Computer Organization and Software Systems

Webinar

Job Sequencing and Collision Prevention

BITS Pilani
Pilani Campus

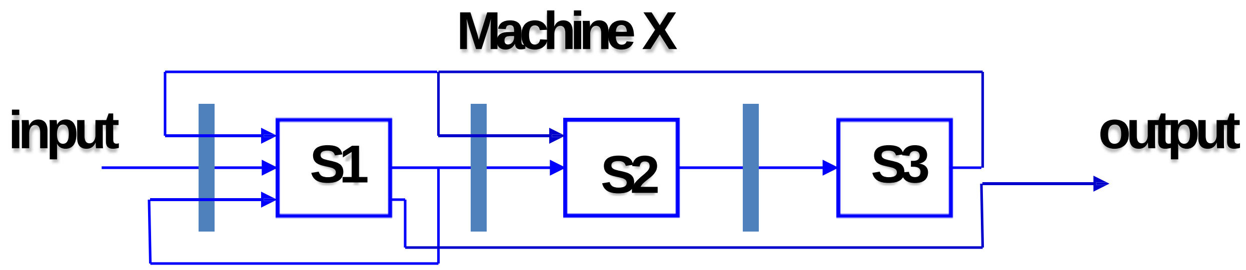
Dr. Lucy J. Gudino
Shamanth N.

Pipeline

- Linear vs Nonlinear pipelines:
  Linear pipeline
  - A linear pipeline processor is a cascade of Processing Stages which are linearly connected to perform fixed function over a stream of data flowing from one end to the other.
  - Linear pipeline are static pipeline because they are used to perform fixed functions

  Nonlinear pipeline
  - Non-Linear pipeline are dynamic pipeline because they can be reconfigured to perform variable functions at different times.
  - Non-Linear pipeline allows feed-forward and feedback connections in addition to the streamline connection
- Reservation Table - The utilization pattern of successive stages in a pipeline is specified by a Reservation Table

Reservation Table



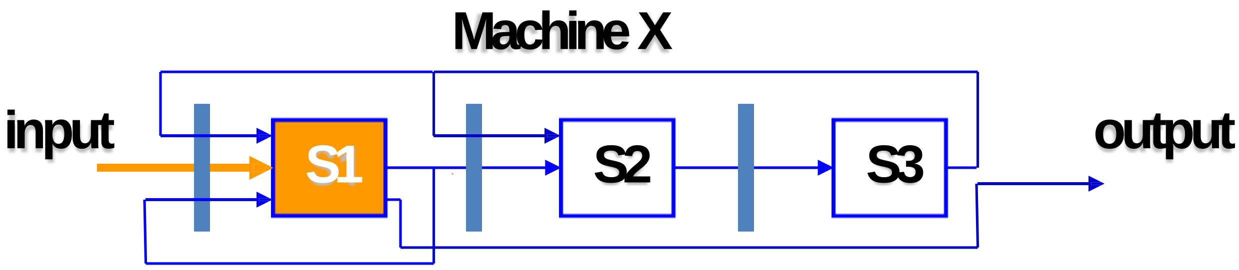
Reservation Table

Time →

Stage ↓

	0	1	2	3	4	5	6	7
S1	X	X					X	X
S2			X		X			
S3				X		X		

Reservation Table



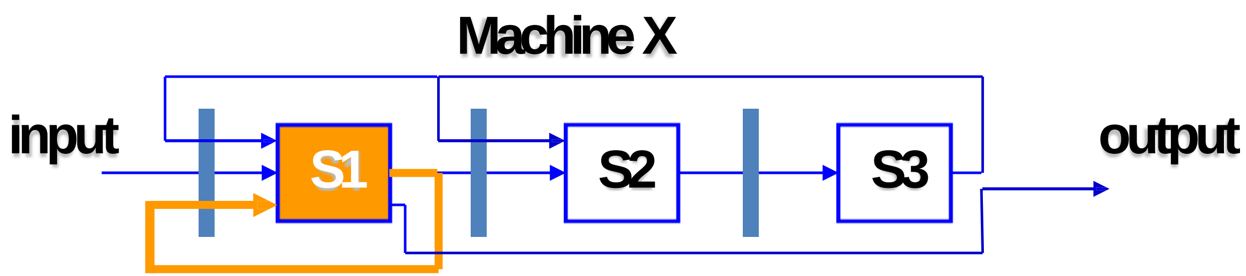
Reservation Table

Time →

Stage ↓

	0	1	2	3	4	5	6	7
S1	X	X					X	X
S2			X		X			
S3				X		X		

Reservation Table



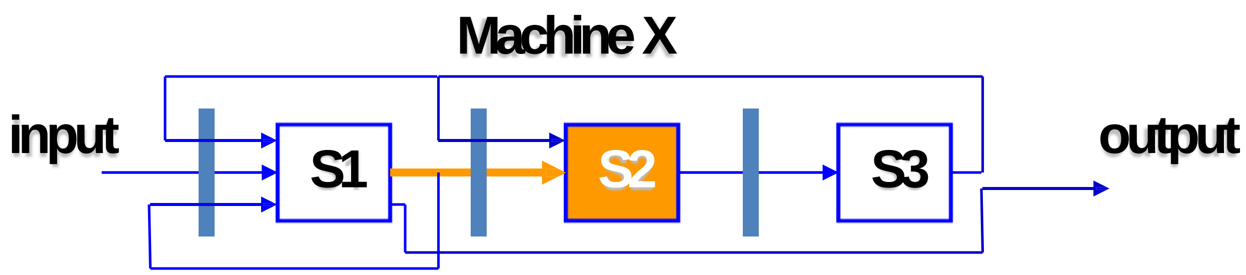
Reservation Table

Time →

Stage ↓

	0	1	2	3	4	5	6	7
S1	X	X					X	X
S2			X		X			
S3				X		X		

Reservation Table



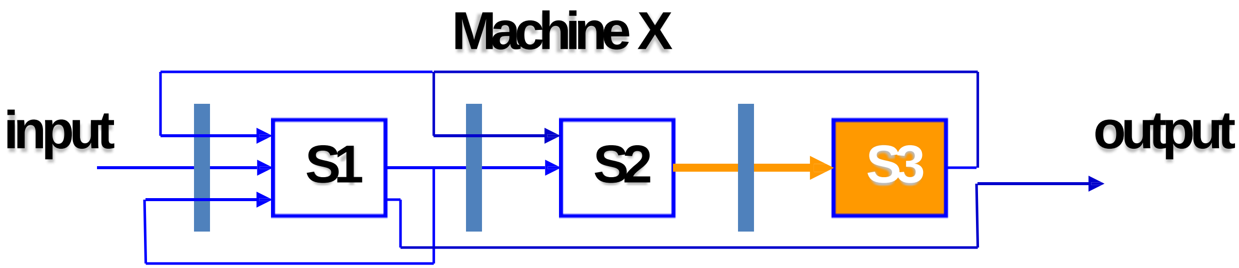
Reservation Table

Time →

Stage ↓

	0	1	2	3	4	5	6	7
S1	X	X					X	X
S2			X		X			
S3				X		X		

Reservation Table



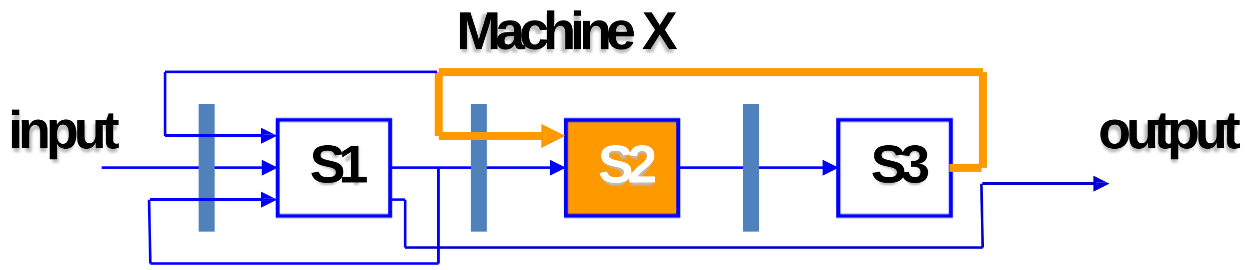
Reservation Table

Time →

Stage ↓

	0	1	2	3	4	5	6	7
S1	X	X					X	X
S2			X		X			
S3				X		X		

Reservation Table



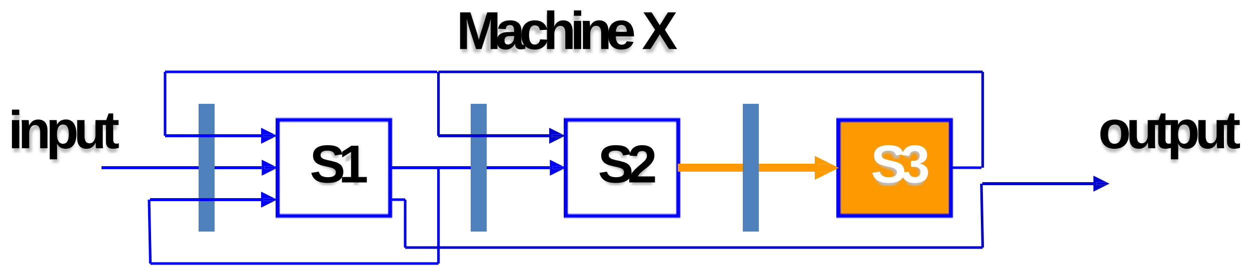
Reservation Table

Time →

Stage ↓

	0	1	2	3	4	5	6	7
S1	**X**	**X**					X	X
S2			**X**		**X**			
S3				**X**		X		

Reservation Table



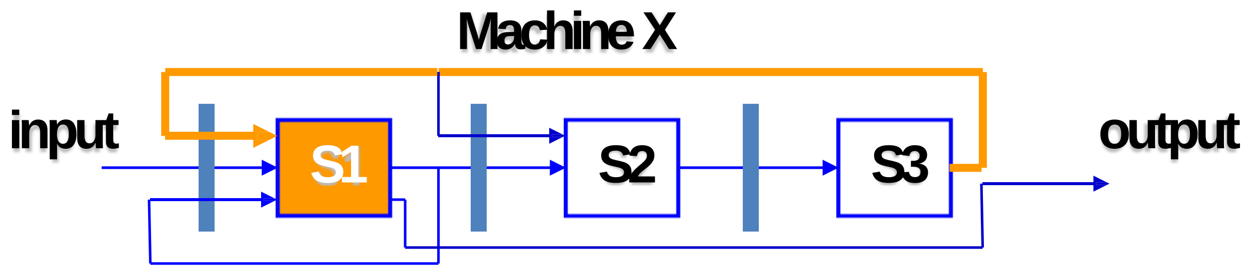
Reservation Table

Time →

Stage ↓

	0	1	2	3	4	5	6	7
S1	**X**	**X**					X	X
S2			**X**		**X**			
S3				**X**		**X**		

Reservation Table



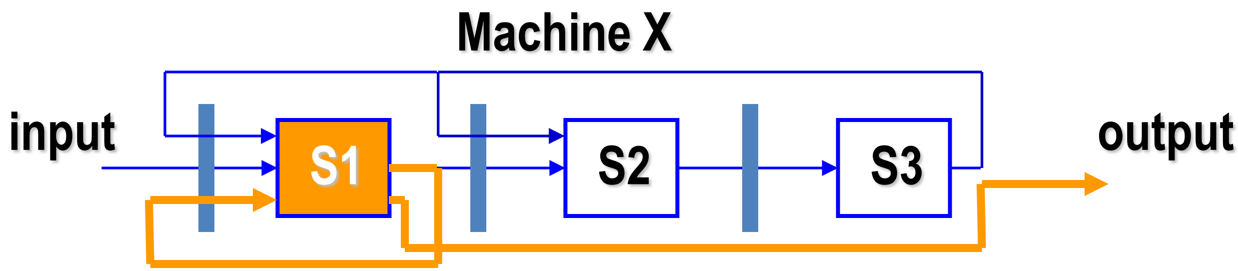
Reservation Table

Time →

Stage ↓

	0	1	2	3	4	5	6	7
S1	**X**	**X**					X	X
S2			**X**		**X**			
S3				**X**		**X**		

Reservation Table



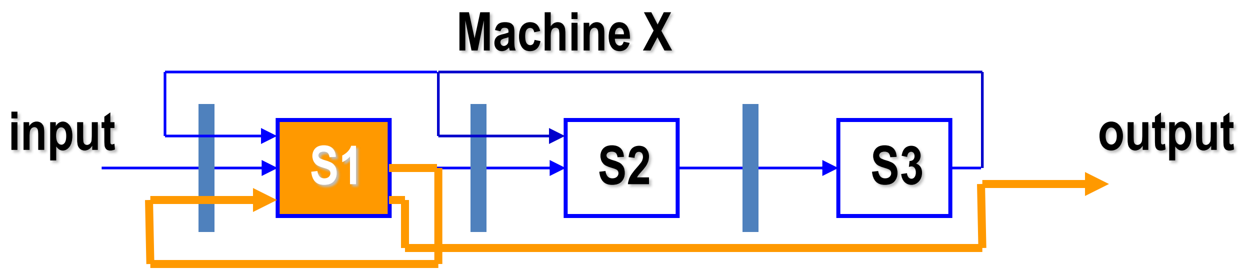
Reservation Table

Time →

Stage ↓

	0	1	2	3	4	5	6	7
S1	X	X					X	X
S2			X		X			
S3				X		X		

Reservation Table



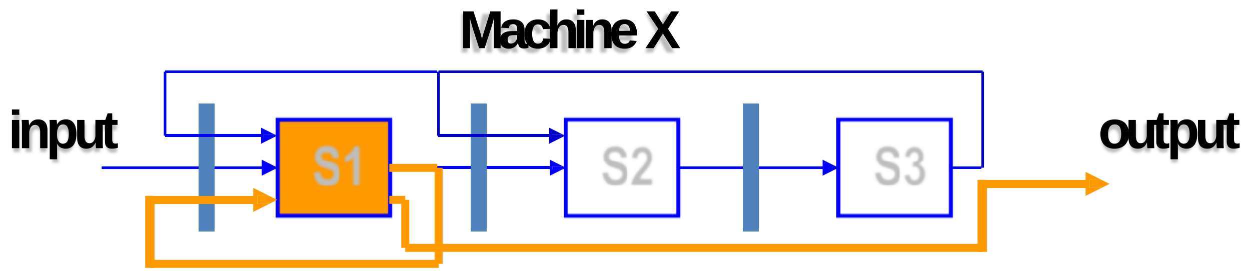
Reservation Table

Time →

Stage ↓

	0	1	2	3	4	5	6	7
S1	X	X					X	X
S2			X		X			
S3				X		X		

Reservation Table



Reservation Table

Time →

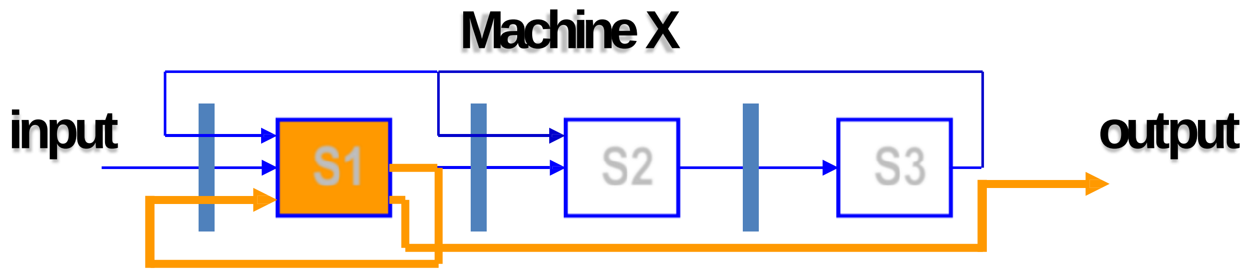
Stage ↓

	0	1	2	3	4	5	6	7
S1	X	X					X	X
S2			X		X			
S3				X		X		

Stage ↓

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
X	X					X	X												
		X		X															
			X		X														

Reservation Table



Reservation Table

Time →

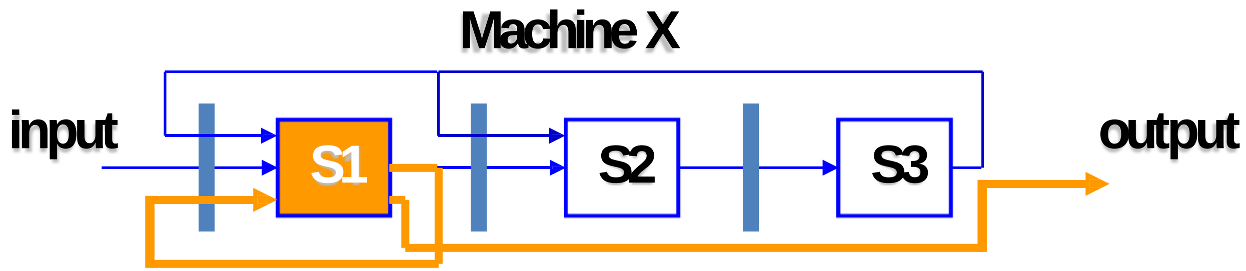
Stage ↓

	0	1	2	3	4	5	6	7
S1	X	X					X	X
S2			X		X			
S3				X		X		

Stage ↓

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
X	X					X	X												
		X		X															
			X		X														

Reservation Table



Reservation Table

Time →

Stage ↓

	0	1	2	3	4	5	6	7
S1	X	X					X	X
S2			X		X			
S3				X		X		

Stage ↓

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
S1	X	X					X	X	Y	Y					Y	Y				
S2			X		X						Y		Y							
S3				X		X						Y		Y						

Job Sequencing and Collision Prevention

BITS Pilani
Pilani Campus

Terms and definitions

- Collision: when two or more initiations attempt to use the same stage at the same time
- Initiation : Launching of an operation
- Latency: The number of cycles that elapse between two initiations or number of time units between two initiations
- Latency Sequence: The sequence of latencies between successive initiations
- Latency cycle: A latency sequence that repeats itself
- Procedure to choose a latency sequence is called Control Strategy

Steps

1. Find forbidden set of latencies
2. Prepare collision vector
3. Construct the state diagram
4. Find the list of simple cycles and greedy cycles
5. Find, the minimum average latency

1. Forbidden set of latencies

- Forbidden set F = {L1, L2,.....Lr}, contains all possible latencies that cause collision between two initiations

1. Forbidden set of latencies

	0	1	2	3	4	5	6	7	8	Latency
S1	X								X	8
S2		X	X					X		1,6,5
S3				X						-
S4					X	X				1
S5							X	X		1

8-0

2-1,7-1,7-2

5-4

7-6

Forbidden Set of latencies

F ={1, 5, 6, 8}

Forbidden set of latencies (contd..)

	0	1	2	3	4	5	6	7	8	8	10	11	12	13
S1	X								x					
S2		x	x					x						
S3				X										
S4					x	X								
s5							x	X						

3. Collision Vector

- Is a binary vector
- $C = (C_n, \ldots\ldots C_2, C_1)$ where $C_i = 1$ if $i \in F$

	0	1	2	3	4	5	6	7	8	Latency
S1	X								X	8
S2		X	X					X		1,6,5
S3				X						-
S4					X	X				1
S5							X	X		1

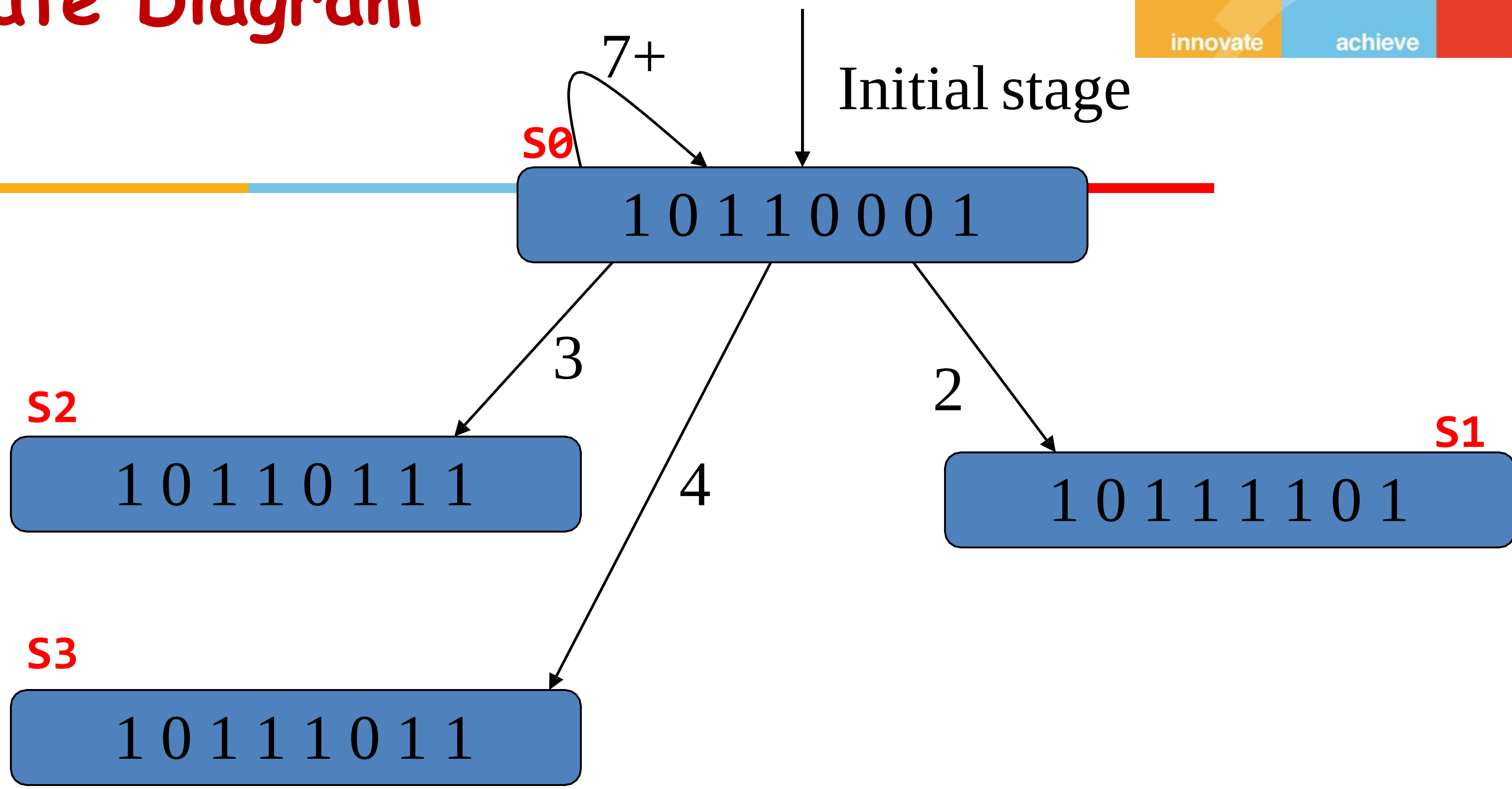
F ={1,5,6, 8}

C = (C8 C7 C6 C5 C4 C3 C2 C1)

= (1 0 1 1 0 0 0 1)

= Initial Vector

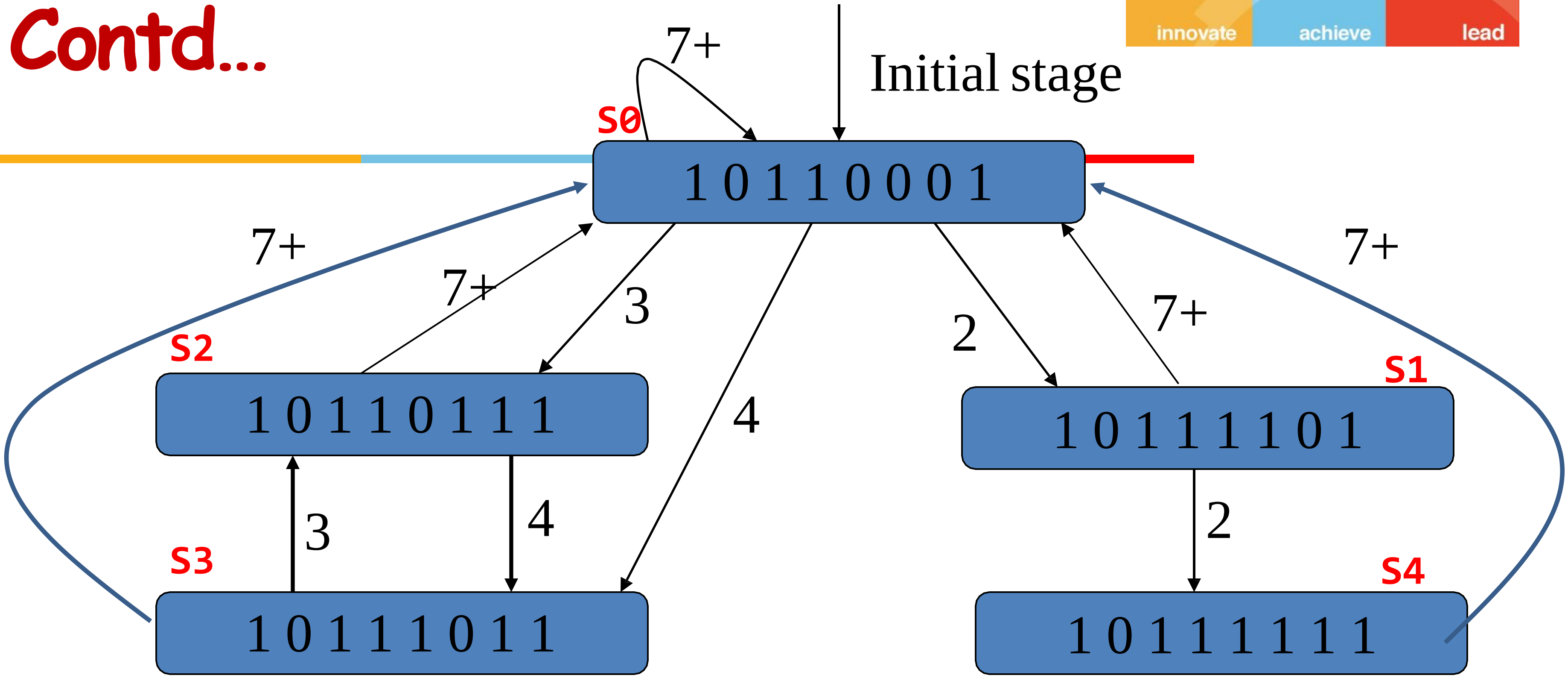
3. State Diagram



Shift right the bits of initial Vector, If shift out bit is zero, do bitwise OR of shift right value with initial vector, else do nothing

```
t0: 1011 0001       : Initial vector  : S0
t1: 0101 1000 ──→ 1 : Nothing to do
t2: 0010 1100 ──→ 0 : 0010 1100 + 1011 0001 => 1011 1101 : S1
t3: 0001 0110 ──→ 0 : 0001 0110 + 1011 0001 => 1011 0111 : S2
t4: 0000 1011 ──→ 0 : 0000 1011 + 1011 0001 => 1011 1011 : S3
t5: 0000 0101 ──→ 1 : Nothing to do
t6: 0000 0010 ──→ 1 : Nothing to do
t7: 0000 0001 ──→ 0 : 0000 0001 + 1011 0001 => 1011 0001 : S0
```

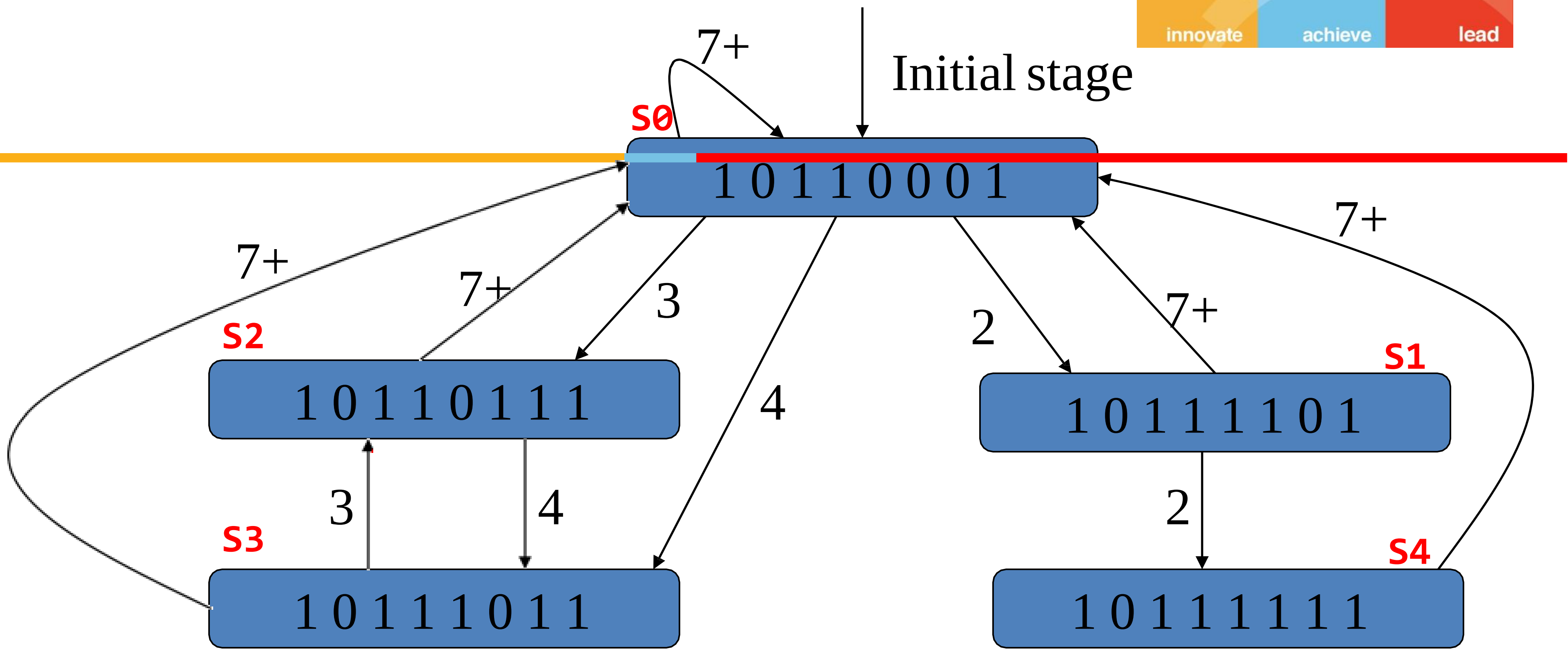
Contd…



Similar to shifting of bits of state S0, Vectors of S1, S2, S3 and S4 is also done.

Important Terms

- **Average latency** of a latency cycle is obtained by dividing the sum of all latencies by the number of latencies along the cycle
- **Constant cycle**: is a latency cycle which contains only one latency value
- **Simple Cycle**: is a latency cycle in which each state appears only once
- **Greedy Cycle**: whose edges are all made with minimum latencies from their respective starting states
- **Minimum average latency(MAL)**
- **Throughput** = inverse of MAL



Simple Cycle

S0:

(2, 7) : Avg. 4.5

(2,2,7): Avg. 3.6

(4,3,7): Avg. 4.6

(3,7) : Avg. 5

(7) : Avg. 7

(4,7): Avg. 5.5

S2:

(4,3) : Avg. 3.5

S3:

(3,4) : Avg. 3.5

Constant Cycle(Cycle with only one edge) : 7

Greedy Cycle:

(Whose edges are all made with minimum latency from their respective starting states)

(2,2,7)	: Avg. 3.6
(2,7,2)	: Avg. 3.6
(4,3)	: Avg. 3.5
(3,4)	: Avg. 3.5

Final Greedy Cycle :

(2,2,7)	: Avg. 3.6
(3,4)	**: Avg. 3.5**

Final Greedy Cycle :

(2,2,7) : Avg. 3.6

(3,4) : Avg. 3.5

Next task can be schedule from time = 3 or time 4

	0	1	2	3	4	5	6	7	8	9	10	11	12	13
S1	X								x					
S2		x	x					x						
S3				X										
S4					x	X								
s5							x	X						

	0	1	2	3	4	5	6	7	8
S1	X								X
S2		X	X					X	
S3				X					
S4					X	X			
S5							X	X	