 39

π Mạch tuần tự

- › Các ngõ ra ở trạng thái kế tiếp vừa phụ thuộc vào trạng thái hiện tại của ngõ vào, vừa phụ thuộc trạng thái hiện tại của ngõ ra
- › Khi các ngõ vào thay đổi trạng thái, các ngõ ra không thay đổi ngay mà chờ đến khi có xung đồng hồ
- › Mạch tuần tự có tính đồng bộ và tính nhớ → cơ sở để thiết kế các bộ nhớ
- › Cơ sở thiết kế mạch tuần tự là dựa trên Flip – Flop
- › Một số mạch tuần tự: thanh ghi, bộ đếm

π Flip – Flop (FF)

- › FF là mạch dao động đa hài hai trạng thái bền, được xây dựng trên cơ sở các cổng logic và hoạt động theo một bảng trạng thái cho trước
- › Một FF thường có:
 - Một hoặc hai ngõ vào dữ liệu, một ngõ vào xung đồng hồ
 - Hai ngõ ra, thường ký hiệu Q (ngõ ra chính) và \bar{Q} (ngõ ra phụ)

π Các loại Flip-Flop

Name	Graphical Symbol	Truth Table															
S-R		<table border="1"> <thead> <tr> <th>S</th> <th>R</th> <th>Q_{n+1}</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>Q_n</td> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> </tr> <tr> <td>1</td> <td>1</td> <td>-</td> </tr> </tbody> </table>	S	R	Q_{n+1}	0	0	Q_n	0	1	0	1	0	1	1	1	-
S	R	Q_{n+1}															
0	0	Q_n															
0	1	0															
1	0	1															
1	1	-															
J-K		<table border="1"> <thead> <tr> <th>J</th> <th>K</th> <th>Q_{n+1}</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>Q_n</td> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> </tr> <tr> <td>1</td> <td>1</td> <td>$\overline{Q_n}$</td> </tr> </tbody> </table>	J	K	Q_{n+1}	0	0	Q_n	0	1	0	1	0	1	1	1	$\overline{Q_n}$
J	K	Q_{n+1}															
0	0	Q_n															
0	1	0															
1	0	1															
1	1	$\overline{Q_n}$															
D		<table border="1"> <thead> <tr> <th>D</th> <th>Q_{n+1}</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> </tr> <tr> <td>1</td> <td>1</td> </tr> </tbody> </table>	D	Q_{n+1}	0	0	1	1									
D	Q_{n+1}																
0	0																
1	1																

Figure 11.27 Basic Flip-Flops

π Thanh ghi song song và thanh ghi dịch

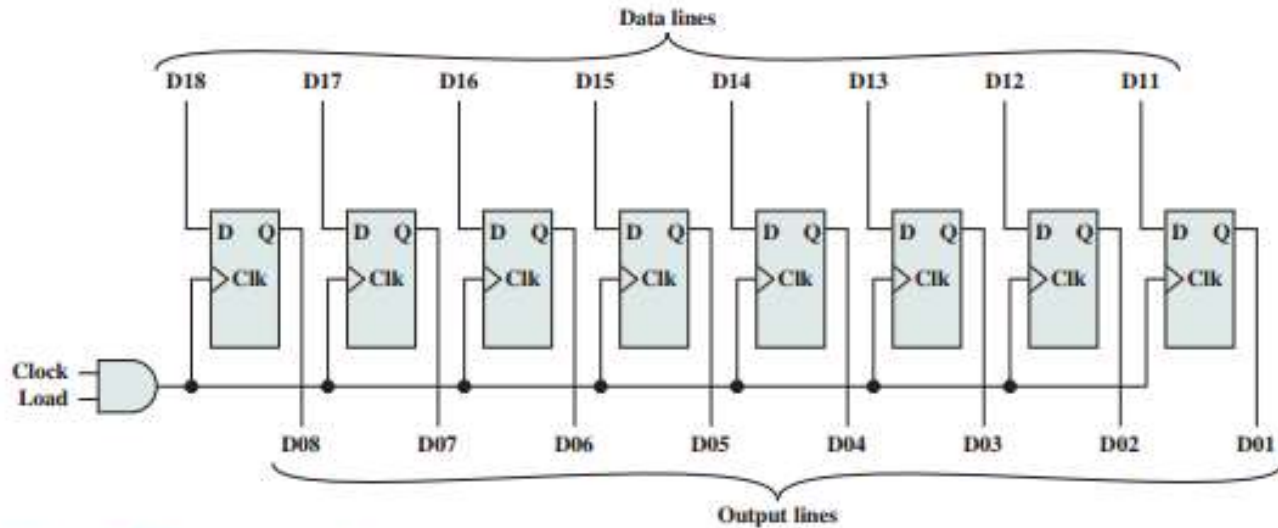


Figure 11.28 8-Bit Parallel Register

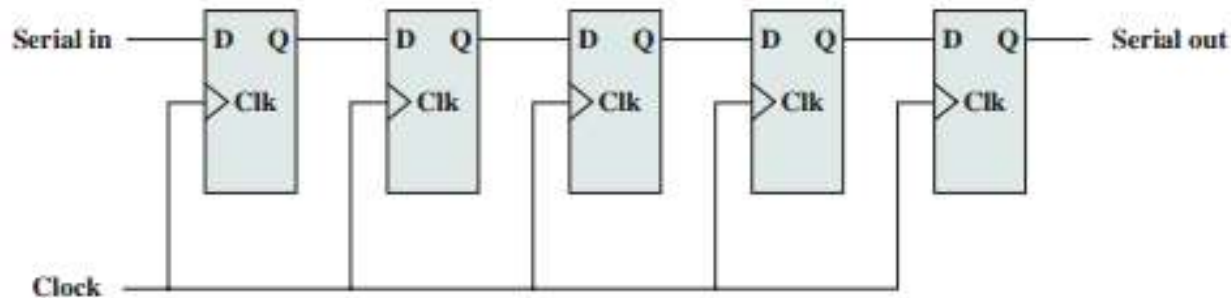
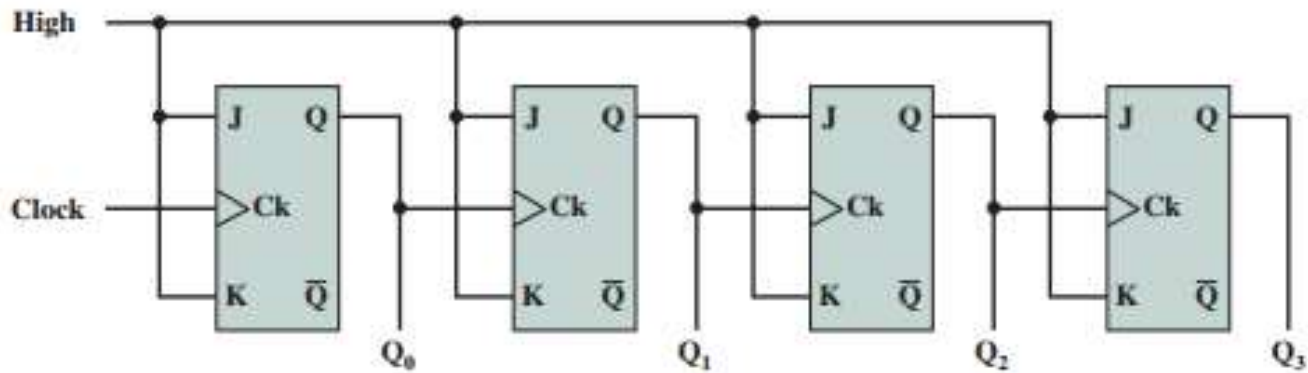
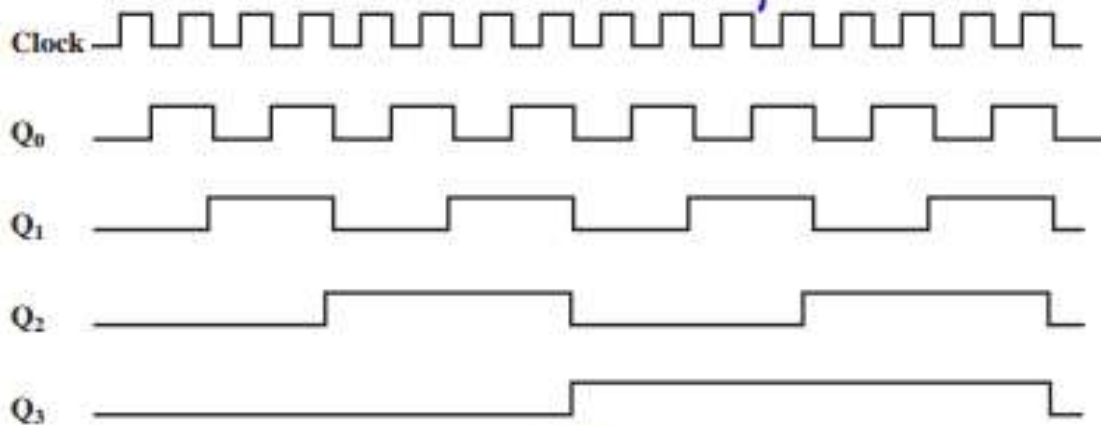


Figure 11.29 5-Bit Shift Register

π Bộ đếm nối tiếp



(a) Sequential circuit



(b) Timing diagram

Figure 11.30 Ripple Counter

π Bộ đếm song song

(a) Truth table

C	B	A	Jc	Kc	Jb	Kb	Ja	Ka
0	0	0	0	d	0	d	1	d
0	0	1	0	d	1	d	d	1
0	1	0	0	d	d	0	1	d
0	1	1	1	d	d	1	d	1
1	0	0	d	0	0	d	1	d
1	0	1	d	0	1	d	d	1
1	1	0	d	0	d	0	1	d
1	1	1	d	1	d	1	d	1

(c) Logic diagram

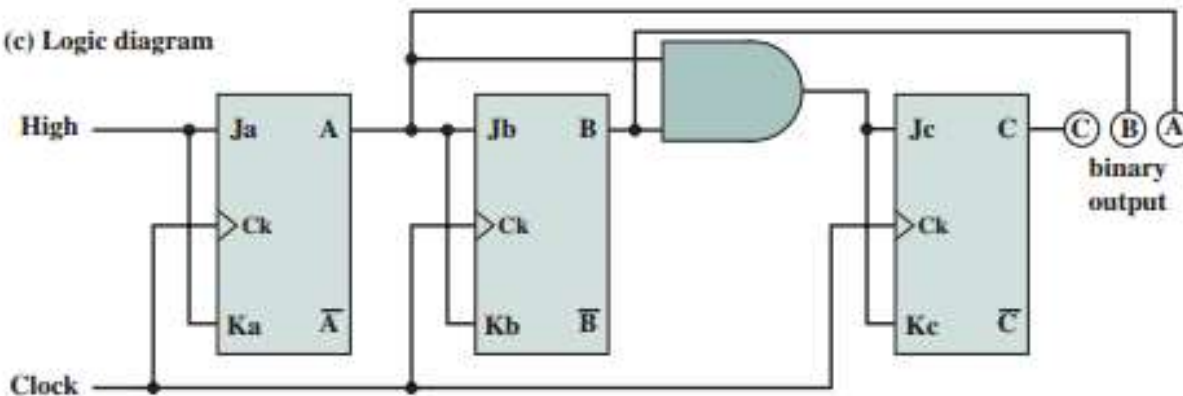


Figure 11.31 Design of a Synchronous Counter