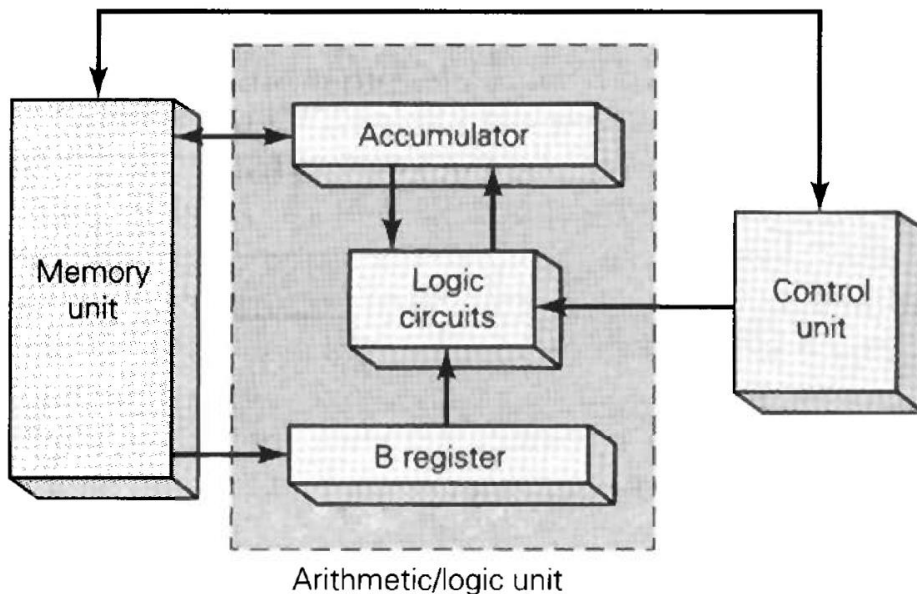


Chương 4

Các Phép Toán & Mạch Số Học



Nội dung

- Biểu diễn số có dấu
- Các phép toán số học
- Đơn vị số học và luận lý (ALU)

Phép cộng nhị phân

- Phép cộng (**Addition**) là phép toán quan trọng nhất trong các hệ thống số
 - Phép trừ (**Subtraction**), phép nhân (**multiplication**) và phép chia (**division**) được hiện thực bằng cách sử dụng phép cộng
 - Luật cơ bản:
 - $0 + 0 = 0$
 - $1 + 0 = 1$
 - $1 + 1 = 10 = 0 + \text{carry of 1 into next position}$
 - $1 + 1 + 1 = 11 = 1 + \text{carry of 1 into next position}$

- Ví dụ

$$\begin{array}{r}
 0 \ 1 \ 1 \ (3) \\
 + \ 1 \ 1 \ 0 \ (6) \\
 \hline
 1 \ 0 \ 0 \ 1 \ (9)
 \end{array}$$

$$\begin{array}{r}
 1 \ 0 \ 0 \ 1 \ (9) \\
 + \ 1 \ 1 \ 1 \ 1 \ (15) \\
 \hline
 1 \ 1 \ 0 \ 0 \ 0 \ (24)
 \end{array}$$

$$\begin{array}{r}
 1 \ 1.0 \ 1 \ 1 \ (3.375) \\
 + \ 1 \ 0.1 \ 1 \ 0 \ (2.750) \\
 \hline
 1 \ 1 \ 0.0 \ 0 \ 1 \ (6.125)
 \end{array}$$

Biểu diễn số có dấu (1)

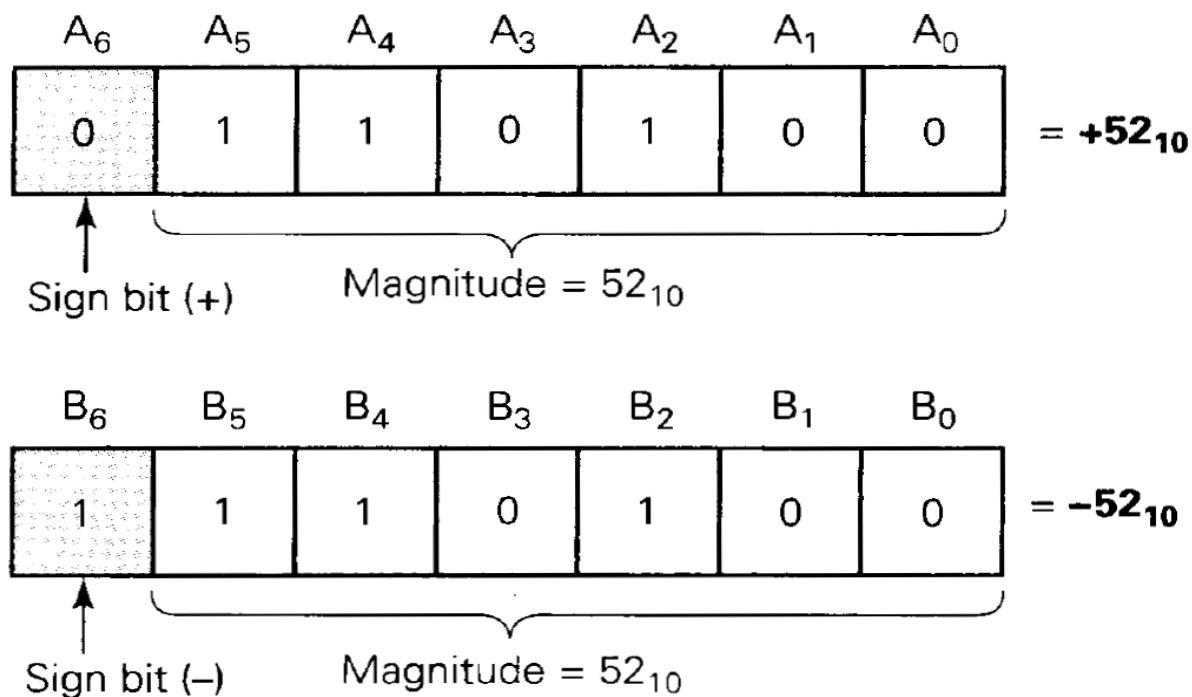
- Bit dấu (**sign bit**)

0: dương (positive)

1: âm (negative)

- Lượng số (**magnitude**)

- Hệ thống **sign-magnitude**



Biểu diễn số có dấu (2)

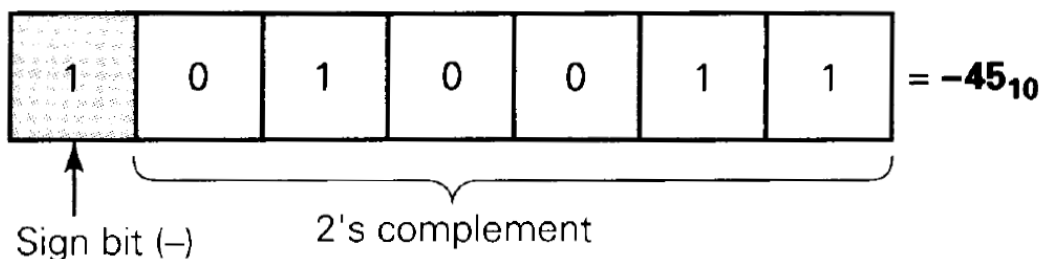
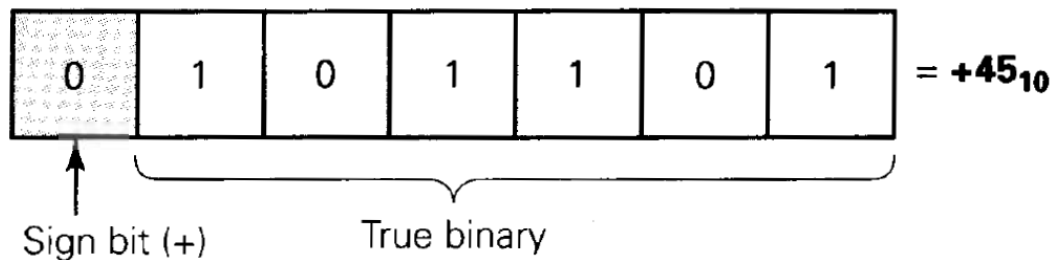
- Hệ thống *sign-magnitude* tuy đơn giản nhưng thông thường không được sử dụng do việc hiện thực mạch phức tạp hơn các hệ thống khác
- Dạng bù-1 (**1's-Complement Form**)
 - Chuyển mỗi bit của số nhị phân sang dạng bù
 - Ví dụ: $101101_2 \rightarrow 010010$ (số bù-1)
- Dạng bù-2 (**2's-Complement Form**)
 - Cộng **1** vào vị trí bit **LSB** (trọng số nhỏ nhất) của số **bù-1**
 - Ví dụ:

<i>Số bù-1</i>		010010
Cộng 1	+	1
<i>Số bù-2</i>		010011

Biểu diễn số có dấu sử dụng bù-2

■ Quy tắc

- **Số dương** (positive): lượng số (magnitude) biểu diễn dưới dạng **số nhị phân đúng**, **bit dấu** bằng **0** (bit trọng số cao nhất - MSB)
- **Số âm** (negative): lượng số biểu diễn dưới dạng số **bù-2**, **bit dấu** bằng **1** (bit MSB)



Biểu diễn số có dấu sử dụng bù-2

- Hệ thống bù-2 được sử dụng để biểu diễn số có dấu vì nó cho phép thực hiện phép toán trừ bằng cách sử dụng phép toán cộng
 - Các máy tính số sử dụng cùng một mạch điện cho cộng và trừ
→ tiết kiệm phần cứng
- **Phủ định** (negation): đổi từ số dương sang số âm hoặc từ số âm sang số dương
 - **Phủ định của 1 số nhị phân có dấu là bù-2 của số đó**
 - Ví dụ:

+9	01001	số có dấu
- 9	10111	phủ định (bù-2)
+9	01001	phủ định lần 2 (bù-2)

Trường hợp đặc biệt của bù-2

- Bit dấu bằng 1, N bit lượng số bằng 0: số thập phân tương đương là -2^N
 - Ví dụ:
$$1000 = -2^3 = -8$$
$$10000 = -2^4 = -16$$
$$100000 = -2^5 = -32$$
- Bit dấu bằng 0, N bit lượng số bằng 1: số thập phân tương đương là $+(2^N - 1)$
 - Ví dụ: $0111 = +(2^3 - 1) = +7$
- Khoảng giá trị có thể biểu diễn bằng hệ thống bù-2 với N bit lượng số là
 -2^N đến $+(2^N - 1)$

Phép cộng trong hệ thống bù-2 (1)

■ Luật cộng

- Cộng 2 số **bù-2** theo luật cộng cơ bản (cộng cả bit dấu)
- Loại bỏ **bit nhớ** (carry) ở **vị trí cuối cùng** của phép cộng (sinh ra bởi phép cộng 2 bit dấu)

Trường hợp 1

			bit dấu
+9	→	0	1001
+4	→	0	0100
<hr/>			
+13		0	1101

Trường hợp 2

			bit dấu
+9	→	0	1001
-4	→	1	1100
<hr/>			
+5		0	0101

~~1~~
 ↑
carry

Phép cộng trong hệ thống bù-2 (2)

Trường hợp 3

		bit đầu	
-9	→	1	0111
+4	→	0	0100
<hr/>			
-5		1	1011

Trường hợp 4

		bit đầu	
-9	→	1	0111
-4	→	1	1100
<hr/>			
-13		1	0011
			carry

-9	→	1	0111
+9	→	0	1001
<hr/>			
0		1	0000
carry			bit đầu

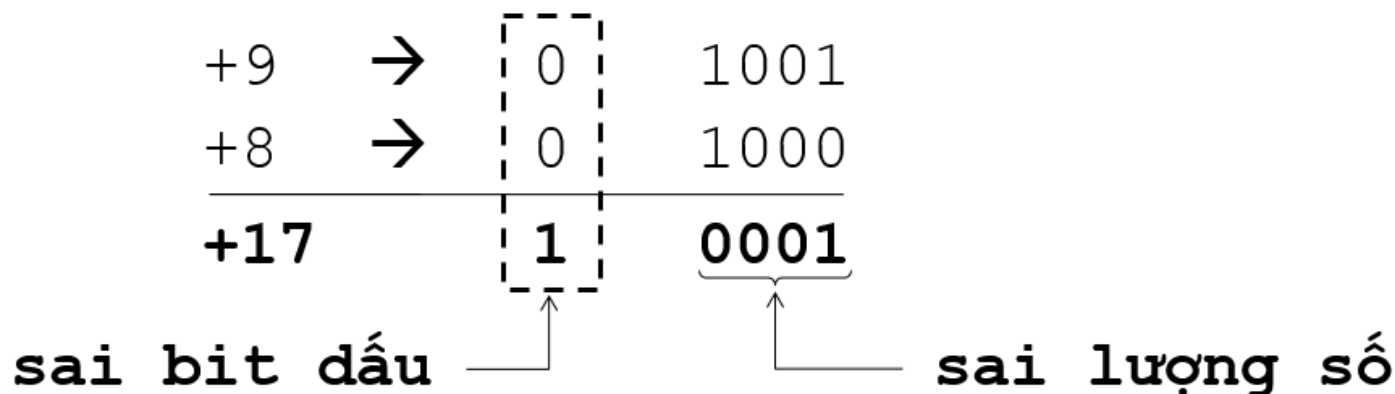
Trường hợp 5

Phép trừ trong hệ thống bù-2

- Phép toán trừ trong hệ thống bù-2 được thực hiện thông qua phép toán cộng
- Trình tự thực hiện
 - Phủ định số trừ
 - Cộng giá trị thu được vào số bị trừ
- Ví dụ

$$\begin{aligned}
 +9 - 4 &= +9 + (-4) &= 01001 + 11100 \\
 &&= \cancel{1}00101 = +5 \\
 -9 - 4 &= -9 + (-4) &= 10111 + 11100 \\
 &&= \cancel{1}10011 = -13 \\
 +9 - 9 &= +9 + (-9) &= 01001 + 10111 \\
 &&= \cancel{1}00000 = 0
 \end{aligned}$$

Tràn số học (Arithmetic Overflow)



- Điều kiện tràn: cộng 2 số dương hoặc 2 số âm
- Phát hiện tràn
 - Hiện tượng tràn được phát hiện bằng cách kiểm tra bit dấu của kết quả phép cộng so với các bit dấu của các toán hạng
 - Phép trừ: tràn chỉ có thể xảy ra khi số trừ và số bị trừ có bit dấu khác nhau

Bài tập

- Thực hiện các phép toán sau trong hệ bù 2.

a. $-9 + -6$

d. $14 + (-17)$

b. $17 - 16$

e. $15 - 21$

c. $-15 - (-36)$

f. $-15 - 17$

- Thiết kế mạch phát hiện tràn số

- Phép cộng hai số nhị phân trong hệ bù 2: $S = A + B$ với S_s , A_s , B_s là các bit dấu của kết quả S và các thừa số A, B .
- Phép trừ hai số nhị phân trong hệ bù 2: $S = A - B$ với S_s , A_s , B_s là các bit dấu của kết quả S và các thừa số A, B .

Phép toán nhân (multiplication)

- Thao tác nhân 2 số nhị phân được thực hiện theo cách tương tự nhân 2 số thập phân

1001	← Số bị nhân = 9_{10}
1011	← Số nhân = 11_{10}
<hr/>	
1001	} Tích thành phần (lần lượt dịch trái)
1001	
0000	
1001	
<hr/>	
1100011	← Kết quả = 99_{10}

Phép nhân trong hệ thống bù-2

1. Nếu số nhân và số bị nhân đều **dương**
 - Nhân bình thường
2. Nếu số nhân và số bị nhân là các số **âm**
 - Chuyển 2 số sang số dương sử dụng bù-2
 - Nhân bình thường
 - Kết quả là 1 số dương với **bit dấu bằng 0**
3. Nếu **1 trong 2** số là số **âm**
 - Chuyển số âm sang số dương sử dụng bù-2
 - Nhân bình thường
 - Kết quả được **chuyển sang dạng bù-2, bit dấu bằng 1**

Phép toán chia (Division)

- Phép chia 2 số nhị phân được thực hiện theo cách tương tự chia 2 số thập phân

$$\begin{array}{r}
 0011 \\
 11 \overline{) 1001} \\
 \underline{011} \\
 0011 \\
 \underline{11} \\
 0
 \end{array}$$

$9 \div 3 = 3$

$$\begin{array}{r}
 0010.1 \\
 100 \overline{) 1010.0} \\
 \underline{100} \\
 100 \\
 \underline{100} \\
 0
 \end{array}$$

$10 \div 4 = 2.5$

- Phép chia 2 số có dấu được xử lý theo cách tương tự phép nhân 2 số có dấu

Phép cộng BCD (1)

■ Trình tự cộng 2 số BCD

- Sử dụng phép cộng nhị phân thông thường để cộng các nhóm mã BCD cho từng vị trí ký số BCD
- Ứng với mỗi vị trí, nếu tổng ≤ 9 , kết quả không cần sửa lỗi
- Nếu tổng của 2 ký số > 9 , kết quả được cộng thêm 6 (0110) để sửa lỗi, thao tác này **luôn tạo bit nhớ** (carry) cho vị trí ký số kế tiếp

45	0100	0101	← BCD for 45
+ <u>33</u>	+ <u>0011</u>	<u>0011</u>	← BCD for 33
78	0111	1000	← BCD for 78

Phép cộng BCD (2)

	0110	← BCD for 6
	+ 0111	← BCD for 7
	1101	← invalid sum
	0110	← add 6 for correction
<hr/>		
0001	0011	← BCD for 13
<u>1</u>	<u>3</u>	

47	0100	0111	← BCD for 47
+ 35	+ 0011	0101	← BCD for 35
82	0111	1100	← invalid sum in first digit
	1 ←	0110	← add 6 to correct
	<hr/>		
1000	0010	← correct BCD sum	
<u>8</u>	<u>2</u>		

Số học thập lục phân (1)

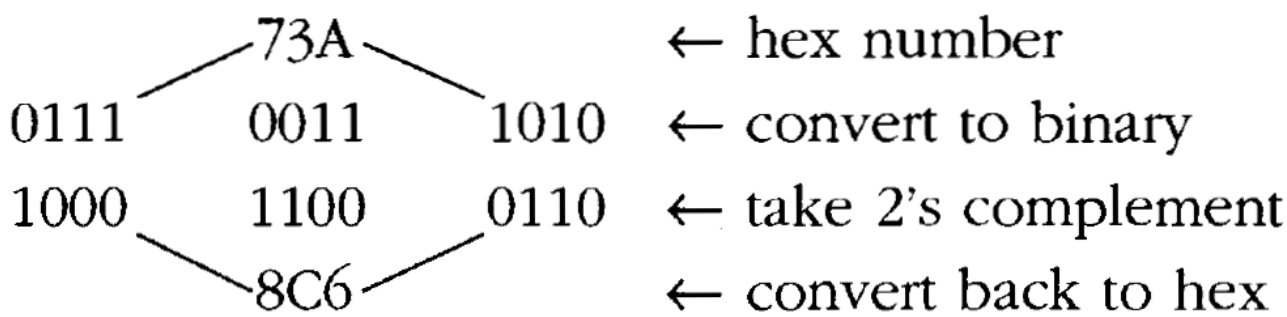
- Phép cộng 2 số thập lục phân được thực hiện theo cách tương tự phép cộng 2 số thập phân
 - Cộng 2 ký số hex dưới dạng thập phân
 - Nếu tổng ≤ 15 , biểu diễn trực tiếp bằng ký số hex
 - Nếu tổng ≥ 16 , trừ cho 16 và nhớ 1 vào vị trí ký số tiếp theo

$$\begin{array}{r}
 58 \\
 + 24 \\
 \hline
 7C
 \end{array}
 \qquad
 \begin{array}{r}
 3AF \\
 + 23C \\
 \hline
 5EB
 \end{array}$$

- Phép trừ 2 số thập lục phân
 - Chuyển số trừ sang dạng bù-2 và đem cộng vào số bị trừ
 - Loại bỏ bit nhớ sinh ra do phép cộng 2 ký số ở vị trí cuối cùng (nếu có)

Số học thập lục phân (2)

- Chuyển số hex sang dạng bù-2
 - Số hex → số nhị phân → dạng bù-2 → số hex



- Trừ mỗi ký số hex, lấy kết quả cộng thêm 1

F	F	F
<u>-7</u>	<u>-3</u>	<u>-A</u>
8	C	5
		+1
<hr/>		
8	C	6

$592_{16} - 3A5_{16}$

592
+ C5B
<hr/>
11ED

↑ Disregard carry

Số học thập lục phân (3)

- Dạng biểu diễn thập lục phân của các số có dấu

Hex Address	Stored Binary Data	Hex Value	Decimal Value
4000	00111010	3A	+58
4001	11100101	E5	−29
4002	01010111	57	+87
4003	10000000	80	−128

- ▣ Số có trọng số cao nhất (**MSD** – most significant digit) ≥ 8 , số được biểu diễn là số âm
- ▣ Nếu **MSD** ≤ 7 , số được biểu diễn là số dương

Bài tập

1. Thực hiện phép cộng trên các số BCD

a. $74 + 23$

c. $147 + 380$

b. $623 + 599$

2. Thực hiện phép toán trên số HEX

a. $3E91 - 2F93$

d. $F000 - FFFE$

b. $ABC + DEF$

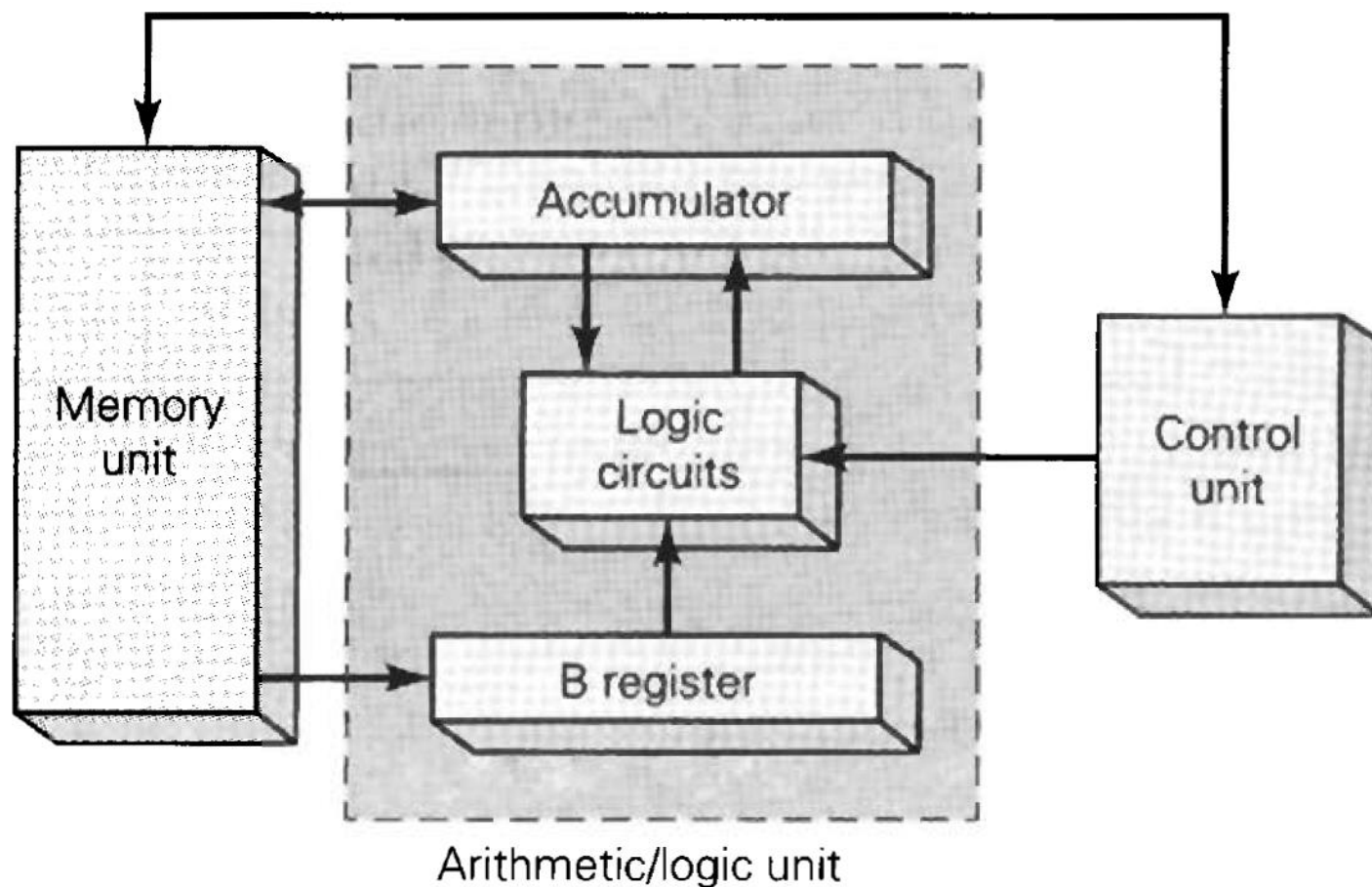
e. $3E91 + 2F93$

c. $91B - 6F2$

f. $2F00 - 4000$

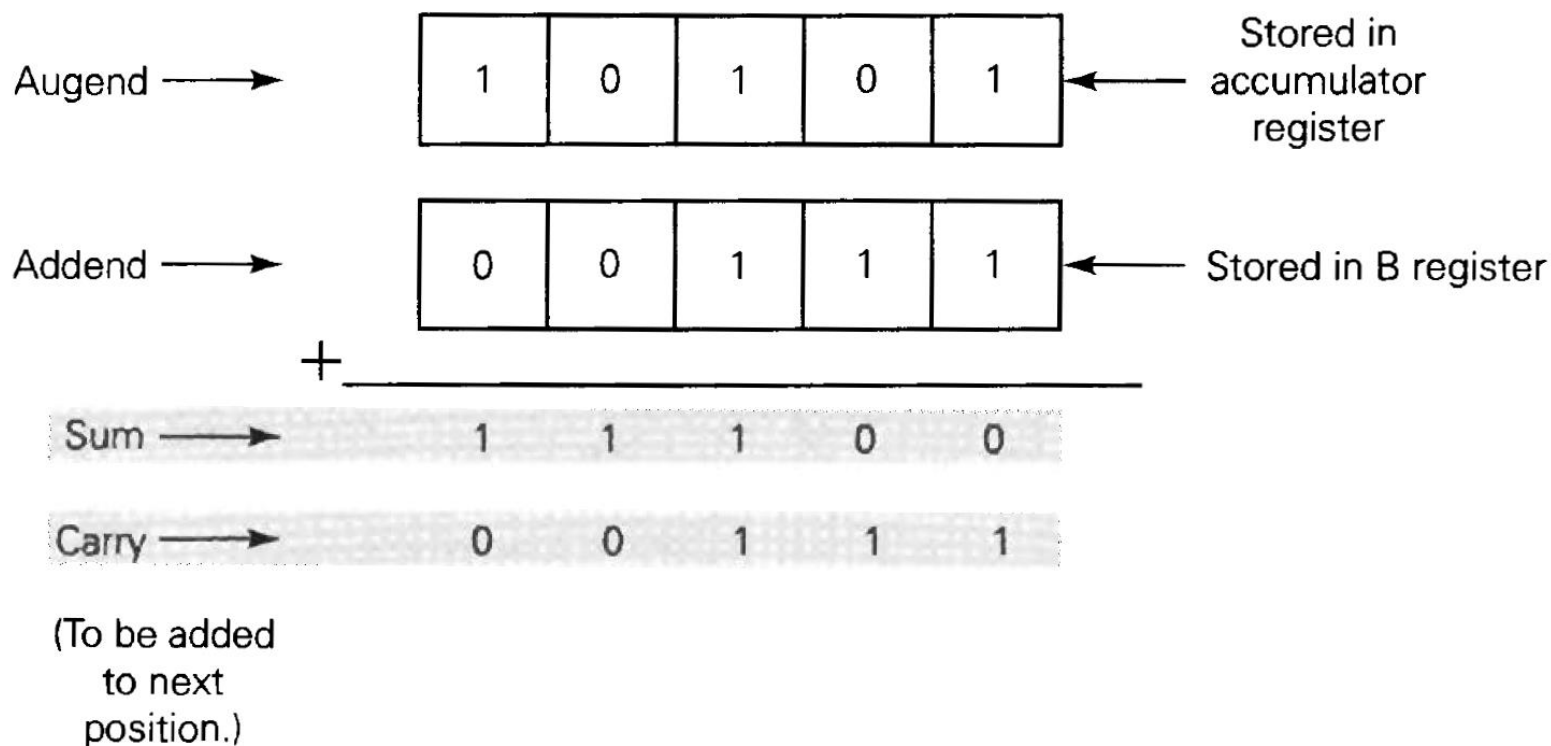
3. Một tài liệu kỹ thuật đi kèm một thiết bị số cho biết rằng, vùng nhớ khả dụng của thiết bị nằm trong khoảng địa chỉ dạng HEX 0200 đến 03FF và 4000 đến 7FD0. Hãy cho biết số ô nhớ khả dụng tổng cộng của thiết bị này.

Đơn vị số học và luận lý (ALU)



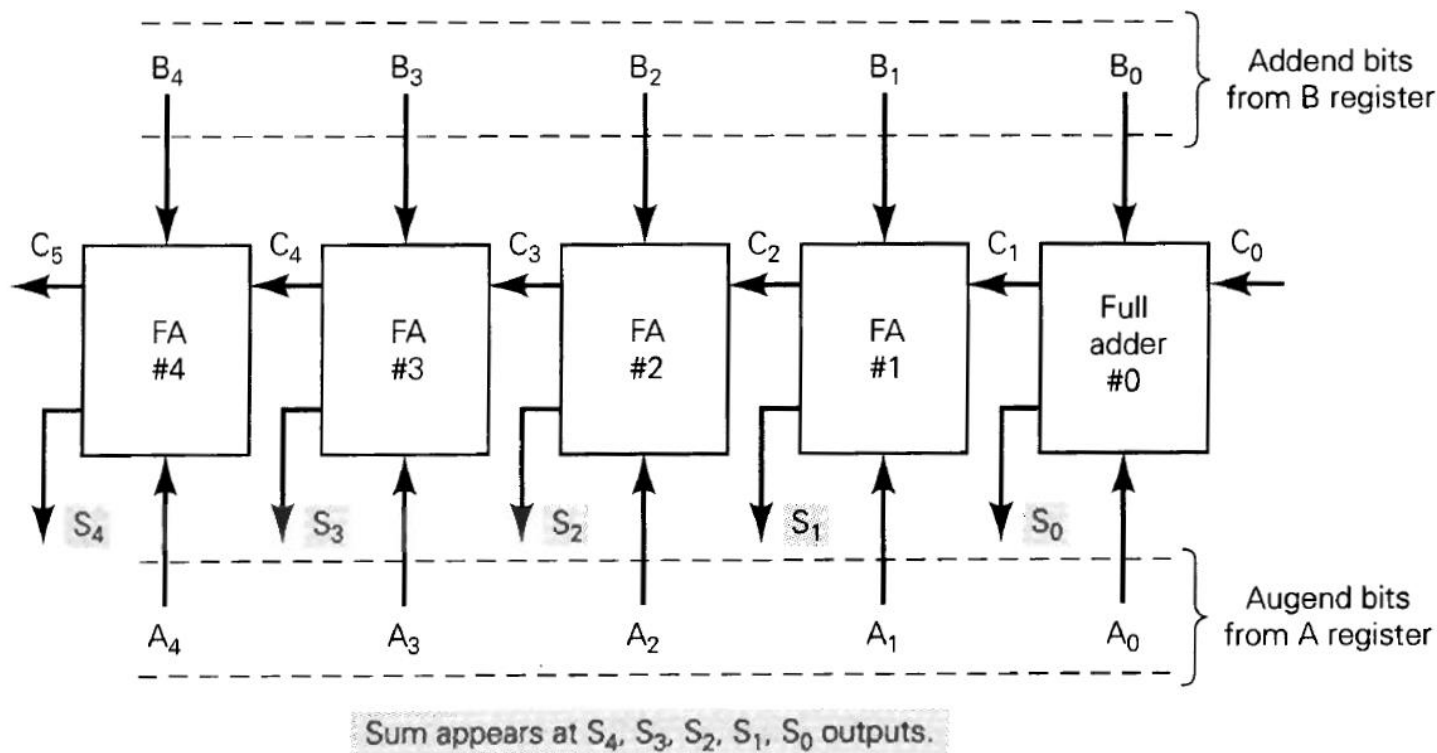
Mạch cộng nhị phân song song

- Toán hạng 1 (số bị cộng): lưu trong thanh ghi tích lũy (accumulator – A)
- Toán hạng 2 (số cộng): lưu trong thanh ghi B



Mạch cộng nhị phân song song

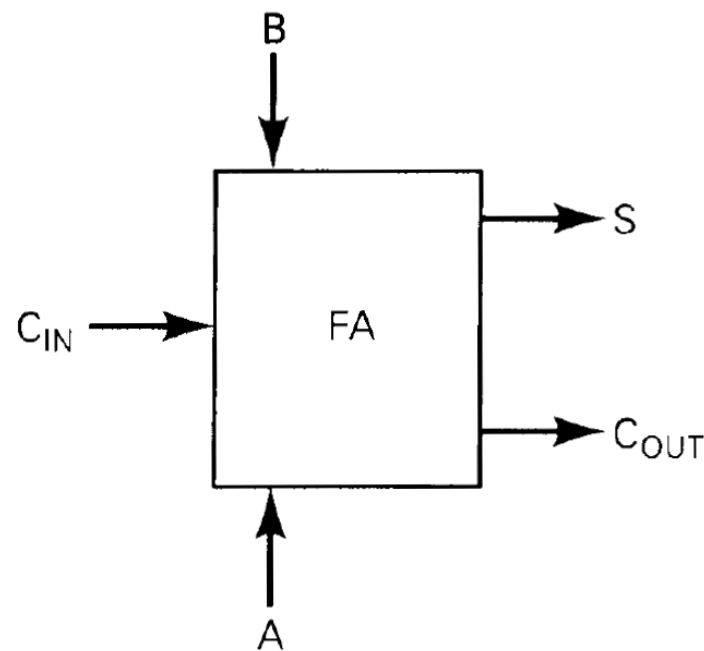
- Toạ hạng 1 và 2 được đưa vào mạch cộng toàn phần (**full adder**)
- Thao tác cộng trên mỗi bit được thực hiện đồng thời



Thiết kế mạch cộng toàn phần

- Lập bảng sự thật cho mạch cộng toàn phần
- Rút gọn biểu thức mạch cộng toàn phần bằng phương pháp đại số hoặc bìa Karnaugh
- Mạch hoàn chỉnh

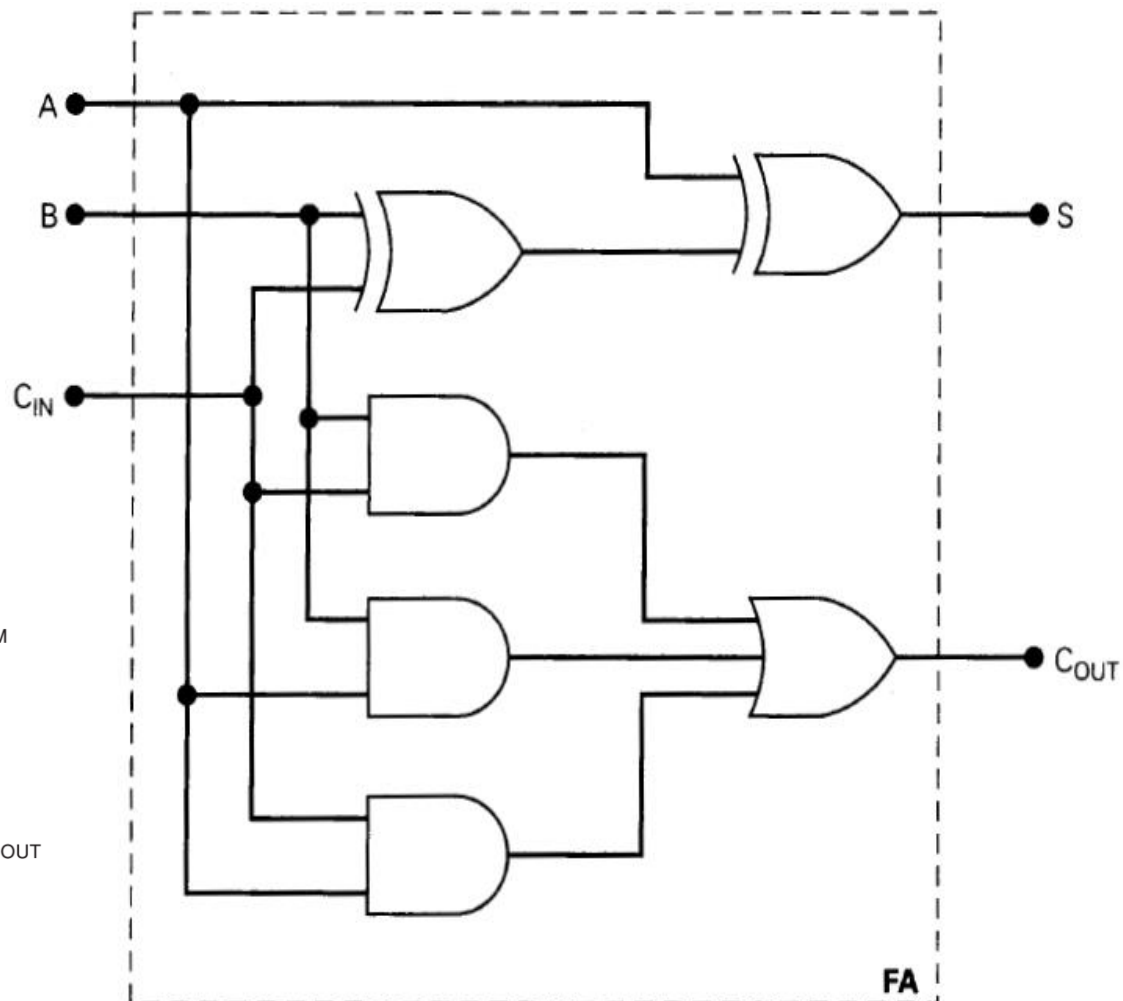
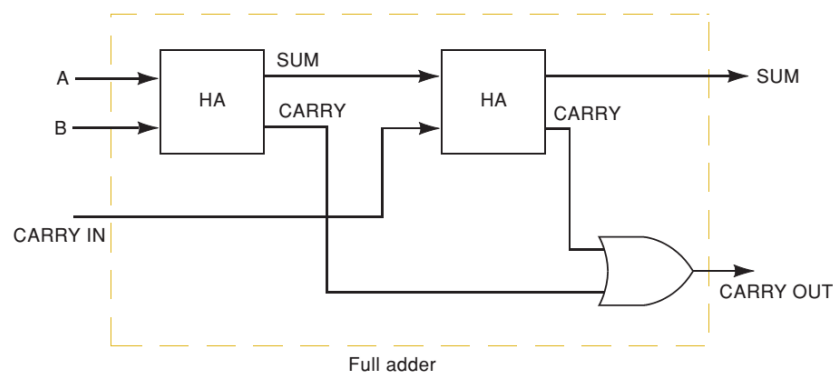
A	B	C_{IN}	S	C_{OUT}
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1



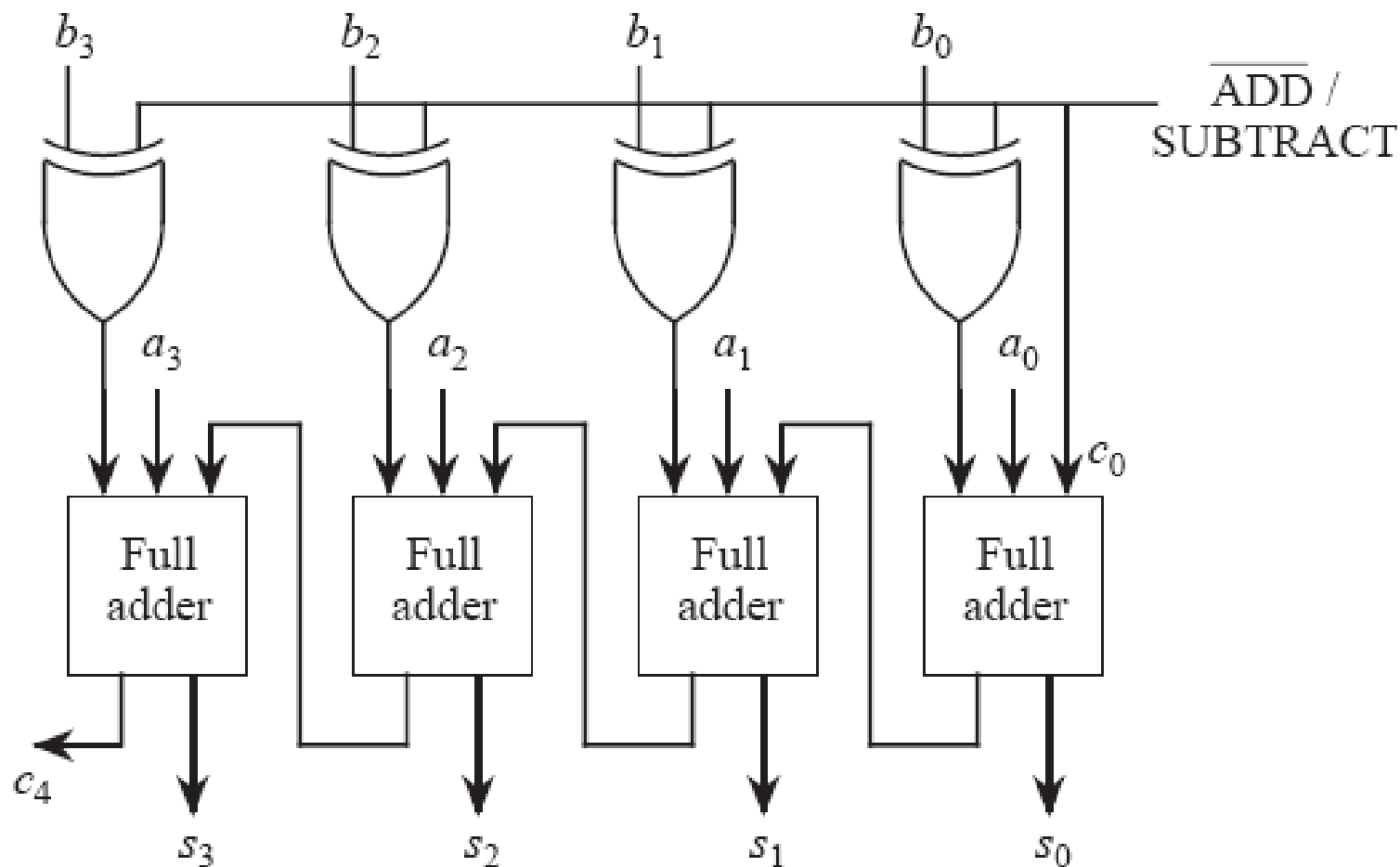
Thiết kế mạch cộng toàn phần

$$S = A \oplus (B \oplus C_{IN})$$

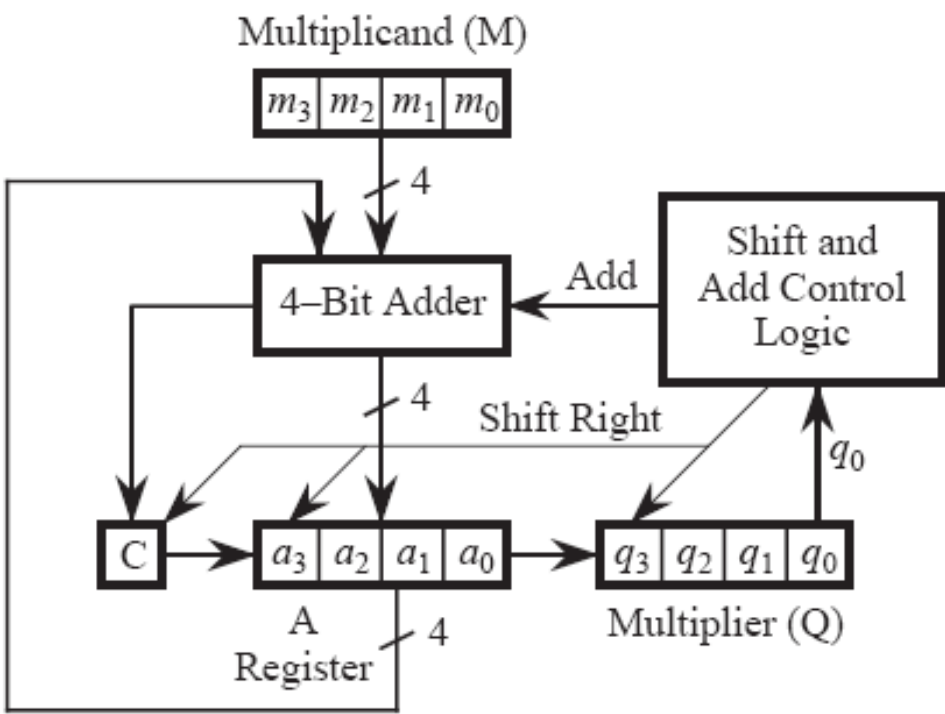
$$C_{OUT} = BC_{IN} + AC_{IN} + AB$$



Mạch cộng/trừ toàn phần



Mạch nhân nhị phân



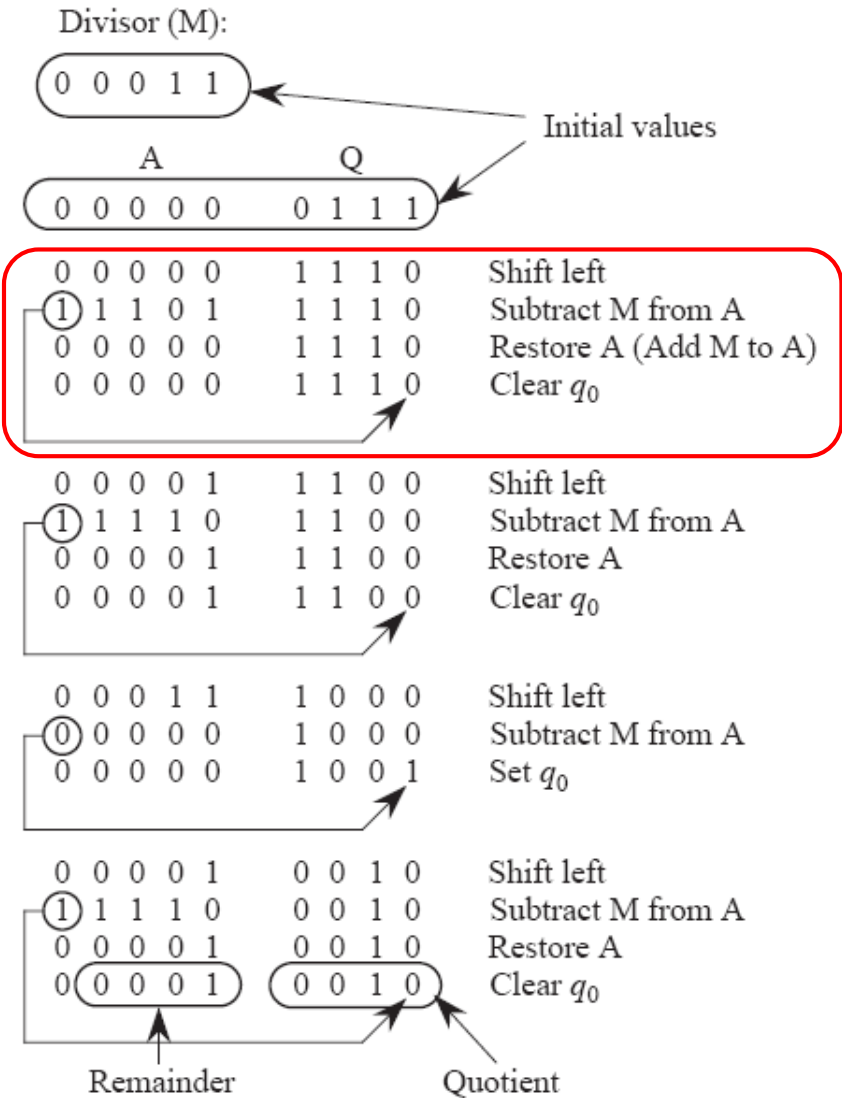
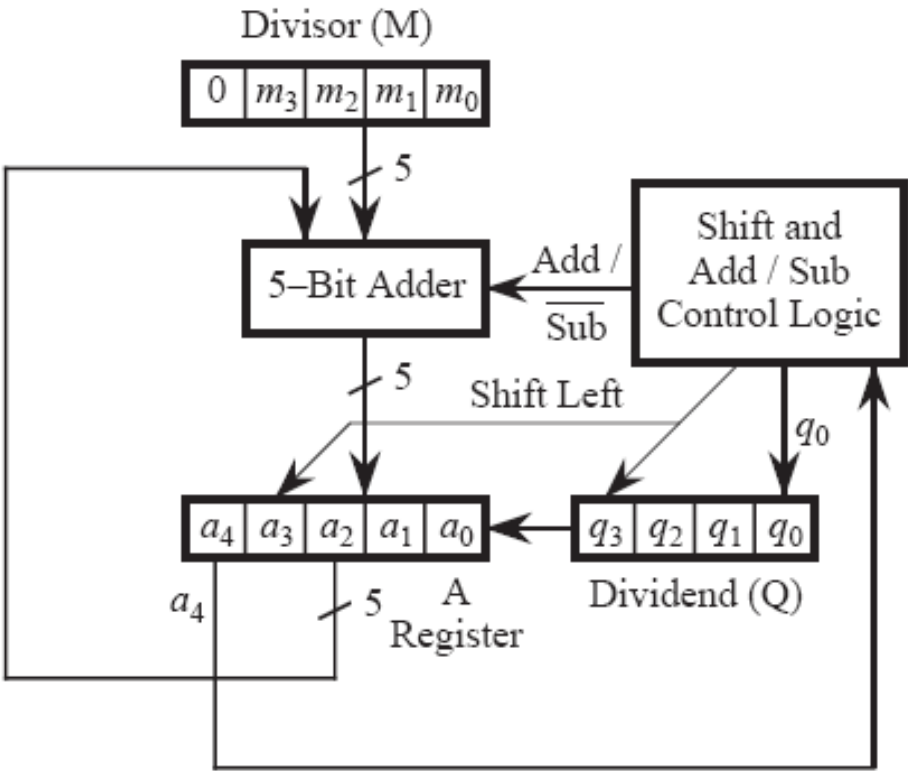
Multiplicand (M):

Initial values

C	A	Q	
0	0 0 0 0	1 0 1 1	
0	1 1 0 1	1 0 1 1	Add M to A
0	0 1 1 0	1 1 0 1	Shift
1	0 0 1 1	1 1 0 1	Add M to A
0	1 0 0 1	1 1 1 0	Shift
0	0 1 0 0	1 1 1 1	Shift (no add)
1	0 0 0 1	1 1 1 1	Add M to A
0	1 0 0 0	1 1 1 1	Shift

Product

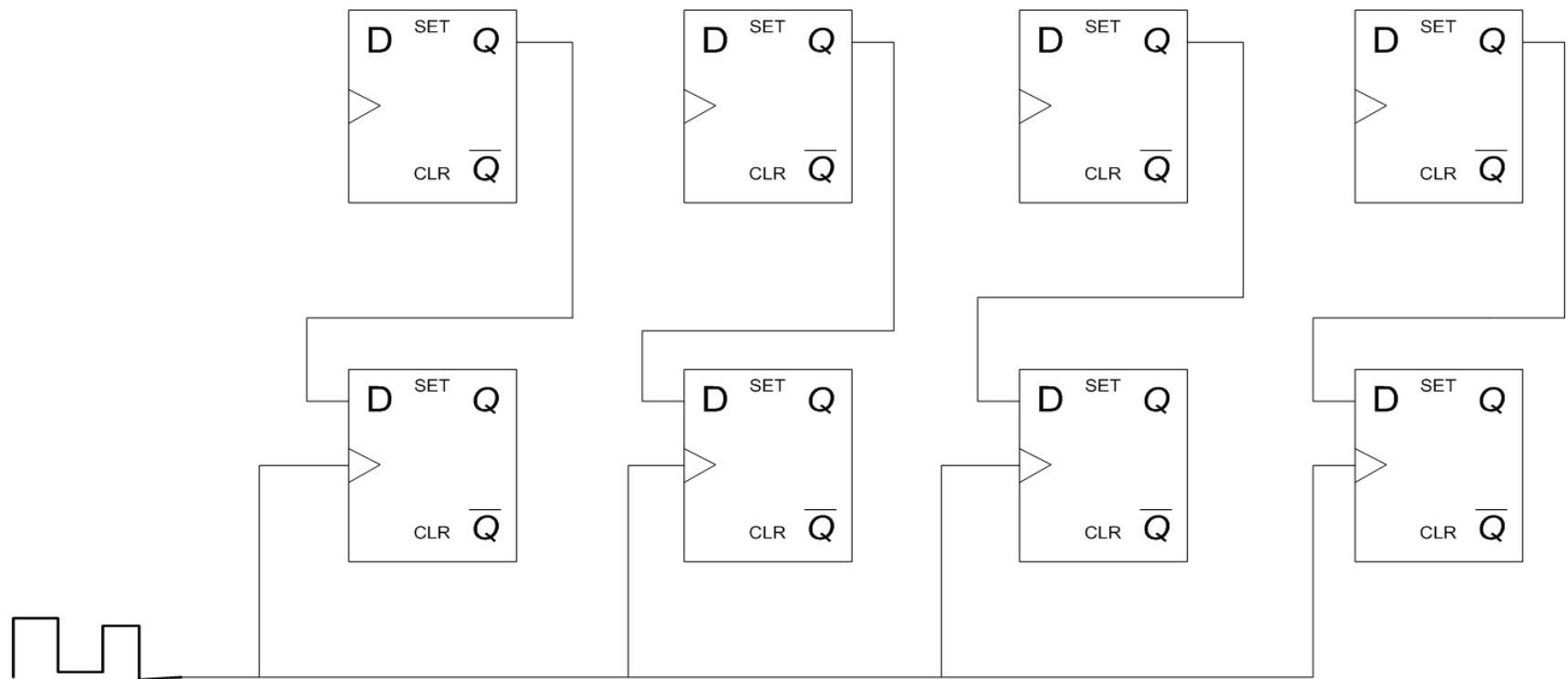
Mạch chia nhị phân



Truyền dữ liệu song song

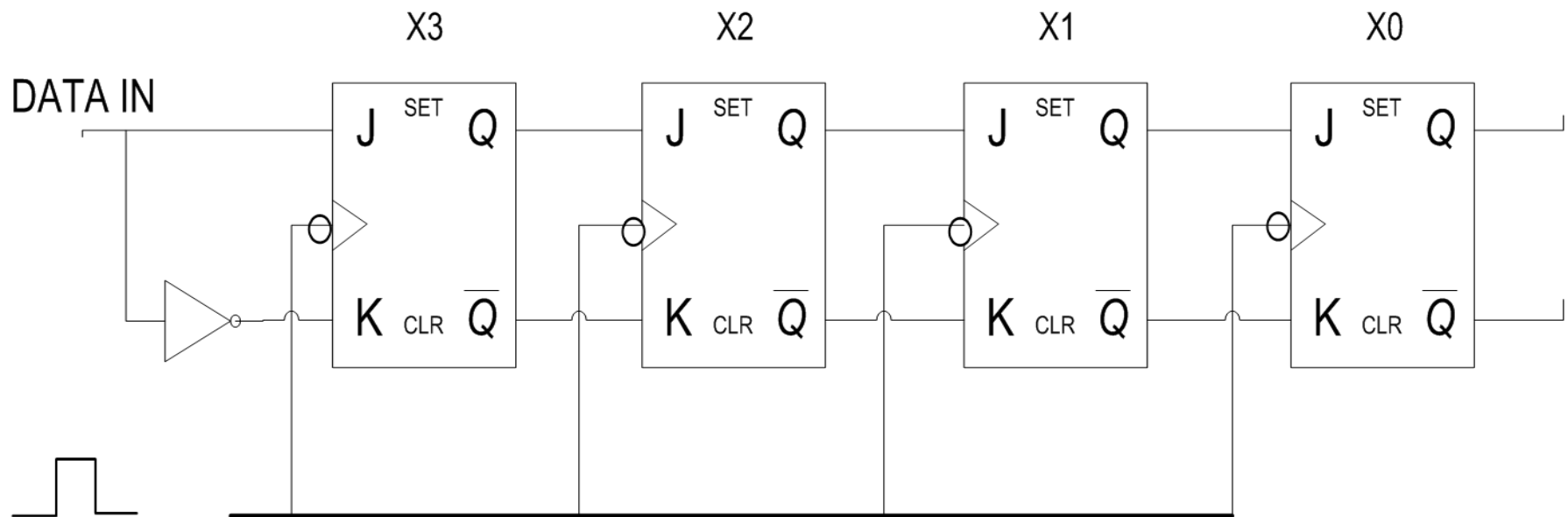
- Truyền dữ liệu song song (Parallel transfer)

Register X



Register y

Thanh ghi dịch (shift register)



Tổng kết

- Biểu diễn số có dấu: Sign-magnitude, bù 1, bù 2
- Số học nhị phân với số có dấu
 - Cộng, trừ, nhân, chia số bù 2
- Cộng trừ số BCD
- Biểu diễn số có dấu dạng HEX
- Đơn vị số học và luận lý ALU
 - Mạch cộng, trừ
 - Mạch nhân
 - Mạch chia

Đọc thêm

- Chương 6: **Digital Arithmetic: Operations and Circuits** trong sách **Digital System** của Ronal Tocci