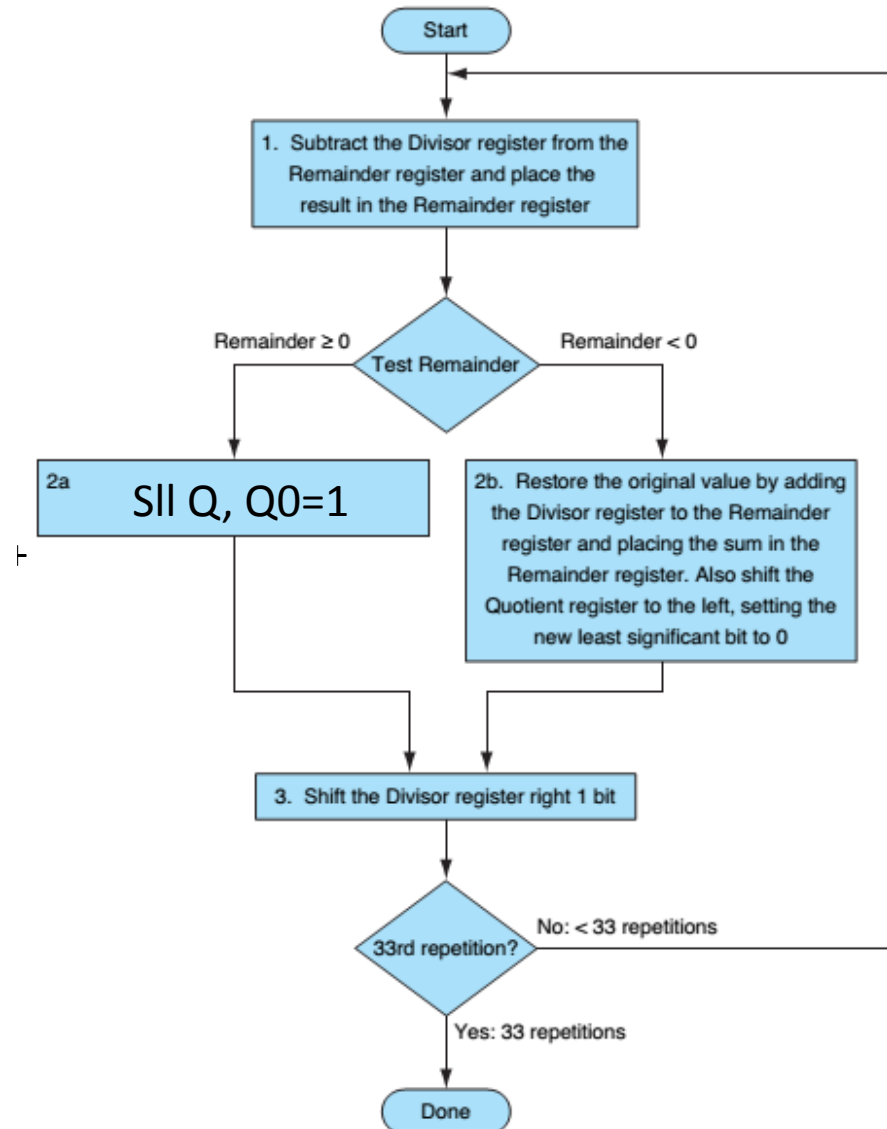
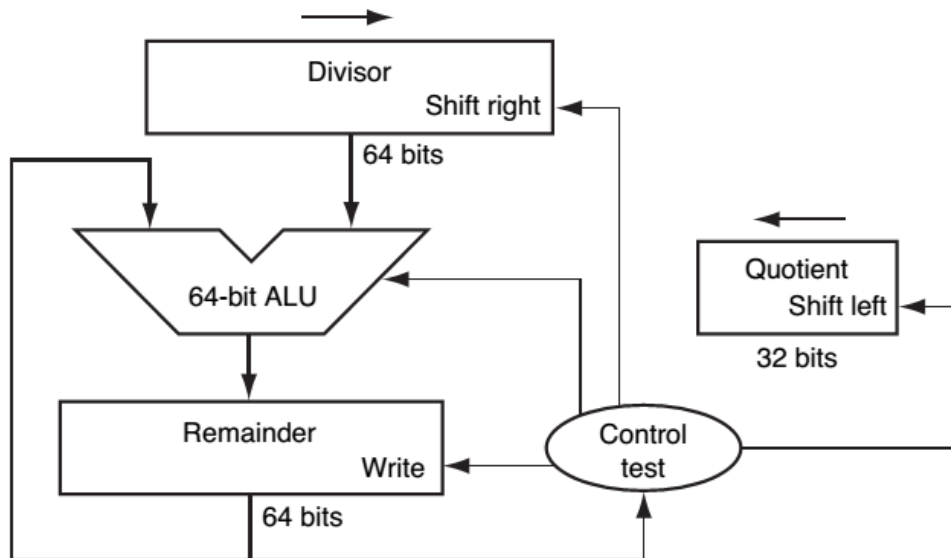


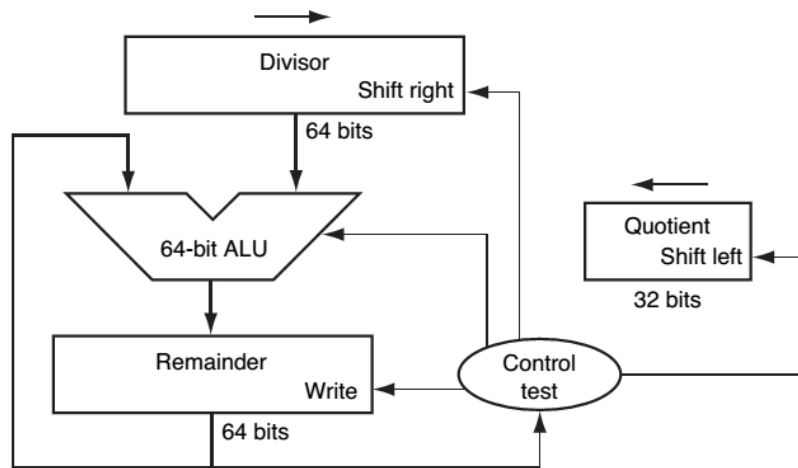
Ví dụ cho phép chia (2 ví dụ)

Ví dụ 1:

Thực hiện phép chia $50_{(8)}/23_{(8)}$ (sử dụng số 6 bit không dấu) theo cấu trúc phần cứng như hình



Lưu đồ giải thuật đi kèm cho cấu trúc phần cứng



Ví dụ 1:

$$50_{(8)} / 23_{(8)} = ?$$

$$\text{Dividend} = 50_8 = 101\ 000_2$$

$$\text{Divisor} = 23_8 = 010\ 011_2$$

Cấu trúc phần cứng như hình vẽ là đang làm việc trên phép chia số 32 bits

Có: thanh ghi divisor 64 bits

thanh ghi quotient là 32 bits

thanh ghi remainder là 64 bits

Ví dụ 1 yêu cầu phép chia dùng số 6 bits không dấu, sử dụng cấu trúc phần cứng tương tự như hình, vậy các thanh ghi trong ví dụ cần được khởi tạo với số bit tương ứng:

=> thanh ghi divisor 12 bits (giá trị khởi tạo **010011000000** – 6 bits cao là giá trị của divisor, 6 bits thấp đưa 0 vào)

thanh ghi quotient là 6 bits (giá trị khởi tạo 000000)

thanh ghi remainder là 12 bits (giá trị khởi tạo 000000**101000** - 6 bits cao đưa 0 vào, 6 bits thấp đưa dividend vào)

-Sau khi khởi tạo xong. Mỗi vòng lặp (iteration) sẽ gồm 3 bước:

- B1. Lấy toàn bộ remainder trừ divisor (hiệu lưu đè lên giá trị remainder hiện đang có)
- B2. Kiểm tra hiệu vừa tính ở trên là âm hay dương (kiểm tra bit cao nhất, nếu 1 là âm, nếu 0 là dương):

Nếu âm:

- Dịch trái quotient 1 bit
- Thêm 0 vào bit 0 của quotient (thật ra thao tác này không cần, vì dịch trái 1 bit mặc định đã thêm bit 0 vào bit 0 của nó)
- Lấy giá trị hiện tại của remainder cộng với divisor, tổng lưu lại vào remainder

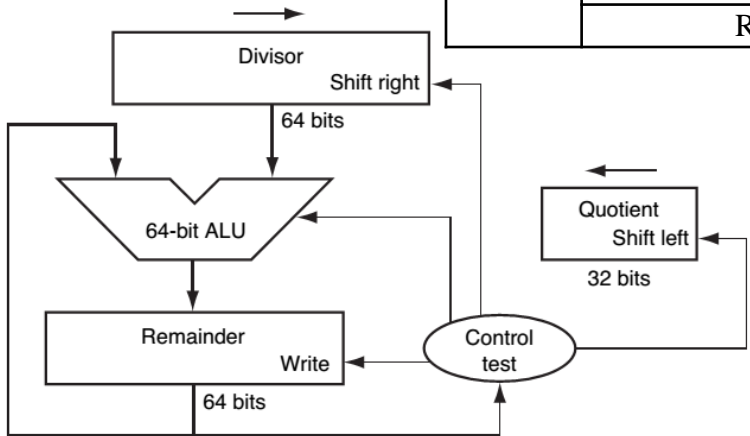
Nếu dương:

- Dịch trái quotient 1 bit
- Chuyển bit 0 của quotient thành 1
- B3. Dịch phải Divisor 1 bit
- Số vòng lặp cho giải thuật này đúng bằng số bit dùng biểu diễn + 1 (ví dụ 1 yêu cầu dùng số 6 bit, thì có 7 vòng lặp)
- Sau khi kết thúc số vòng lặp, giá trị trong thanh ghi quotient chính là kết quả phép chia, giá trị trong remainder là phần dư

Step	Action	Quotient	Divisor	Remainder
0	Initial Vals (Giá trị khởi tạo)	000 000	010 011 000000	000000 101000

Ví dụ 1:
 $50_{(8)} / 23_{(8)} =$
 $?$
 Dividend =
 $50_8 =$
 $101\ 000_2$
 Divisor =
 $23_8 =$
 $010\ 011_2$

Step	Action	Quotient	Divisor	Remainder
0	Initial Vals	000 000	010 011 000 000	000 000 101 000
1	Rem = Rem – Div	000 000	010 011 000 000	1 01 101 101 000
	Rem < 0, R + D, Q<<	000 000	010 011 000 000	000 000 101 000
	Rshift Div	000 000	001 001 100 000	000 000 101 000
2	Rem = Rem – Div	000 000	001 001 100 000	1 10 111 001 000
	Rem < 0, R + D, Q<<	000 000	001 001 100 000	000 000 101 000
	Rshift Div	000 000	000 100 110 000	000 000 101 000
3	Rem = Rem – Div	000 000	000 100 110 000	1 11 011 111 000
	Rem < 0, R + D, Q<<	000 000	000 100 110 000	000 000 101 000
	Rshift Div	000 000	000 010 011 000	000 000 101 000
4	Rem = Rem – Div	000 000	000 010 011 000	1 11 110 010 000
	Rem < 0, R + D, Q<<	000 000	000 010 011 000	000 000 101 000
	Rshift Div	000 000	000 001 001 100	000 000 101 000
5	Rem = Rem – Div	000 000	000 001 001 100	1 11 110 111 100
	Rem < 0, R + D, Q<<	000 000	000 001 001 100	000 000 101 000
	Rshift Div	000 000	000 000 100 110	000 000 101 000
6	Rem = Rem – Div	000 000	000 000 100 110	0 00 000 000 010
	Rem < 0, R + D, Q<<, Q ₀ = 1	000 001	000 000 100 110	000 000 000 010
	Rshift Div	000 001	000 000 010 011	000 000 000 010
7	Rem = Rem – Div	000 001	000 000 010 011	1 11 111 101 111
	Rem < 0, R + D, Q<<	000 010	000 000 010 011	000 000 000 010
	Rshift Div	000 010	000 000 001 101	000 000 000 010



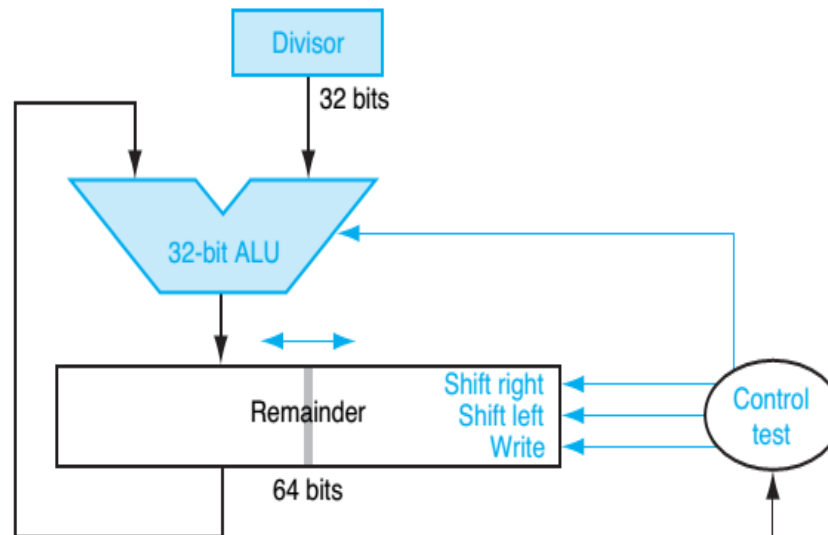
Thương số Phần dư

<<: Ký hiệu của dịch trái (ở đây là dịch trái 1 bit)

Ví dụ cho phép chia

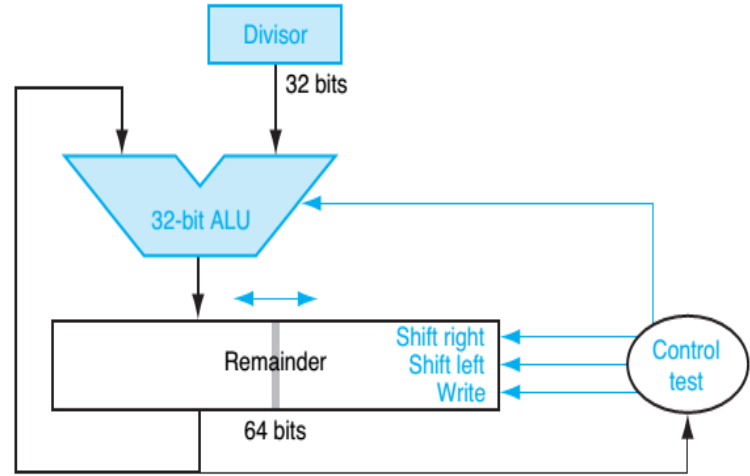
Ví dụ 2:

Thực hiện phép chia $50_{(8)}/23_{(8)}$ (sử dụng số 6 bit không dấu) theo cấu trúc phần cứng như hình (đề bài giống ví dụ 1, nhưng sử dụng cấu trúc phần cứng khác)



Trong cấu trúc phần cứng này, thanh ghi Remainder còn được gọi là Remainder/Quotient

Ví dụ 2:
 $50_{(8)} / 23_{(8)} = ?$
 Dividend = $50_8 = 101\ 000_2$
 Divisor = $23_8 = 010\ 011_2$



Cấu trúc phần cứng như hình vẽ là đang làm việc trên phép chia số 32 bits
 Có: thanh ghi divisor 32 bits
 thanh ghi remainder hay còn gọi là remainder/quotient là 64 bits
 Ví dụ 2 yêu cầu phép chia dùng số 6 bits không dấu, sử dụng cấu trúc phần cứng tương tự như hình, vậy các thanh ghi trong ví dụ cần được khởi tạo:
 ⇒ thanh ghi divisor 6 bits (giá trị khởi tạo **010011**)
 thanh ghi remainder hoặc remainder/quotient là 12 bits (giá trị khởi tạo 000000**101000** - 6 bits cao đưa 0 vào, 6 bits thấp đưa dividend vào)

- Sau khi khởi tạo xong. Mỗi vòng lặp (iteration) sẽ gồm 3 bước:
- B1. Dịch trái 1 bit Remainder/Quotient
 - B2. Lấy nửa cao của Remainder/Quotient trừ Divisor, giá trị lưu lại vào nửa cao, còn nửa thấp giữ nguyên
 - B3. Kiểm tra giá trị của Remainder/Quotient
- Nếu âm:**
- Lấy giá trị hiện tại của nửa cao Remainder/Quotient cộng với divisor, tổng lưu lại vào nửa cao của Remainder/Quotient, nửa thấp giữ nguyên
- Nếu dương:**
- Chuyển bit 0 của Remainder/Quotient thành 1
 - Số vòng lặp cho giải thuật này đúng bằng số bit dùng biểu diễn (ví dụ 1 yêu cầu dùng số 6 bit, thì có 6 vòng lặp)
 - Sau khi kết thúc số vòng lặp, giá trị nửa cao trong thanh ghi Remainder/Quotient chính là phần dư, nửa thấp chính kết quả phép chia

Step	Action	Divisor	Remainder/Quotient
0	Initial Vals	010 011	000 000 101 000

Step	Action	Divisor	Remainder/Quotient
0	Initial Vals	010 011	000 000 101 000
1	R<<	010 011	000 001 010 000
	Rem = Rem – Div	010 011	101 110 010 000
	Rem < 0, R + D	010 011	000 001 010 000
2	R<<	010 011	000 010 100 000
	Rem = Rem – Div	010 011	101 111 100 000
	Rem < 0, R + D	010 011	000 010 100 000
3	R<<	010 011	000 101 000 000
	Rem = Rem – Div	010 011	110 010 000 000
	Rem < 0, R + D	010 011	000 101 000 000
4	R<<	010 011	001 010 000 000
	Rem = Rem – Div	010 011	110 111 000 000
	Rem < 0, R + D	010 011	001 010 000 000
5	R<<	010 011	010 100 000 000
	Rem = Rem – Div	010 011	000 001 000 000
	Rem > 0, R0 = 1	010 011	000 001 000 001
6	R<<	010 011	000 010 000 010
	Rem = Rem – Div	010 011	101 111 000 010
	Rem < 0, R + D	010 011	000 010 000 010

101110 = 000001 - 010011

Phần dư

Thương