EECS4210 Architecture & Hardware for DSP

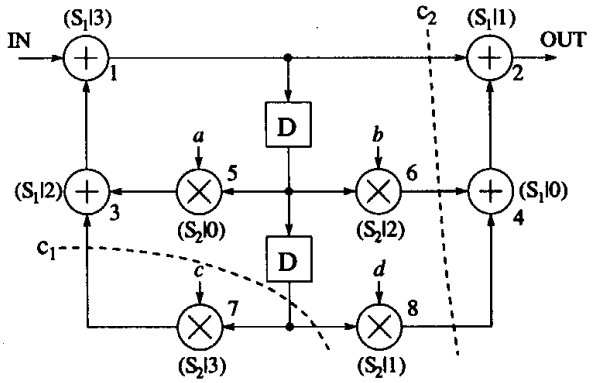
Chapter 8

Activities

Instructor: Prof. Peter Lian
Department of Electrical Engineering & Computer Science
Lassonde School of Engineering
York University

Activity 1

Given the biquad filter below, (1) find folding equations, (2) can it be folded? If not, retiming it.

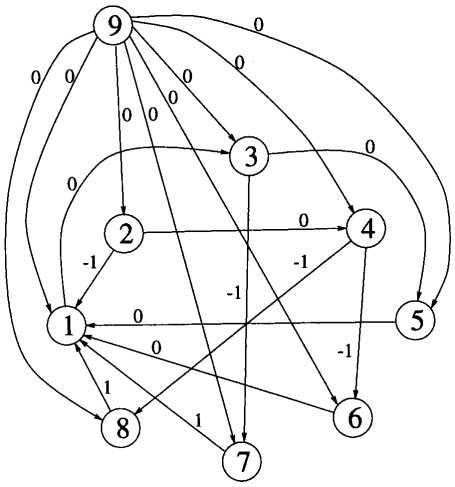


Activity 1 Solution

$$D_F(U \xrightarrow{e} V) = Nw(e) - P_u + v - u \qquad r(U) - r(V) \leq \left\lfloor \frac{D_F(U \xrightarrow{e} V)}{N} \right\rfloor$$

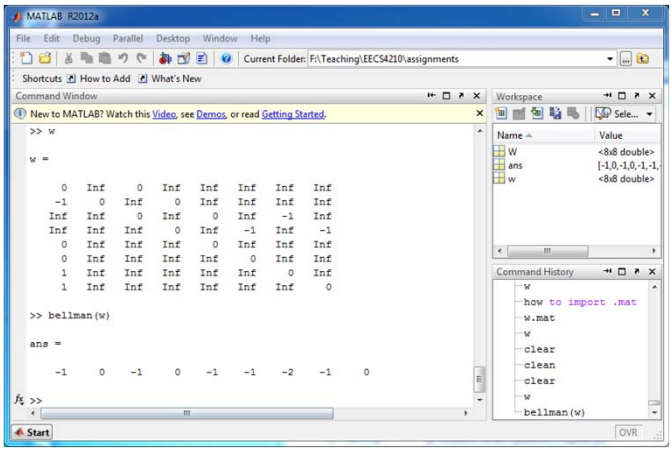
Edge	Folding Equation	Retiming for Folding Constraint
$1 \rightarrow 2$	$D_F(1 \rightarrow 2) = -3$	$r(1) - r(2) \leq -1$
$1 \rightarrow 5$	$D_F(1 \rightarrow 5) = 0$	$r(1) - r(5) \leq 0$
$1 \rightarrow 6$	$D_F(1 \rightarrow 6) = 2$	$r(1) - r(6) \leq 0$
$1 \rightarrow 7$	$D_F(1 \rightarrow 7) = 7$	$r(1) - r(7) \leq 1$
$1 \rightarrow 8$	$D_F(1 \rightarrow 8) = 5$	$r(1) - r(8) \leq 1$
$3 \rightarrow 1$	$D_F(3 \rightarrow 1) = 0$	$r(3) - r(1) \leq 0$
$4 \rightarrow 2$	$D_F(4 \rightarrow 2) = 0$	$r(4) - r(2) \leq 0$
$5 \rightarrow 3$	$D_F(5 \rightarrow 3) = 0$	$r(5) - r(3) \leq 0$
$6 \rightarrow 4$	$D_F(6 \rightarrow 4) = -4$	$r(6) - r(4) \leq -1$
$7 \rightarrow 3$	$D_F(7 \rightarrow 3) = -3$	$r(7) - r(3) \leq -1$
$8 \rightarrow 4$	$D_F(8 \rightarrow 4) = -3$	$r(8) - r(4) \leq -1$

Activity 1 Solution



Activity 1 Solution

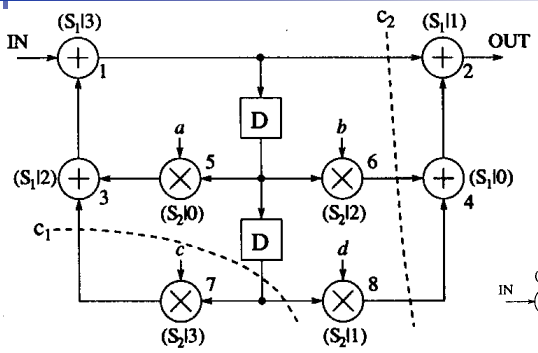
From constraint graph (or inequality equations), we can find the weight matrix. Then we can use Bellman-Ford Algorithm to find the solution, i.e. r(1)=-1, r(2)=0, r(3)=-1, r(4)=0, r(5)=-1, r(6)=-1, r(7)=-2, r(8)=-1, as shown below.



Activity 1 Solution

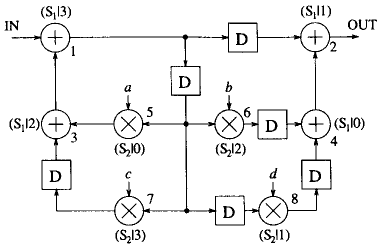
- One solution is found from the constraint graph using Bellman-Ford Algorithm
- r(1)=-1 r(2)=0 r(3)=-1 r(4)=0
- r(5)=-1 r(6)=-1 r(7)=-2 r(8)=-1

Activity 1 Solution



r(1)=-1 r(2)=0 r(3)=-1 r(4)=0
r(5)=-1 r(6)=-1 r(7)=-2 r(8)=-1
Apply retiming property IV (add 1 to each retiming value, we have:

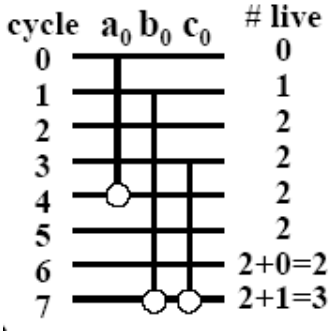
r(1)=0 r(2)=1 r(3)=0 r(4)=1
r(5)=0 r(6)=0 r(7)=-1 r(8)=0



Retimed DFG

Activity 2

Given the linear lifetime chart below, (1) derive the data allocation using forward-backward register allocation; (2) synthesis the architecture.



Activity 2 Solution

cycle	input	R1	R2	R3	output
0	a				
1	b	a			
2		b	a		
3			b	a	
4	c			b	
5		c			
6			c		
7				ⓒ	c

cycle	input	R1	R2	R3	output
0	a				
1	b	a			
2		b	a		
3			b	a	
4	c		ⓐ	b	a
5		c	b		
6			c	b	
7			ⓑ	ⓒ	b, c