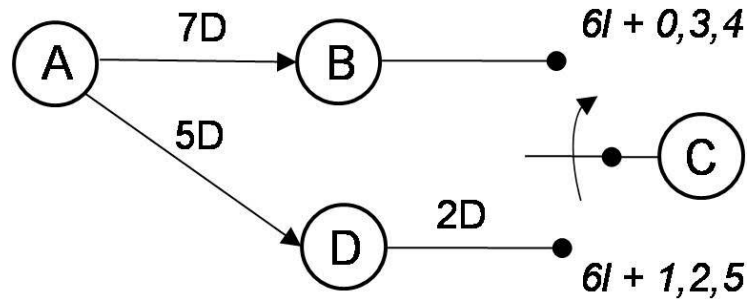


**Một số bài tập ôn thi HK**

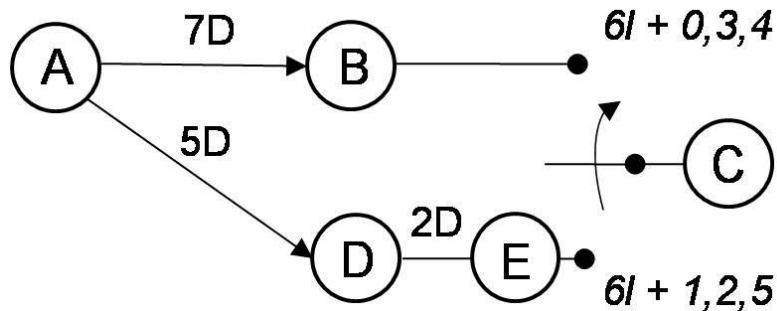
(Ngoài các dạng BT như KT tại lớp, SV xem thêm các dạng BT sau – BT 7, 8, và 9 không có thi)

1. Hãy thực hiện trải DFG sau ra 2 lần ( $J=2$ ).



**BG.**

Vì có phần tử Delay nối với khóa, ta thêm nút giả E:



Thực hiện trải ra DFG cho các nút A, B, D và E theo bảng sau:

$i$	$w$	$[U_i \rightarrow V] (i + w) \% 2$	$w_{\text{new}} = \lfloor (i + w) / 2 \rfloor$
0	7	1	3
1	7	0	4
0	5	1	2
1	5	0	3
0	2	0	1
1	2	1	1

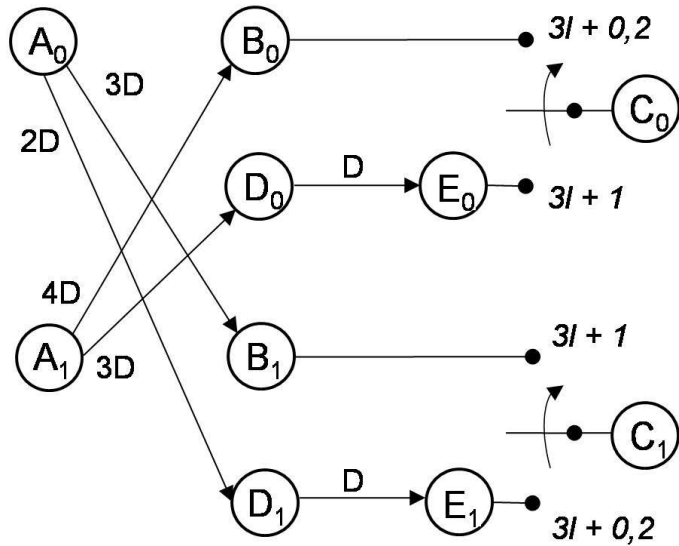
Thực hiện trải ra DFG cho các nút B, E và C với khóa theo các thời điểm sau:

$$Wl + u = J( W'l + \lfloor u/J \rfloor ) + (u \% J)$$

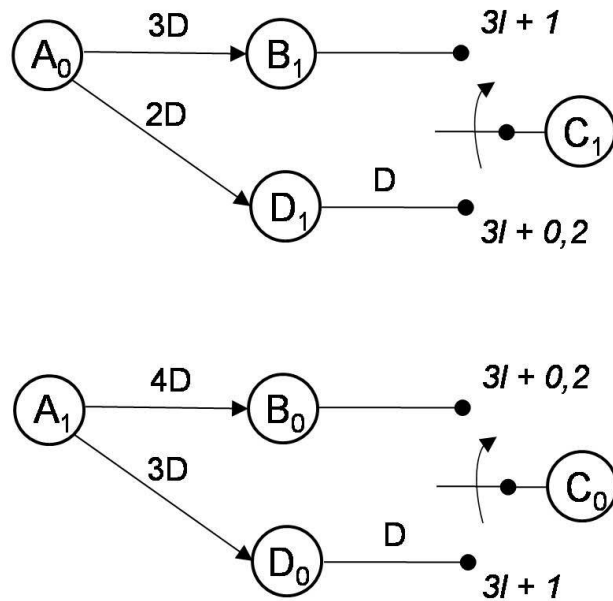
Ta có:  $W = W'J \Rightarrow W' = W/J = 6/2 = 3$ . Từ đó ta có bảng sau:

Vị trí khóa	Các thời điểm chuyển mạch
Bật lên	$6l + 0 = 2(3l + 0) + 0$ $6l + 3 = 2(3l + 1) + 1$ $6l + 4 = 2(3l + 2) + 0$
Bật xuống	$6l + 1 = 2(3l + 0) + 1$ $6l + 2 = 2(3l + 1) + 0$ $6l + 5 = 2(3l + 2) + 1$

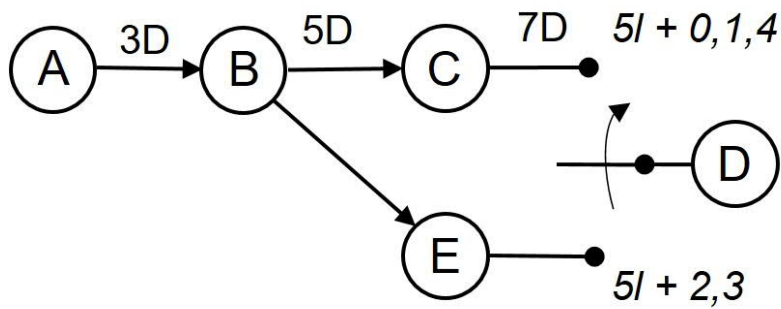
**Kết quả của unfolding khi vẫn còn nút giả:**



**Sau khi bỏ các nút giả và sắp xếp lại, ta có kết quả sau cùng:**

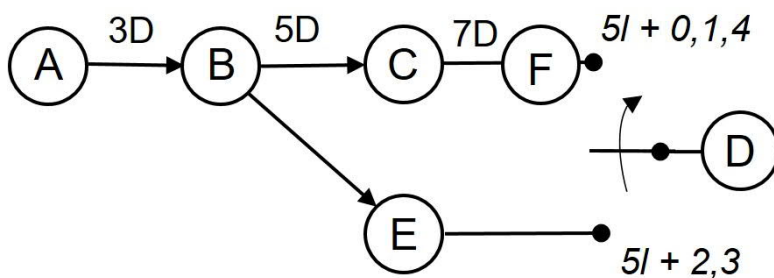


**2. Hãy thực hiện mở (trải) DFG sau ra 3 lần ( $J=3$ ):**



**BG.**

**Vì có phần tử Delay nối với khóa, ta thêm nút giả F:**



Vì chiều dài word  $W = 5$  không phải là bội số của  $J = 3$ , ta phải tìm  $L =$  bội chung nhỏ nhất của  $W$  và  $J = \text{lcm}(5, 3) = 15$ .

Thay thế các thời điểm chuyển đổi  $Wl + u$  bằng  $L/W$  thời điểm

$Ll + u + wW$ , với  $w = 0..L/W - 1 = 0..2$

tức là chu kỳ đã được thay đổi từ  $W$  thành  $L$ .

Thời điểm chuyển mạch cũ	Thời điểm chuyển mạch mới	Vị trí khóa
$5l + 0$	$15l + 0 + 0 \times 5 = 15l + 0$ $15l + 0 + 1 \times 5 = 15l + 5$ $15l + 0 + 2 \times 5 = 15l + 10$	Bật lên
$5l + 1$	$15l + 1 + 0 \times 5 = 15l + 1$ $15l + 1 + 1 \times 5 = 15l + 6$ $15l + 1 + 2 \times 5 = 15l + 11$	Bật lên
$5l + 4$	$15l + 4 + 0 \times 5 = 15l + 4$ $15l + 4 + 1 \times 5 = 15l + 9$ $15l + 4 + 2 \times 5 = 15l + 14$	Bật lên
$5l + 2$	$15l + 2 + 0 \times 5 = 15l + 2$ $15l + 2 + 1 \times 5 = 15l + 7$ $15l + 2 + 2 \times 5 = 15l + 12$	Bật xuống
$5l + 3$	$15l + 3 + 0 \times 5 = 15l + 3$ $15l + 3 + 1 \times 5 = 15l + 8$ $15l + 3 + 2 \times 5 = 15l + 13$	Bật xuống

Thực hiện trải ra DFG theo bảng sau với  $J = 3$ :

$$U_i \xrightarrow{W} V_i \Rightarrow U_i \xrightarrow{W'} V_k$$

i	w	k = $(i + w) \% 3$	w' = $\lfloor (i + w) / 3 \rfloor$
0	3	0	1
1	3	1	1
2	3	2	1
0	5	2	1
1	5	0	2
2	5	1	2
0	7	1	2
1	7	2	2
2	7	0	3

Thực hiện trải ra DFG cho các nút B, E và C với khóa dùng W mới = 12

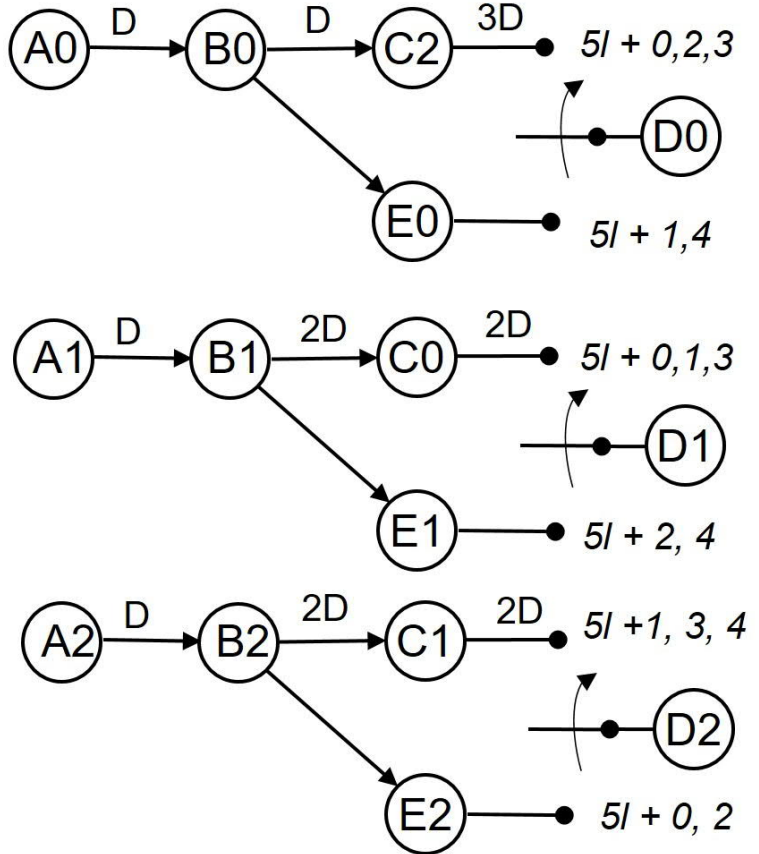
$$Wl + u = J( W'l + \lfloor u/J \rfloor ) + (u \% J)$$

Ta có:  $W = W'J \Rightarrow W' = W/J = 15/3 = 5$ . Từ đó ta có bảng sau:

**Bảng thời điểm chuyển**

**Kết quả sau cùng khi bỏ các nút giả**

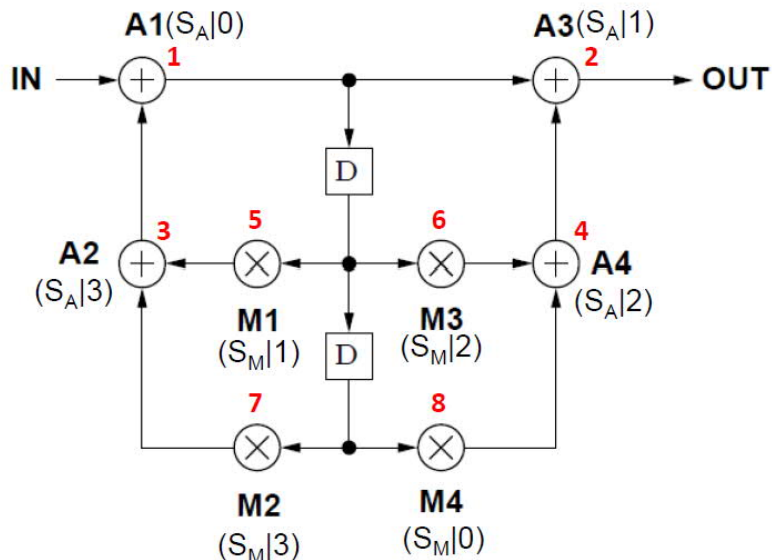
Vị trí khóa	Các thời điểm chuyển mạch
<b>Bật lên</b>	$15l + 0 = 3(5l + 0) + 0$ $15l + 1 = 3(5l + 0) + 1$ $15l + 4 = 3(5l + 1) + 1$ $15l + 5 = 3(5l + 1) + 2$ $15l + 6 = 3(5l + 2) + 0$ $15l + 9 = 3(5l + 3) + 0$ $15l + 10 = 3(5l + 3) + 1$ $15l + 11 = 3(5l + 3) + 2$ $15l + 14 = 3(5l + 4) + 2$
<b>Bật xuống</b>	$15l + 2 = 3(5l + 0) + 2$ $15l + 3 = 3(5l + 1) + 0$ $15l + 7 = 3(5l + 2) + 1$ $15l + 8 = 3(5l + 2) + 2$ $15l + 12 = 3(5l + 4) + 0$ $15l + 13 = 3(5l + 4) + 1$



3. Thực hiện gấp DFG sau với hệ số gấp N = 4. Giả sử các bộ cộng có pipeline 1 tầng, các bộ nhân có pipeline 2 tầng và thời gian tính toán  $T_A = 1$  ut và  $T_M = 2$  ut. Cho trước 2 tập gấp của các bộ nhân và cộng như sau:

$$S_M = \{M4, M1, M3, M2\}$$

$$S_A = \{A1, A3, A4, A2\}$$



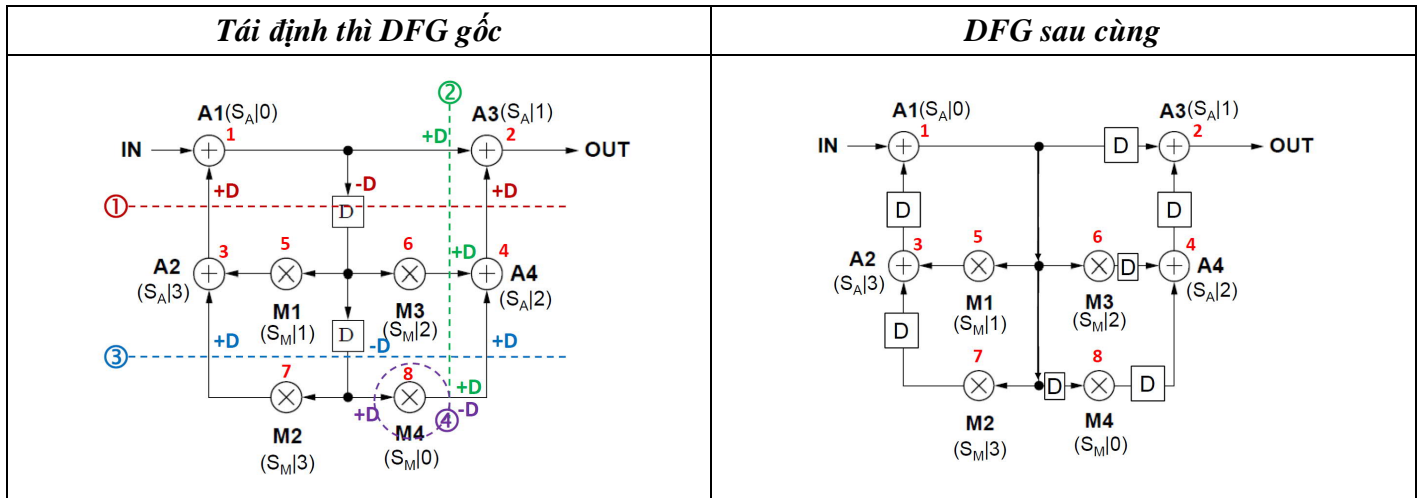
BG.

- Trước hết ta đánh số (màu đỏ) các nút trên hình 3 và ghi tập gấp cùng thứ tự của nút trong tập gấp kế bên nút đó.

<b>Phương trình gấp <math>D_F</math></b>	<b>Tái định thì cho ràng buộc gấp</b>
$D_F(U \xrightarrow{e} V) = Nw(e) - P_u + v - u$	$r(U) - r(V) \leq \left\lfloor \frac{D_F(U \xrightarrow{e} V)}{N} \right\rfloor$
$D_F(1 \rightarrow 2) = 4(0) - 1 + 1 - 0 = 0$	$r(1) - r(2) \leq 0$
$D_F(1 \rightarrow 5) = 4(1) - 1 + 1 - 0 = 4$	$r(1) - r(5) \leq 1$
$D_F(1 \rightarrow 6) = 4(1) - 1 + 2 - 0 = 5$	$r(1) - r(6) \leq 1$
$D_F(1 \rightarrow 7) = 4(2) - 1 + 3 - 0 = 10$	$r(1) - r(7) \leq 2$
$D_F(1 \rightarrow 8) = 4(2) - 1 + 0 - 0 = 7$	$r(1) - r(8) \leq 1$
<b><math>D_F(3 \rightarrow 1) = 4(0) - 1 + 0 - 3 = -4</math></b>	<b><math>r(3) - r(1) \leq -1</math></b>
<b><math>D_F(4 \rightarrow 2) = 4(0) - 1 + 1 - 2 = -2</math></b>	<b><math>r(4) - r(2) \leq -1</math></b>
$D_F(5 \rightarrow 3) = 4(0) - 2 + 3 - 1 = 0$	$r(5) - r(3) \leq 0$
<b><math>D_F(6 \rightarrow 4) = 4(0) - 2 + 2 - 2 = -2</math></b>	<b><math>r(6) - r(4) \leq -1</math></b>
<b><math>D_F(7 \rightarrow 3) = 4(0) - 2 + 3 - 3 = -2</math></b>	<b><math>r(7) - r(3) \leq -1</math></b>
$D_F(8 \rightarrow 4) = 4(0) - 2 + 2 - 0 = 0$	$r(8) - r(2) \leq 4$

Với bảng trên ta thấy việc tái định thì sẽ phải xảy ra với các tập cắt đi qua các nhánh  $3 \rightarrow 1$ ,  $4 \rightarrow 2$ ,  $6 \rightarrow 4$  và  $7 \rightarrow 3$ ;  $r(u) - r(v) \leq -1$  với tất cả nhánh này  $\Rightarrow$  DFG sau cùng phải có 1D tại các nhánh đó.

Sau khi tái định thì (các đường màu đứt nét có ghi thứ tự của các tập cắt kế bên) và tính các D tại mỗi nhánh ta có kết quả sau:



**Chú ý:** Nếu kiểm chứng lại với DFG đã tái định thì, ta thấy chúng thỏa điều kiện gấp:

$D_F$	$r(u) - r(v)$
$D_F(1 \rightarrow 2) = 4(1) - 1 + 1 - 0 = 4$	$r(1) - r(2) \leq 1$
$D_F(1 \rightarrow 5) = 4(0) - 1 + 1 - 0 = 0$	$r(1) - r(5) \leq 0$
$D_F(1 \rightarrow 6) = 4(0) - 1 + 2 - 0 = 1$	$r(1) - r(6) \leq 0$
$D_F(1 \rightarrow 7) = 4(0) - 1 + 3 - 0 = 2$	$r(1) - r(7) \leq 0$
$D_F(1 \rightarrow 8) = 4(1) - 1 + 0 - 0 = 3$	$r(1) - r(8) \leq 0$
$D_F(3 \rightarrow 1) = 4(1) - 1 + 0 - 3 = 0$	$r(3) - r(1) \leq 0$
$D_F(4 \rightarrow 2) = 4(1) - 1 + 1 - 2 = 2$	$r(4) - r(2) \leq 0$
$D_F(5 \rightarrow 3) = 4(0) - 2 + 3 - 1 = 0$	$r(5) - r(3) \leq 0$
$D_F(6 \rightarrow 4) = 4(1) - 2 + 2 - 2 = 2$	$r(6) - r(4) \leq 0$
$D_F(7 \rightarrow 3) = 4(1) - 2 + 3 - 3 = 2$	$r(7) - r(3) \leq 0$
$D_F(8 \rightarrow 4) = 4(1) - 2 + 2 - 0 = 4$	$r(8) - r(4) \leq 1$

- Ta thực hiện việc gấp (folding) DFG đã được tái định thì với  $N = 4$   
 Tính thời gian  $T_{in}$  và  $T_{out}$  của mỗi nút để tối thiểu hóa thành ghi:

$D_F$	Nút	$T_{in} \rightarrow T_{out}$
$D_F(1 \rightarrow 2) = 4(1) - 1 + 1 - 0 = 4$	1	1 → 5
$D_F(1 \rightarrow 5) = 4(0) - 1 + 1 - 0 = 0$	2	-----
$D_F(1 \rightarrow 6) = 4(0) - 1 + 2 - 0 = 1$	3	4 → 4
$D_F(1 \rightarrow 7) = 4(0) - 1 + 3 - 0 = 2$	4	3 → 5
$D_F(1 \rightarrow 8) = 4(1) - 1 + 0 - 0 = 3$	5	3 → 3
$D_F(3 \rightarrow 1) = 4(1) - 1 + 0 - 3 = 0$	6	4 → 6
$D_F(4 \rightarrow 2) = 4(1) - 1 + 1 - 2 = 2$	7	5 → 7
$D_F(5 \rightarrow 3) = 4(0) - 2 + 3 - 1 = 0$	8	2 → 6
$D_F(6 \rightarrow 4) = 4(1) - 2 + 2 - 2 = 2$		
$D_F(7 \rightarrow 3) = 4(1) - 2 + 3 - 3 = 2$		
$D_F(8 \rightarrow 4) = 4(1) - 2 + 2 - 0 = 4$		

**Chú ý:** Bảng trên dùng công thức cho T như sau:

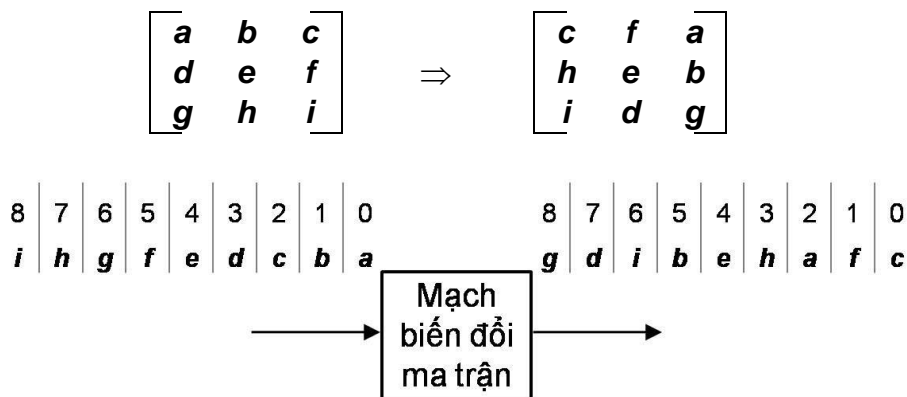
Thời gian vào  $T_{in}$  và ra  $T_{out}$  tương ứng cho nút U

$$T_{in} = u + P_u \text{ và } T_{out} = u + P_u + \max_v \{D_F(U \rightarrow V)\}$$

4. Thiết kế mạch biến đổi ma trận 3 x 3 với 1 ngõ vào và 1 ngõ ra, theo quy tắc sau:

Ma trận vào

Ma trận ra



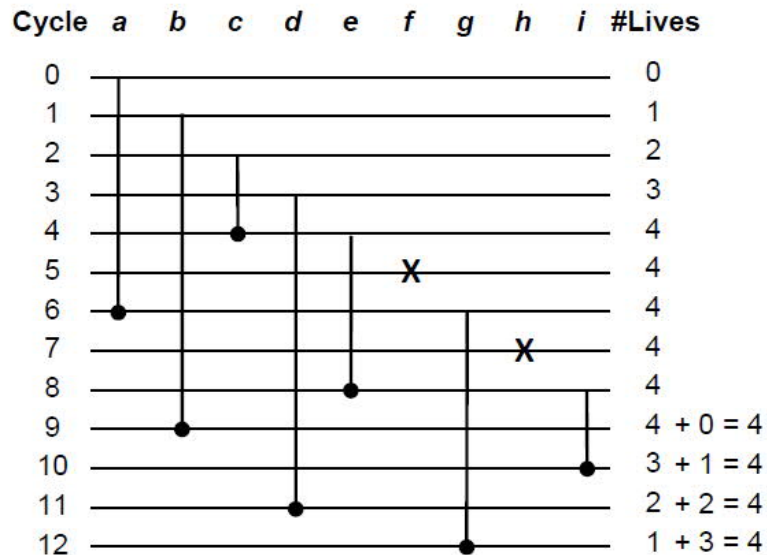
- a) Lập bảng thời gian sống và giản đồ thời gian sống, từ đó suy ra số thanh ghi cần cho mạch này là bao nhiêu?  
 b) Lập bảng cấp phát thanh ghi thuận-nghịch. (Không cần vẽ mạch)

BG.

- Bảng thời gian sống:**

Mẫu	$T_{in}$	$T_{zout}$	$T_{diff}$	$T_{out}$	Đời sống
<b>a</b>	0	2	2	6	0 → 6
<b>b</b>	1	5	4	9	1 → 9
<b>c</b>	2	0	-2	4	2 → 4
<b>d</b>	3	7	4	11	3 → 11
<b>e</b>	4	4	0	8	4 → 8
<b>f</b>	5	1	-4	5	5 → 5
<b>g</b>	6	8	-2	12	6 → 12
<b>h</b>	7	3	-4	7	7 → 7
<b>i</b>	8	6	-2	10	8 → 10

- Giản đồ thời gian sống:



Như vậy số thanh ghi cần sử dụng là 4.

- Bảng cấp phát thanh ghi thuận-ngược

Cycle	Input	R1	R2	R3	R4	Output
0	a					
1	b	a				
2	c	b	a			
3	d	c	b	a		
4	e	d	c	b	a	c
5	f	e	d	a	b	f
6	g	b	e	d	a	a
7	h	g	b	e	d	h
8	i	d	g	b	e	e
9		i	d	g	b	b
10			i	d	g	i
11				g	d	d
12					g	g

5. Xét biểu diễn không gian ở hình 7.35 để tính hoạt động lọc sau:

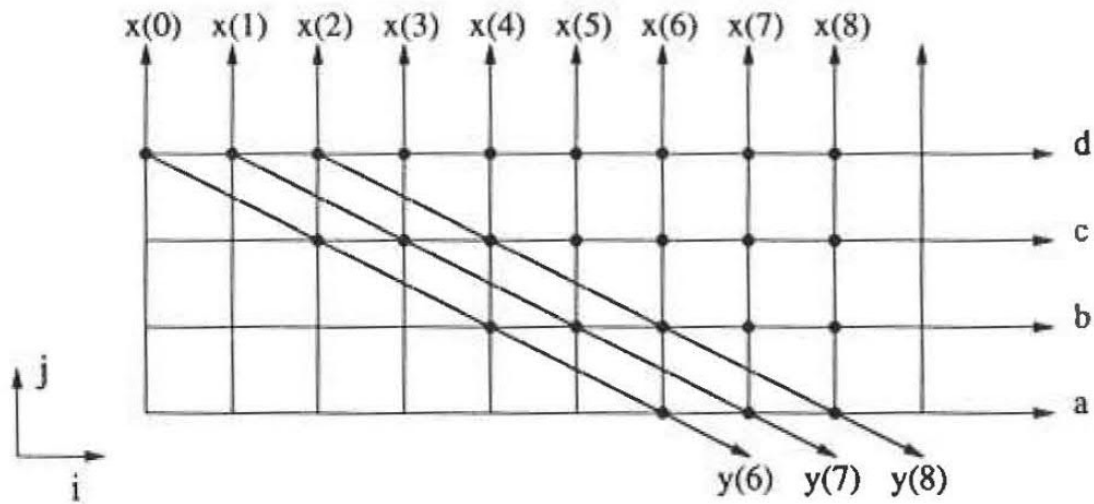
$$y(n) = ax(n) + bx(n - 2) + cx(n - 4) + dx(n - 6) \quad (7.33)$$

- Lập công thức mô tả RIA tương đương cho giải thuật này. Vẽ DG được rút gọn và viết các bất đẳng thức định thời biểu nếu phép toán nhân và cộng cần 2 t.u.. Giả sử rằng các hệ số nhập và trọng số có thể được truyền theo cách quảng bá (broadcast) và định thời biểu tuyến tính.
- Chọn vector chiều  $d^T = [1 \ 0]$ . Chọn vector không gian bộ xử lý p. Có được nghiệm của các bất đẳng thức định thì để kiến trúc thu tâ, được thiết kế cần số phần tử Delay ít nhất. Tính HUE của kiến trúc và vẽ kiến trúc thu tâm.

BG.

Từ DG trên ta tìm được các cạnh cơ bản của x, y và w như sau:

$$e_x = [0 \ 1]^T; \quad e_y = [2 \ -1]^T; \quad \text{và} \quad e_h = [1 \ 0]^T$$



Hình 7.35 DG

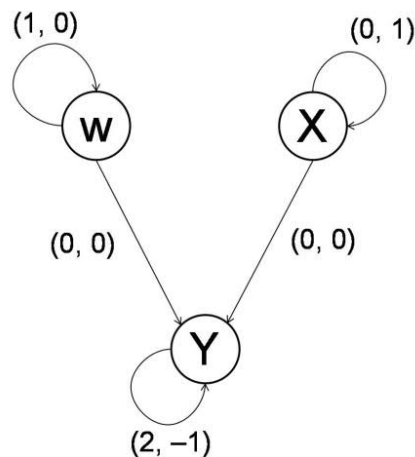
Theo hình vẽ DG ta suy ra được mô tả RIA cho các trọng số  $w$ , ngõ vào  $x$  và ngõ ra như sau:

$$W(i, j) = W(i-1, j)$$

$$X(i, j) = X(i, j-1)$$

$$Y(i, j) = Y(i-2, j+1) + W(i, j) X(i, j)$$

Và RDG:



Giả sử ta dùng định thời biểu tuyến tính, khi đó có các bất đẳng thức định thời biểu sau:

- $W \rightarrow W: [s1 \ s2] \times [1 \ 0]^T \geq 0$  hay  $s1 \geq 0$  (giả sử thời gian truyền thông = 0)
- $W \rightarrow Y: [s1 \ s2] \times [0 \ 0]^T \geq 0$  hay  $0 \geq 0$
- $X \rightarrow X: [s1 \ s2] \times [0 \ 1]^T \geq 0$  hay  $s2 \geq 0$
- $X \rightarrow Y: [s1 \ s2] \times [0 \ 0]^T \geq 0$  hay  $0 \geq 0$
- $Y \rightarrow Y: [s1 \ s2] \times [2 \ -1]^T \geq 2$  hay  $2s1 - s2 \geq 2$

Nếu chọn:

- $s2 = 0$  thì  $s1 = 1 \Rightarrow s = [1 \ 0]^T$
- $s2 = 1$  thì chọn  $s1 = 2 \Rightarrow s = [2 \ 1]^T$
- $s2 = 2$  thì chọn  $s1 = 2 \Rightarrow s = [2 \ 2]^T$

Để giảm số phần tử Delay trong kiến trúc thu tâm, ta chọn  $s = [1 \ 0]^T$

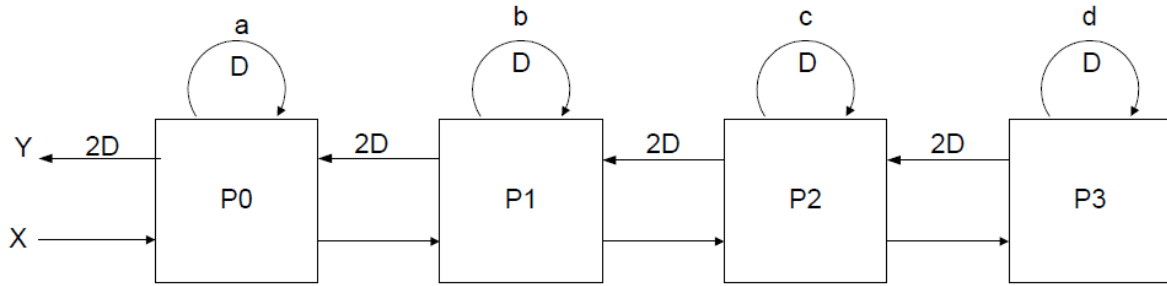
Từ  $d = [1 \ 0]^T \Rightarrow p = [0 \ 1]^T$  (do  $p^T d = 0$ )

Với  $p^T = [0 \ 1]$  và  $s^T = [1 \ 0]$ , ta có bảng sau:

$e$	$p^T e$	$s^T e$
$e_x = [0 \ 1]^T$	1	0
$e_y = [2 \ -1]^T$	-1	2
$e_h = [1 \ 0]^T$	0	1



Từ đó ta có kiến trúc tâm thu như sau:



6. Xét 1 bộ lọc có phương trình ngõ ra:  $y(n) = ax(n-1) + bx(n-3) + cx(n-5)$

- Hãy vẽ DG của bộ lọc này. Lập công thức mô tả RIA (chuẩn ngõ ra) tương đương cho giải thuật này. Từ đó suy ra các cạnh cơ bản  $e_x$ ,  $e_w$  và  $e_y$ ?
- Cho trước  $d = [1 \ 0]^T$ , tìm  $p$  và  $s$ .  
(Giả sử  $d$ ,  $p$  và  $s$  phải thỏa các điều kiện:  $p^T d = 0$ ;  $s^T d \neq 0$  và  $s^T e \geq 0$ )
- Vẽ mảng tâm thu cần thiết kế.

BG.

a) Ta có:

$$y(n) = ax(n-1) + bx(n-3) + cx(n-5)$$

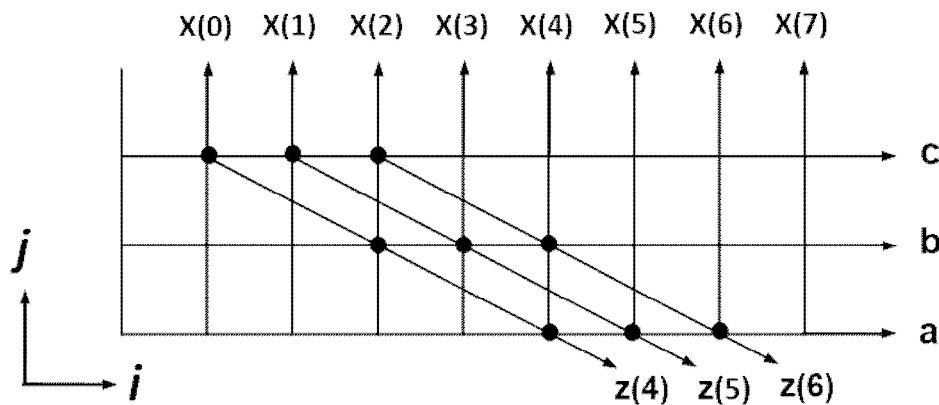
Nếu đặt  $y(n) = z(n-1)$ , khi đó:

$$z(n-1) = ax(n-1) + bx(n-3) + cx(n-5)$$

hay

$$z(n) = ax(n) + bx(n-2) + cx(n-4)$$

Suy ra DG:



Mô tả RIA (chuẩn ngõ ra): (gọi các trọng số  $a, b, c$  là  $w_0, w_1, w_2$ )

$$Z(i, j) = Z(i-2, j+1) + W(i, j) \cdot X(i, j) \Rightarrow (i, j) - (i-2, j+1) = (2, -1)$$

$$W(i, j) = W(i-1, j) \Rightarrow (i, j) - (i-1, j) = (1, 0)$$

$$X(i, j) = X(i, j-1) \Rightarrow (i, j) - (i, j-1) = (0, 1)$$

Suy ra

$$e_x = [0 \ 1]^T, \quad e_w = [1 \ 0]^T, \quad \text{và} \quad e_z = [2 \ -1]^T$$

b) Tìm  $p$  và  $s$  khi  $d = [1 \ 0]^T$

$$\bullet \text{ Điều kiện } p^T d = 0 \Rightarrow [p_1 \ p_2] [1 \ 0]^T = 0 \Rightarrow p_1 = 0.$$

$$\text{Nếu chọn } p_2 = 1 \Rightarrow p = [0 \ 1]^T$$

$\bullet$  Điều kiện của  $s$ :

$$\text{➤ } s^T d \neq 0 \Rightarrow [s_1 \ s_2] [1 \ 0]^T \neq 0 \Rightarrow s_1 \neq 0.$$

$$\text{➤ } s^T e \geq 0:$$

$E$	$s^T e \geq 0$
$e_x = [0 \ 1]^T$	$s_2 \geq 0$
$e_w = [1 \ 0]^T$	$s_1 \geq 0$
$e_z = [2 \ -1]^T$	$2s_1 - s_2 \geq 0$

Kết hợp các bất đẳng thức trên, ta chọn  $s_1 = 1 \Rightarrow 0 \leq s_2 \leq 2s_1$  hay  $0 \leq s_2 \leq 2$

Như vậy chọn  $s_2 = 1 \Rightarrow s = [1 \ 1]^T$ .

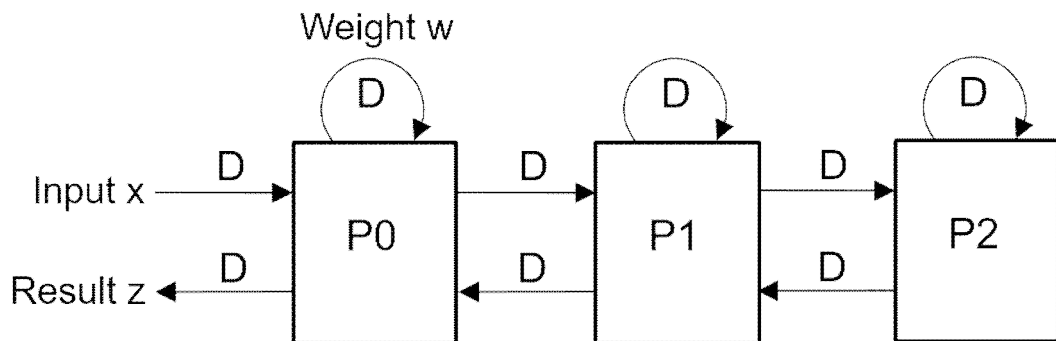
Kết luận:  $p = [0 \ 1]^T$  và  $s = [1 \ 1]^T$ .

c) Mạng tâm thu cần thiết kế:

Bảng ánh xạ cạnh với  $p^T = [0 \ 1]$  và  $s^T = [1 \ 1]$ :

$e$	$p^T e$	$s^T e$
$e_x = [0 \ 1]^T$	1	1
$e_w = [1 \ 0]^T$	0	1
$e_z = [2 \ -1]^T$	-1	1

Sơ đồ mạng tâm thu:



7. Thiết kế mạch lọc FIR có 3 nhánh [rẽ] (3-tap FIR filter) với hệ số là  $h_0 = 11$ ,  $h_1 = 14$ , và  $h_2 = 7$ .

- Với số học phân bố (distributed arithmetic), hãy cho biết cần dùng bảng tra cứu (ROM hoặc bảng LUT của FPGA) có kích thước bao nhiêu và nội dung của bảng này.
- Với cải tiến dùng mã CSD và vẽ sơ đồ khối cài đặt FIR này. Viết mã Verilog cho mạch này.

**ĐS.**

a) Trường hợp dữ liệu lớn nhất là tổng các hệ số  $h$  là:

$$11 + 14 + 7 = 32 \text{ và } 2^5 - 1 < 32 < 2^6 - 1$$

Suy ra bảng tra cứu phải có độ rộng dữ liệu là 6 và số đường địa chỉ là 3 (do có 3 tap).

Nếu gọi các bit địa chỉ là  $A_2 A_1 A_0$  thì nội dung tại ô nhớ là  $A_2 h_2 + A_1 h_1 + A_0 h_0$

Từ đó ta có bảng ghi ROM cho FIR có các hệ số  $\{11, 14, 7\}$  như sau:

Địa chỉ (Binary)	Dữ liệu (Decimal)
000	0
001	11
010	14
011	25
100	7
101	18
110	21
111	32

b) Đổi các giá trị hệ số sang dạng CSD:

$$7D = 0111 = 100-1 = 2^3 - 1 \Rightarrow 7 \times X = X \times (2^3 - 1) = X \ll 3 - X$$

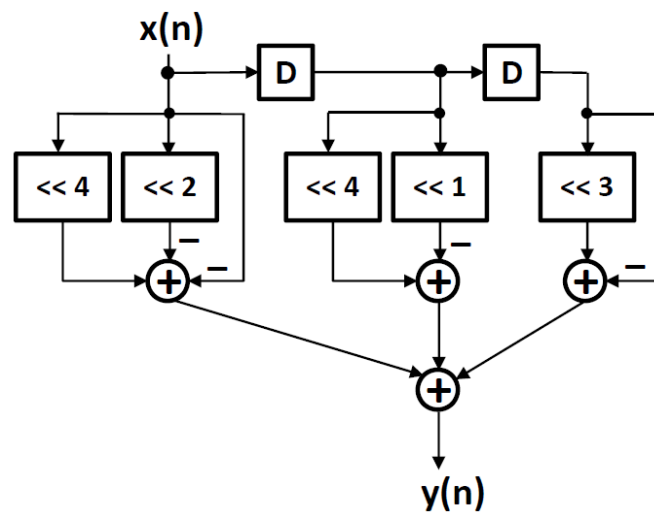
$$11D = 1011 = 110-1 = 10-10-1 = 2^4 - 2^2 - 1 \Rightarrow 11X = X(2^4 - 2^2 - 1) = X \ll 4 - X \ll 2 - X$$

$$14D = 1110 = 100-10 = 2^4 - 2^1 \Rightarrow 14X = X(2^4 - 2^1) = X \ll 4 - X \ll 1$$

Như vậy ta dùng các khối dịch trái và cộng/trừ để có được sơ đồ khối cài đặt cho bộ lọc này.

Với bộ lọc FIR:  $y(n) = 11x(n) + 14x(n-1) + 7x(n-2)$

Ta có sơ đồ khối sau:



8. Thiết kế mạch lọc FIR có 3 nhánh [rẽ] (3-tap FIR filter)

$$y(n) = h_0x(n) + h_1x(n-1) + h_2x(n-2)$$

với các hệ số là  $h_0 = 107$ ,  $h_1 = 105$ , và  $h_2 = 121$ .

- Với số học phân bố (distributed arithmetic), hãy cho biết cần dùng bảng tra cứu (ROM hoặc bảng LUT của FPGA) có kích thước bao nhiêu và nội dung của bảng này.
- Viết mã Verilog cho mạch lọc này. (Giả sử ngõ vào mỗi lần cung cấp 3 mẫu dữ liệu liên tiếp:  $x(n)$ ,  $x(n-1)$  và  $x(n-2)$ .)

**BG.**

a) Trường hợp dữ liệu lớn nhất là tổng các hệ số  $h$  là:

$$107 + 105 + 121 = 333 \text{ và } 2^8 - 1 (= 255) < 333 < 2^9 - 1 (= 511)$$

Suy ra bảng tra cứu phải có độ rộng dữ liệu là 9 và số đường địa chỉ là 3 (do có 3 tap).

Nếu gọi các bit địa chỉ là  $A_2A_1A_0$  thì nội dung tại ô nhớ là  $A_2h_2 + A_1h_1 + A_0h_0$

Như vậy ROM hoặc LUT có kích thước:  $2^3 \times 9$

Từ đó ta có bảng ghi ROM cho FIR có các hệ số  $\{109, 107, 111\}$  như sau:

Địa chỉ (Binary)	Dữ liệu (Decimal)
000	0
001	107
010	105
011	212
100	121
101	228
110	226
111	333

b) Mã Verilog cho hệ cần thiết kế:  $y(n) = h_0x(n) + h_1x(n-1) + h_2x(n-2)$

```
module(clk1, clk2, reset, xn, xn_1, xn_2, yn);
input clk1, clk2, reset;  input [7:0] xn, xn_1, xn_2;
output [15:0] yn; reg [15:0] yn ;  reg [7:0] dn, dn_1, dn_2;
reg [7:0] counter; reg [15:0] sum;
wire [5:0] data;
// có 9 clock clk2 giữa 2 cạnh lên của clk1
rom d1(clk2, {dn_2[0], dn_1[0], dn[0]}, data);
```

```
always@(posedge clk1)
```

```
begin
```

```
    dn <= xn;
```

```
    dn_1 <= xn_1;
```

```
    dn_2 <= xn_2;
```

```
end
```

```

always@(negdge clk2)
begin
    if (reset)
        begin
            counter <= 8;
            sum <= 0;
        end
    else
        if (counter == 0)
            yn <= sum;
        else
            begin
                counter <= counter - 1;
                sum <= sum + (data << counter);
            end
        end
end

always@(posdge clk2)
begin
    dn <= dn >> 1;
    dn_1 <= dn_1 >> 1;
    dn_2 <= dn_2 >> 1;
end
endmodule

module rom (en, addr, data); // tính A2h2 + A1h1 + A0h0
input en; input [2:0] addr;
output [8:0] data; reg [8:0] data;

always @(posedge en)
case(addr)
    3'b000: data = 0;
    3'b001: data = 107;
    3'b010: data = 105;
    3'b011: data = 212;
    3'b100: data = 121;
    3'b101: data = 228;
    3'b110: data = 226;
    3'b111: data = 333;
    default: data = 'bX;
endcase
endmodule

```

### 9. Thiết kế mạch lọc FIR có 3 nhánh [rẽ] (3-tap FIR filter)

$$y(n) = h_0x(n) + h_1x(n-1) + h_2x(n-2)$$

- Giả sử  $h_0 = 47$ ,  $h_1 = 55$ , và  $h_2 = 31$ , áp dụng mã CSD cho hệ này và vẽ sơ đồ khối tương ứng.
- Viết mã Verilog cho câu a). (Giả sử ngõ vào mỗi lần cung cấp 3 mẫu dữ liệu liên tiếp:  $x(n)$ ,  $x(n-1)$  và  $x(n-2)$ .)

**BG.**

**a) CSD:**

**Đổi các giá trị hệ số sang dạng CSD:**

$$\begin{aligned}
 47D &= 1\ 0\ 1\ 1\ 1\ 1 = 1\ 1\ 0\ 0\ 0\ -1 = 1\ 0\ -1\ 0\ 0\ 0\ -1 = 2^6 - 2^4 - 1 \\
 &\Rightarrow 55 \times X = X \times (2^6 - 2^4 - 1) = X \ll 6 - X \ll 4 - X
 \end{aligned}$$

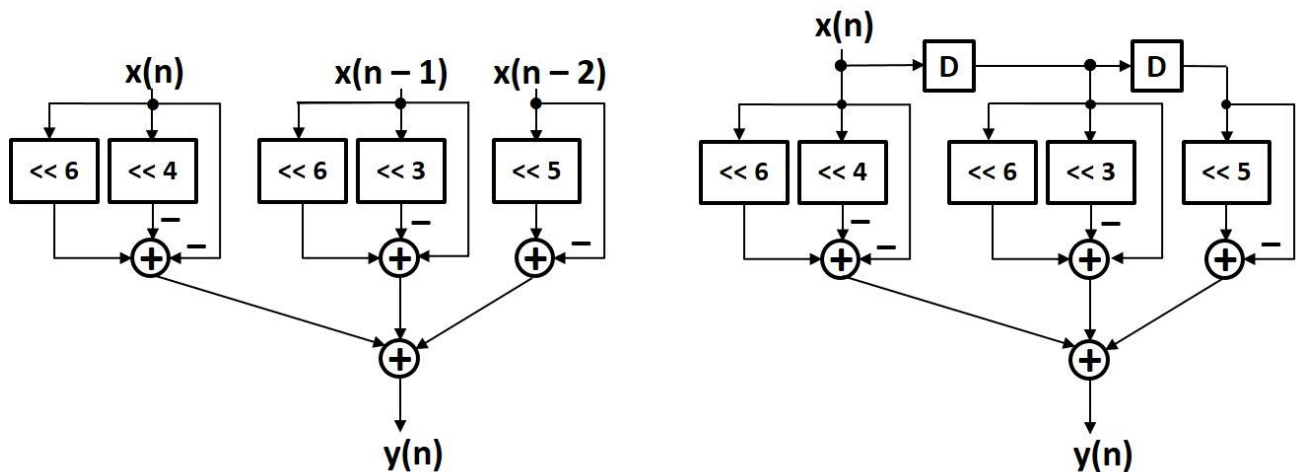
$$\begin{aligned}
 55D &= 1\ 1\ 0\ 1\ 1\ 1 = 1\ 1\ 1\ 0\ 0\ -1 = 1\ 0\ 0\ -1\ 0\ 0\ -1 = 2^6 - 2^3 - 1 \\
 &\Rightarrow 55X = X(2^6 - 2^3 - 1) = X \ll 6 - X \ll 3 - X
 \end{aligned}$$

$$31D = 1\ 1\ 1\ 1\ 1\ 1 = 1\ 0\ 0\ 0\ 0\ -1 = 2^5 - 1 \Rightarrow 31X = X(2^5 - 1) = X \ll 5 - X$$

Như vậy ta dùng các khối dịch trái và cộng/trừ để có được sơ đồ khối cài đặt cho bộ lọc này.

Với bộ lọc FIR:  $y(n) = 47x(n) + 55x(n-1) + 31x(n-2)$

Ta có sơ đồ khối sau:



b) Mã Verilog cho câu a): (giả sử hằng số và dữ liệu là số 8 bit)

```
module filter(clk, xn, xn_1, xn_2, yn);
input clk;
input [7:0] xn, xn_1, xn_2;
output [17:0] yn;  reg [17:0] yn;
reg [7:0] dn, dn_1, dn_2;
reg [15:0] d0, d1, d2;

always@(posedge clk)
begin
    dn <= xn;
    dn_1 <= xn_1;
    dn_2 <= xn_2;
end

always@(dn or dn_1 or dn_2)
begin
    // d0 <= dn << 6 - dn << 4 - dn;
    d0 <= {dn, 6'b000000} - {dn, 4'b0000} - dn;
    // d1 <= dn_1 << 6 - dn_1 << 3 - dn_1;
    d1 <= {dn_1, 6'b000000} - {dn_1, 3'b000} - dn_1;
    // d2 <= dn_2 << 5 - dn_2;
    d2 <= {dn_2, 5'b000000} - dn_2;
    yn <= d0 + d1 + d2;
end
endmodule
```

10. Hãy viết mã Verilog cho các bài toán khác: (xem thêm các mã Verilog trong “Case studies”)

- bộ lọc FIR dạng trực tiếp, chuyển vị, pipeline, song song, ...
- bộ lọc IIR dạng trực tiếp, tái định thì, ...
- hệ thống unfolding
- hệ thống folding
- cài đặt bảng cấp phát thanh ghi thuận nghịch
- kiến trúc tâm thu
- ...