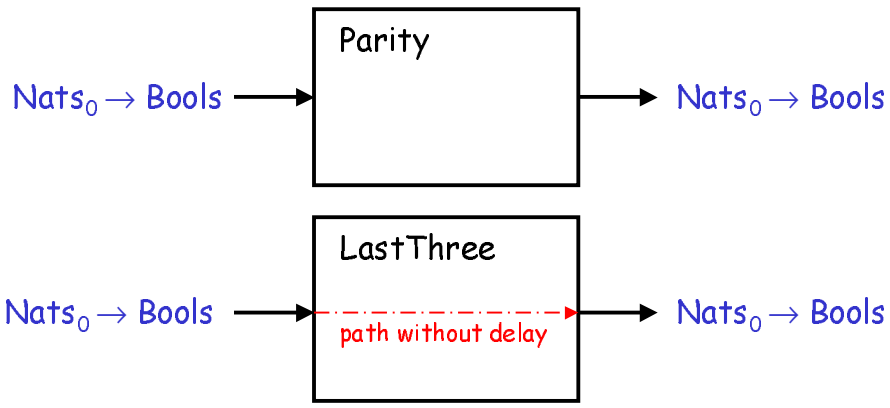
Feedback

EECS 20

Lecture 11 (February 9, 2001)

Tom Henzinger



The Parity System :

States [Parity] = { true, false }

initialState [Parity] = true

nextState [Parity] (q,x) = $(q \neq x)$

output [Parity] (q,x) = q

The LastThree System :

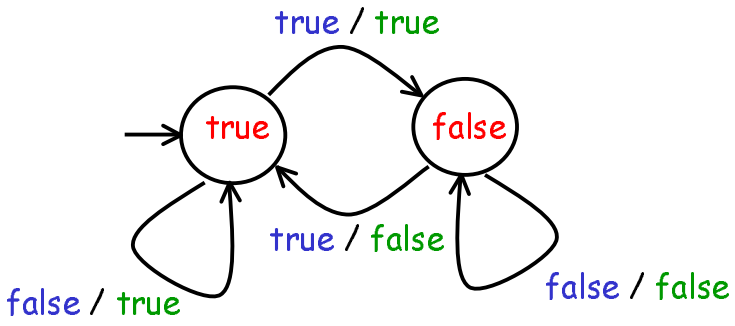
States [LastThree] = { 0, 1, 2 }

initialState [LastThree] = 0

nextState [LastThree] (q,x) = $\begin{cases} 0 & \text{if } \neg x \\ \min(q+1, 2) & \text{if } x \end{cases}$

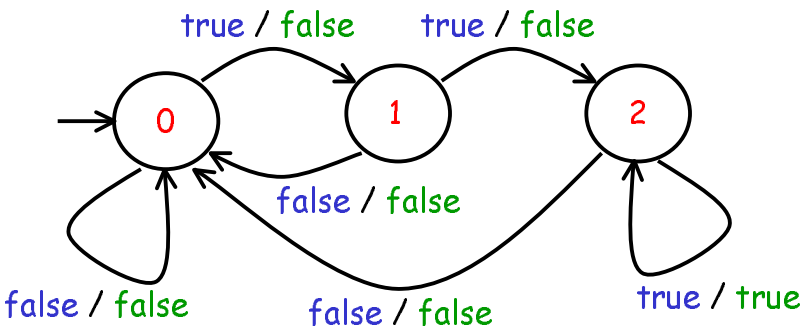
output [LastThree] (q,x) = $((q = 2) \wedge x)$

Parity System



States = Bools
Inputs = Bools
Outputs = Bools

LastThree System



States = { 0, 1, 2 }
Inputs = Bools
Outputs = Bools

A Run of the LastThree System

Time	0	1	2	3	4
Input	t	t	t	t	f
Output					
State					

A Run of the LastThree System

Time	0	1	2	3	4
Input	t	t	t	t	f
Output					
State	0				

A Run of the LastThree System

Time	0	1	2	3	4
Input	t	t	t	t	f
Output	f				
State	0	1			

A Run of the LastThree System

Time	0	1	2	3	4
Input	t	t	t	t	f
Output	f	f			
State	0	1	2		

A Run of the LastThree System

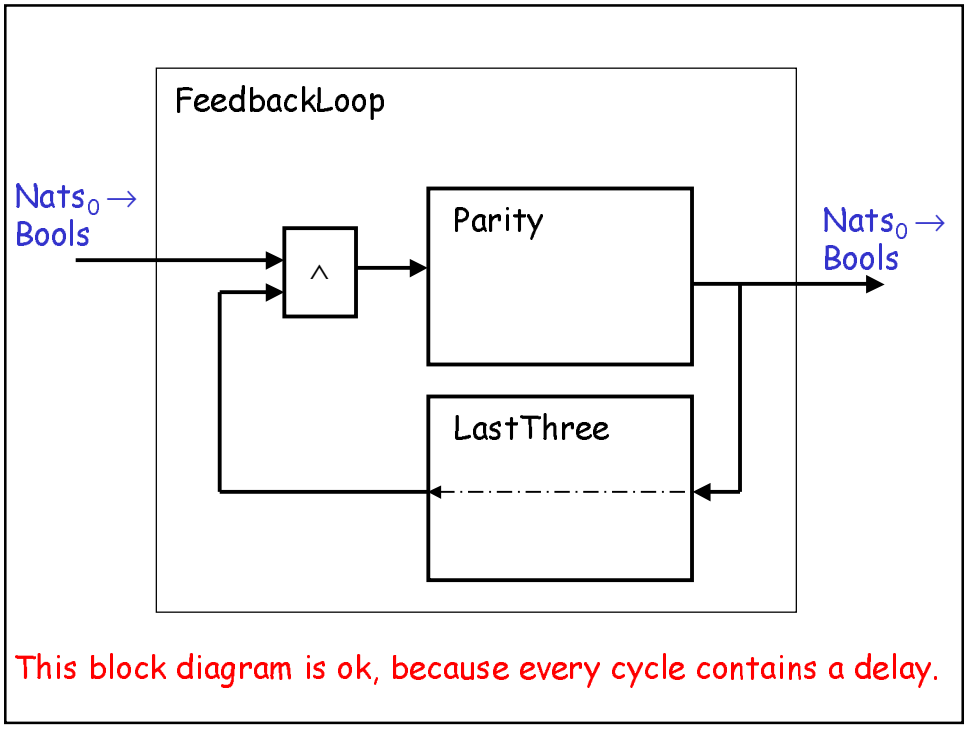
Time	0	1	2	3	4
Input	t	t	t	t	f
Output	f	f	t		
State	0	1	2	2	

A Run of the LastThree System

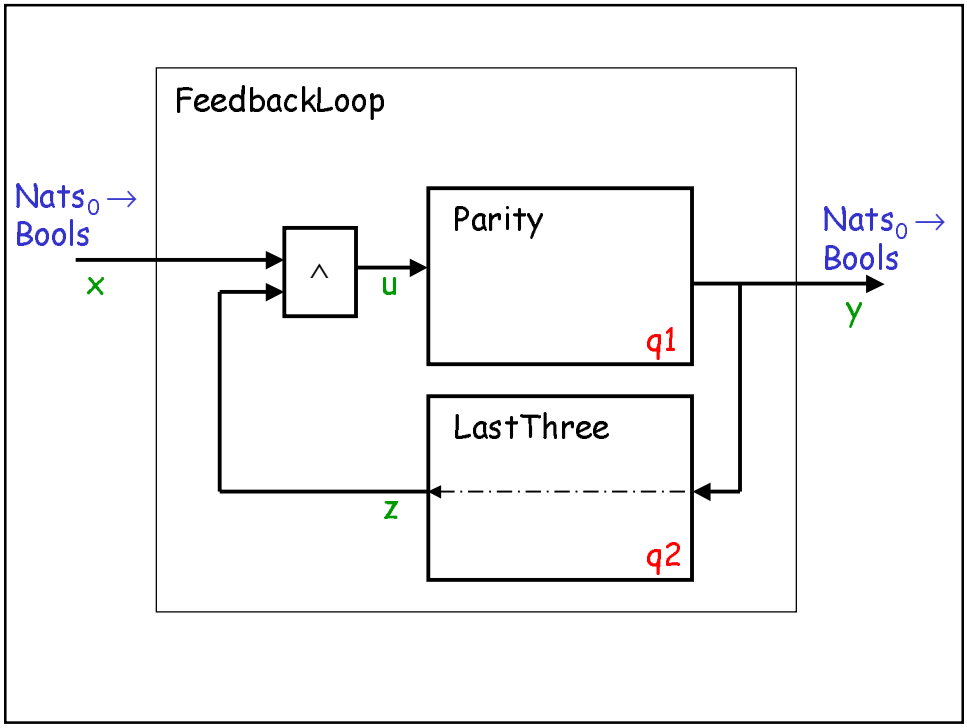
Time	0	1	2	3	4
Input	t	t	t	t	f
Output	f	f	t	t	
State	0	1	2	2	2

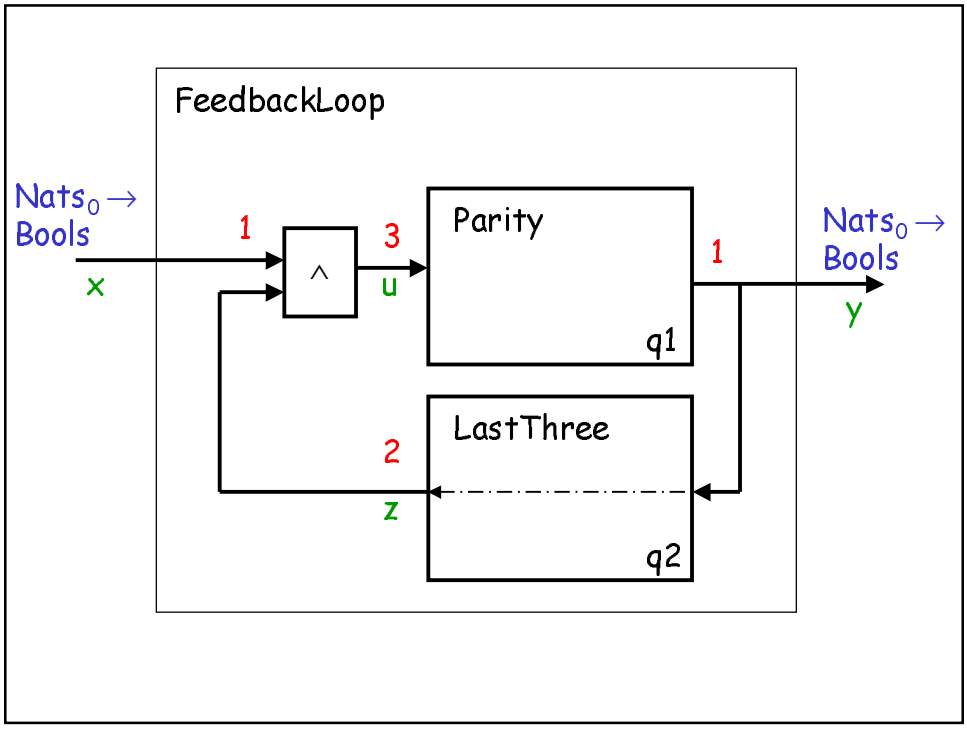
A Run of the LastThree System

Time	0	1	2	3	4	
Input	t	t	t	t	f	
Output	f	f	t	t	f	
State	0	1	2	2	2	0



This block diagram is ok, because every cycle contains a delay.





A Run of the FeedbackLoop System

	Time	0	1	2	3	4
	Input x	t	t	f	t	f
1	Output y					
2	Aux z					
3	Aux u					
	State q1					
	State q2					

A Run of the FeedbackLoop System

	Time	0	1	2	3	4
	Input x	t	t	f	t	f
1	Output y					
2	Aux z					
3	Aux u					
	State q1 t					
	State q2 0					

A Run of the FeedbackLoop System

	Time	0	1	2	3	4
	Input x	t	t	f	t	f
1	Output y	t				
2	Aux z					
3	Aux u					
	State q1 t					
	State q2 0					

A Run of the FeedbackLoop System

	Time	0	1	2	3	4
	Input x	t	t	f	t	f
1	Output y	t				
2	Aux z	f				
3	Aux u					
	State q1 t					
	State q2 0					

A Run of the FeedbackLoop System

	Time	0	1	2	3	4
	Input x	t	t	f	t	f
1	Output y	t				
2	Aux z	f				
3	Aux u	f				
	State q1 t					
	State q2 0					

A Run of the FeedbackLoop System

	Time	0	1	2	3	4
	Input x	t	t	f	t	f
1	Output y	t				
2	Aux z	f				
3	Aux u	f				
	State q1 t	t				
	State q2 0	1				

A Run of the FeedbackLoop System

	Time	0	1	2	3	4
	Input x	t	t	f	t	f
1	Output y	t	t			
2	Aux z	f				
3	Aux u	f				
	State q1 t	t				
	State q2 0	1				

A Run of the FeedbackLoop System

	Time	0	1	2	3	4
	Input x	t	t	f	t	f
1	Output y	t	t			
2	Aux z	f	f			
3	Aux u	f				
	State q1 t	t				
	State q2 0	1				

A Run of the FeedbackLoop System

	Time	0	1	2	3	4
	Input x	t	t	f	t	f
1	Output y	t	t			
2	Aux z	f	f			
3	Aux u	f	f			
	State q1 t	t				
	State q2 0	1				

A Run of the FeedbackLoop System

	Time	0	1	2	3	4
	Input x	t	t	f	t	f
1	Output y	t	t			
2	Aux z	f	f			
3	Aux u	f	f			
	State q1	t	t	t		
	State q2	0	1	2		

A Run of the FeedbackLoop System

Time	0	1	2	3	4
Input x	t	t	f	t	f
Output y	t	t	t	t	f
Aux z	f	f	t	t	f
Aux u	f	f	f	t	f
State q1	t	t	t	t	f	f
State q2	0	1	2	2	2	0

The FeedbackLoop System

Inputs [FeedbackLoop] = Bools

Outputs [FeedbackLoop] = Bools

States [FeedbackLoop]

= States [Parity] × States [LastThree]

= { true, false } × { 0, 1, 2 }

initialState [FeedbackLoop]

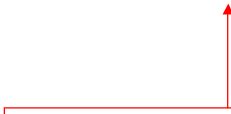
= (initialState [Parity], initialState [LastThree])

= (true, 0)

The FeedbackLoop System, continued

output [FeedbackLoop] ((q1, q2) , x)

= output [Parity] (q1, any)



The value of this does not matter !

The FeedbackLoop System, continued

output [FeedbackLoop] ((q1, q2) , x)

= output [Parity] (q1, any)

Let this be y .

nextState [FeedbackLoop] ((q1, q2) , x) $_2$

= nextState [LastThree] (q2, y)

The FeedbackLoop System, continued

output [FeedbackLoop] ((q1, q2) , x)

= output [Parity] (q1, any)

Let this be y .

nextState [FeedbackLoop] ((q1, q2) , x) $_2$

= nextState [LastThree] (q2, y)

Let z = output [LastThree] (q2, y) .

The FeedbackLoop System, continued

output [FeedbackLoop] ((q1, q2) , x)

= output [Parity] (q1, any)

Let this be y .

nextState [FeedbackLoop] ((q1, q2) , x) $_2$

= nextState [LastThree] (q2, y)

Let z = output [LastThree] (q2, y) .

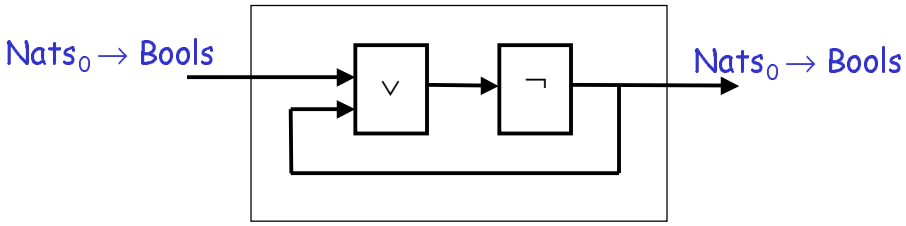
nextState [FeedbackLoop] ((q1, q2) , x) $_1$

= nextState [Parity] (q1, x $\wedge$ z)

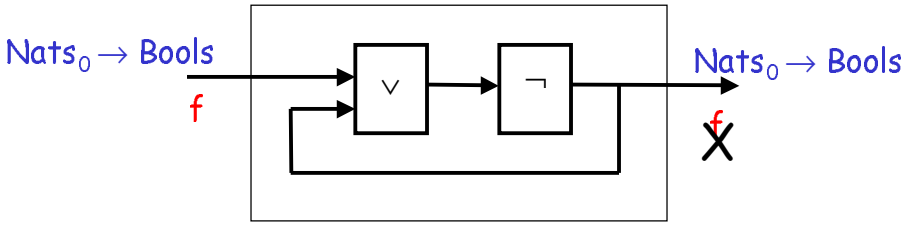
If every cycle contains a delay, then all values can be computed.

If there is a cycle without delay, then some values may not be well-defined.

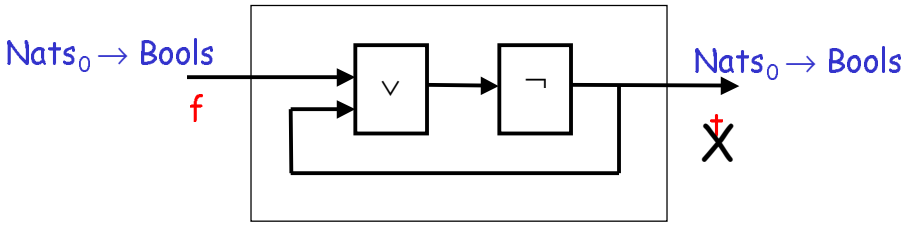
An Ill-Defined System



An Ill-Defined System

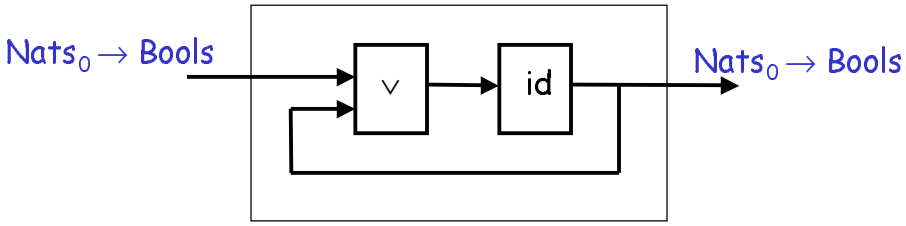


An Ill-Defined System

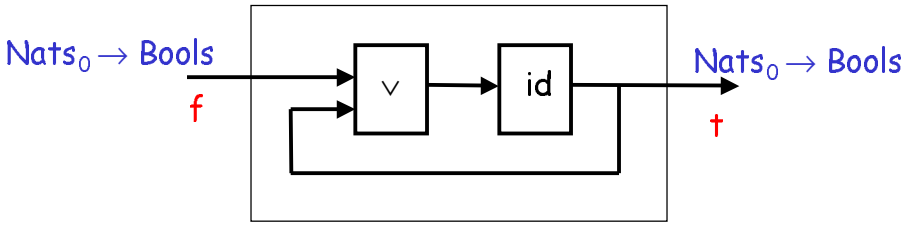


No legal output !

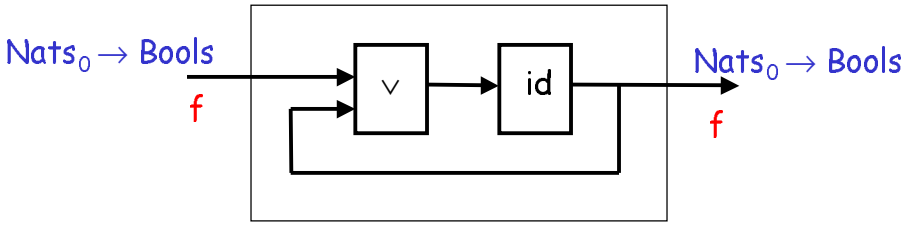
Another Ill-Defined System



Another Ill-Defined System



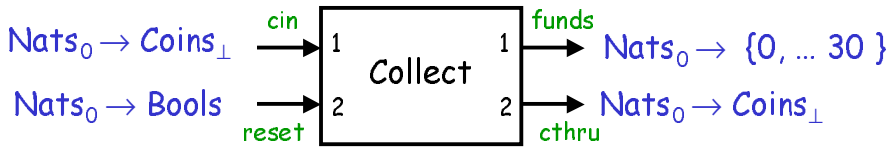
Another Ill-Defined System



Multiple legal outputs !

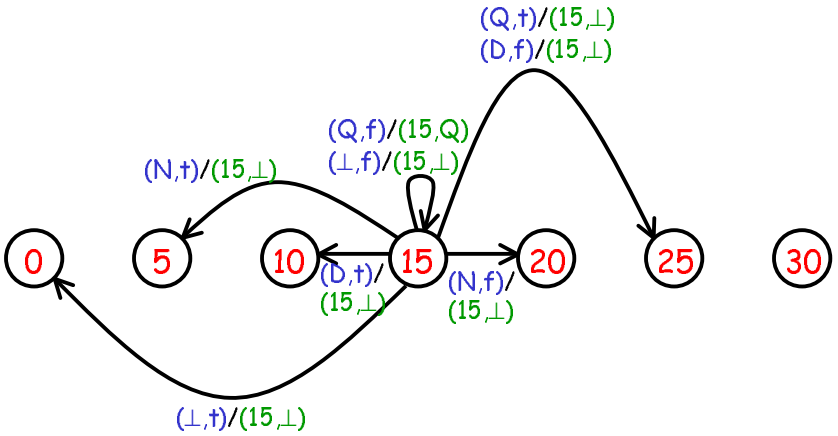
Example: Vending Machine

Coin Collector

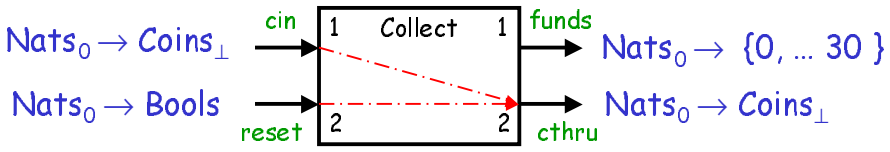


Let Coins = { Nickel, Dime, Quarter } .

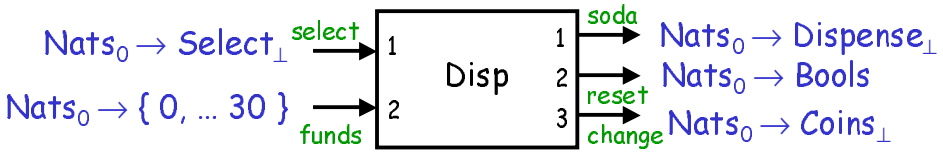
Coin Collector



Coin Collector



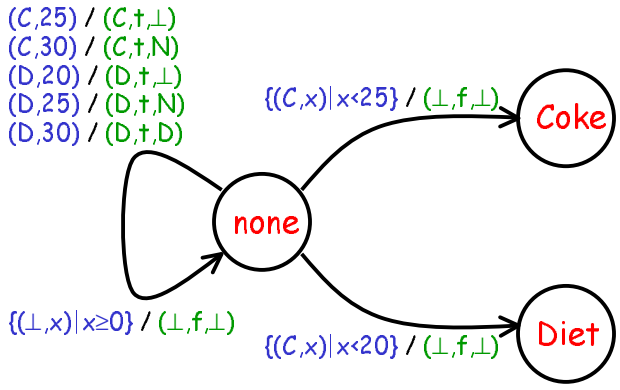
Soda Dispenser



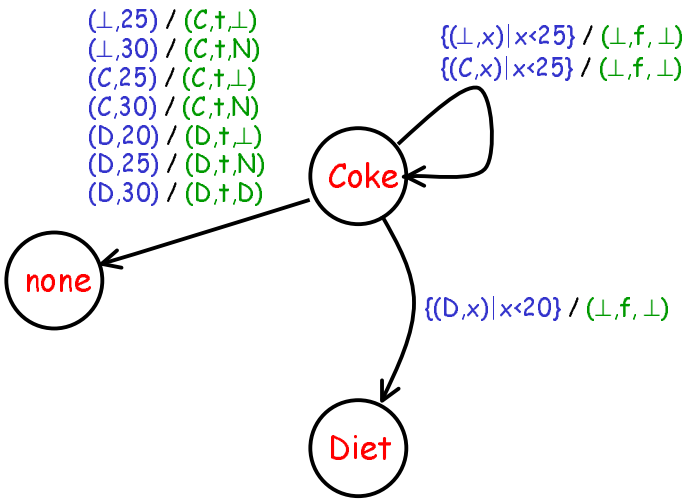
Let Select = { selectCoke, selectDiet } .

Let Dispense = { dispenseCoke, dispenseDiet } .

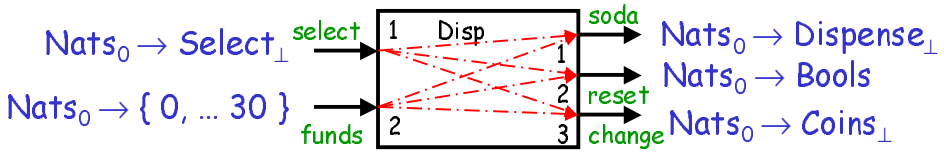
Soda Dispenser



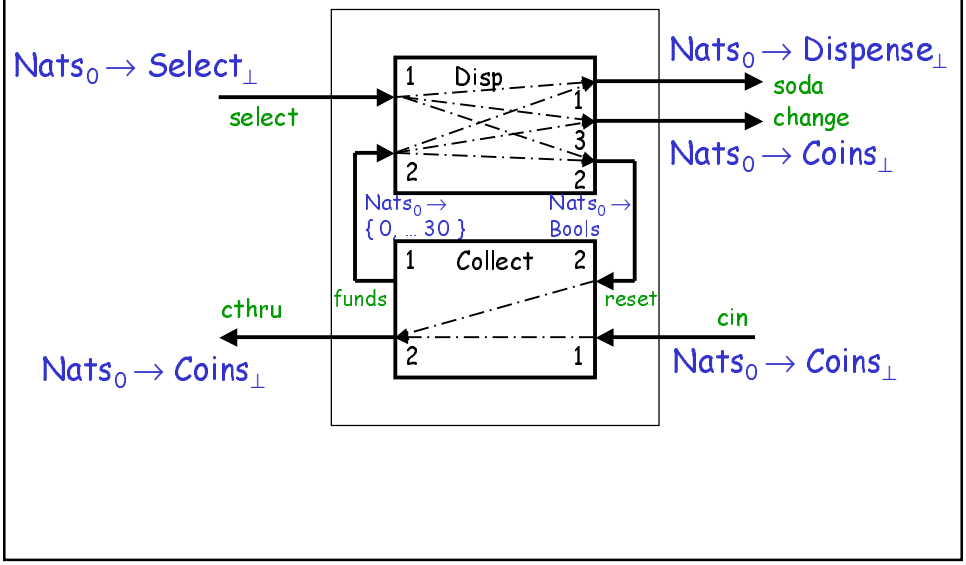
Soda Dispenser



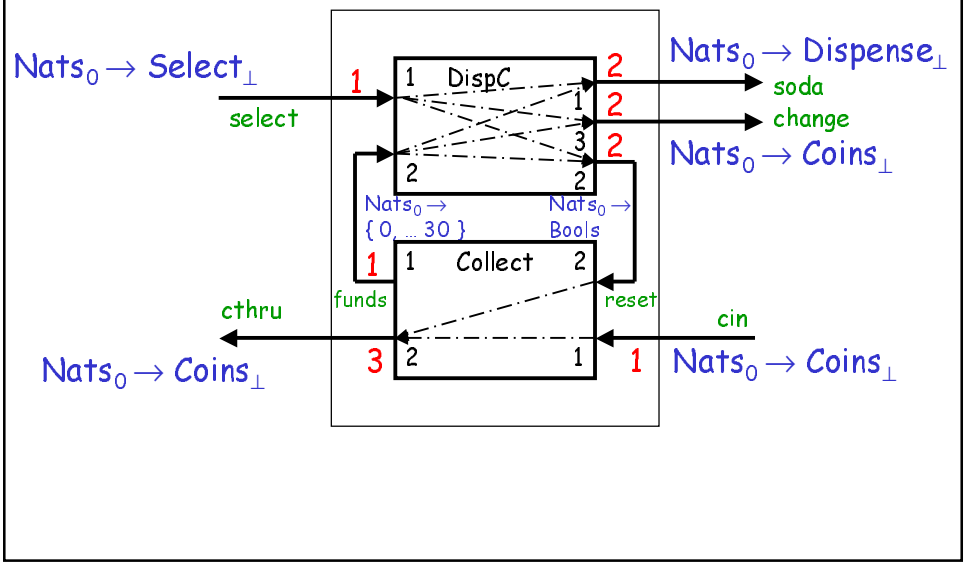
Soda Dispenser



Vending Machine



Vending Machine



	Time	0	1	2	3	4
	cin	N	⊥	Q	Q	D
	select	⊥	C	⊥	D	C
1	funds					
2	soda					
2	change					
2	reset					
3	cthru					
	State Coll	0				
	State Disp	n				

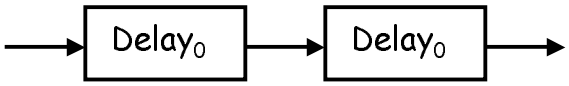
Time	0	1	2	3	4
cin	N	⊥	Q	Q	D
select	⊥	C	⊥	D	C
funds	0	5	5	0	0
soda	⊥	⊥	C	D	⊥
change	⊥	⊥	N	N	⊥
reset	f	f	t	t	f
cthru	⊥	⊥	⊥	⊥	⊥
State Coll	0	5	5	0	0	10
State Disp	n	n	C	n	n	C

Quiz

1. Draw the transition diagram of the system

$$Delay_0 : [\, Nats_0 \to Bins \,] \to [\, Nats_0 \to Bins \,]$$

$$\forall\, x \in [\, Nats_0 \to Bins \,],\ \forall\, y \in Nats_0,$$

$$(\, Delay_0\,(x))\,(y) = \begin{cases} 0 & \text{if } y = 0 \\ x\,(y\text{-}1) & \text{if } y > 0 \end{cases}$$

2. Draw the transition diagram of the system