

# Computer Systems II

Li Lu

Room 605, CaoGuangbiao Building

[li.lu@zju.edu.cn](mailto:li.lu@zju.edu.cn)

<https://person.zju.edu.cn/lynnluli>



# Scheduling of Nonlinear pipelining



# Linear vs. Nonlinear Pipelining

**Linear pipelining:** Each section of the pipelining is connected serially without feedback loop. When data passes through each segment in the pipelining, each segment can only flow once at most.

What about nonlinear pipelining?



# Linear vs. Nonlinear Pipelining

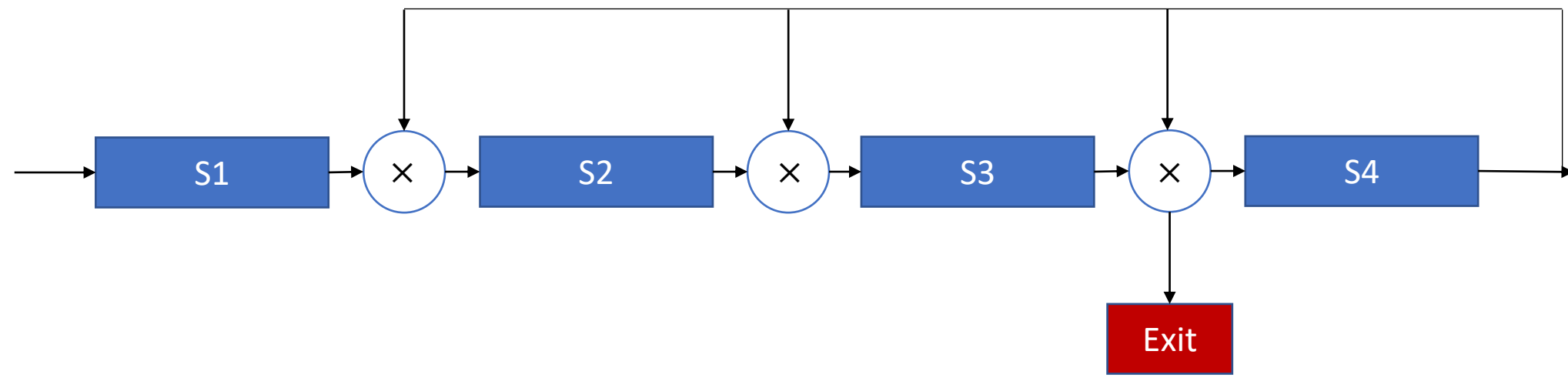
**Nonlinear pipelining:** In addition to the serial connection, there is also a feedback loop in the pipelining.

Scheduling problem of nonlinear pipelining.

Determine when to introduce a new task to the pipelining, so that the task will not conflict with the task previously entering the pipelining.

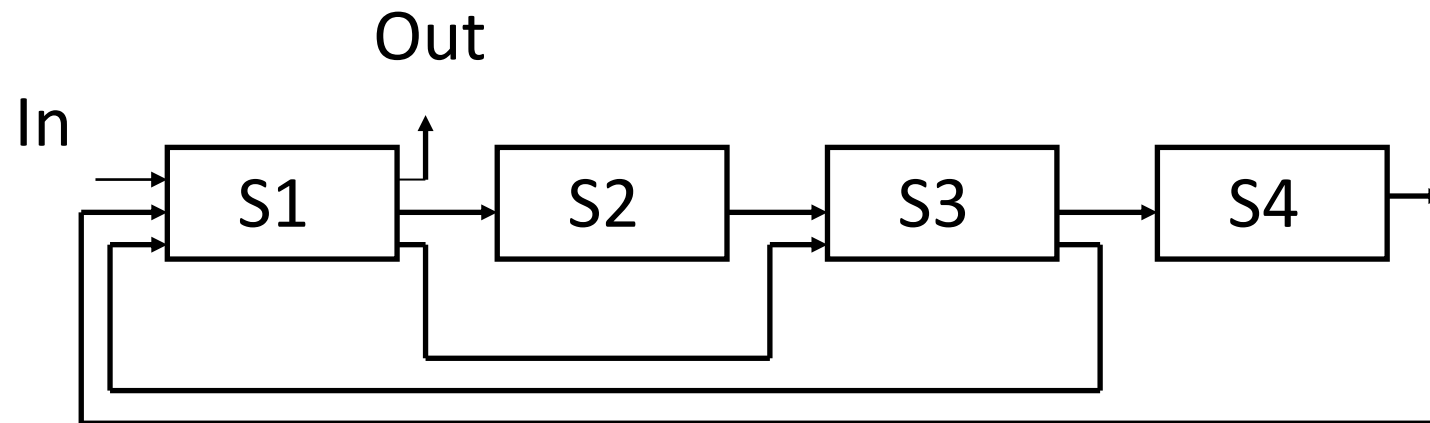


# Nonlinear pipelining



Task:  $\rightarrow S_1 \rightarrow S_2 \rightarrow S_3 \rightarrow S_4 \rightarrow S_2 \rightarrow S_3 \rightarrow S_4 \rightarrow S_3 \rightarrow$





Reservation Table:

	1	2	3	4	5	6	7
S1	✓			✓			✓
S2		✓			✓		
S3		✓				✓	
S4			✓				



# Extending Clock Cycles

Reservation Table:

	1	2	3	4	5	6	7	8	9	10	11	...
S1	1			1			1					
S2		1			1							
S3		1				1						...
S4			1									



# Following Instructions Scheduling

Reservation Table:

	1	2	3	4	5	6	7	8	9	10	11	...
S1	1	2		1	2		1	2				
S2		1	2		1	2						
S3		1	2			1	2					...
S4			1	2								





# Following Instructions Scheduling

Reservation Table:

	1	2	3	4	5	6	7	8	9	10	11	...
S1	1	2	3	1	2	3	1	2	3			
S2		1	2	3	1	2	3					
S3		1	2	3		1	2	3				...
S4			1	2	3							



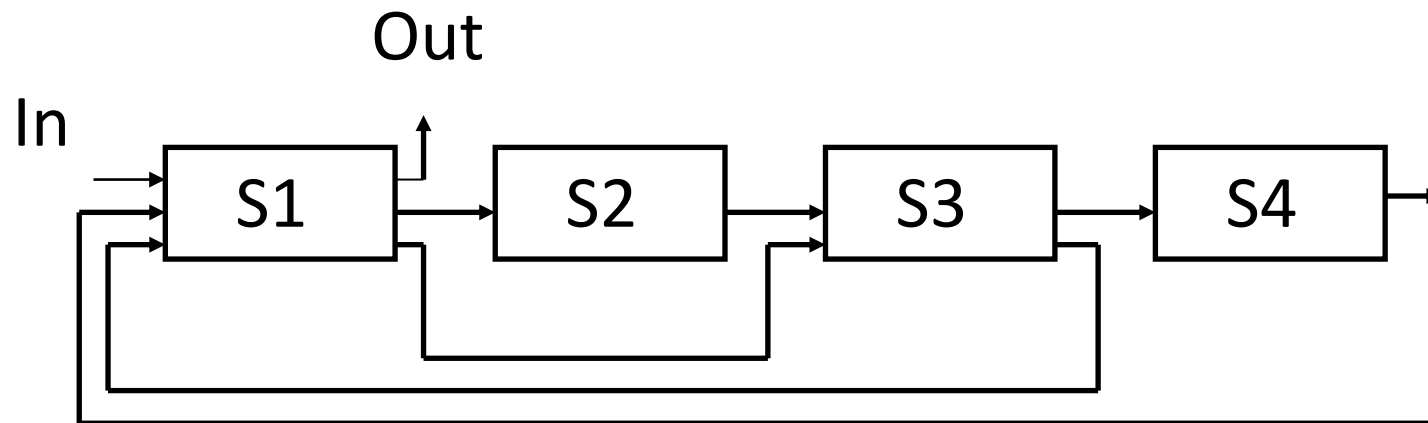
# Following Instructions Scheduling

Reservation Table:

	1	2	3	4	5	6	7	8	9	10	11	...
S1	1	2	3	14	25	3	14	2 5	3	4	5	
S2		1	2	3	14	25	3	4	5			
S3		1	2	3	4	15	2	3	4	5		...
S4			1	2	3	4	5					

Hazard! Any other scheduling?





Reservation Table:

	1	2	3	4	5	6	7	8	9	10	11	...
S1	1		2	1		2	1		2			
S2		1		2	1		2					
S3		1		2		1		2				...
S4			1		2							



# Another Scheduling

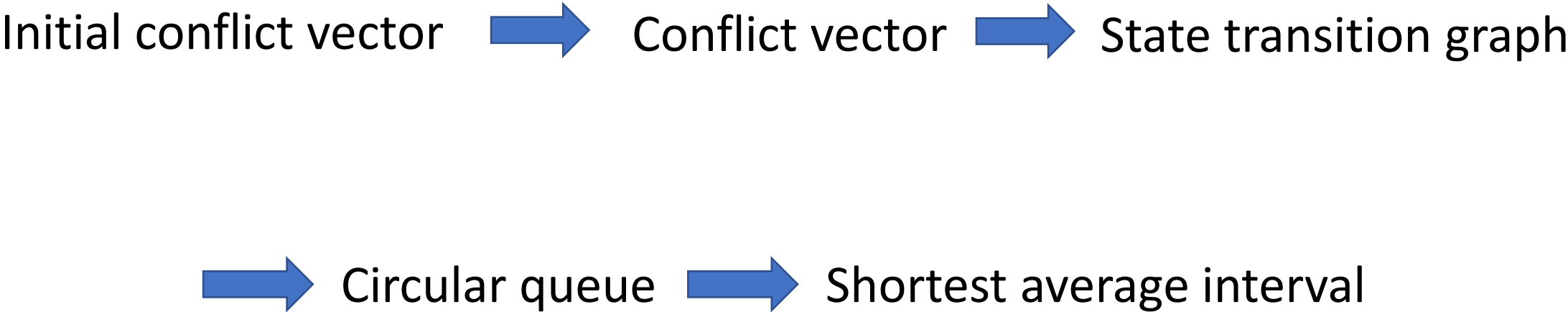
Reservation Table:

	1	2	3	4	5	6	7	8	9	10	11	...
S1	1		2	1	3	2	1 4	3	2 5	4	3 6	
S2		1		2	1	3	2	4	3	5	4	
S3		1		2		1 3		2 4		3 5		...
S4			1		2		3		4		5	

Still hazard! Need scheduling



# Schedule of Nonlinear pipelining without hazards



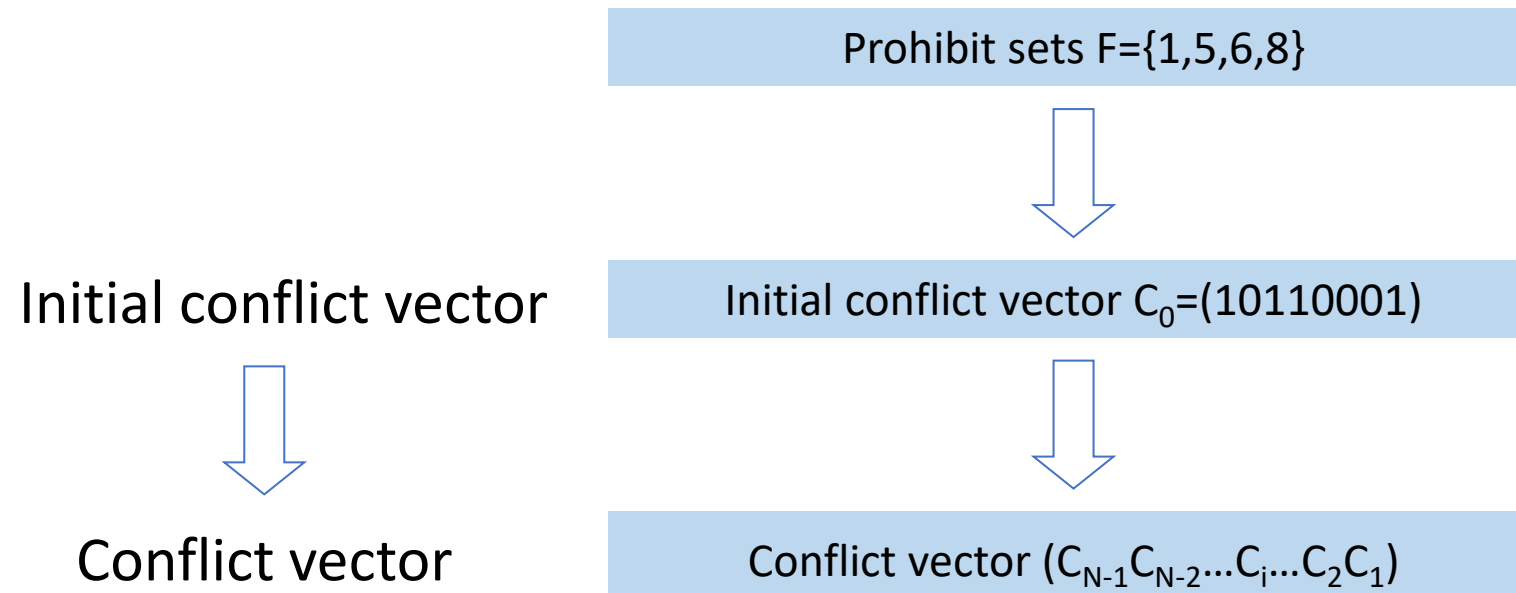
# Initial conflict vector

Reservation table for a 5-stage non-linear pipeline

	n									
k		1	2	3	4	5	6	7	8	9
	1	✓								✓
	2		✓	✓					✓	
	3				✓					
	4					✓	✓			
	5							✓	✓	



# Initial conflict vector



# Conflict vector

- Initial conflict vector  $C_0=(10110001)$  **CCV=Current Conflict vector**

Interval	Initial	
<b>CCV</b>	<b>10110001</b>	
<b>1→</b>	<b>10110001</b>	





# Conflict vector

- Initial conflict vector  $C_0=(10110001)$  **CCV=Current Conflict vector**

Interval	Initial	2
<b>CCV</b>	1011000 <u>1</u>	<b>10111101</b>
1→	10110001	00101100
2→		10110001



# Conflict vector

- Initial conflict vector  $C_0=(10110001)$  **CCV=Current Conflict vector**

Interval	Initial	2	2
<b>CCV</b>	1011000 <u>1</u>	101111 <u>0</u> 1	<b>10111111</b>
1→	10110001	00101100	00001011
2→		10110001	00101100
3→			10110001



# Conflict vector

Any other scheduling?

- Initial conflict vector  $C_0=(10110001)$  CCV=Current Conflict vector

Interval	Initial	2	2	7
CCV	1011000 <u>1</u>	101111 <u>0</u> 1	1 <u>0</u> 111111	10110001
1→	10110001	00101100	00001011	00000000
2→		10110001	00101100	00000000
3→			10110001	00000001
4→				10110001



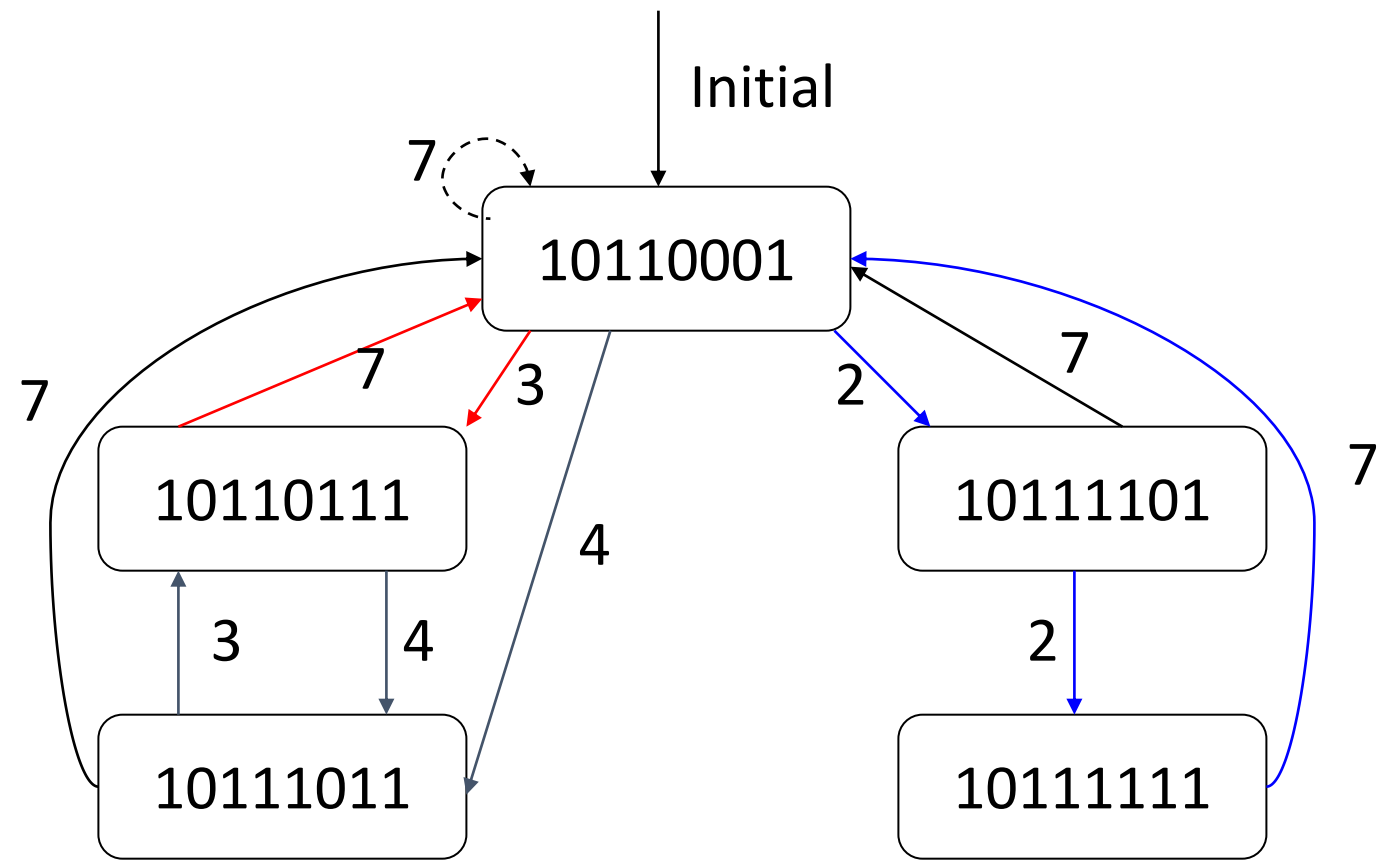
# Conflict vector

- Initial conflict vector  $C_0=(10110001)$  **CCV=Current Conflict vector**

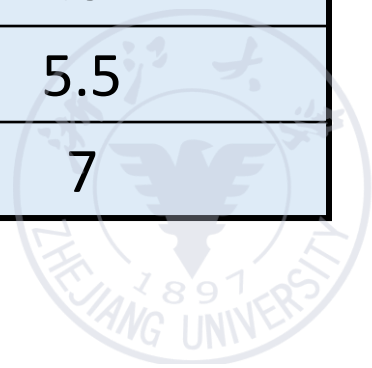
Interval	Initial	2	7
<b>CCV</b>	<b>1011000<u>1</u></b>	<b>1<u>0</u>111101</b>	<b>10110001</b>
1→	10110001	00101100	00000000
2→		10110001	00000000
3→			10110001



# State transition graph

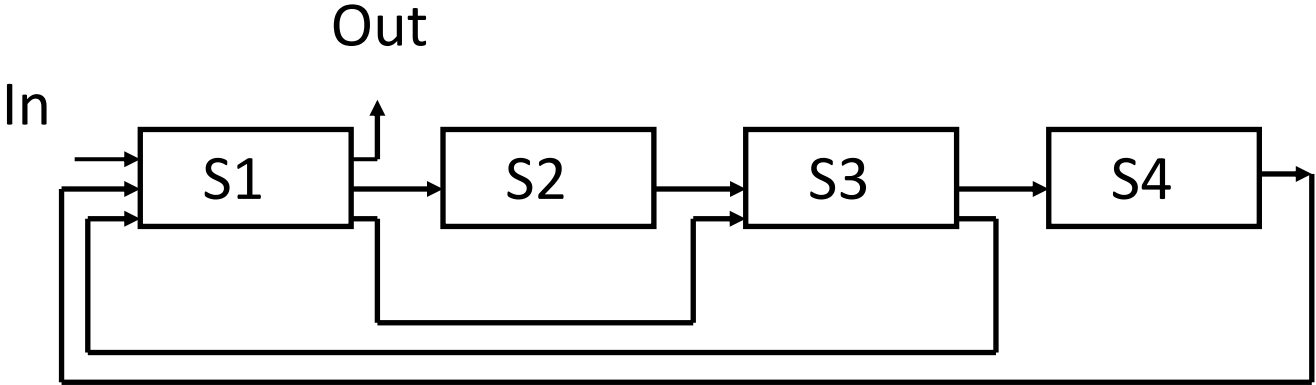


Circular queue	Shortest average interval
2,2,7	3.67
2,7	4.5
3,4	3.5
4,3	3.5
3,4,7	4.67
3,7	5
4,3,7	4.67
4,7	5.5
7	7



# Question?

How to determine the scheduling of the previous pipeline?



	1	2	3	4	5	6	7
S1	✓			✓			✓
S2		✓			✓		
S3		✓				✓	
S4			✓				

