Digital Circuit Design

Li Bai

Module 6

- Non integer representation
  - Fix-point
  - Float-point

- Multiplication process
  - Bit-wise multiplication
  - Shift operation (represent numbers as sum of multiple of 2^n)
  - Use multiplication IP core (block diagram instantiation)

- Sequential Logic
  - S/R Latch
  - D latch
  - D flip-flop

Fix-point represenation

- m:f represenation

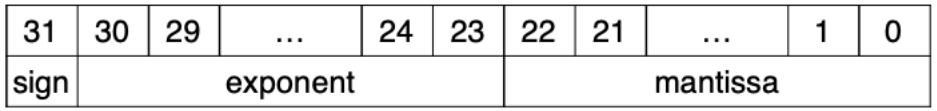
- Example: for an *n-bit* number with *m-bits* "before" the decimal point (the integer component) and *f-bits* for the fraction
  (n = m + f):

Float point representation

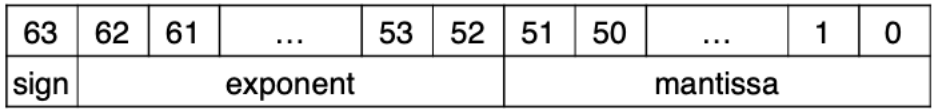
- Mantissa and exponents
- Signed bit
- Mantissa (map to number decimal number 1-9)
- Exponents using excess form
  - 8-bit represent 256
  - excess form (-127, 127) mapped to 8'b0 to 8'b1

IEEE 754-1985 Standard

32-bit floating point number



64-bit floating point number



Example

Normalized Form

Let's illustrate with an example, suppose that the 32-bit pattern is `1 1000 0001 011 0000 0000 0000 0000 0000`, with:

- `S = 1`
- `E = 1000 0001`
- `F = 011 0000 0000 0000 0000 0000`

In the *normalized form*, the actual fraction is normalized with an implicit leading 1 in the form of `1.F`. In this example, the actual fraction is `1.011 0000 0000 0000 0000 0000` or `1.375D`.

The sign bit represents the sign of the number, with `S=0` for positive and `S=1` for negative number. In this example with `S=1`, this is a negative number, i.e., `-1`

In normalized form, the actual exponent is `E-127` (so-called excess-127 or bias-127). This is because we need to represent both positive and negative expo scheme could provide actual exponent of -127 to 128. In this example, `E-127=129-127=2D`.

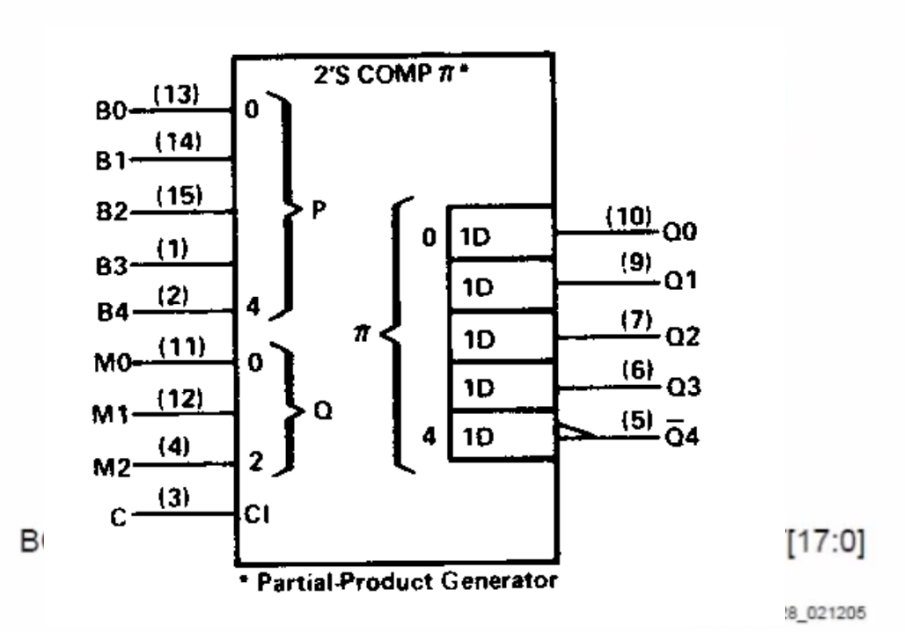
Hence, the number represented is `-1.375×2^2=-5.5D`.

Binary multiplication

- Convert one of the multiplication number into binary, then do the shift <<

- Example: A*10 = A* 8 + A* 2= A<<3 + A<<1

Multiplication IC

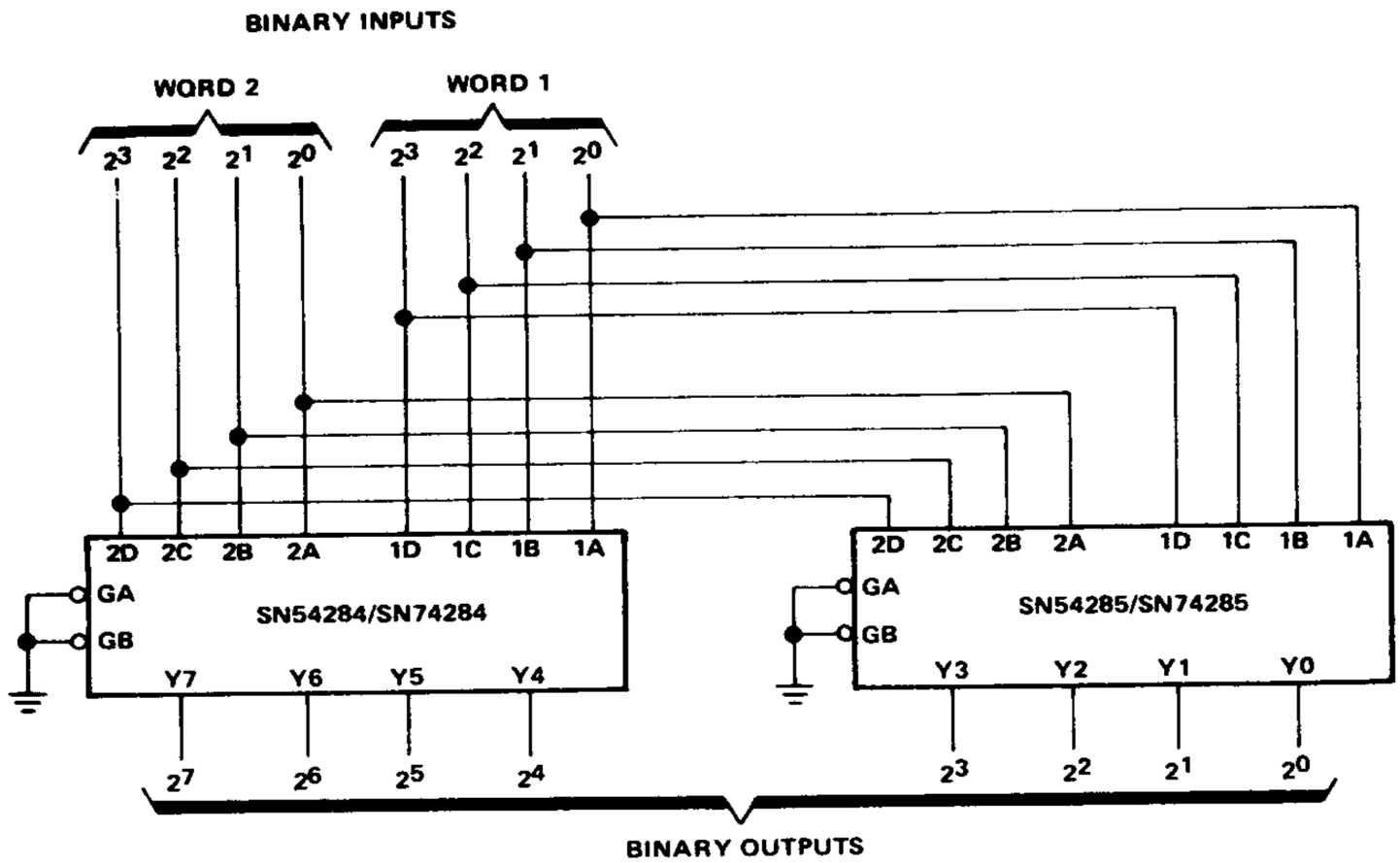


SN54LS261

FUNCTION TABLE

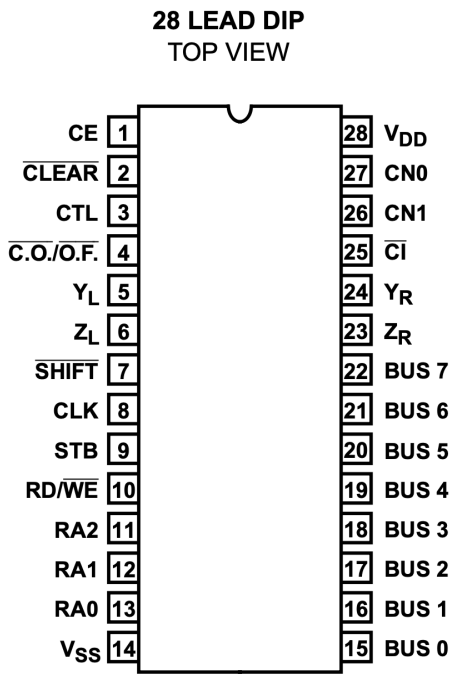
INPUTS				OUTPUTS				
LATCH CONTROL C	MULTIPLIER M2	M1	M0	$\bar{Q}4$	Q3	Q2	Q1	Q0
L	X	X	X	$\bar{Q}4_0$	$Q3_0$	$Q2_0$	$Q1_0$	$Q0_0$
H	L	L	L	H	L	L	L	L
H	L	L	H	$\bar{B}4$	B4	B3	B2	B1
H	L	H	L	$\bar{B}4$	B4	B3	B2	B1
H	L	H	H	$\bar{B}4$	B3	B2	B1	B0
H	H	L	L	B4	$\bar{B}3$	$\bar{B}2$	$\bar{B}1$	$\bar{B}0$
H	H	L	H	B4	$\bar{B}4$	$\bar{B}3$	$\bar{B}2$	$\bar{B}1$
H	H	H	L	B4	$\bar{B}4$	$\bar{B}3$	$\bar{B}2$	$\bar{B}1$
H	H	H	H	H	L	L	L	L

Mutliplication IC

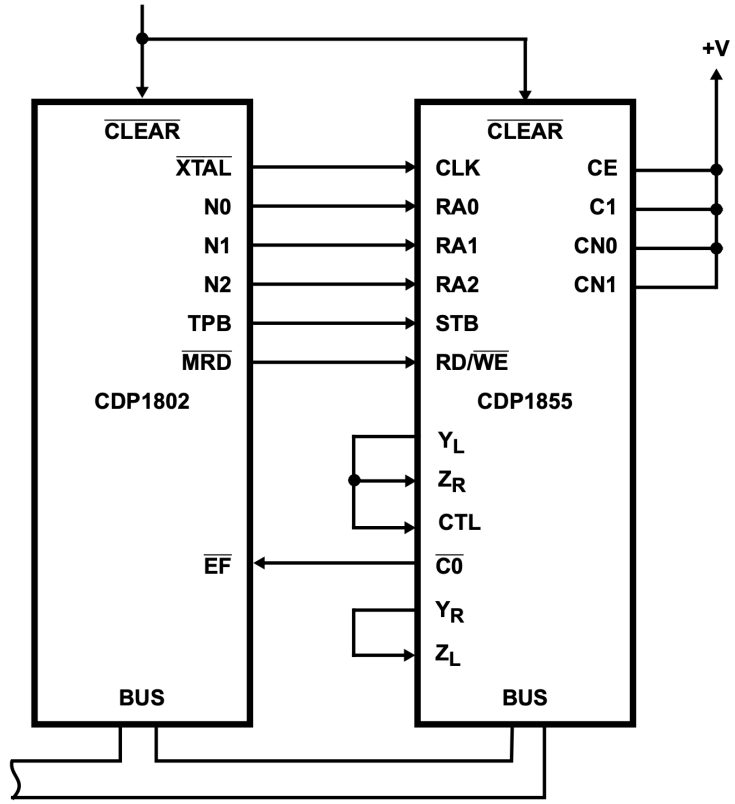


Another one ...

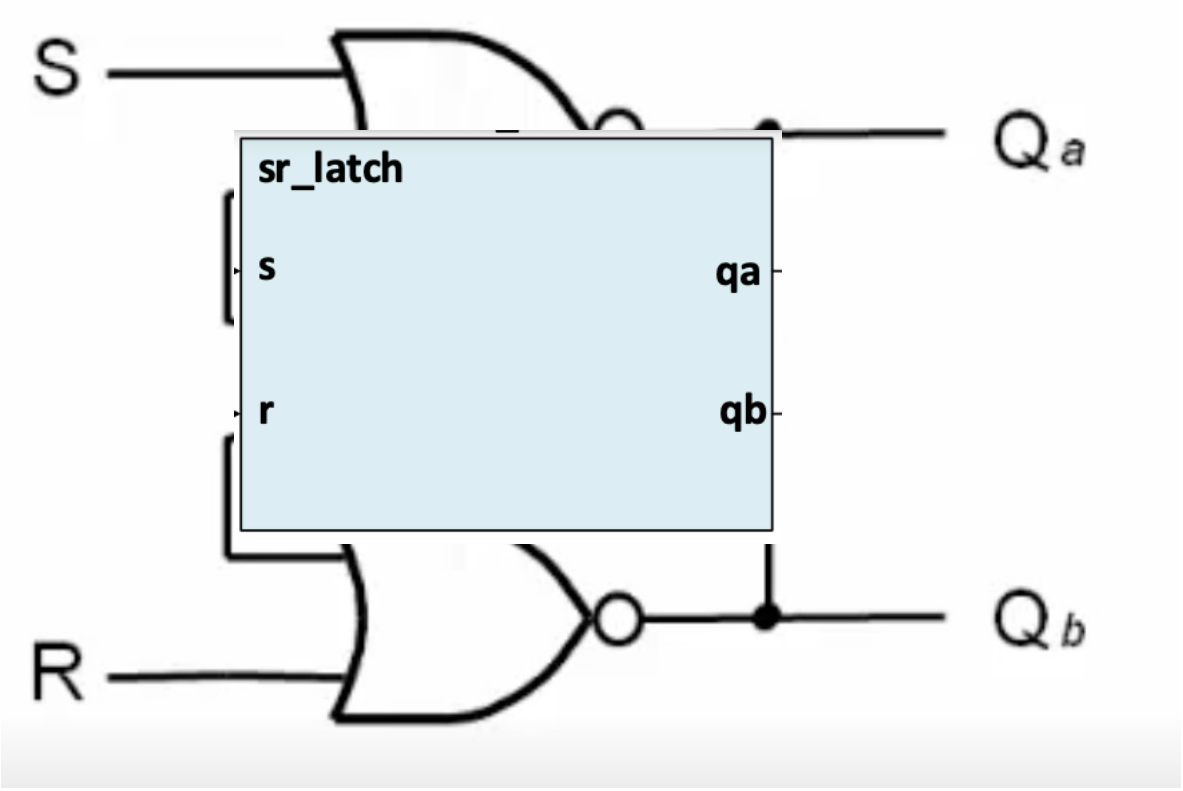
Pinout



Circuit Configuration



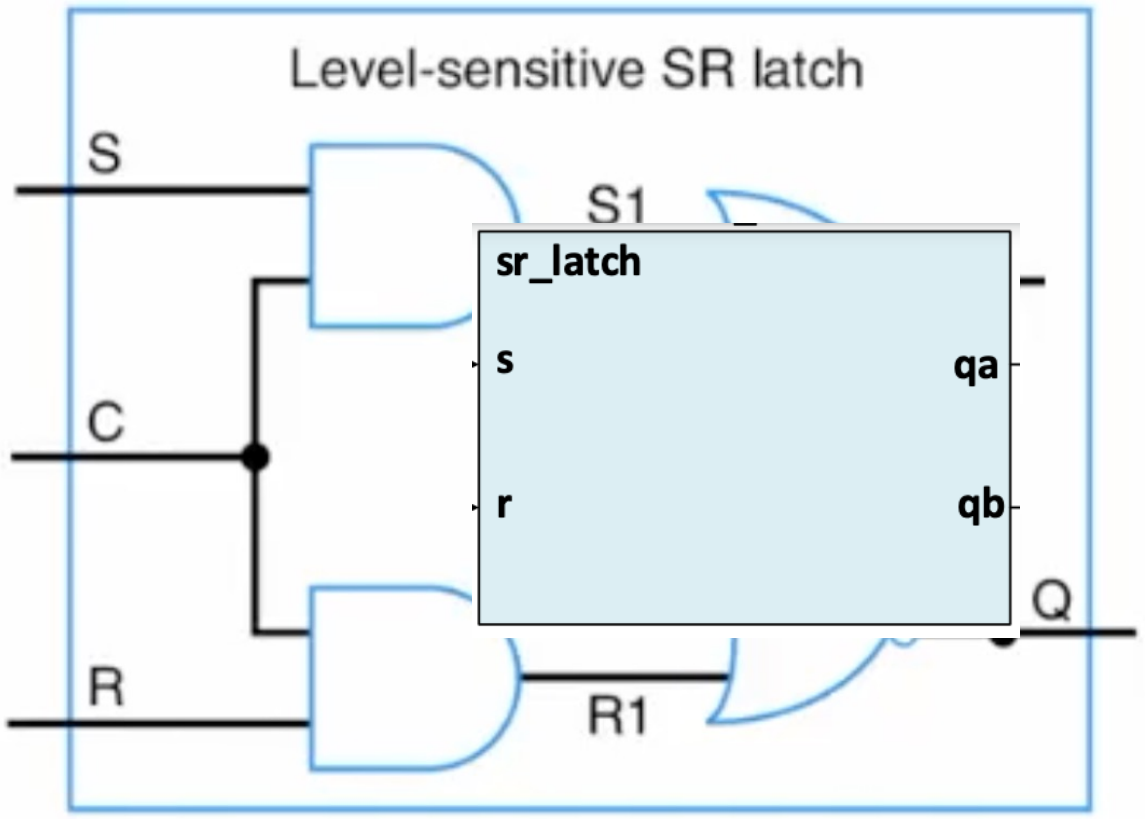
SR Latch



Qa=~Qb

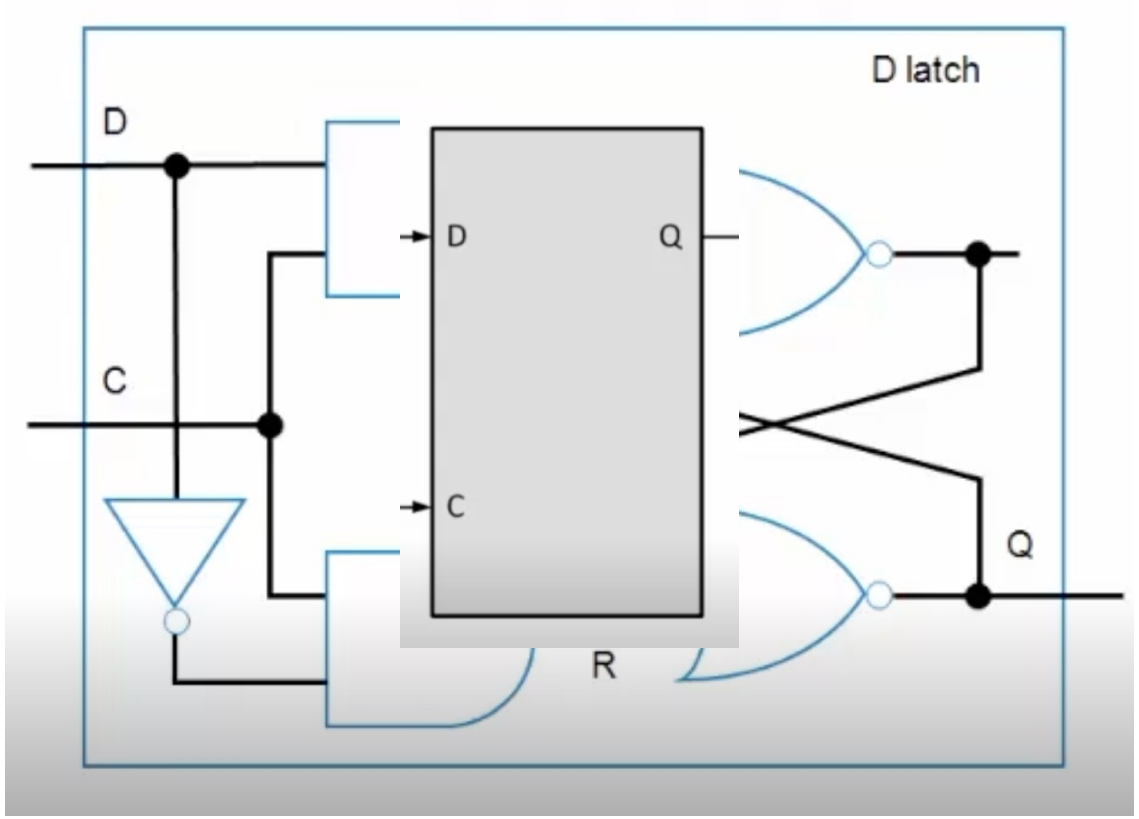
S	R	Qa	Qb	Condition
0	0	0/1	1/0	Hold
0	1	1	0	Reset
1	0	0	1	Set
1	1	0	0	Illegal

Level Sensitive SR Latch



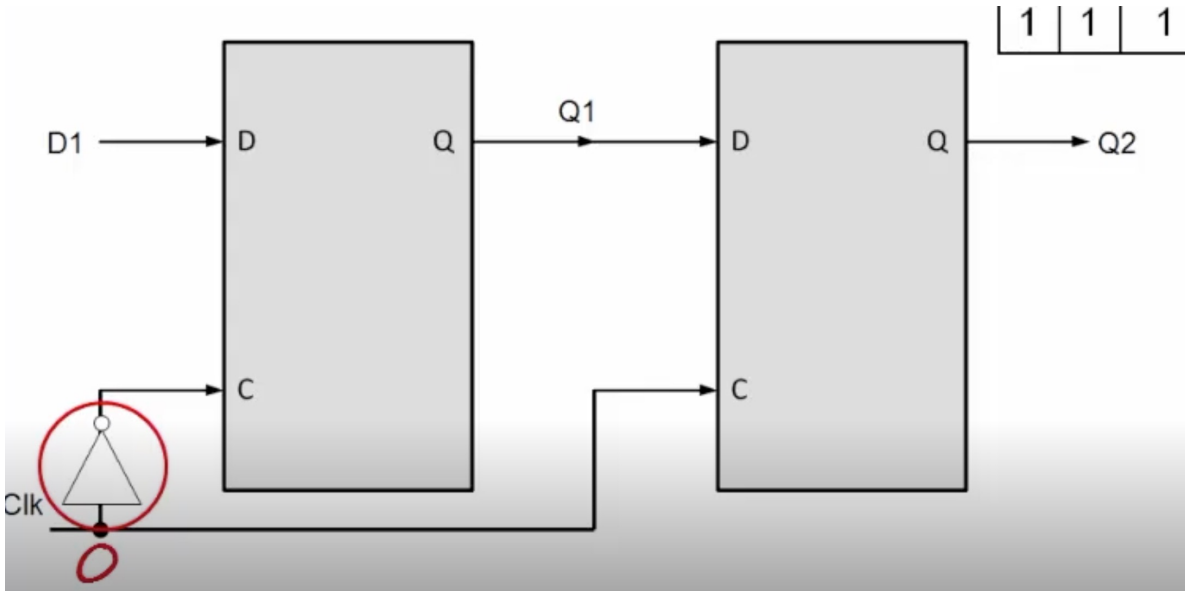
C	S	R	Q	Condition
0	x	x	1/0	Hold
1	0	0	1/0	Hold
1	0	1	0	Reset
1	1	0	1	Set
1	1	1	0	Illegal

Level Sensitive D Latch



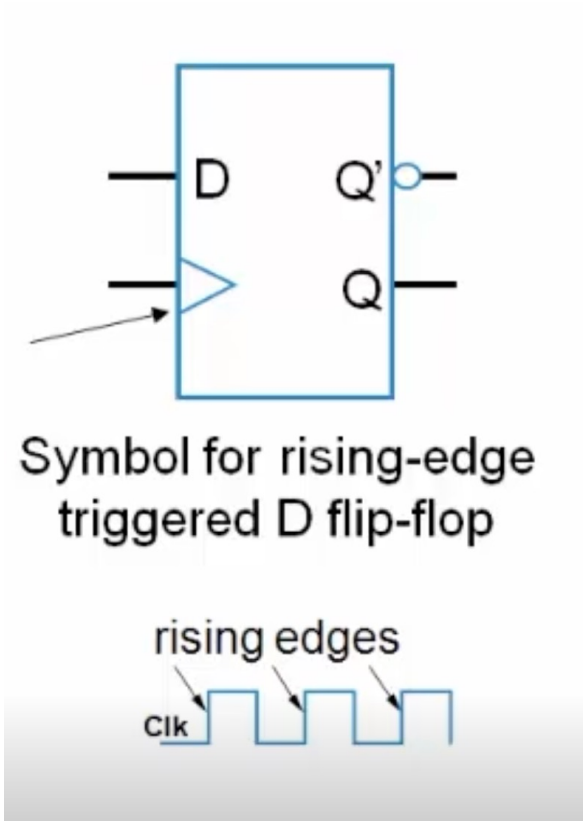
C	D	Q	Condition
0	x	1/0	Hold
1	0	0	Q follows D
1	1	1	

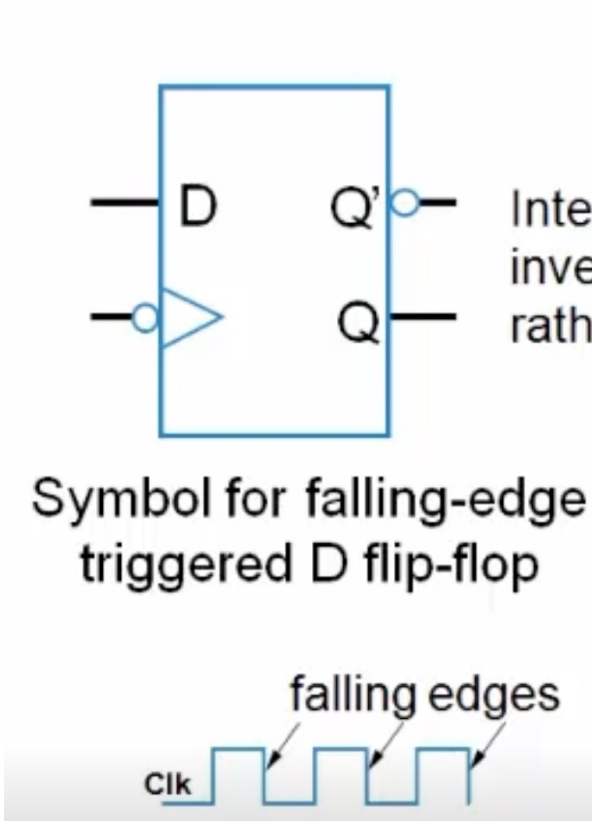
Edge Sensitive D-Latch
D Flip-Flop



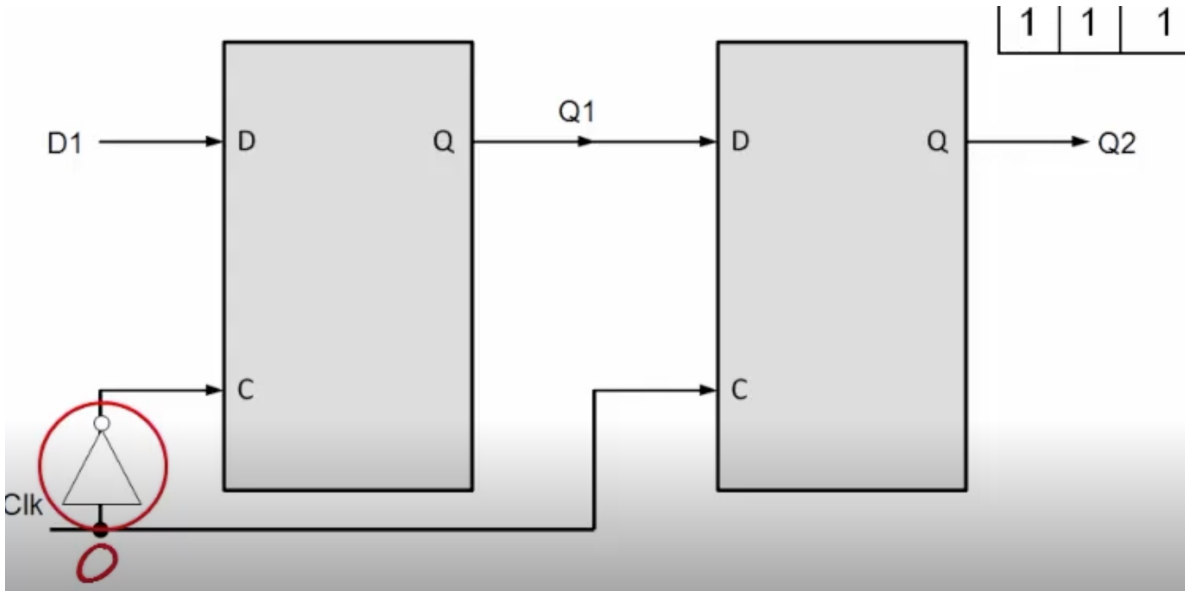
Clk	D	Q	Condition
↡	x	Q_p	Hold
↑	0	0	Q follows D
↑	1	1	

Edge Trigger Symbol





Edge Sensitive D-Latch



Clk	D	Q	Condition
↑	x	Qp	Hold
↑	0	0	Q follows D
↑	1	1	

Edge Trigger Symbol