

Instructor: Robert Dick
 Office: L477 Tech
 Email: dickrp@northwestern.edu
 Phone: 847-467-2298

TA: Neal Oza
 Office: Tech. Inst. L375
 Phone: 847-467-0033
 Email: neoza@u.northwestern.edu

TT: David Bild
 Office: Tech. Inst. L470
 Phone: 847-401-2083
 Email: d-bild@northwestern.edu



NORTHWESTERN
UNIVERSITY

Word description to state diagram

- We can enumerate the inputs on which an apple should be released

$$ddd + ddq + dq + qd + qq$$

$$d(dd + dq + q) + q(d + q)$$

$$d(d(d + q) + q) + q(d + q)$$

For $d, i = 0$, for $q, i = 1$

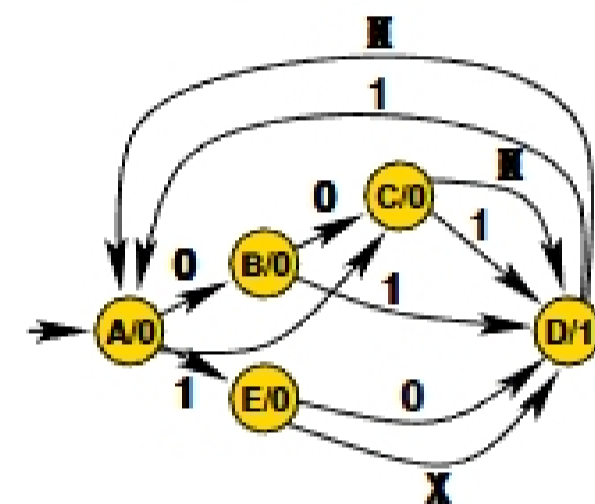
$$0(0(0 + 1) + 1) + 1(0 + 1)$$

Word description to state diagram

- Design a vending machine controller that will release (output signal r) an apple as soon as 30¢ have been inserted
- The machine's sensors will clock your controller when an event occurs. The machine accepts only dimes (input signal d) and quarters (input signal q) and does not give change
- When an apple is removed from the open machine, it indicates this by clocking the controller with an input of d
- The sensors use only a single bit to communicate with the controller

Word description to state diagram

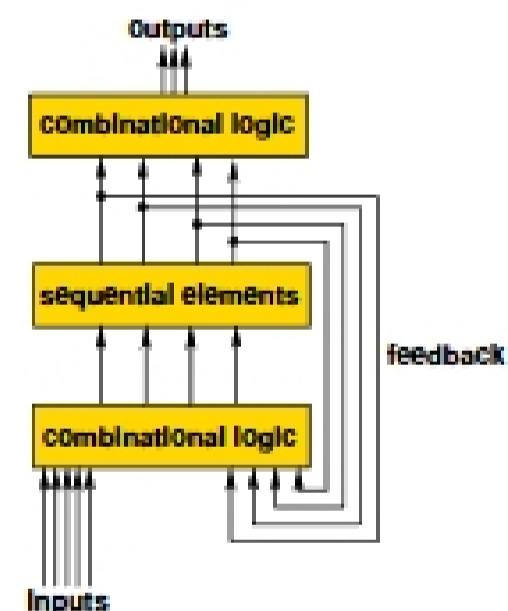
$$0(0(0 + 1) + 1) + 1(0 + 1)$$



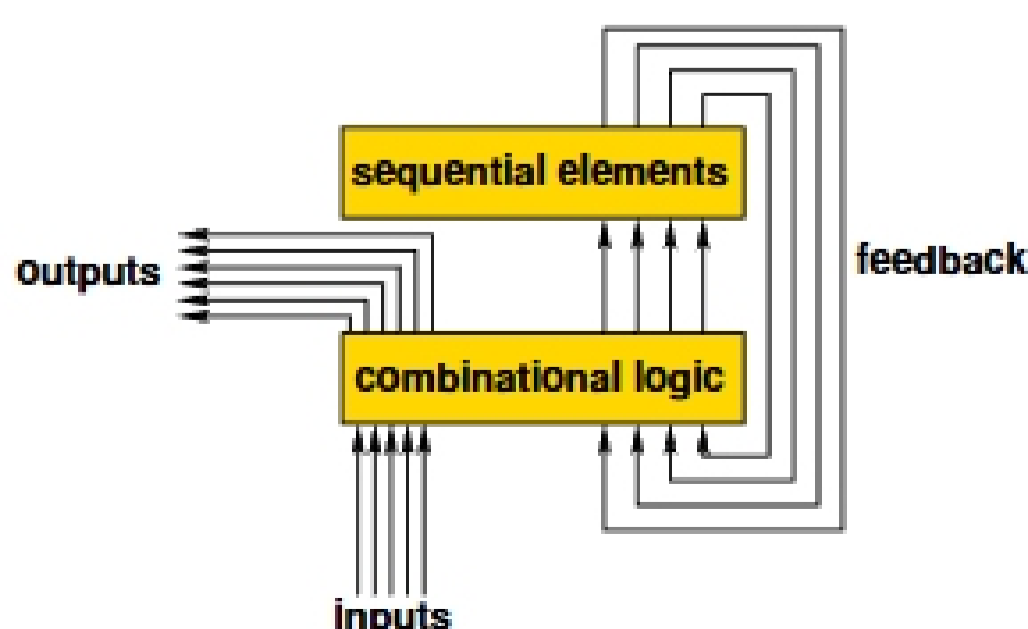
State diagram to state table

Current state	next state		output (r)
	$i=0$	$i=1$	
A	B	E	0
B	C	D	0
C	D	D	0
D	A	A	1
E	D	D	0

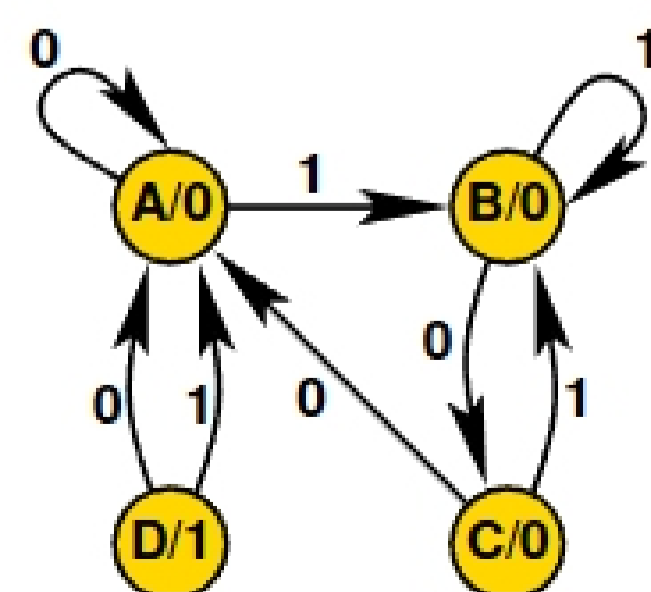
Moore block diagram



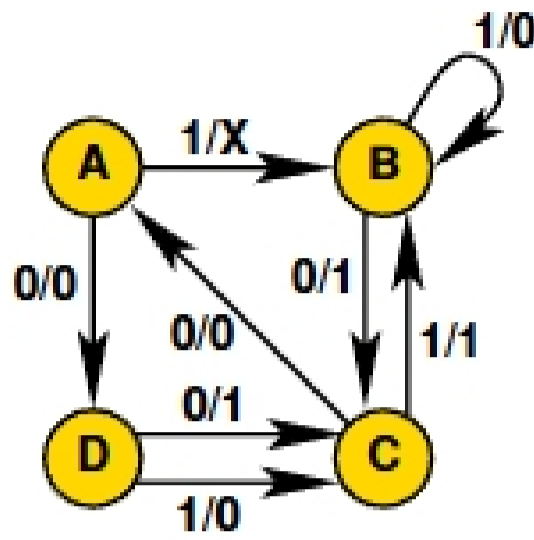
Mealy block diagram



Moore FSMs



Mealy FSMs

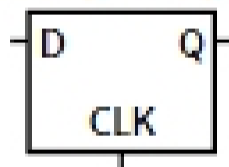


FSM design summary

- Specify requirements in natural form
- Manually derive state diagram
 - Automatic way to go from English to FSM, however more theory required
 - Can minimize state count, however, more theory also required
 - See me if you want more information on this, or take a compilers course and a graduate-level switching theory course, or take my ECE 303
- Assign values to states to minimize logic complexity
- Optimize implementation of state and output functions

Review: Clocking conventions

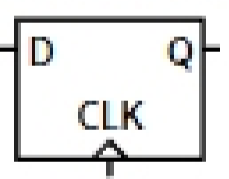
Active-high transparent



Active-low transparent



Positive (rising) edge



Negative (falling) edge



Latch and flip-flop equations

JK $Q^+ = J\bar{Q} + \bar{K}Q$

T $Q^+ = T \oplus Q$

Mealy tabular form

s	s ⁺ /q	
	0	1
A	D/0	B/X
B	C/1	B/0
C	A/0	B/1
D	C/1	C/0

Back to latches

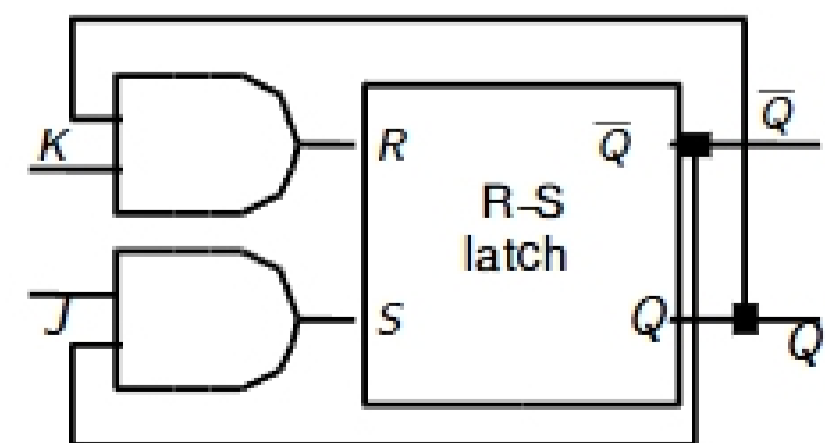
- Latches: Level sensitive
- Flip-flops: Edge-triggered

Latch and flip-flop equations

RS $Q^+ = S + \bar{R}Q$

D $Q^+ = D$

JK latch



Use output feedback to ensure that $RS \neq 11$
 $Q^+ = Q\bar{K} + \bar{Q}J$

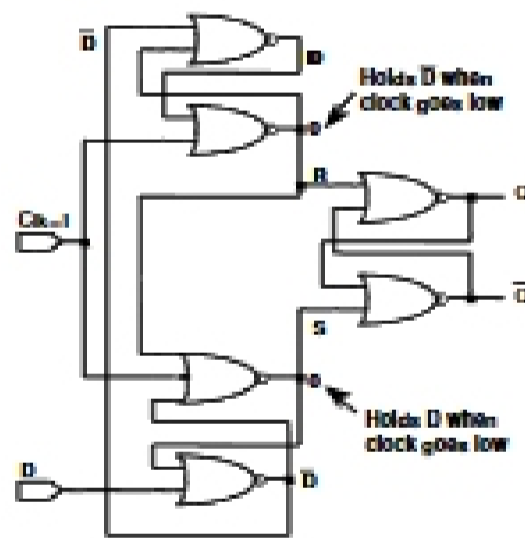
JK latch

J	K	Q	Q ⁺	
0	0	0	0	hold
0	0	1	1	
0	1	0	0	reset
0	1	1	0	
1	0	0	1	set
1	0	1	1	
1	1	0	1	toggle
1	1	1	0	

Falling edge-triggered D flip-flop

- Use two stages of latches
- When clock is high
 - First stage samples input w.o. changing second stage
 - Second stage holds value
- When clock goes low
 - First stage holds value and sets or resets second stage
 - Second stage transmits first stage
- $Q^+ = D$
- One of the most commonly used flip-flops

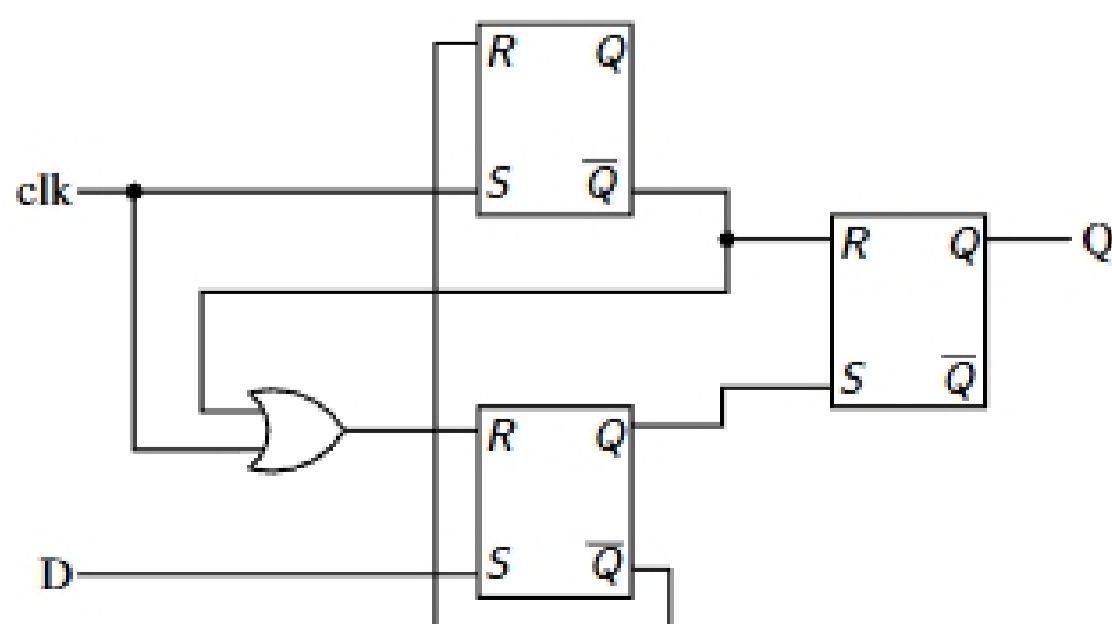
Falling edge-triggered D flip-flop



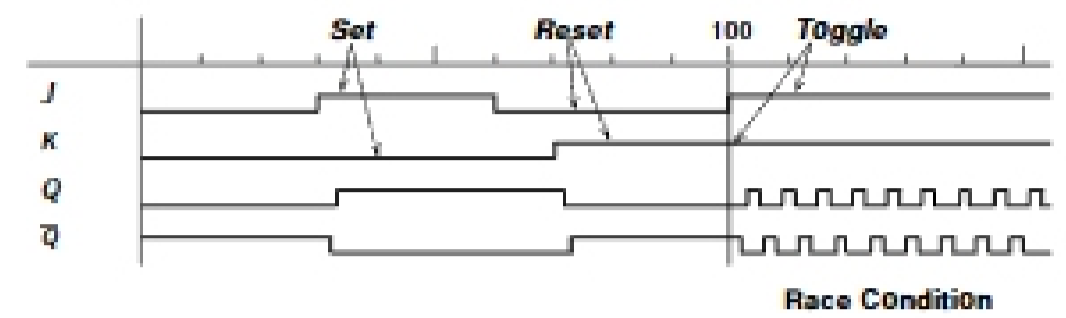
Clock switching

Inputs sampled on falling edge, outputs change after falling edge

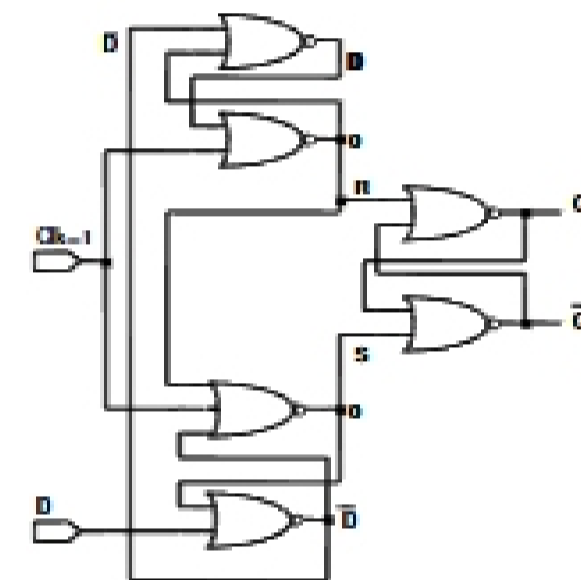
Another view of an edge-triggered DFF



JK race

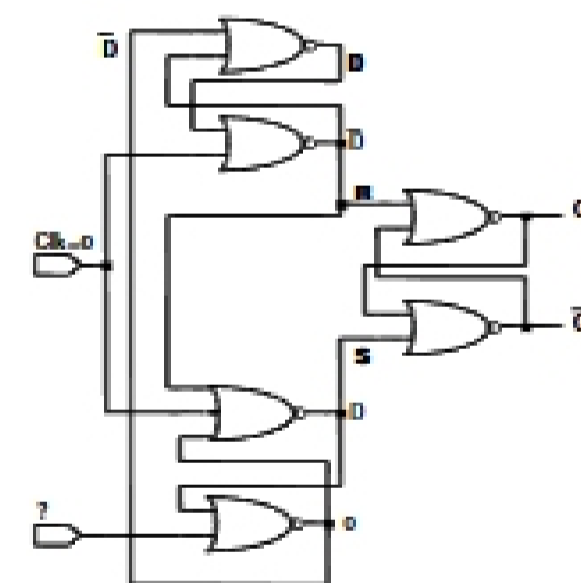


Falling edge-triggered D flip-flop



Clock high

Falling edge-triggered D flip-flop



Clock low

Edge triggered timing

