

CpE358/CS381

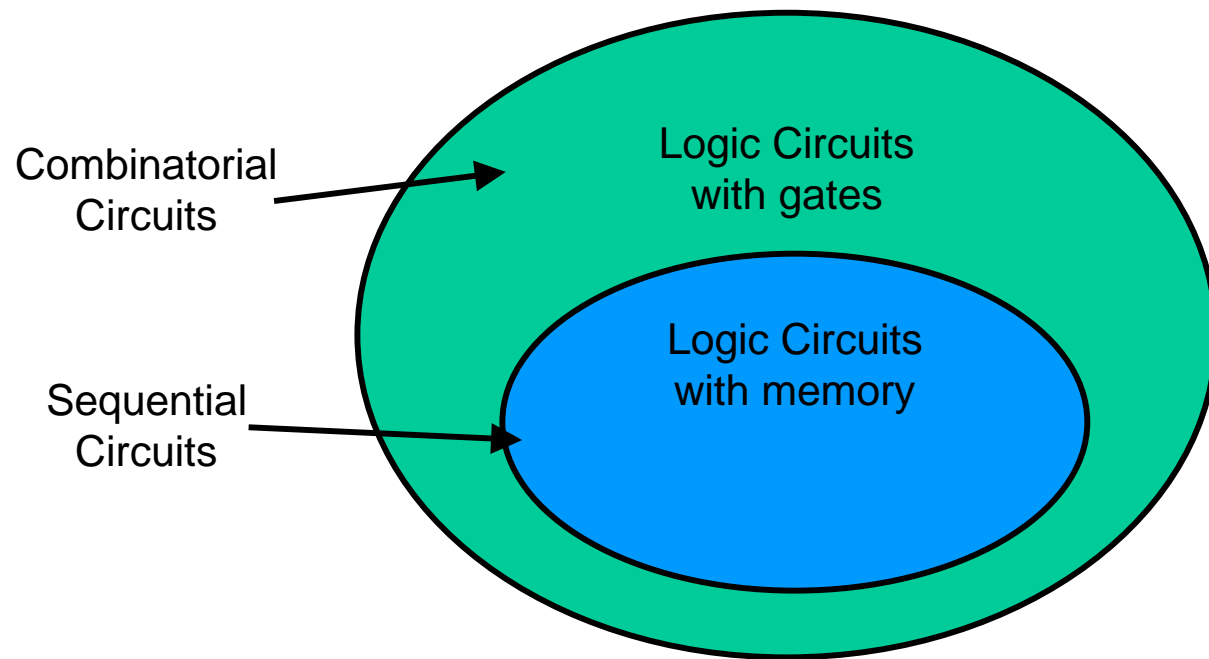
**Switching Theory and
Logical Design**

Class 13

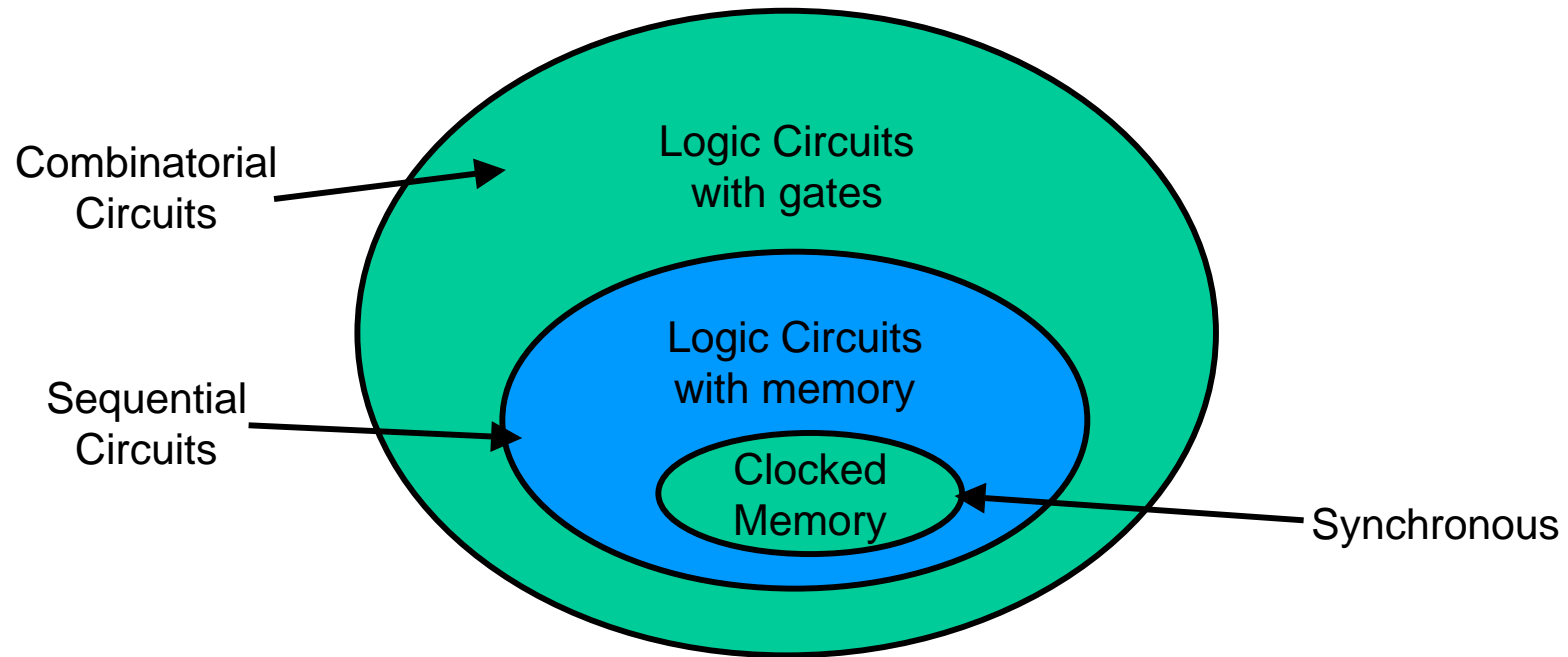
Today

- Fundamental concepts of digital systems (Mano Chapter 1)
- Binary codes, number systems, and arithmetic (Ch 1)
- Boolean algebra (Ch 2)
- Simplification of switching equations (Ch 3)
- Digital device characteristics (e.g., TTL, CMOS)/design considerations (Ch 10)
- Combinatoric logical design including LSI implementation (Chapter 4)
- Flip-flops and state memory elements (Ch 5)
- Sequential logic analysis and design (Ch 5)
- Counters, shift register circuits (Ch 6)
- Hazards, Races, and time related issues in digital design (Ch 9)
- **Synchronous vs. asynchronous design (Ch 9)**
- Memory and Programmable logic (Ch 7)
- Minimization of sequential systems
- Introduction to Finite Automata

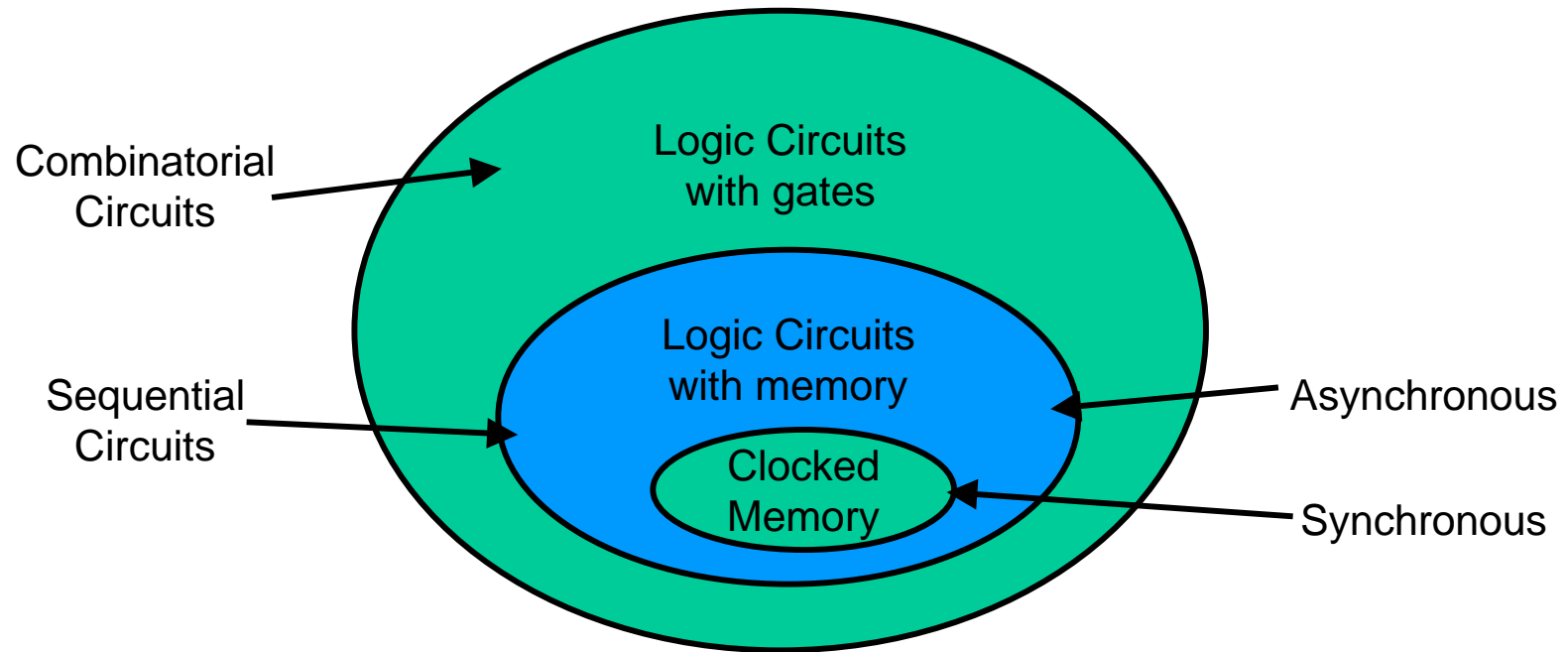
Course Roadmap



Course Roadmap

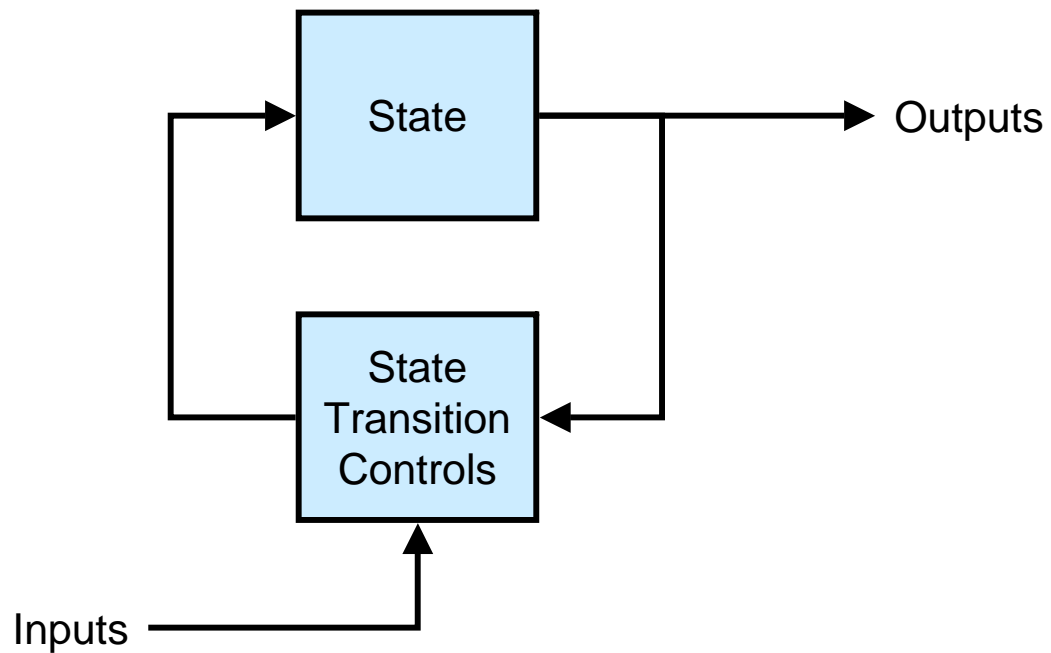


Course Roadmap



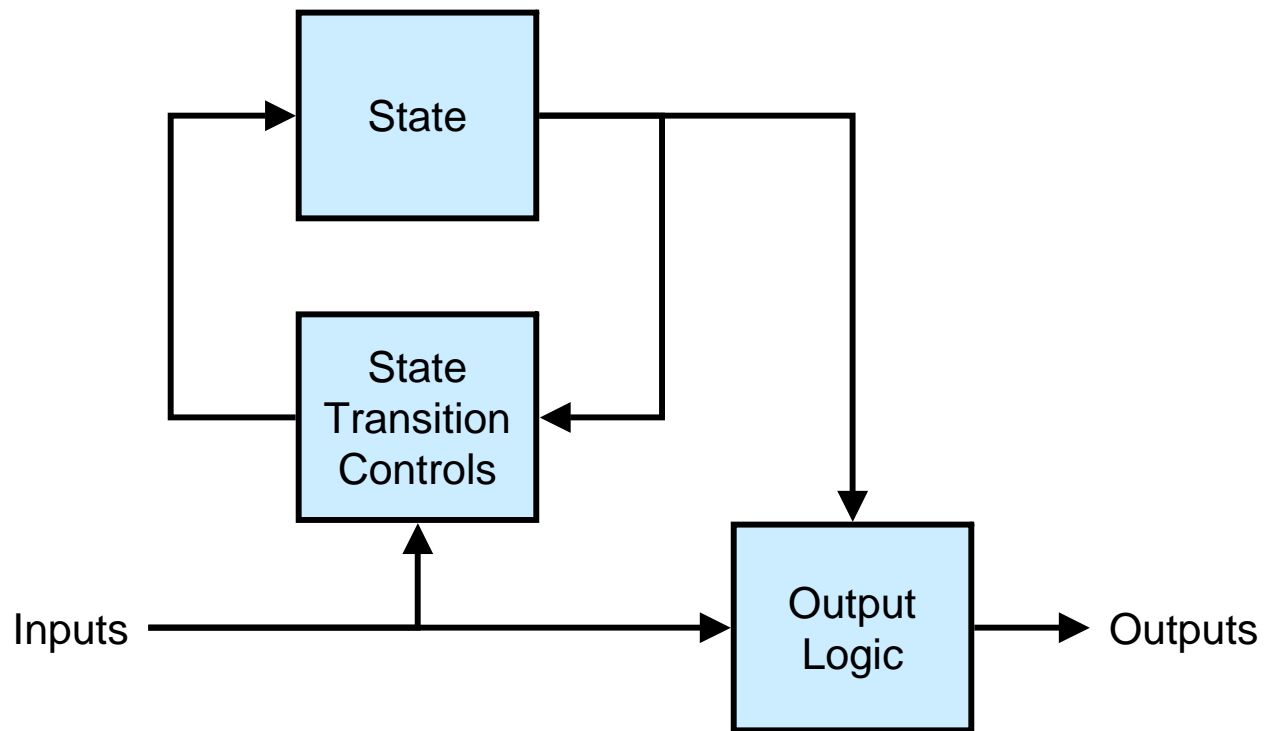
Mealy/Moore Models

- Output is a function of state only = Moore Model

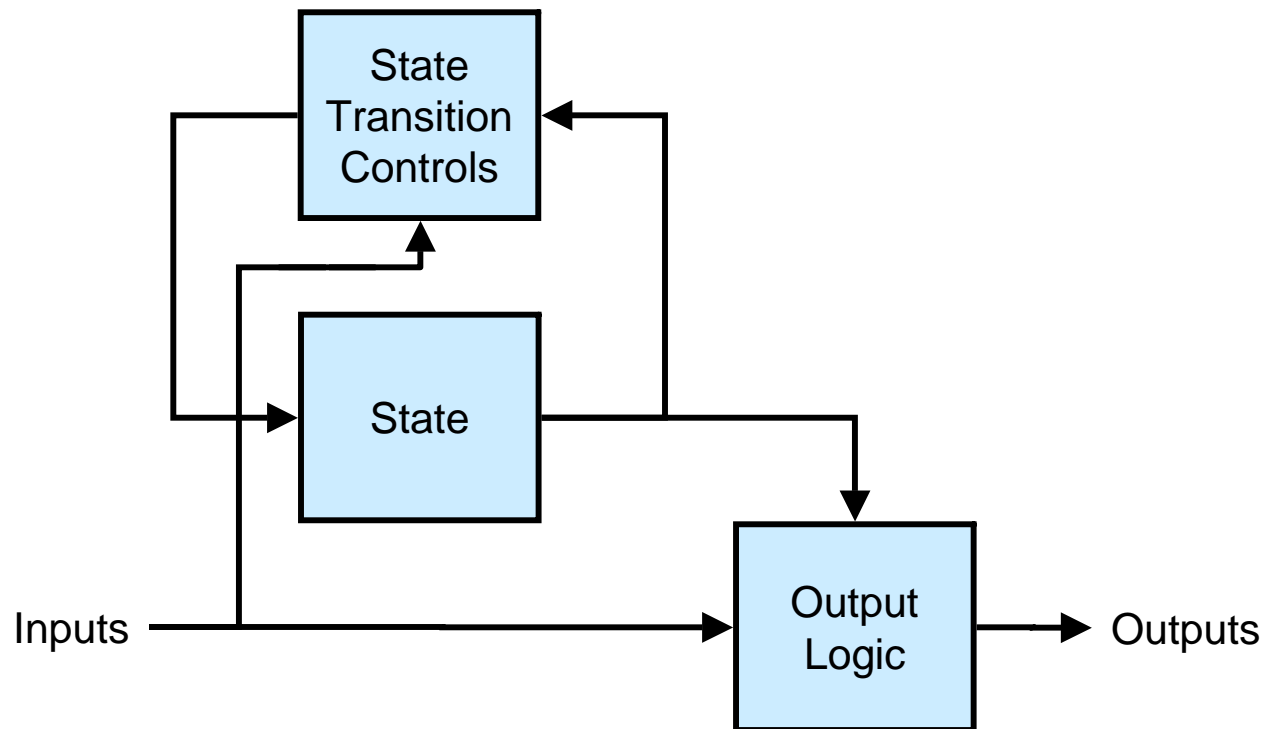


Mealy/Moore Models

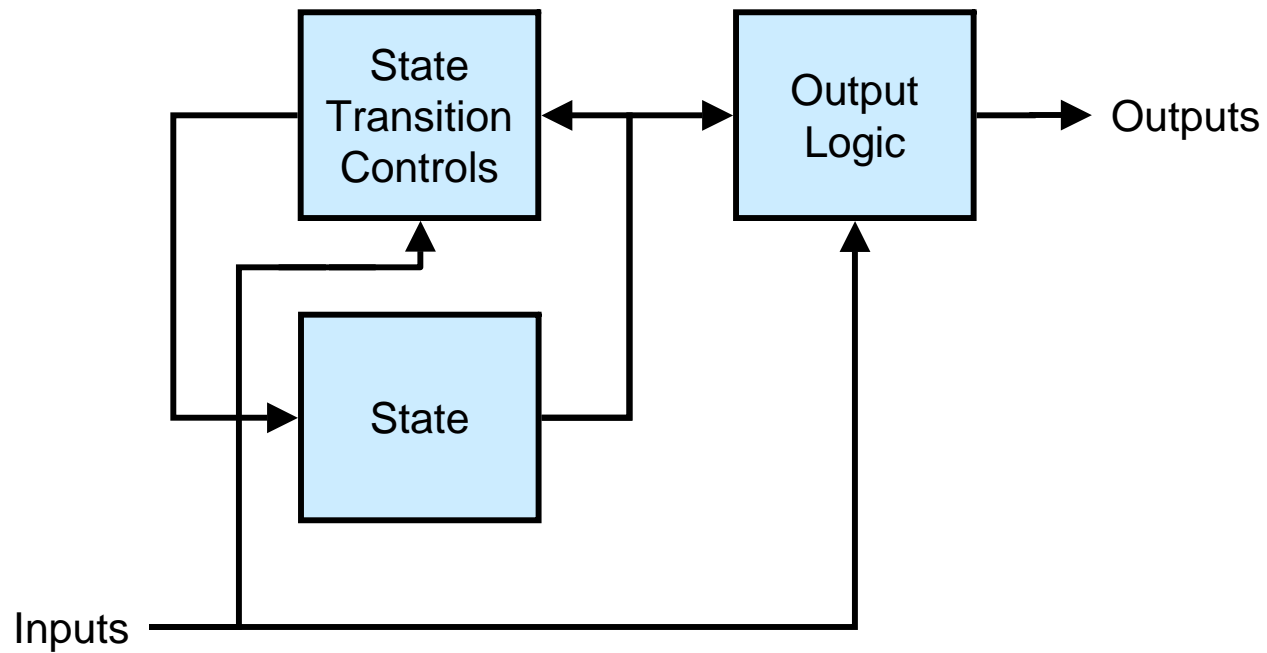
- Output is a function of state and inputs = Mealy Model, a more generic model



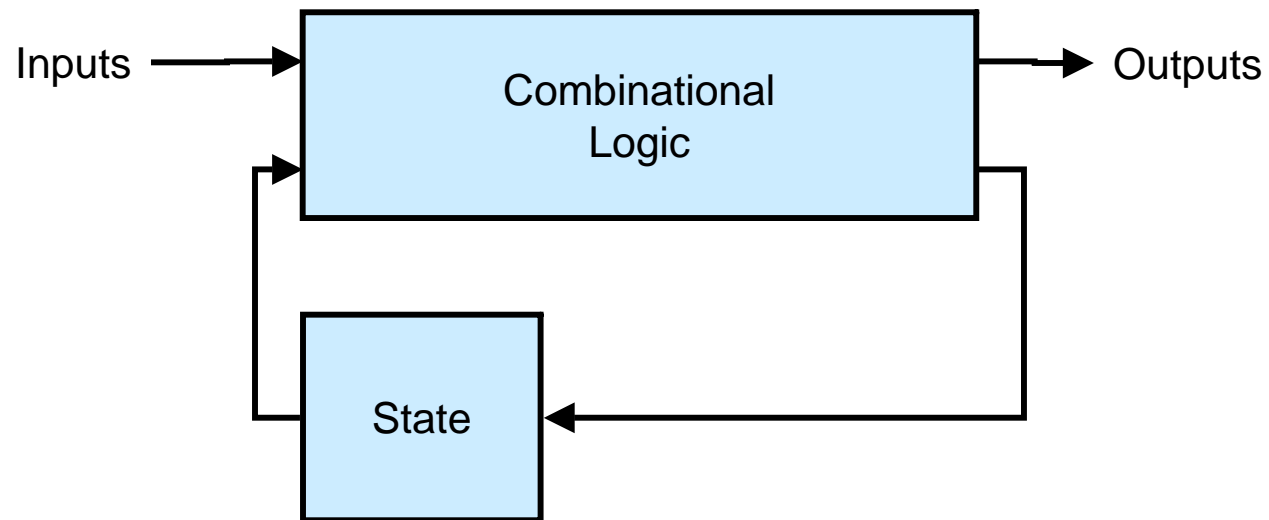
Rearranging Circuit Model



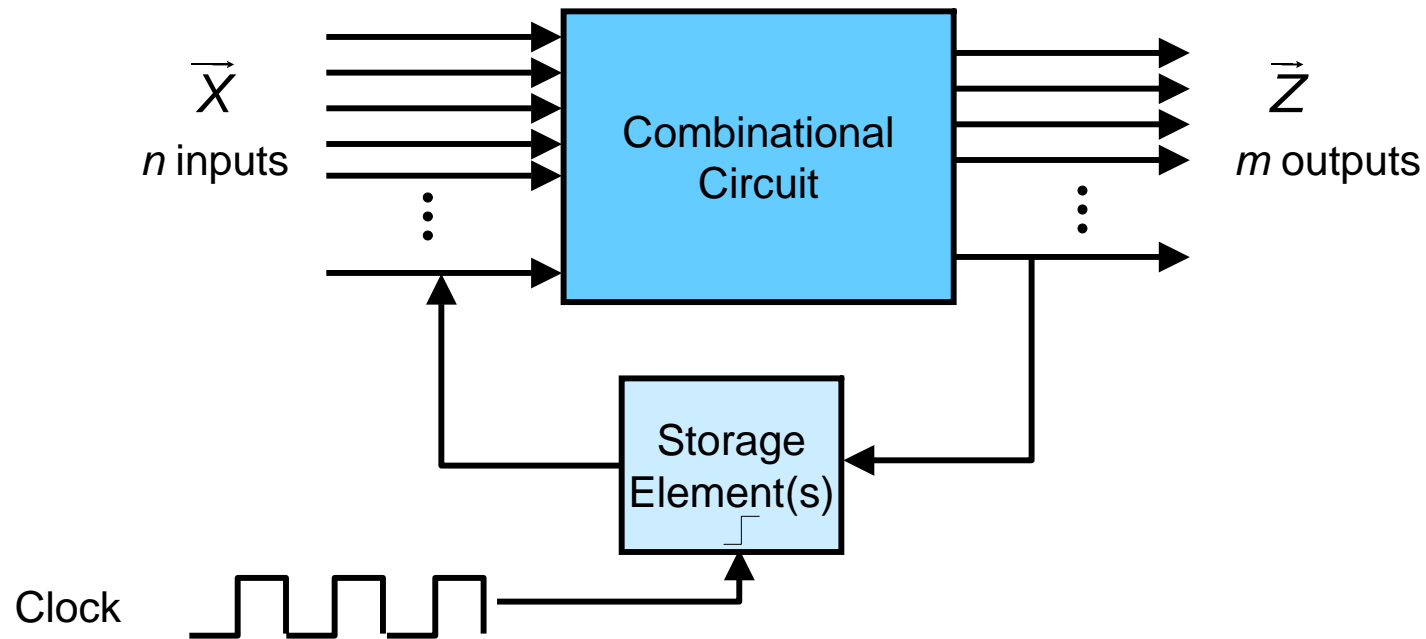
Rearranging Circuit Model



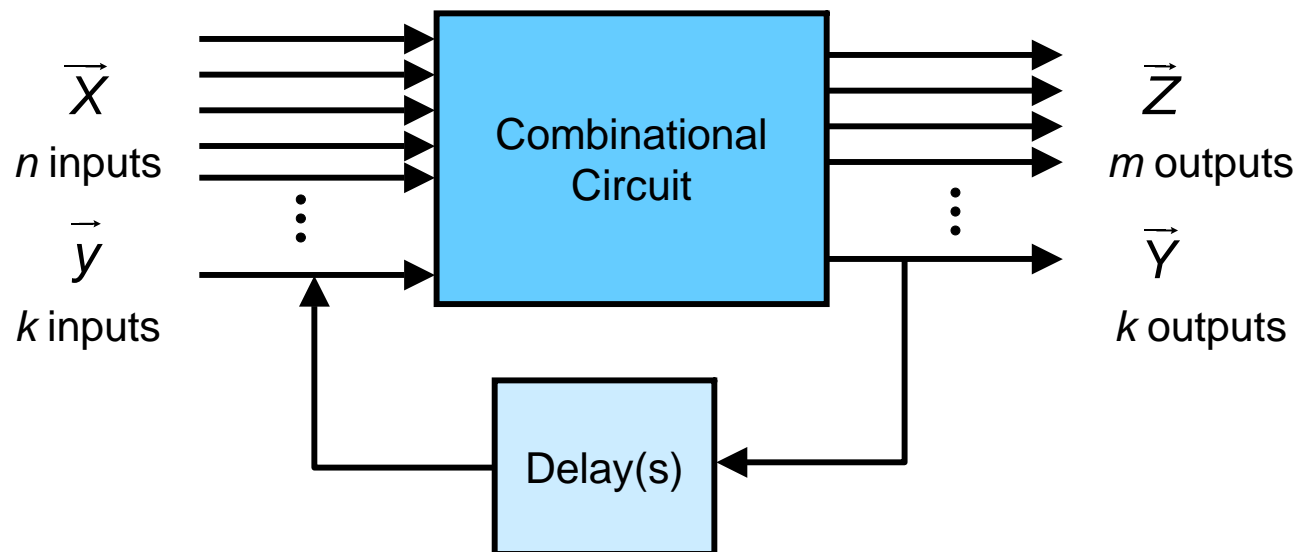
Rearranging Circuit Model



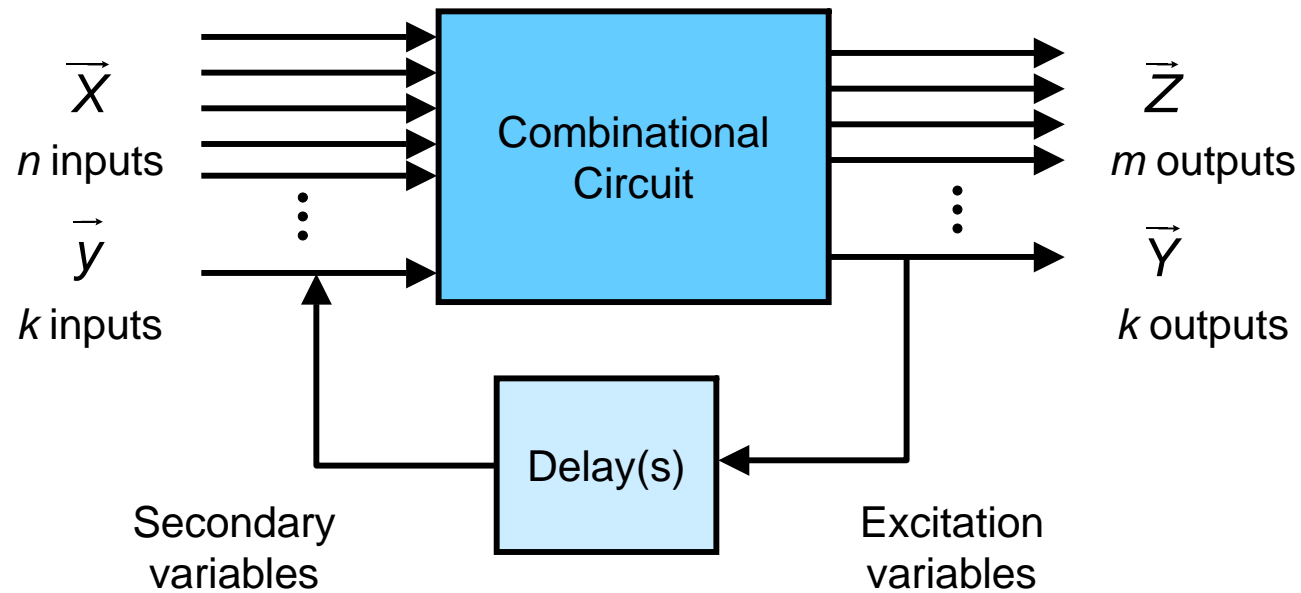
Synchronous Sequential Circuit



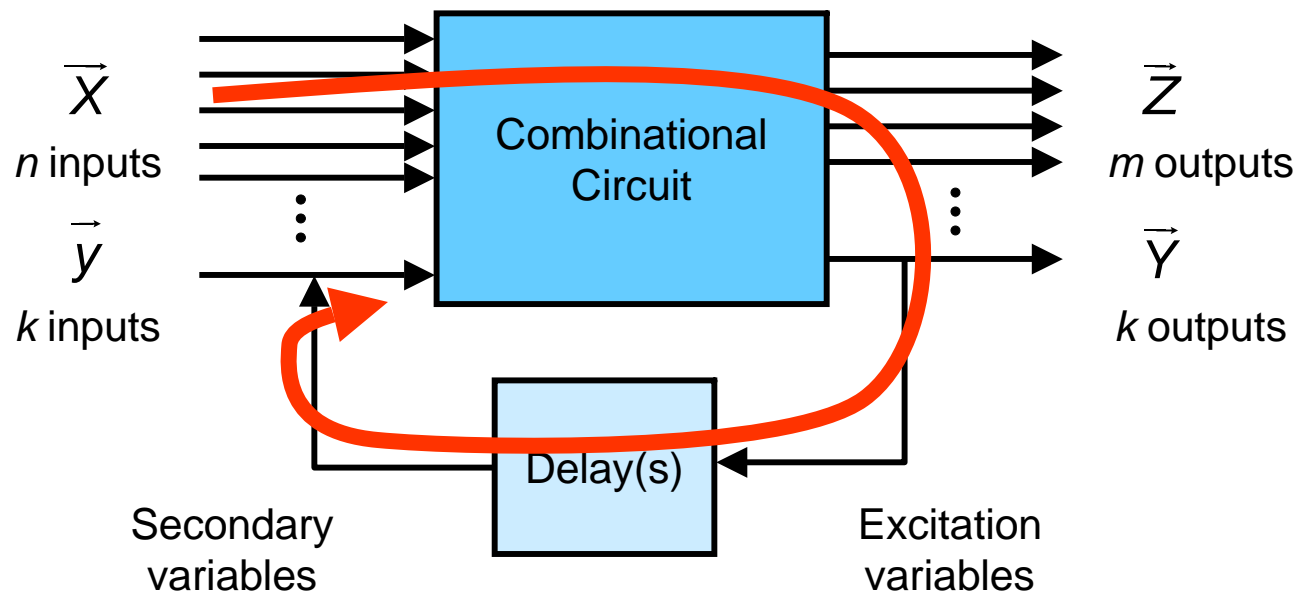
Asynchronous Sequential Circuit



Asynchronous Sequential Circuit



Asynchronous Sequential Circuit

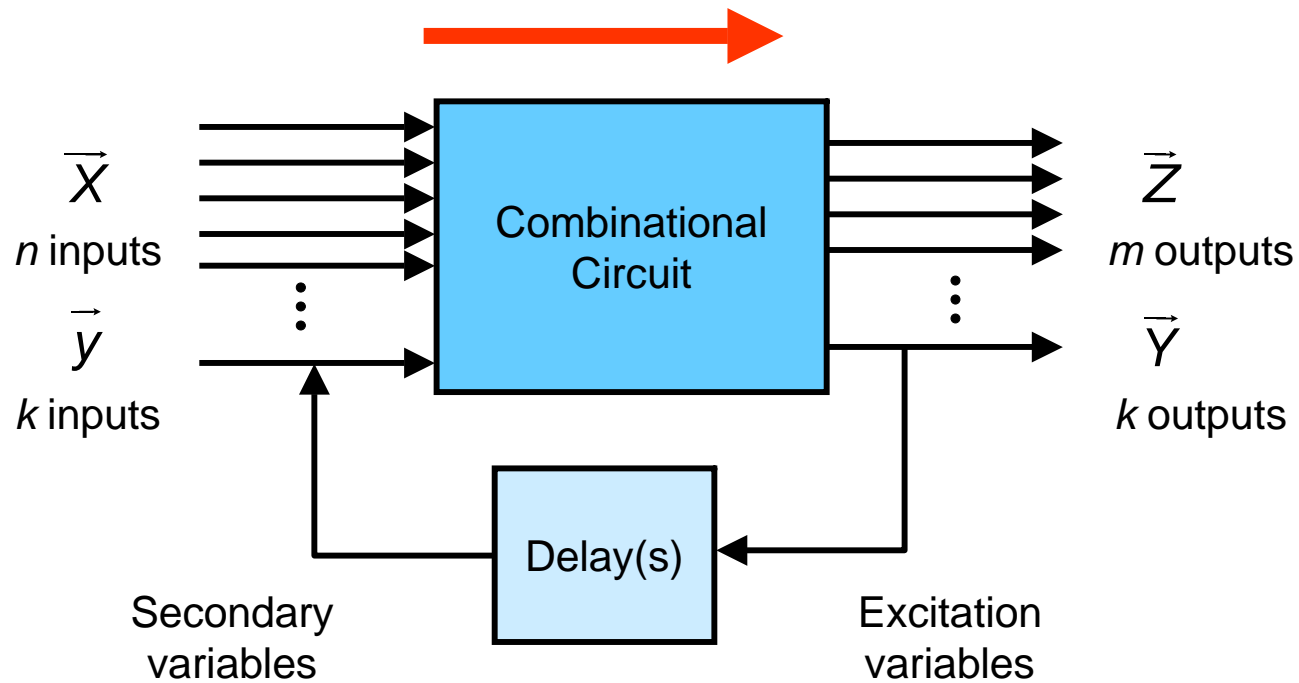


Change of input variable(s) creates (delayed) change of secondary variable(s)

Stability Condition

- With no change in X input, Z output and Y states attain constant values

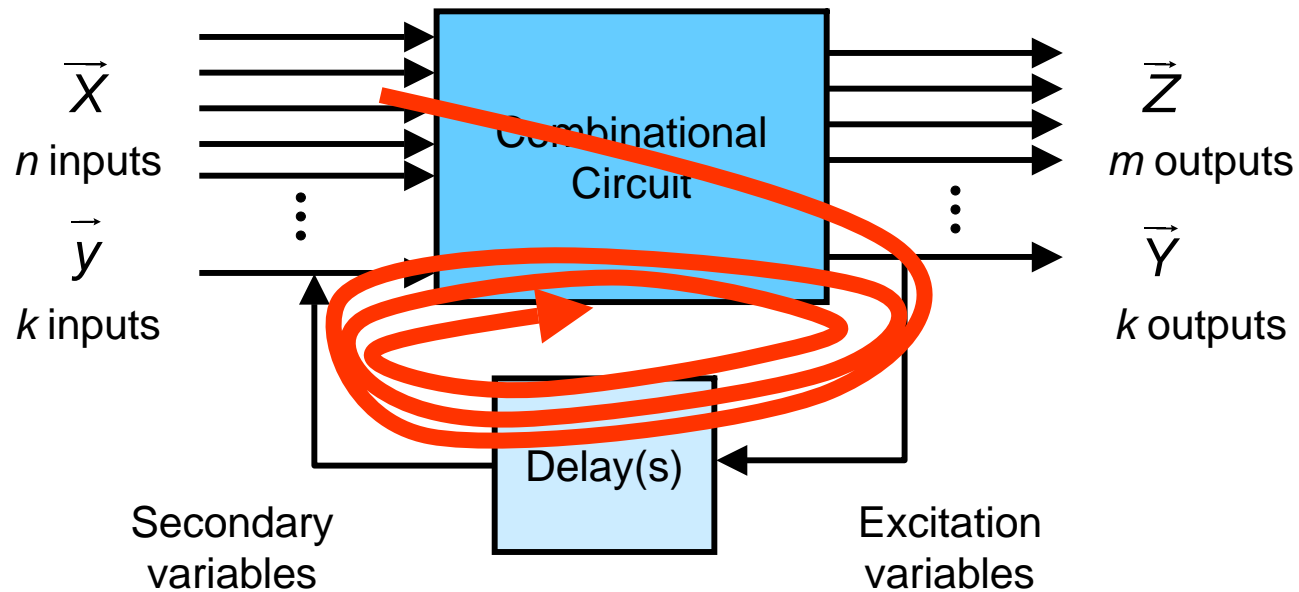
$$\vec{X} = \vec{C}_X \longrightarrow \vec{Z} \rightarrow \vec{C}_Z, \vec{Y} \rightarrow \vec{C}_Y$$



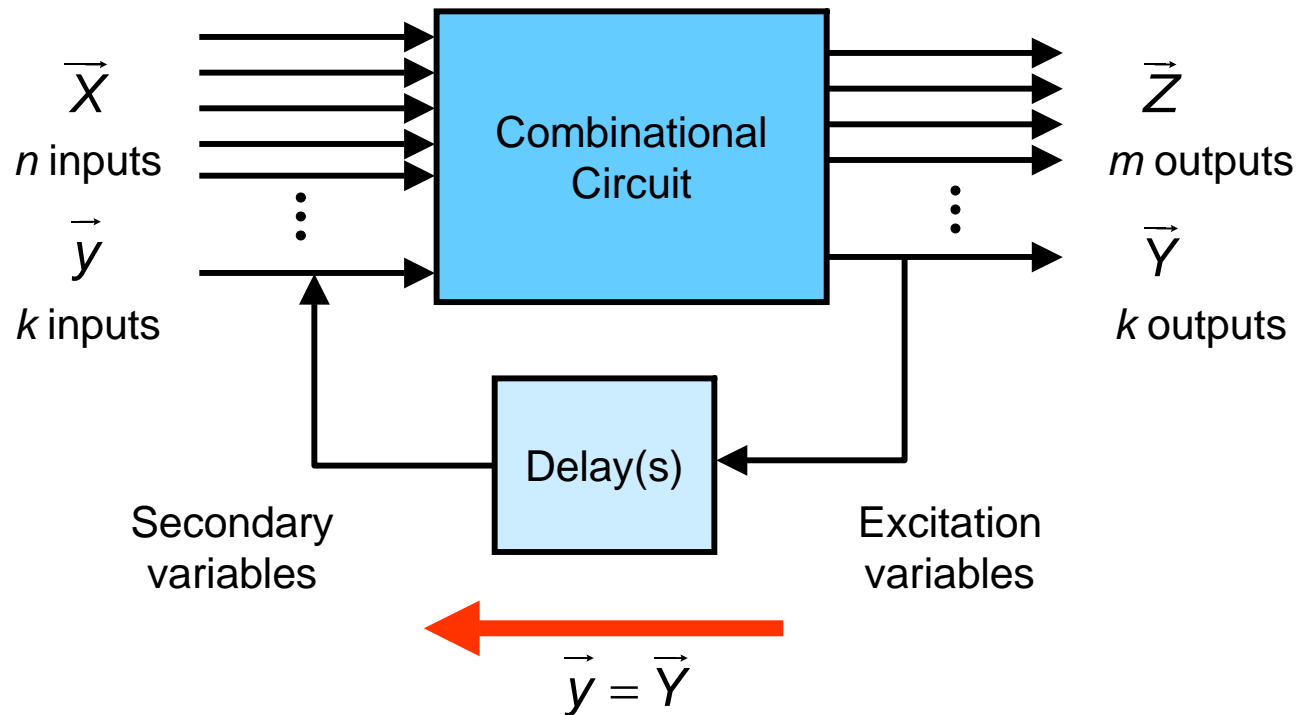
Stability Condition

- With no change in X input, if Y states do not attain constant values, system is unstable

$$\vec{X} = \vec{C}_x \quad \text{but} \quad \vec{Y} \neq \vec{y}$$



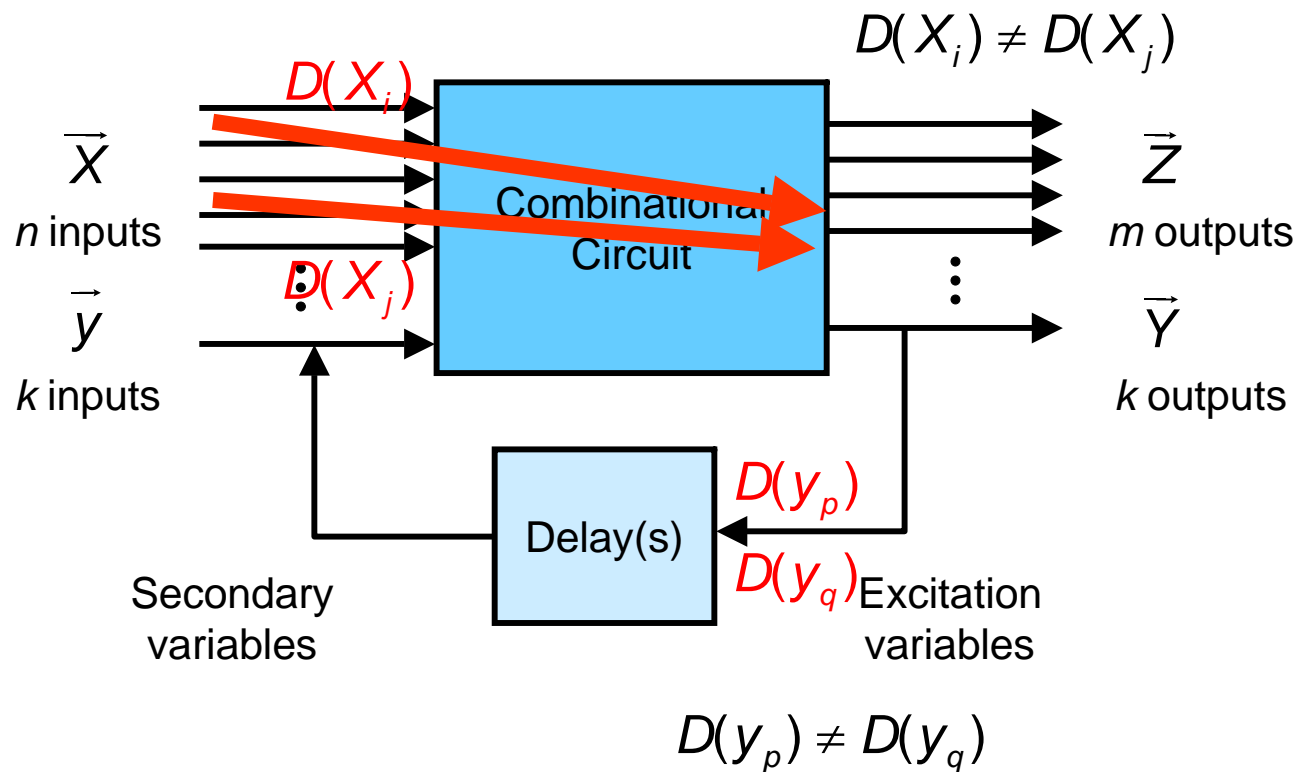
Steady-State Condition



- In steady-state, delays become irrelevant

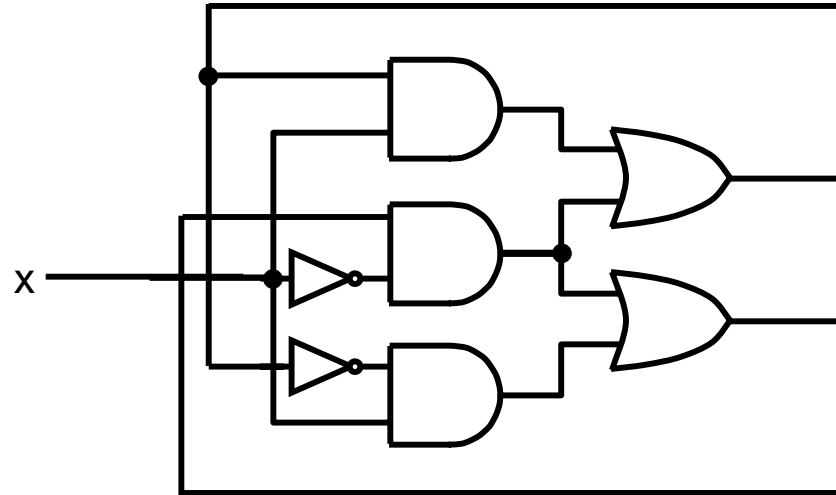
Single Input Changing Condition

- Circuit delays can never be equal, so only one variable can change “at a time” (compared to time to reach stable state)



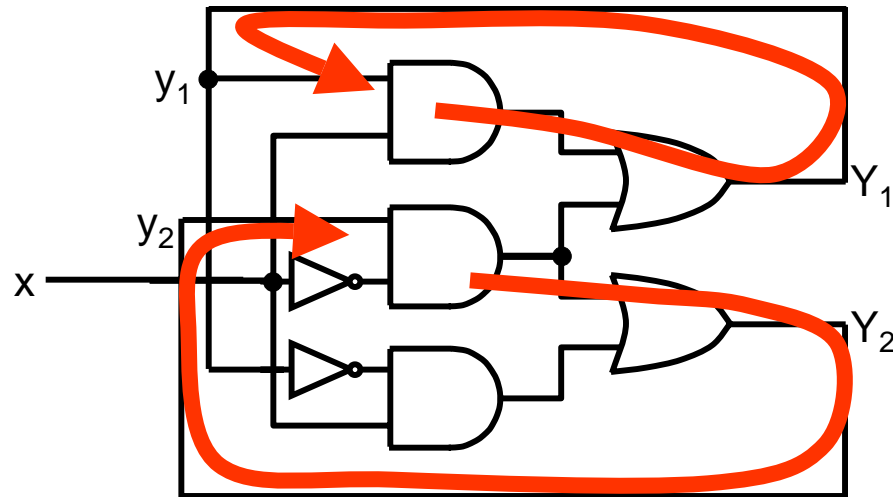
Analysis of Asynchronous Circuits

- Identify feedback loops to be able to find excitation and secondary variables:

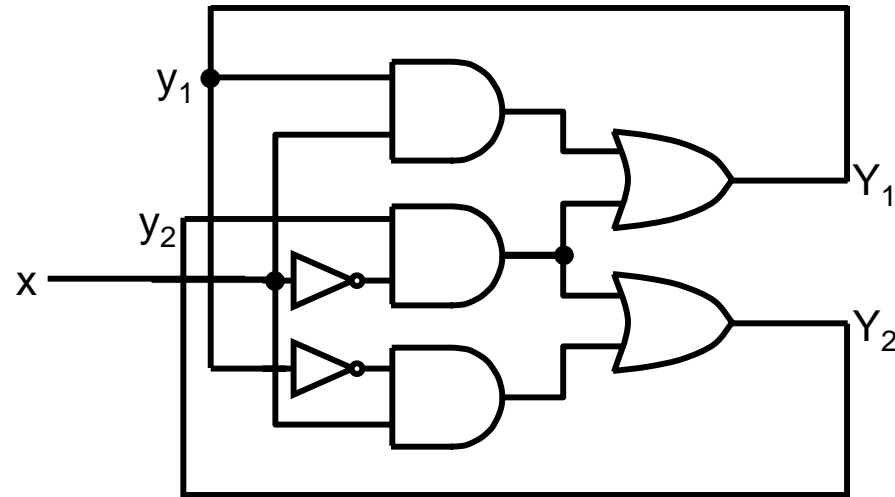


Analysis of Asynchronous Circuits

- Identify feedback loops to be able to find excitation and secondary variables:



Analysis of Asynchronous Circuits



- Write Boolean functions expressing excitation variables in terms of input and secondary variables:

$$Y_1 = (xy_1) + (x'y_2)$$

$$Y_2 = (xy'_1) + (x'y_2)$$

Analysis of Asynchronous Circuits

$$Y_1 = (xy_1) + (x' y_2)$$

$$Y_2 = (xy'_1) + (x' y_2)$$

- Plot excitation variable in a map, as a function of input and secondary variables:

input(s)

	input(s)	
	↙	↘
	0	1
secondary variable(s)	$y_1 y_2 \backslash x$	
	00	0
	01	1
	11	1
	10	0

Y_1

	input(s)	
	↙	↘
	0	1
secondary variable(s)	$y_1 y_2 \backslash x$	
	00	0
	01	1
	11	1
	10	0

Y_2

Analysis of Asynchronous Circuits

$$Y_1 = (xy_1) + (x'y_2)$$

$$Y_2 = (xy'_1) + (x'y_2)$$

- Plot excitation variable in a map, as a function of input and secondary variables.
- Create transition table from excitation variable map:

$y_1y_2 \backslash x$	0	1
00	0	0
01	1	0
11	1	1
10	0	1

Y_1

$y_1y_2 \backslash x$	0	1
00	0	1
01	1	1
11	1	0
10	0	0

Y_2

$y_1y_2 \backslash x$	0	1
00	00	01
01	11	01
11	11	10
10	00	10

Transition
table

Analysis of Asynchronous Circuits

$$Y_1 = (xy_1) + (x'y_2)$$

$$Y_2 = (xy'_1) + (x'y_2)$$

- Plot excitation variable in a map, as a function of input and secondary variables.
- Create transition table from excitation variable map.
- And mark the stable conditions:

$y_1y_2 \backslash x$	0	1
00	0	0
01	1	0
11	1	1
10	0	1

Y_1

$y_1y_2 \backslash x$	0	1
00	0	1
01	1	1
11	1	0
10	0	0

Y_2

$y_1y_2 \backslash x$	0	1
00	00	01
01	11	01
11	11	10
10	00	10

Transition table

Analysis of Asynchronous Circuits

$y_1y_2 \backslash x$	0	1
00	0	0
01	1	0
11	1	1
10	0	1

Y_1

$y_1y_2 \backslash x$	0	1
00	0	1
01	1	1
11	1	0
10	0	0

Y_2

$y_1y_2 \backslash x$	0	1
00	00	01
01	11	01
11	11	10
10	00	10

Transition
table

- Consider the effect of a change in the input variable
 $x=0, y_1y_2=00$, stable state

Input, x	State y_1y_2	Y_1Y_2	Condition
0	00	00	Stable State
1	00	01	Transitory condition
1	01	01	Stable State

Analysis of Asynchronous Circuits

$y_1y_2 \backslash x$	0	1
00	0	0
01	1	0
11	1	1
10	0	1

Y_1

$y_1y_2 \backslash x$	0	1
00	0	1
01	1	1
11	1	0
10	0	0

Y_2

$y_1y_2 \backslash x$	0	1
00	00	01
01	11	01
11	11	10
10	00	10

Transition
table

- State table for asynchronous circuit

Present state	Next state $x=0$	Next state $x=1$
00	00	01
01	11	01
10	00	10
11	11	10

Analysis of Asynchronous Circuits

$y_1y_2 \backslash x$	0	1
00	0	0
01	1	0
11	1	1
10	0	1

Y_1

$y_1y_2 \backslash x$	0	1
00	0	1
01	1	1
11	1	0
10	0	0

Y_2

$y_1y_2 \backslash x$	0	1
00	00	01
01	11	01
11	11	10
10	00	10

Transition table

- State table for asynchronous circuit

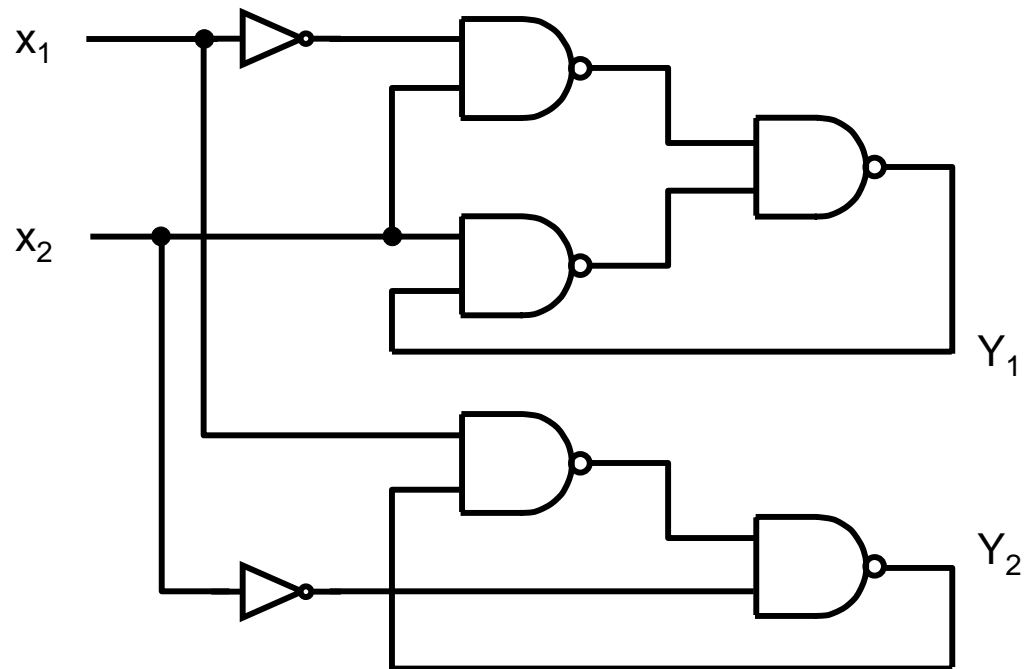
Present state	Next state $x=0$	Next state $x=1$
00	00	01
01	11	01
10	00	10
11	11	10

Each line must contain present state for (at least) one of the input conditions to ensure stability

Analysis of Asynchronous Circuits

Problem 9-2

- Derive transition table and determine the sequence of internal states for input sequence 00, 10, 11, 01, 11, 10, 00



Analysis of Asynchronous Circuits

Problem 9-2

$$Y_1 = ((x'_1 x_2)'(x_2 y_1))' = x'_1 x_2 + x_2 y_1$$

$$Y_2 = (x'_2(x_1 y_2))' = x_1 y_2 + x_2$$

- Plot excitation variables in a map, as a function of input and secondary variables:

$x_1 x_2$:	00	01	11	10
00	0	1	0	0
$y_1 y_2$ 01	0	1	1	0
11	0	1	1	0
10	0	1	0	0

Y_1

$x_1 x_2$:	00	01	11	10
00	0	1	1	0
$y_1 y_2$ 01	0	1	1	1
11	0	1	1	1
10	0	1	1	0

Y_2

Analysis of Asynchronous Circuits

Problem 9-2

x_1x_2 :	00	01	11	10
00	0	1	0	0
y_1y_2 01	0	1	1	0
11	0	1	1	0
10	0	1	0	0

Y_1

x_1x_2 :	00	01	11	10
00	0	1	1	0
y_1y_2 01	0	1	1	1
11	0	1	1	1
10	0	1	1	0

Y_2

- Create transition table from excitation variable map.
- And mark the stable conditions:

x_1x_2 :	00	01	11	10
00	00	11	01	00
State 01	00	11	11	01
11	00	11	11	01
10	00	11	10	00

Analysis of Asynchronous Circuits

Problem 9-2

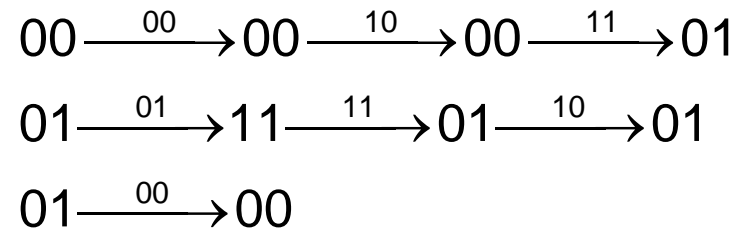
- Transition table:

x_1x_2 :	00	01	11	10
00	00	11	01	00
y_1y_2 01	00	11	11	01
11	00	11	11	01
10	00	11	10	00

- State table:

Present state	Next state, $x_1x_2=$			
	00	01	10	11
00	00	11	00	01
01	00	11	01	11
10	00	11	00	10
11	00	11	11	01

With inputs 00, 10, 11, 01, 11, 10, 00:



Analysis of Asynchronous Circuits

Problem 9-2

- Transition table:

x_1x_2 :	00	01	11	10
00	00	11	01	00
y_1y_2 01	00	11	11	01
11	00	11	11	01
10	00	11	10	00

- Flow table:

Expressed with
Symbolic state
names

x_1x_2 :	00	01	11	10
a	a	c	b	a
State b	a	c	c	b
c	a	c	c	b
d	a	c	d	a

Flow Tables

- Problem 9.2:

x_1x_2 :	00	01	11	10
00	00	11	01	00
y_1y_2 01	00	11	11	01
11	00	11	11	01
10	00	11	10	00

x_1x_2 :	00	01	11	10
a	a	c	b	a
State b	a	c	c	b
c	a	c	c	b
d	a	c	d	a

Previous example:

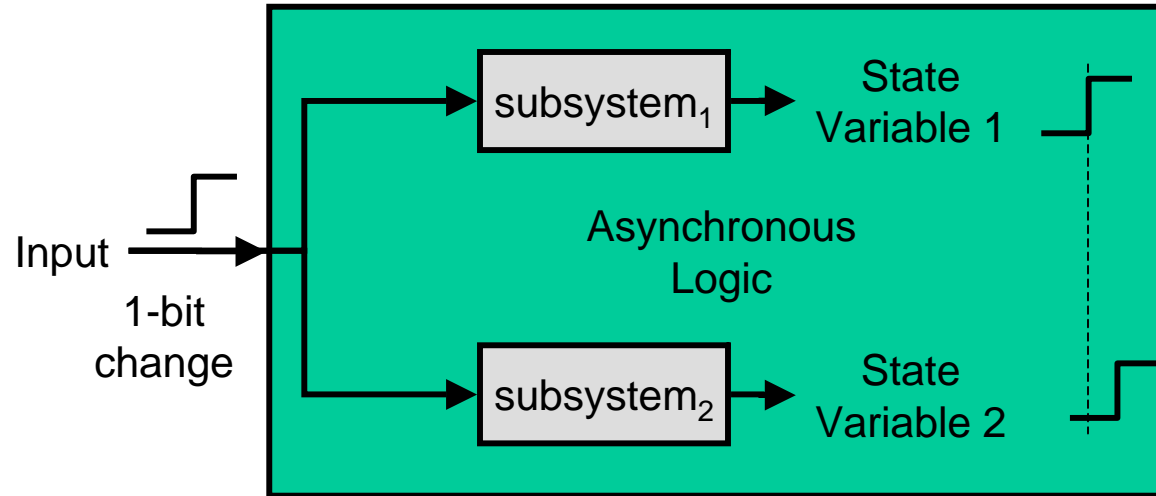
$y_1y_2 \backslash x$	0	1
00	00	01
01	11	01
11	11	10
10	00	10

$y_1y_2 \backslash x$	0	1
A	A	B
B	C	B
C	C	D
D	A	D

Primitive flow table:

1 stable state per row

Back to the Races



- Which occurs first?
- What effect does the order of change have on system operation?

Race Conditions in Asynchronous Circuits

- Consider Problem 9.2:

x_1x_2 :	00	01	11	10
00	00	11	01	00
y_1y_2 01	00	11	11	01
11	00	11	11	01
10	00	11	10	00

- System is in state 00 with input 00, a stable condition.

Race Conditions in Asynchronous Circuits

- Consider Problem 9.2:

x_1x_2 :	00	01	11	10
00	00	11	01	00
y_1y_2 01	00	11	11	01
11	00	11	11	01
10	00	11	10	00

- System is in state 00 with input 00, a stable condition.
- Input is changed to 01, which should lead to state 11
 - But: which variable changes first: y_1 or y_2 ?
 - And does it make any difference to the resulting state?

Race Conditions in Asynchronous Circuits

- Consider Problem 9.2:

x_1x_2 :	00	01	11	10
00	00	11	01	00
y_1y_2 01	00	11	11	01
11	00	11	11	01
10	00	11	10	00

- System is in state 00 with input 00, a stable condition.
- Input is changed to 01, which should lead to state 11
 - But: which variable changes first: y_1 or y_2 ?
 - And does it make any difference to the resulting state?
 - » In this case, if either variable changes before the other, the system may briefly visit state 01 or 10, but the end result is the same – the system's final state is 11
- This is a **noncritical** race

Race Conditions in Asynchronous Circuits

- Consider this transition table:

$x_1x_2:$	0	1
00	00	11
y_1y_2 01		01
11		11
10		10

- System is in state 00 with input 0, a stable condition.
- Input is changed to 1, which should lead to state 11
 - But: which variable changes first: y_1 or y_2 ?
 - And does it make any difference to the resulting state?

Race Conditions in Asynchronous Circuits

- Consider this transition table:

$x_1x_2:$	0	1
00	00	11
y_1y_2 01		01
11		11
10		10

- System is in state 00 with input 0, a stable condition.
- Input is changed to 1, which should lead to state 11
 - But: which variable changes first: y_1 or y_2 ?
 - And does it make any difference to the resulting state?
 - If y_1 changes first: $0\ 0 \rightarrow 1\ 0 \rightarrow \cancel{1\ 1}$

Race Conditions in Asynchronous Circuits

- Consider this transition table:

$x_1x_2:$	0	1
00	00	11
y_1y_2 01		01
11		11
10		10

- System is in state 00 with input 0, a stable condition.
- Input is changed to 1, which should lead to state 11
 - But: which variable changes first: y_1 or y_2 ?
 - And does it make any difference to the resulting state?
 - If y_1 changes first: $0\ 0 \rightarrow 1\ 0 \rightarrow \cancel{1\ 1}$
 - If y_2 changes first: $0\ 0 \rightarrow 0\ 1 \rightarrow \cancel{1\ 1}$

Race Conditions in Asynchronous Circuits

- Consider this transition table:

x_1x_2 :	0	1
00	00	11
y_1y_2 01		01
11		11
10		10

- System is in state 00 with input 0, a stable condition.
- Input is changed to 1, which should lead to state 11
 - But: which variable changes first: y_1 or y_2 ?
 - And does it make any difference to the resulting state?
 - If y_1 changes first: $0\ 0 \rightarrow 1\ 0 \rightarrow \cancel{1\ 1}$
 - If y_2 changes first: $0\ 0 \rightarrow 0\ 1 \rightarrow \cancel{1\ 1}$
 - If y_1 and y_2 change “simultaneously,” system reaches correct state
- This is a **critical** race

Cycles and Races

- Avoid critical races by directing system through a cycle:

x_1x_2 :	0	1
00	00	11
y_1y_2 01		01
11		11
10		10

Multiple possible stable states between 00 and 11 create the problem

$00 \rightarrow \{01, 10, 11\} ?$

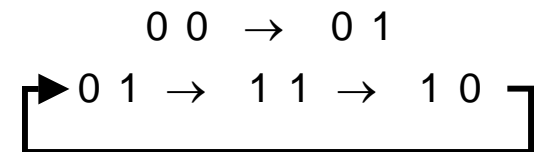
x_1x_2 :	0	1
00	00	01
y_1y_2 01		11
11		11
10		10

Intermediate state is unstable, forcing transition to desired state

$00 \rightarrow 01 \rightarrow 11$

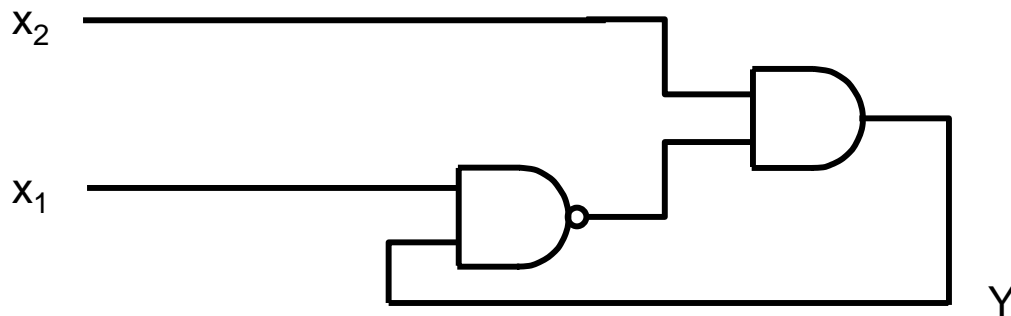
x_1x_2 :	0	1
00	00	01
y_1y_2 01		11
11		10
10		01

But there must be a stable state, or system is unstable



Stability Issues in Asynchronous Circuits

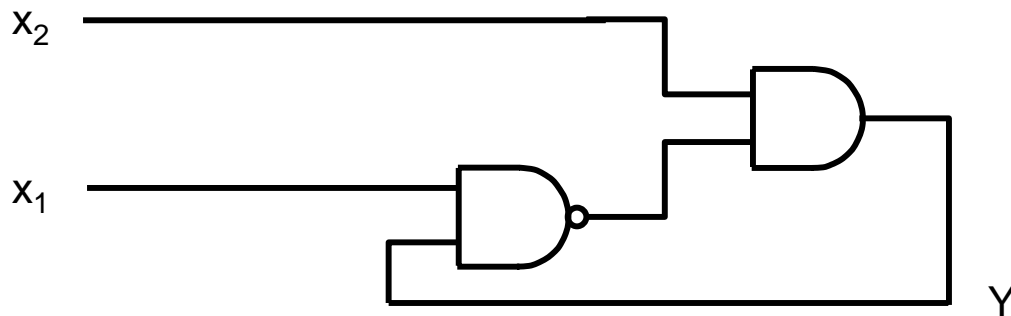
- Consider:



$$Y = ((x'_1 x_2)'(x_2 y_1)')' = x'_1 x_2 + x_2 y'$$

Stability Issues in Asynchronous Circuits

- Consider:



$$Y = ((x'_1 x_2)'(x_2 y_1)')' = x'_1 x_2 + x_2 y'$$

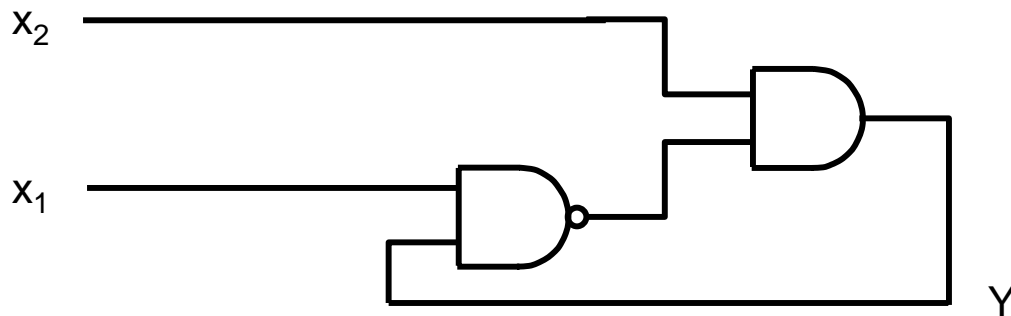
- Transition table:

$x_1 x_2$:	00	01	11	10
0	0	1	1	0
y 1	0	1	0	0

Y

Stability Issues in Asynchronous Circuits

- Consider:



$$Y = ((x'_1 x_2)'(x_2 y_1)')' = x'_1 x_2 + x_2 y'$$

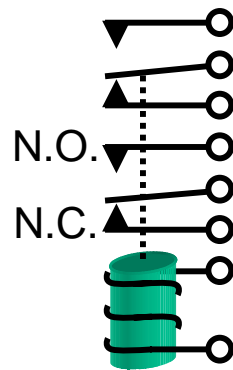
- Transition table:

$x_1 x_2$:	00	01	11	10
0	0	1	1	0
y 1	0	1	0	0

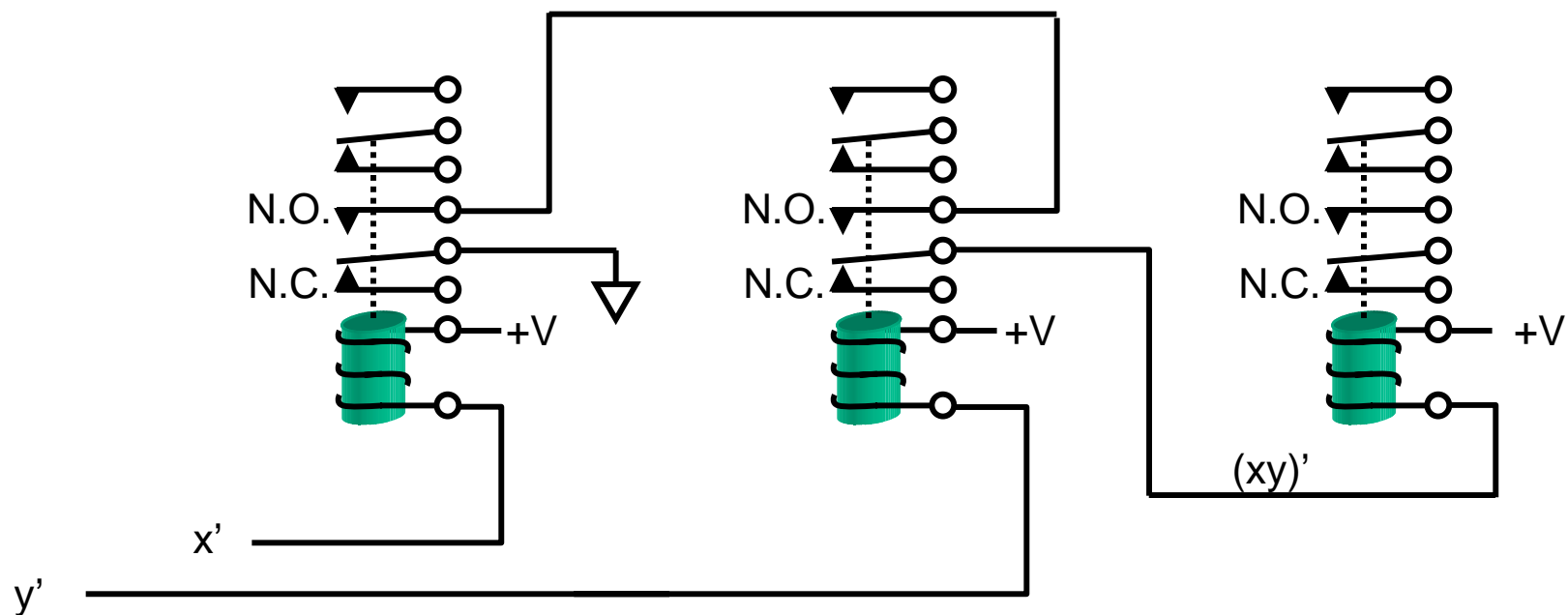
Y

There are no stable states for input 11

“Early” Asynchronous Sequential Logic



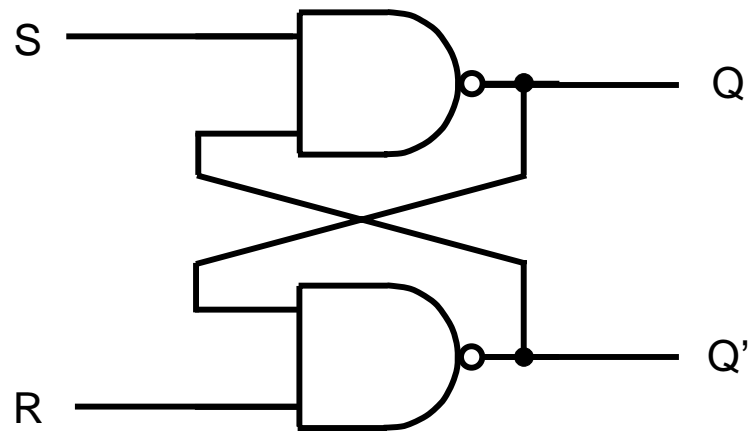
“Early” Asynchronous Sequential Logic



- Some not-so-early asynchronous logic devices:
 - Mechanical pinball machines
 - Automatic transmissions (using fluid and hydraulic relays)

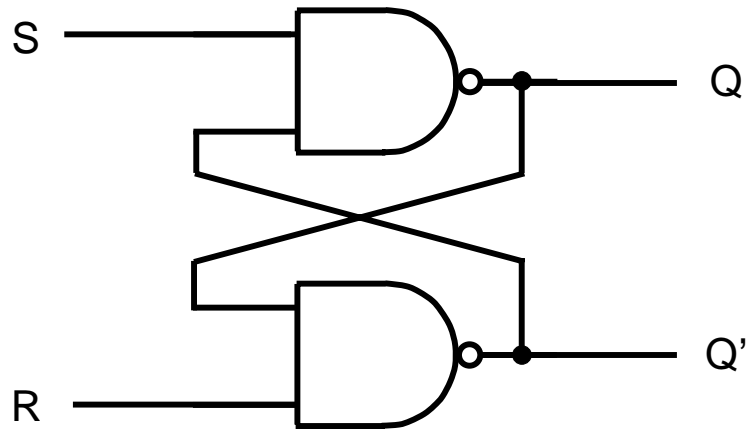
Asynchronous Logic With Latches

- Previously seen circuit

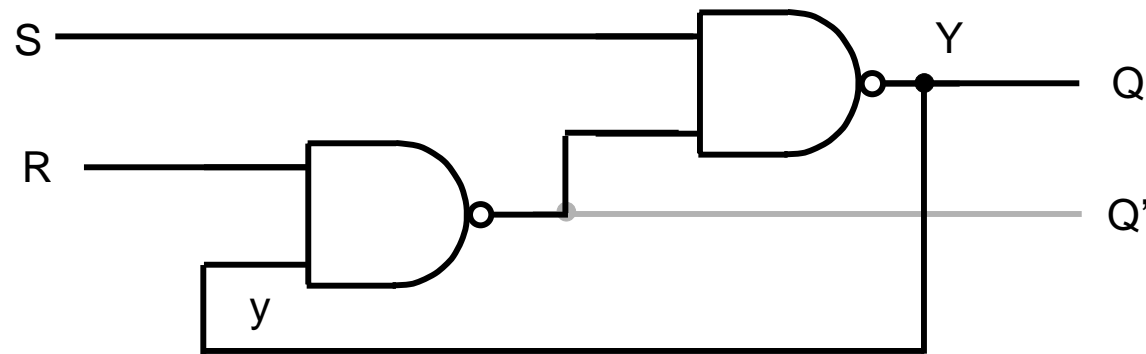


Asynchronous Logic With Latches

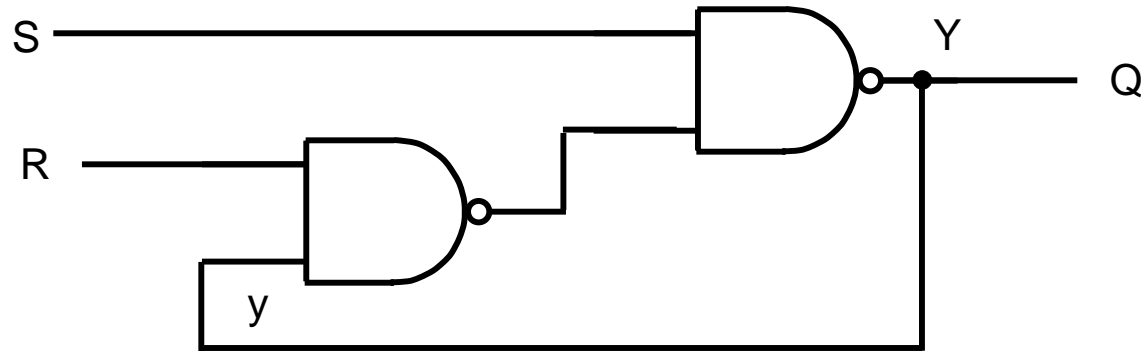
- Previously seen circuit



- With explicit feedback



Asynchronous Logic With Latches



- Transition table:

SR:	00	01	11	10
0	1	1	0	0
y 1	1	1	1	0

Y

$$Y = S' + Ry \text{ if } (S+R)=1$$

Summary

- Fundamental concepts of digital systems (Mano Chapter 1)
- Binary codes, number systems, and arithmetic (Ch 1)
- Boolean algebra (Ch 2)
- Simplification of switching equations (Ch 3)
- Digital device characteristics (e.g., TTL, CMOS)/design considerations (Ch 10)
- Combinatoric logical design including LSI implementation (Chapter 4)
- Flip-flops and state memory elements (Ch 5)
- Sequential logic analysis and design (Ch 5)
- Counters, shift register circuits (Ch 6)
- Hazards, Races, and time related issues in digital design (Ch 9)
- **Synchronous vs. asynchronous design (Ch 9)**
- Memory and Programmable logic (Ch 7)
- Minimization of sequential systems
- Introduction to Finite Automata

Homework 13 – due in Class 15

- Show all work
- Problems 9-6, 9-9