Lecture 8: Flip-flops

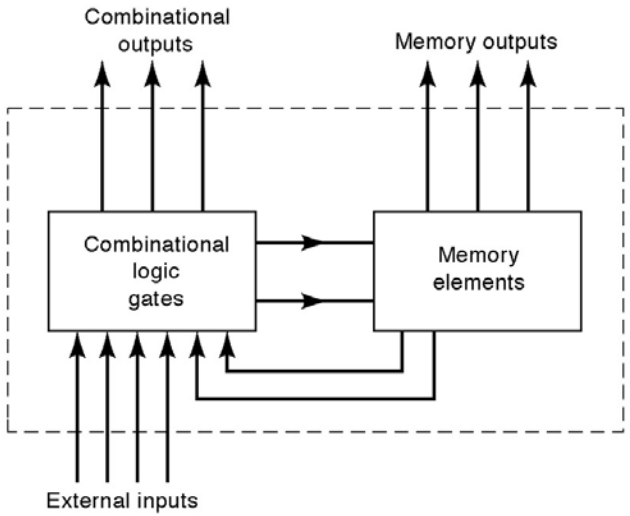
Professor Peter Cheung
Department of EEE, Imperial College London

(Floyd 7.1-7.4)
(Tocci 5.1-5.9)

Points Addressed in this Lecture

- Properties of synchronous and asynchronous sequential circuits
- Overview of flip-flops and latches

General digital system diagram



Properties of Sequential Circuits

- So far we have seen Combinational Logic
  - the output(s) depends only on the current values of the input variables
- Here we will look at Sequential Logic circuits
  - the output(s) can depend on present and also past values of the input and the output variables
- Sequential circuits exist in one of a defined number of states at any one time
  - they move "sequentially" through a defined sequence of transitions from one state to the next
  - The output variables are used to describe the state of a sequential circuit either directly or by deriving state variables from them

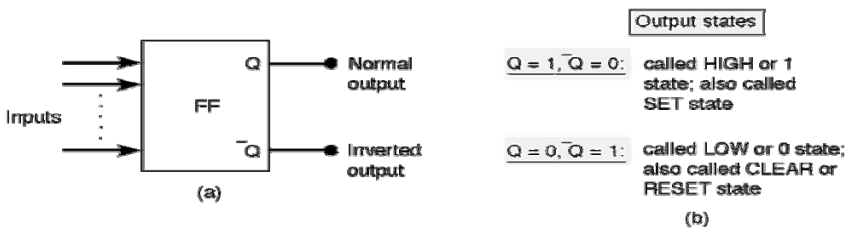
Synchronous and Asynchronous Sequential Logic

- Synchronous
  - the timing of all state transitions is controlled by a common clock
  - changes in all variables occur simultaneously
- Asynchronous
  - state transitions occur independently of any clock and normally dependent on the timing of transitions in the input variables
  - changes in more than one output do not necessarily occur simultaneously
- Clock
  - A clock signal is a square wave of fixed frequency
  - Often, transitions will occur on one of the edges of clock pulses
    - i.e. the rising edge or the falling edge

Flip-Flops

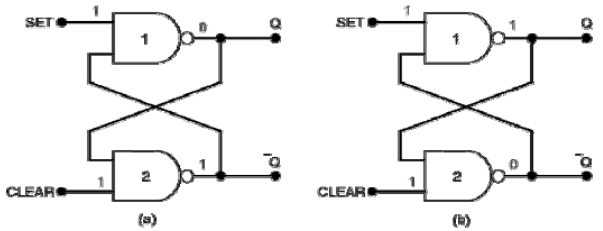
- Flip-flops are the fundamental element of sequential circuits
  - bistable
  - (gates are the fundamental element for combinational circuits)
- Flip-flops are essentially 1-bit storage devices
  - outputs can be set to store either 0 or 1 depending on the inputs
  - even when the inputs are de-asserted, the outputs retain their prescribed value
- Flip-flops have (normally) 2 complimentary outputs
  - Q and $\overline{Q}$
- Three main types of flip-flop
  - R-S J-K D-type

Flip-Flop



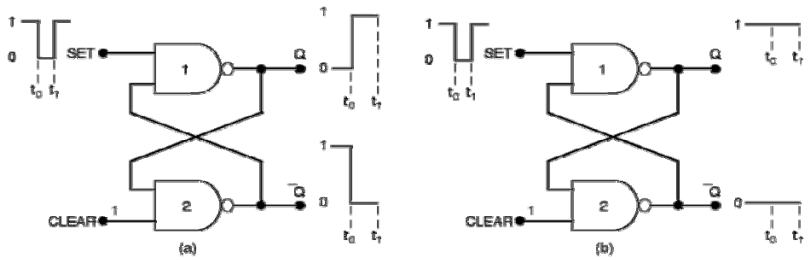
FF = latch = bistable circuit

NAND Gate Latch



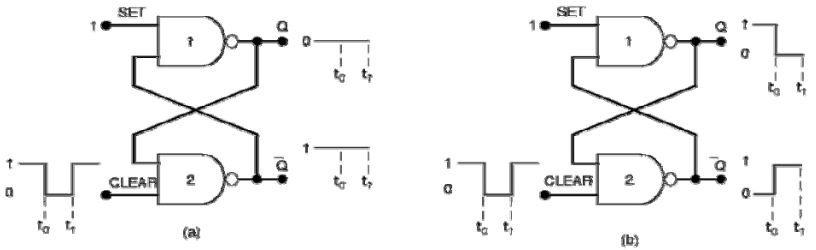
A NAND latch has two possible resting states when SET = CLEAR = 1.

NAND Gate Latch (cont.)



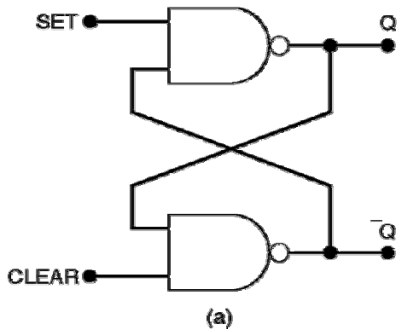
Negative Pulse on SET input put the latch in a HIGH (SET) state

NAND Gate Latch (cont.)



Negative Pulse on CLEAR input put the latch in a LOW (Clear or RESET) state

NAND Gate Latch (cont.)



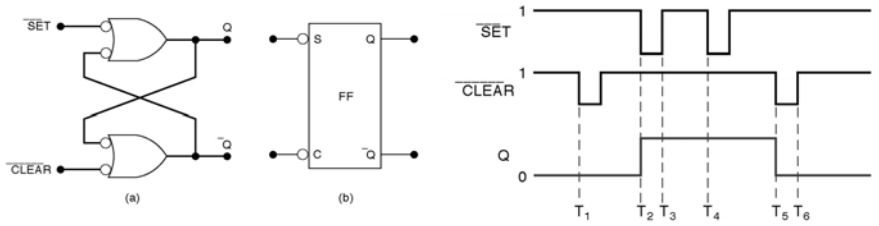
Set	Clear	Output
1	1	No change
0	1	Q = 1
1	0	Q = 0
0	0	Invalid*

*produces $Q = \overline{Q} = 1$

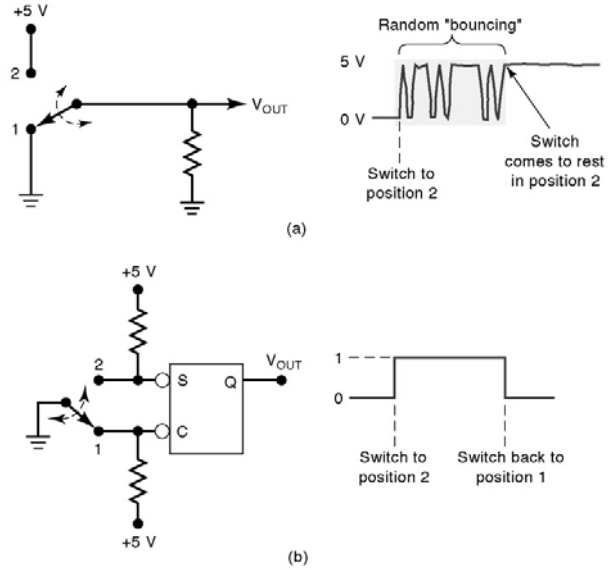
(b)

Truth table for the NAND Set-Clear (Set-Reset or SR) Latch

Alternative representation of SR Latch

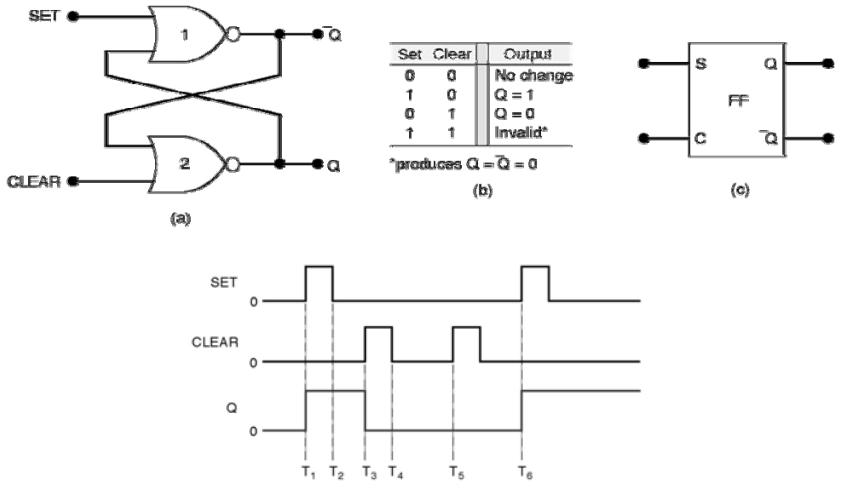


SR Latch to deglitch a switch



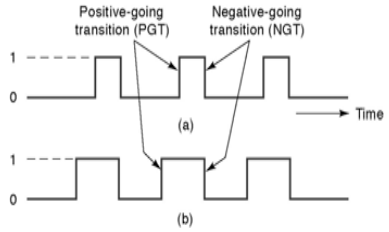
NOR gate Latch

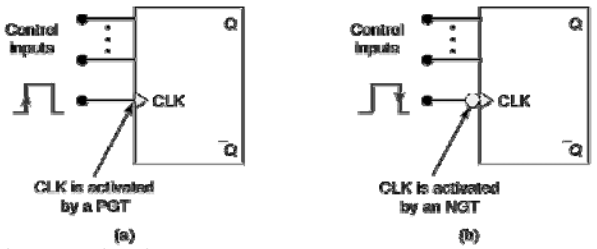
- **Made of two cross-coupled NOR gates**

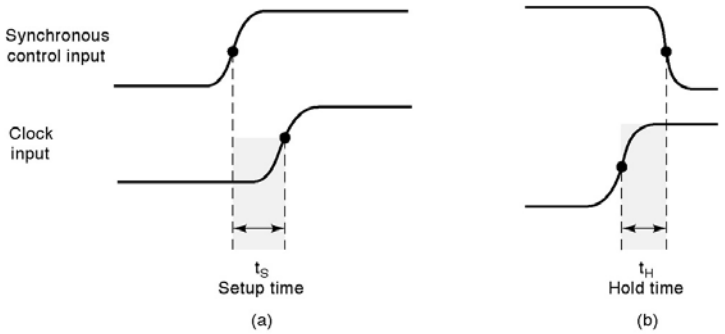


Clock Signals and Clocked FFs

- **Digital systems can operate**
  - **Asynchronously: output can change state whenever inputs change**
  - **Synchronously: output only change state at clock transitions (edges)**
- **Clock signal**
  - **Outputs change state at the transition (edge) of the input clock**
  - **Positive-going transitions (PGT)**
  - **Negative-going transitions (NGT)**

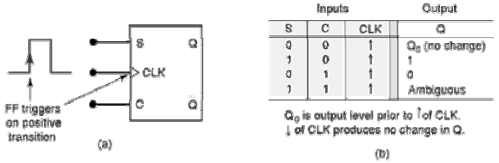




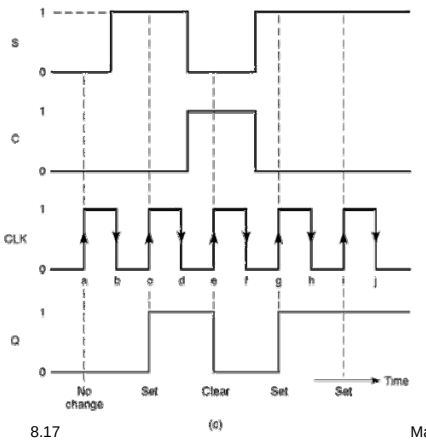


Control inputs must be held stable for (a) a time t_S prior to active clock transition and for (b) a time t_H after the active block transition.

Clocked S-C FF

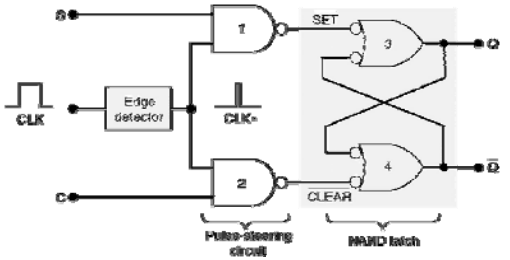


(a) Clocked S-C FF that responds only to the positive-going edge of a clock pulse;
(b) truth table;
(c) Typical waveforms.

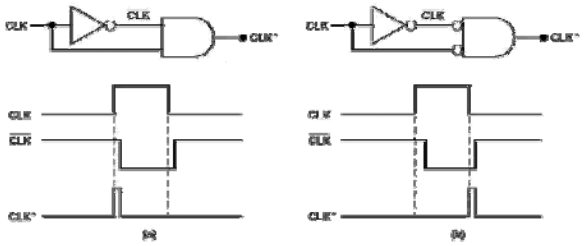


Internal Circuitry of S-C FF

Simplified version of the internal circuitry for an edge-triggered S-C FF

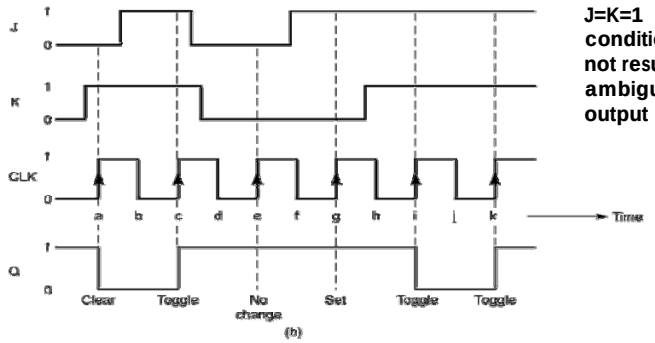


Implementation of edge-detector circuits used in edge-triggered FFs: (a) PGT; (b) NGT. The duration of the *CLK pulses is typically 2-5 *ns***



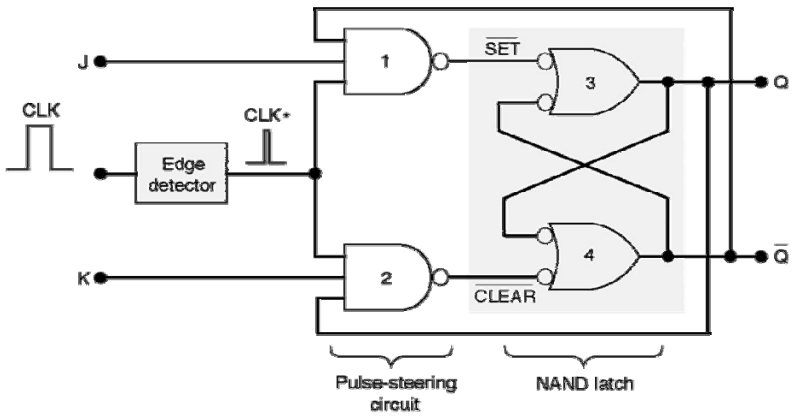
Clocked J-K FF

(a) Clocked J-K flip-flop that responds only to the positive edge of the clock; (b) waveforms.



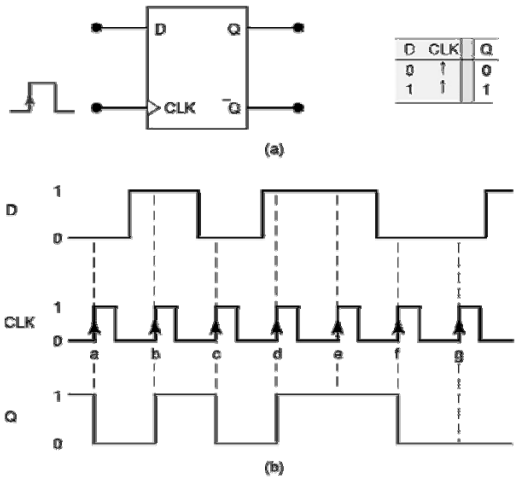
J=K=1 condition does not result in an ambiguous output

Internal circuitry of edge-triggered J-K flip-flop

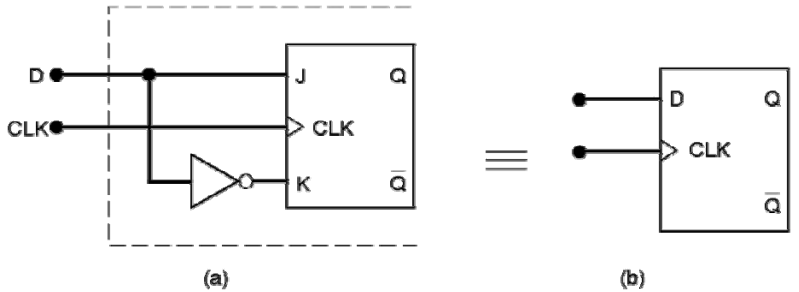


Clocked D Flip-Flop

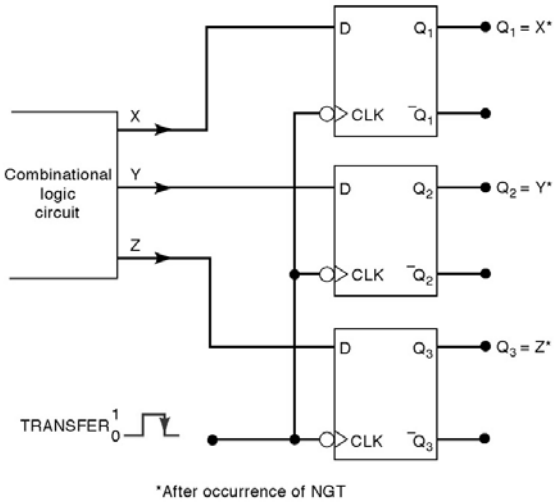
D FF that triggers only on positive-going transitions; (b) waveforms.



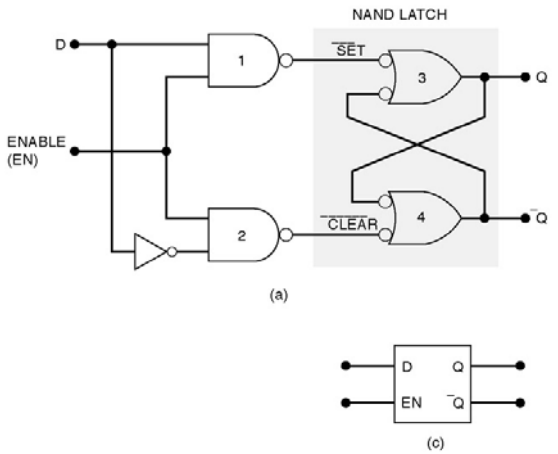
Clocked D Flip-Flop from J-K Flip-Flop



Parallel Data Transfer using D-FF



D Latch (transparent latch)

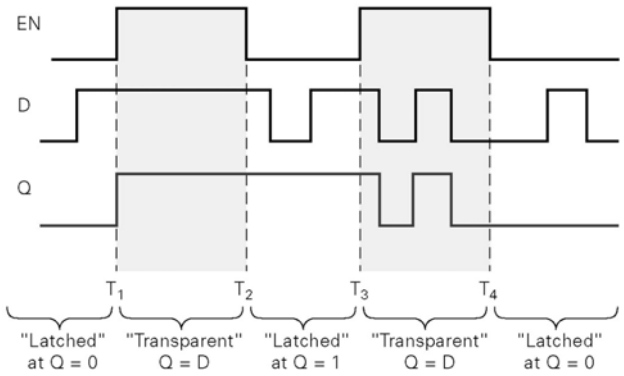


Inputs		Output
EN	D	Q
0	X	Q_0 (no change)
1	0	0
1	1	1

"X" indicates "don't care"
Q_0 is state Q just prior to EN going LOW

(b)

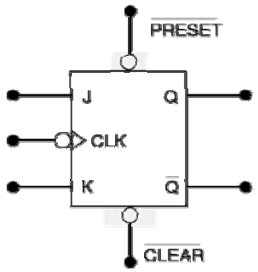
Transparent Latch Timing



Asynchronous Inputs to FF

The S, C, J, K, and D inputs is called synchronous inputs because their effects on the output are synchronized with the *CLK* input.

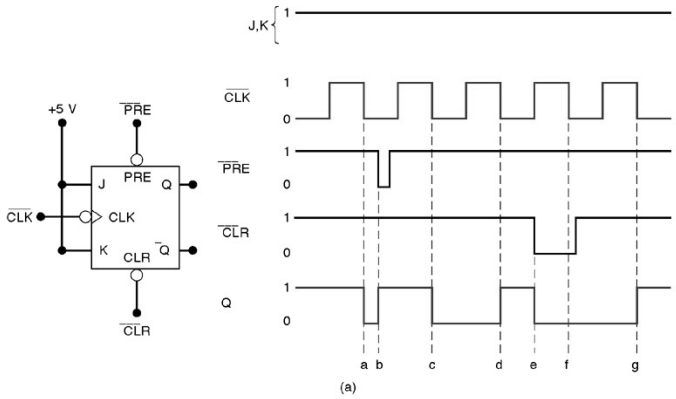
Asynchronous inputs (override inputs) operate independently of the synchronous inputs and clock and can be used to set the FF to 1/0 states *at any time.*



PRESET	CLEAR	FF response
1	1	Clocked operation*
0	1	Q = 1 (regardless of CLK)
1	0	Q = 0 (regardless of CLK)
0	0	Not used

*Q will respond to J, K, and CLK

Asynchronous Inputs cont.



(a)

Point	Operation
a	Synchronous toggle on NGT of CLK
b	Asynchronous set on PRE = 0
c	Synchronous toggle
d	Synchronous toggle
e	Asynchronous clear on CLR = 0
f	CLR over-rides the NGT of CLK
g	Synchronous toggle

(b)

Points Addressed in the next part of the Lecture

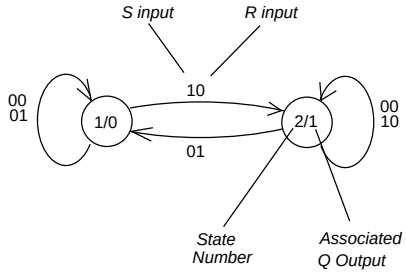
- Introduction to Moore model and Mealy model state diagrams
- State diagrams and state tables of flip-flops
- Timing parameters of flip-flops

State Diagrams

- There are two ways to draw state diagrams:
  - **Moore** model and **Mealy** model
  - we will look first at the Moore model using the Reset-Set (RS) flip-flop as an example
- A Moore model state diagram shows
  - a circle for each of the various "states" a circuit can be in
    - "states" refers to the logic levels around the circuit
  - labels in the circles to show the state number and the associated output for that state
  - an arrow for each possible transition between states
  - labels on the arrows to show the required conditions to transit between states

State Diagram of RS Flip-Flop

- The circuit must be in either state 1 or state 2.



- In state 1
  - Q output = 0
  - an input of S=1, R=0 causes a transition to state 2
  - any other input leaves the circuit in state 1
  - (an input of S=R=1 is not allowed for RS flip-flops)
- In state 2
  - Q output = 1
  - an input of S=0, R=1 causes a transition to state 1
  - any other input leaves the circuit in state 2

State Table of RS-Flip-Flop

- A state table is a tabular form of the state diagram
  - one row for each possible state
- Shows the next state which will be entered for all possible combinations of inputs
- The ordering of the inputs is same as for Karnaugh map

- Example

Present State	Next state inputs: SR			
	00	01	11	10
1	1	1	X	2
2	2	1	X	2

- The circuits is in state 2; the inputs are S=1, R=0. What is the next state?
  - state 2

Assigned State Table of RS-Flip-Flop

- Differs from a State Table by showing the associated outputs not the state numbers
- Example

Present Output	Next output inputs: SR			
	00	01	11	10
0	0	0	X	1
1	1	0	X	1

- The output of the circuits is 1; the inputs are S=1, R=0. What is the next output?
  - 1

Boolean Expression from Assigned State Table

- We continue the example of the RS flip-flop and call the "next output" Q+
- The assigned state table defines the logical relationship between the inputs (S and R) and Q+
  - a Boolean relationship
- Hence we can re-draw the assigned state table as a Karnaugh map

Q\SR	00	01	11	10
0	0	0	X	1
1	1	0	X	1

Q\SR	00	01	11	10
0	0	0	X	1
1	1	0	X	1

- The Boolean expression is then obtained by grouping terms as usual

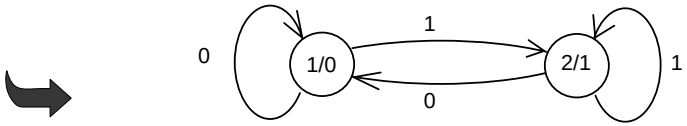
$$Q^+ = Q\overline{R} + S$$

$$\overline{Q^+} = \overline{Q}\overline{S} + R$$

- Such equations are called **characteristic equations**

D-type Implementation

- Moore Model State Diagram



- Assigned State Table

Present Output	Next output	
	0	1
0	0	1
1	0	1

Characteristic Equation

$$Q^+ = D$$

Imperial College
London

Mealy Model State Diagrams

- Similar principles to Moore model but different labelling
- State circles are labelled only with state numbers
- Outputs are written next to inputs on the arrows
- E.g. JK Flip-flop