Cs302

Final Term Diagrams and Important Points

Presented by Moon-Knight

Circuit Diagram of OLMC

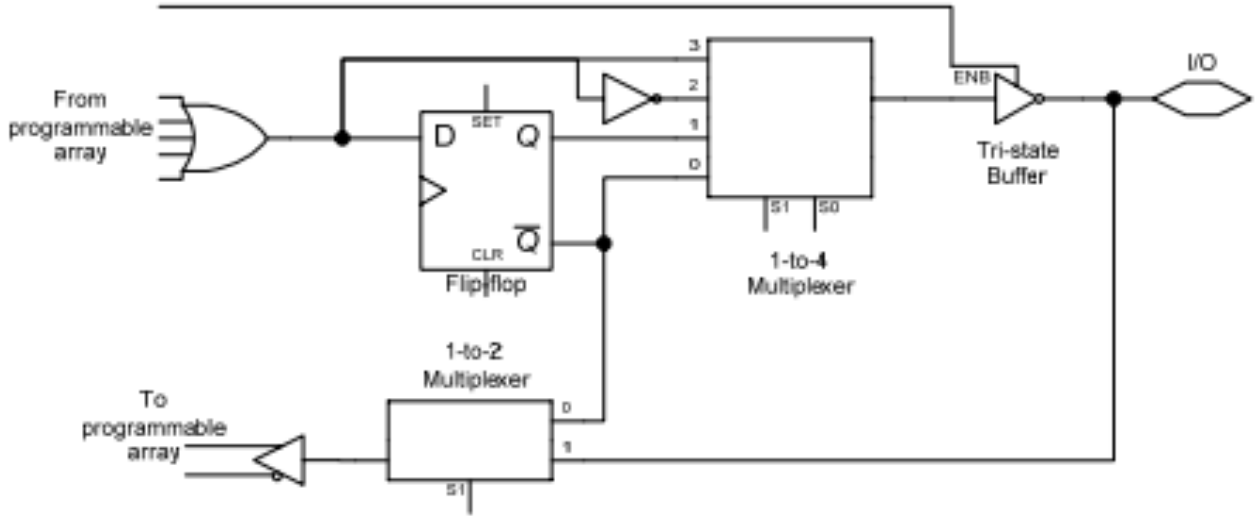
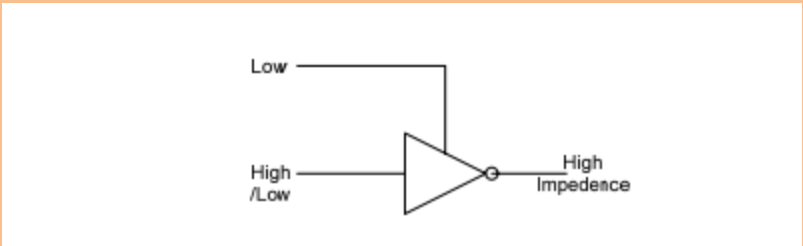


Figure 20.8 Circuit Diagram of OLMC

The four OLMC configurations are

- Combination Mode with active-low output
- Combinational Mode with active-high output
- Registered Mode with active-low output
- Registered Mode with active-high output

Tri-State Buffer in High-Impedance State

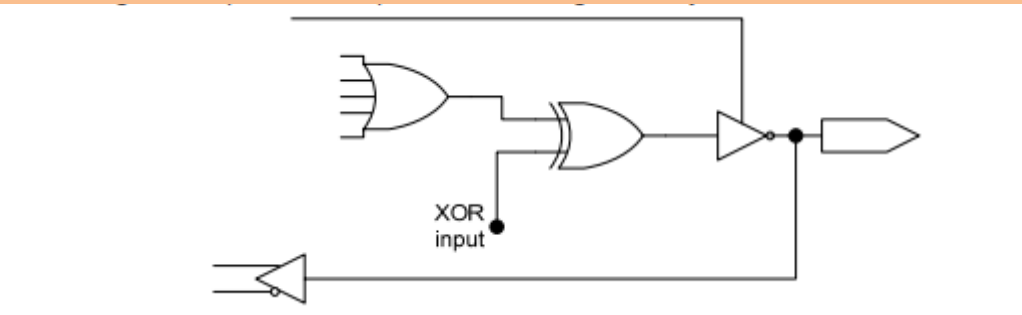


Tri-state Buffer and OLMC output pin

The tri-state buffer connecting the output of the OLMC circuit to the output pin is
controlled through four different sources. The tri-state buffer control input can be connected in
four different ways.

1. Connected to Vcc. The output is always enabled.
2. Connected to GND. The output is disabled and the output pin is configured as an input pin.
3. Connected to the external pin (11) which can be connected to Vcc or GND. The tri-state
buffer is therefore controlled externally by applying an appropriate signal at the pin.
4. Connected to the output of one of the eight AND gates connected to the OLMC. Thus the
tri-state buffer is controlled by a logical expression.

The tri-sate buffer Combinational Input/Output



ABEL

ABEL provides three different text-based methods for describing and entering a logic
design. The three methods are

- Boolean Equations
- Truth Tables
- State Diagrams

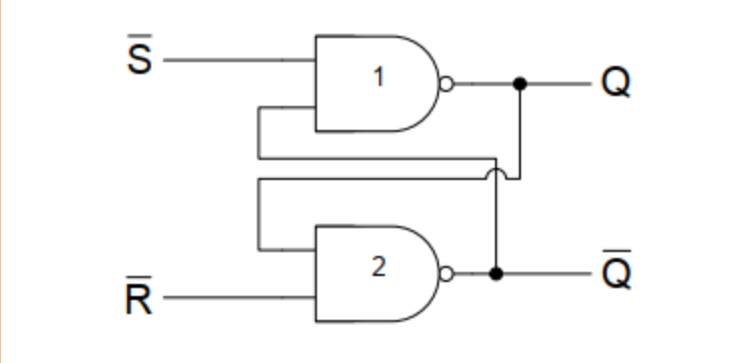
ABEL Symbols for logic operations

Logic Operation	ABEL Symbol
NOT	!
AND	&
OR	#
XOR	$

Boolean and equivalent ABEL Notations

Boolean Notation	ABEL Notation
A	!A
.BA	A&B
+ BA	A#B
⊕ BA	A$B

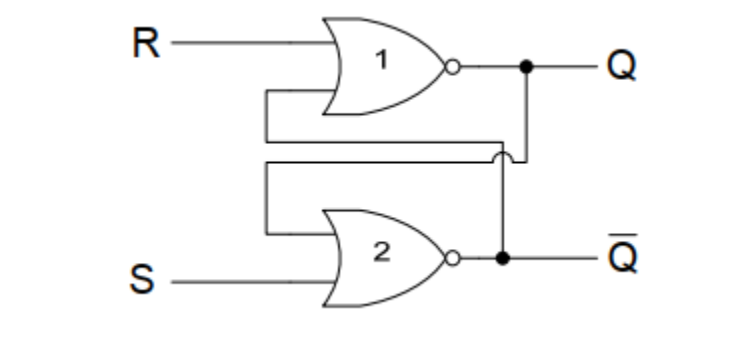
The NAND gate based S-R (Set-Reset) Latch



Truth-Table of NAND based S-R Latch

Input		Output
S	R	Qt+1
0	0	invalid
0	1	1
1	0	0
1	1	Qt

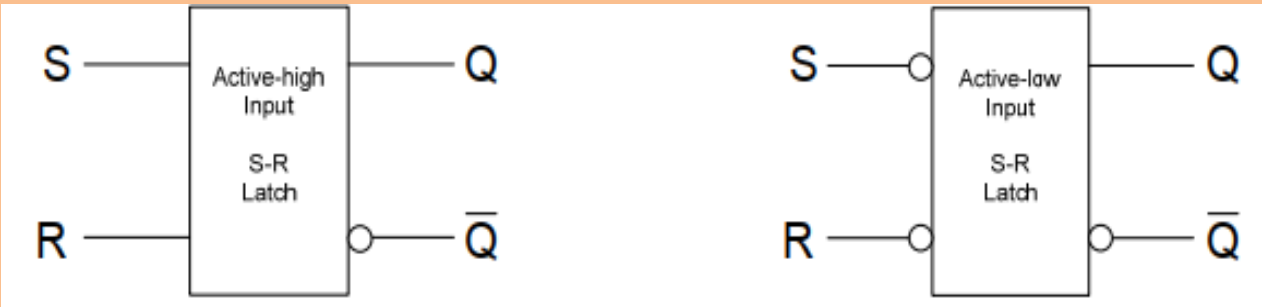
The NOR gate based S-R (Set-Reset) Latch



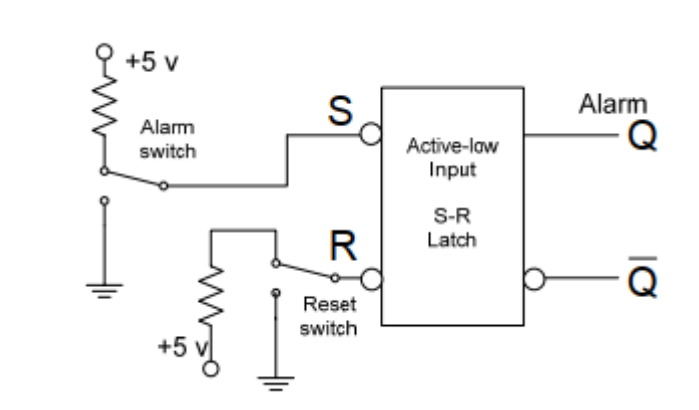
Truth-Table of NOR based S-R Latch

Input output		
S	R	Qt+1
0	0	Qt
0	1	0
1	0	1
1	1	invalid

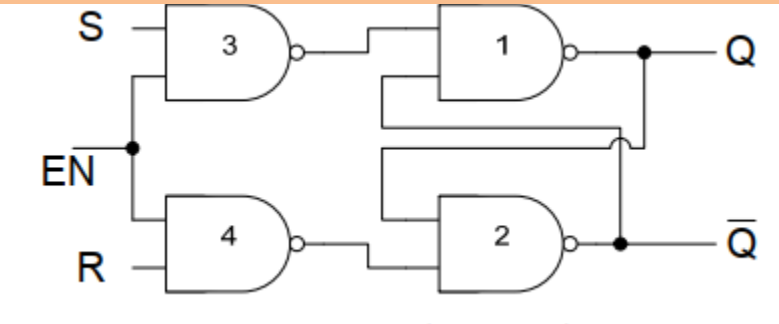
NOR based Active-High and NAND based Active-Low S-R Latches



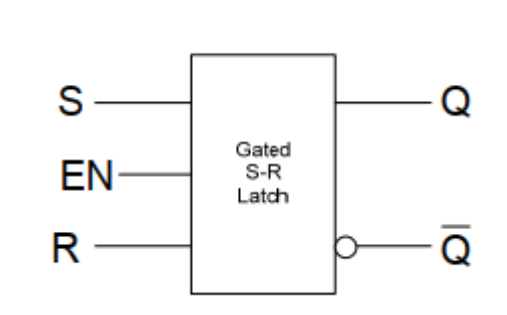
The switch connected through an S-R latch



The circuit diagram of the gated S-R latch



Logic Symbol of a Gated S-R Latch

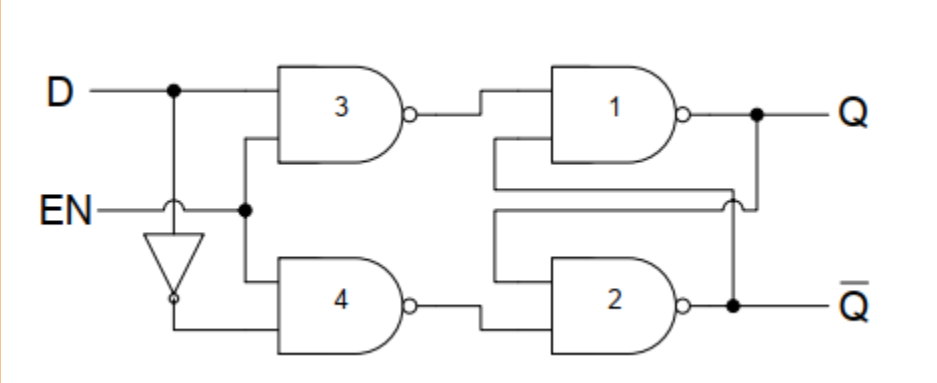


Truth-Table of a gated S-R Latch

input			output
EN	S	R	Qt+1

0	x	x	Qt
1	0	0	Qt
1	0	1	0
1	1	0	1
1	1	1	invalid

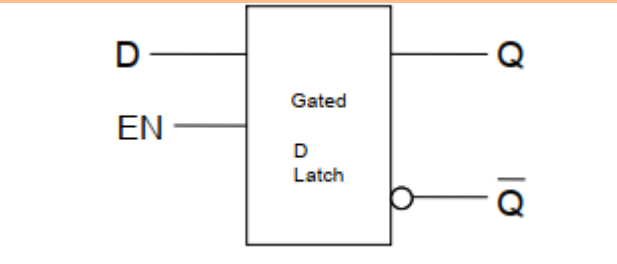
The Gated D Latch



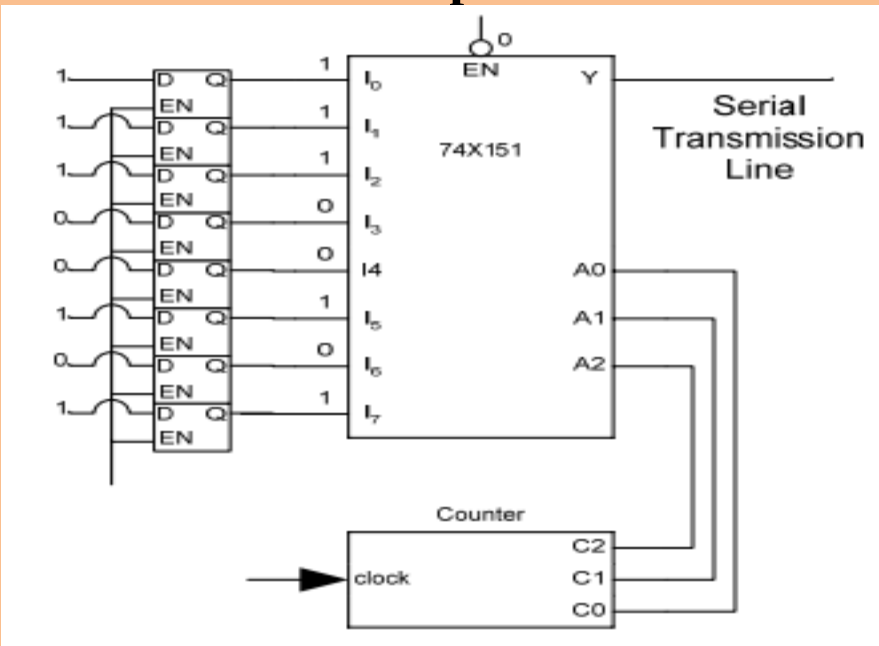
Truth-Table of a gated D Latch

Input		Output
EN	D	Qt+1
0	x	Qt
1	0	0
1	1	1

Logic Symbol of a Gated D Latch



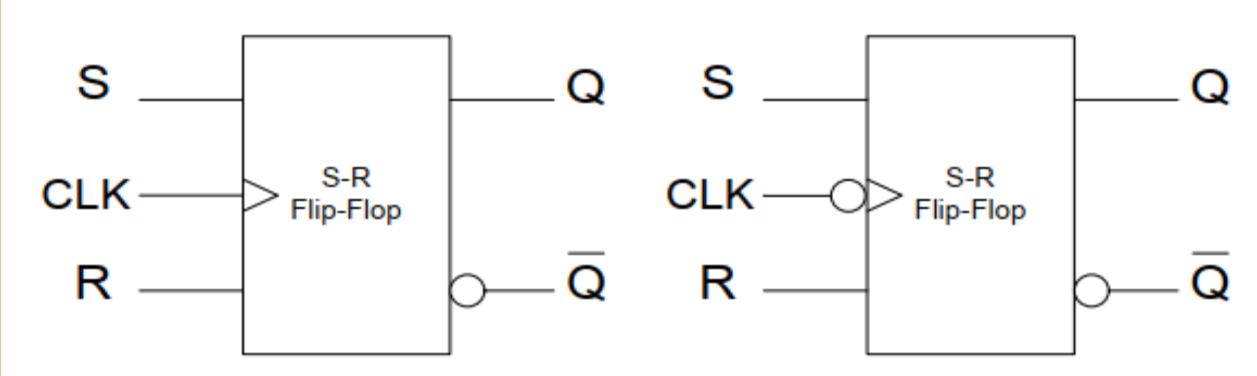
Gated D-latch used to store parallel data



Three different types of edge-triggered flip-flops are generally used in digital logic circuits.

- S-R edge-triggered flip-flop
- D edge-triggered flip-flop
- J-K edge-triggered flip-flop

Logic Symbol of Positive and Negative edge triggered S-R flip-flops

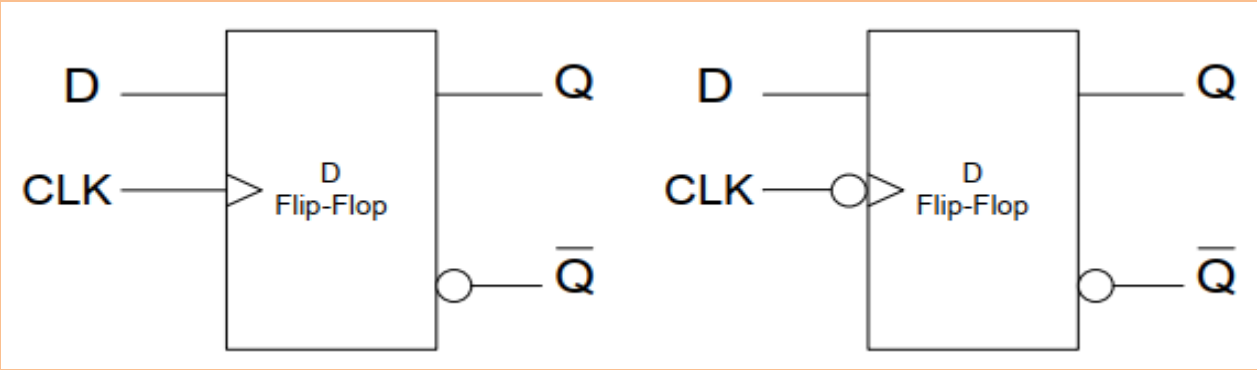


Truth-Table of Positive and Negative Edge triggered S-R flip-flops

Input			Output
CLK	S	R	Q_{t+1}
0	X	X	Q_t
1	X	X	Q_t
↑	0	0	Q_t
↑	0	1	0
↑	1	0	1
↑	1	1	invalid

Input			Output
CLK	S	R	Q_{t+1}
0	x	x	Q_t
1	x	x	Q_t
↓	0	0	Q_t
↓	0	1	0
↓	1	0	1
↓	1	1	invalid

Logic Symbol of Positive and Negative edge triggered D flip-flops

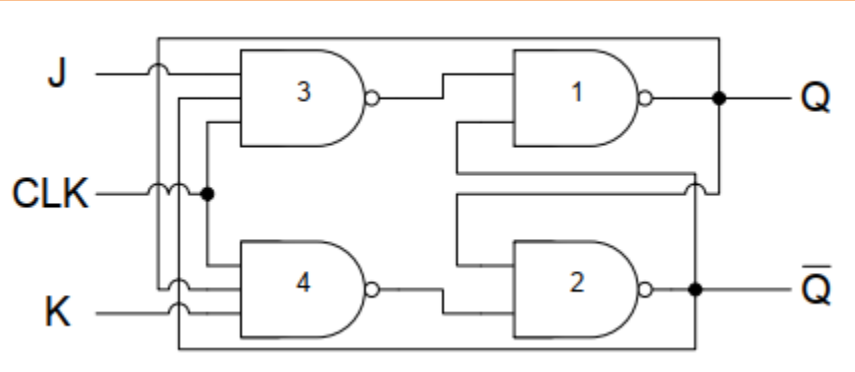


Truth-Table of Positive and Negative Edge triggered D flip-flops

Input		Output
CLK	D	Q_{t+1}
0	X	Q_t
1	X	Q_t
↑	0	0
↑	1	1

Input		Output
CLK	D	Q_{t+1}
0	X	Q_t
1	X	Q_t
↓	0	0
↓	1	1

Edge-Triggered J-K Flip-flop

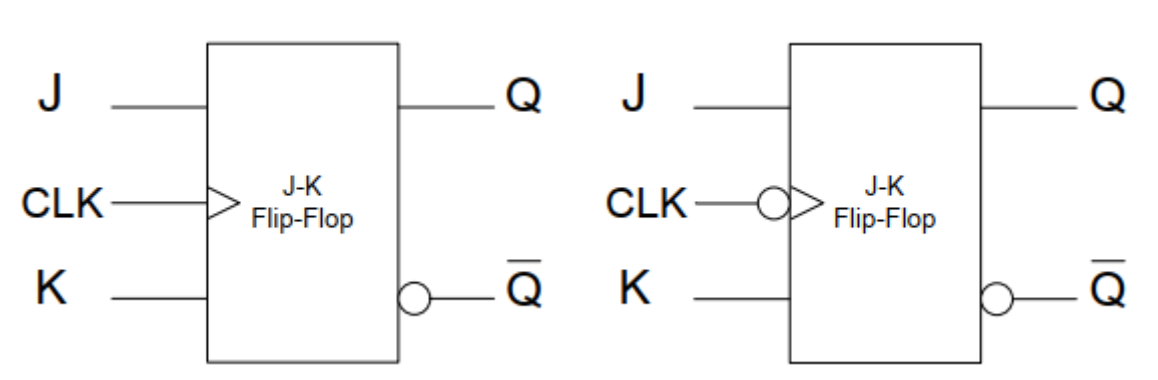


Truth-Table of Positive and Negative Edge triggered J-K flip-flops

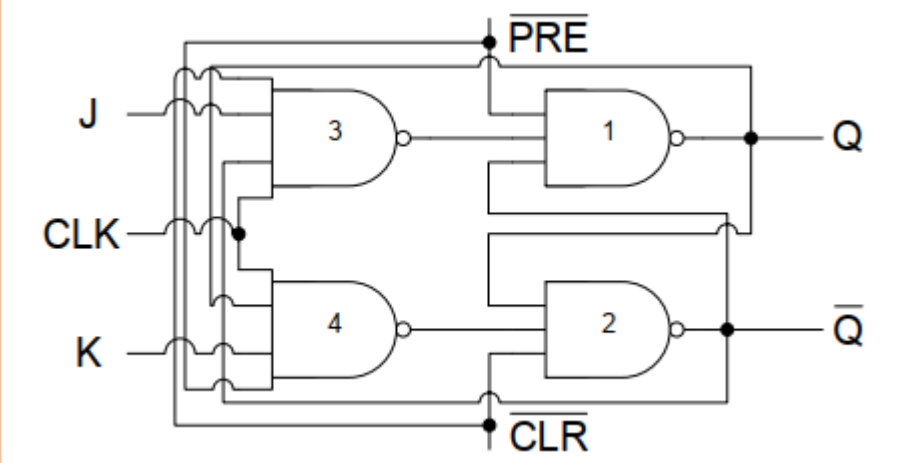
Input			Output
CLK	J	K	Q_{t+1}
0	x	X	Q_t
1	x	X	Q_t
↑	0	0	Q_t
↑	0	1	0
↑	1	0	1
↑	1	1	$\overline{Q}_t$

Input			Output
CLK	J	K	Q_{t+1}
0	x	x	Q_t
1	x	x	Q_t
↓	0	0	Q_t
↓	0	1	0
↓	1	0	1
↓	1	1	$\overline{Q}_t$

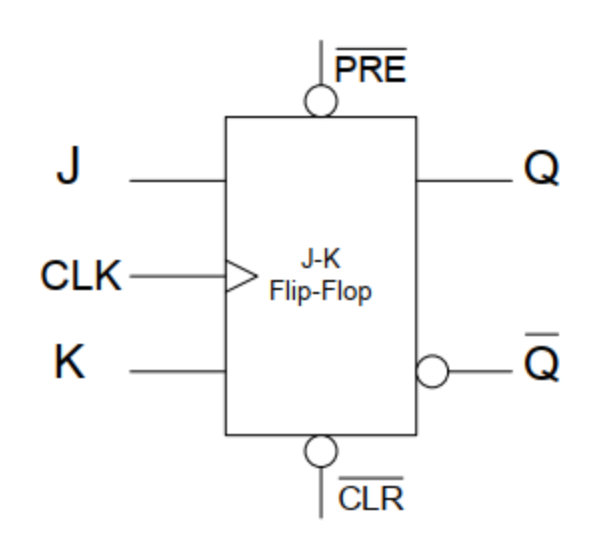
Logic Symbol of Positive and Negative edge triggered J-K flip-flops



J-K flip-flop with Asynchronous Preset and Clear inputs



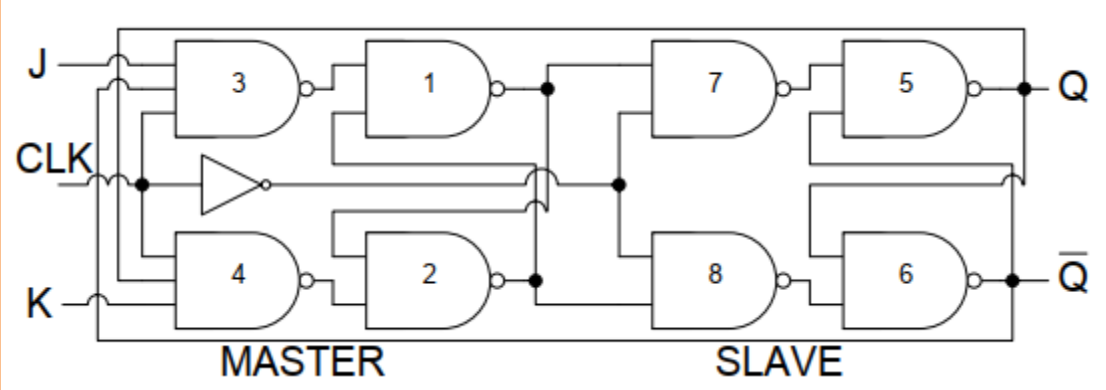
Logic Symbol of a J-K flip-flop with Asynchronous inputs



Truth table of J-K flip-flop with Asynchronous inputs

Input		Output
$\overline{PRE}$	$\overline{CLR}$	Q_{t+1}
0	0	Invalid
0	1	1
1	0	0
1	1	Clocked operation

Master-Slave flip-flop



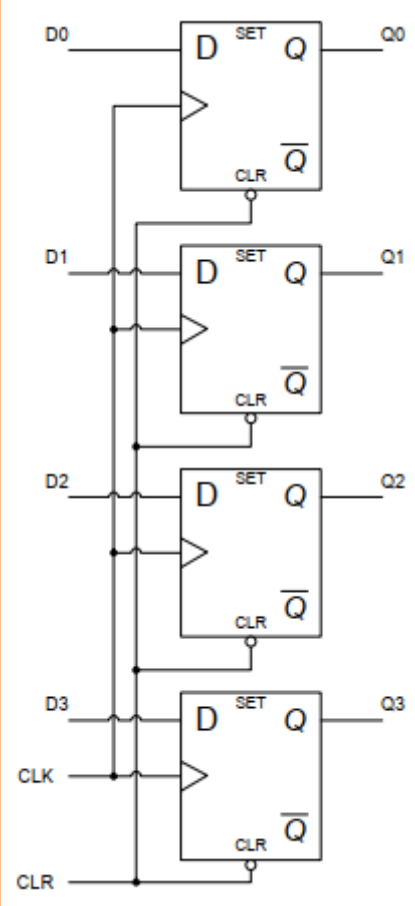
Truth table of the Master-Slave J-K flip-flop

Input			Output
CLK	J	K	Q_{t+1}
Pulse	0	0	Q_t
Pulse	0	1	0
Pulse	1	0	1
Pulse	1	1	$\overline{Q}_t$

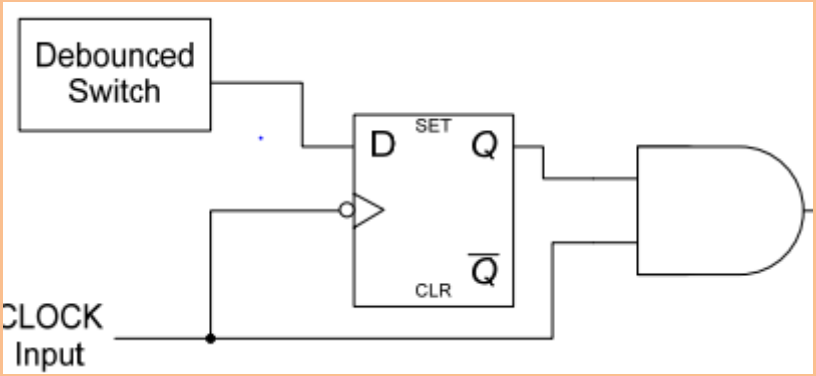
Flip-Flop Operating Characteristics

- Propagation Delay
- Set-up Time
- Hold Time
- Maximum Clock frequency
- Pulse width
- Power Dissipation

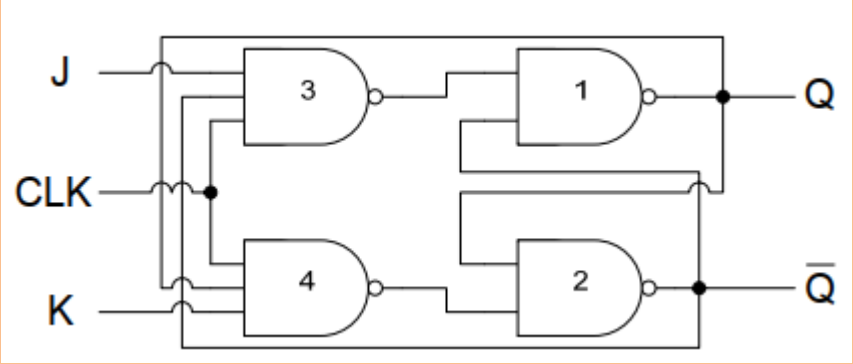
D-flip-flops used for Parallel Data Storage



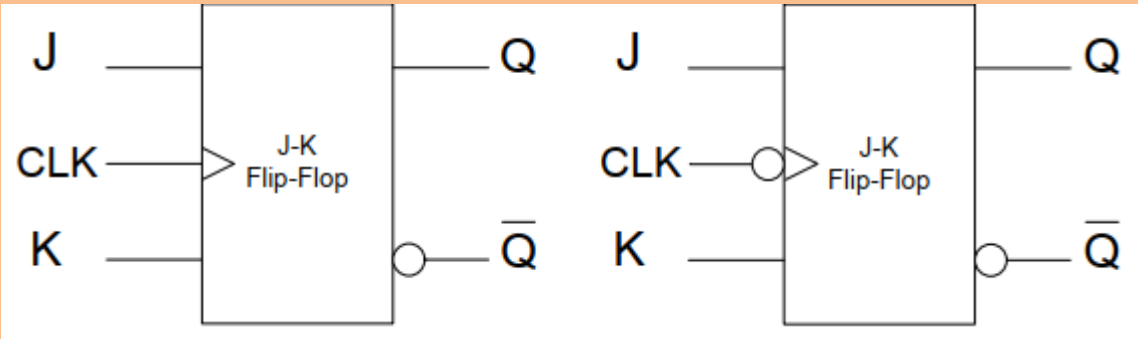
D flip-flop used to synchronize the AND Gate output



Edge-triggered J-K flip-flop



Logic Symbol of Positive and Negative edge triggered J-K flip-flops

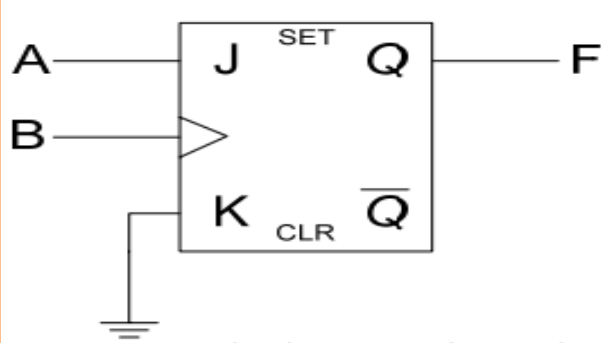


Truth-Table of Positive and Negative Edge triggered J-K flip-flop

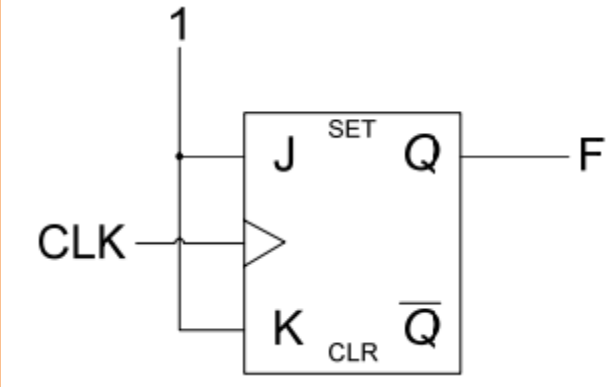
Input			Output
CLK	J	K	Q_{t+1}
0	x	X	Q_t
1	x	X	Q_t
↑	0	0	Q_t
↑	0	1	0
↑	1	0	1
↑	1	1	$\overline{Q}_t$

Input			Output
CLK	J	K	Q_{t+1}
0	x	x	Q_t
1	x	x	Q_t
↓	0	0	Q_t
↓	0	1	0
↓	1	0	1
↓	1	1	$\overline{Q}_t$

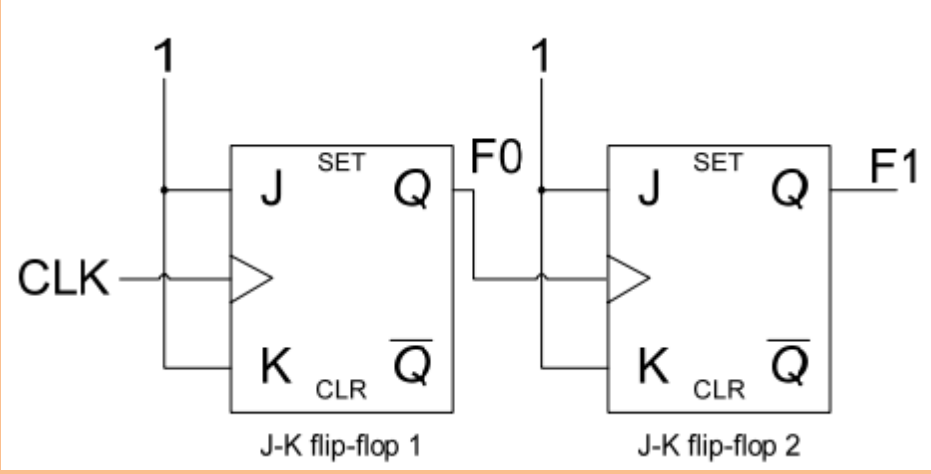
J-K flip-flop used as sequence detector



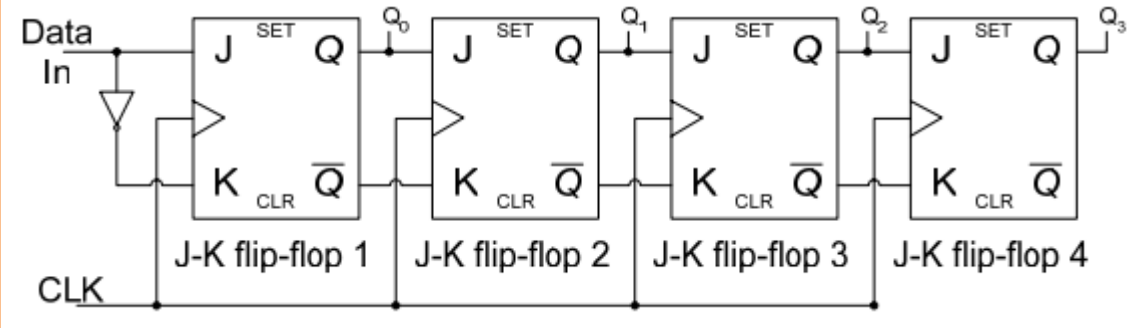
J-K flip-flop used as frequency divider



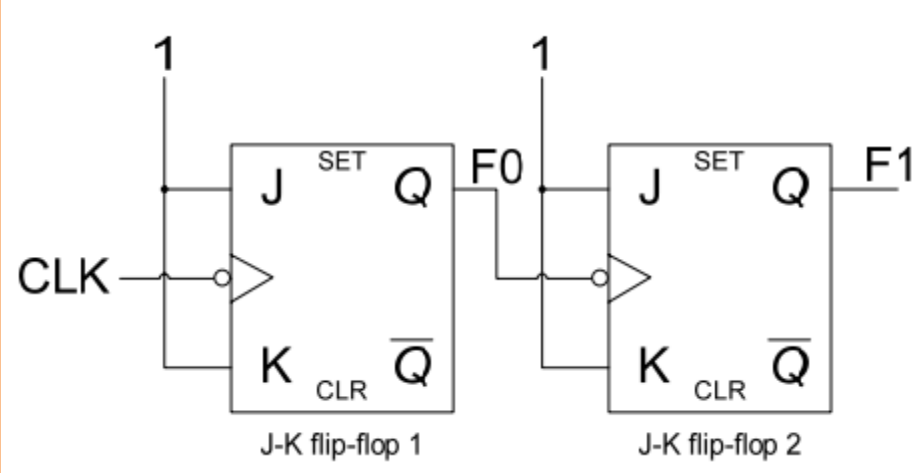
J-K flip-flop connected as divide-by-4 frequency divider



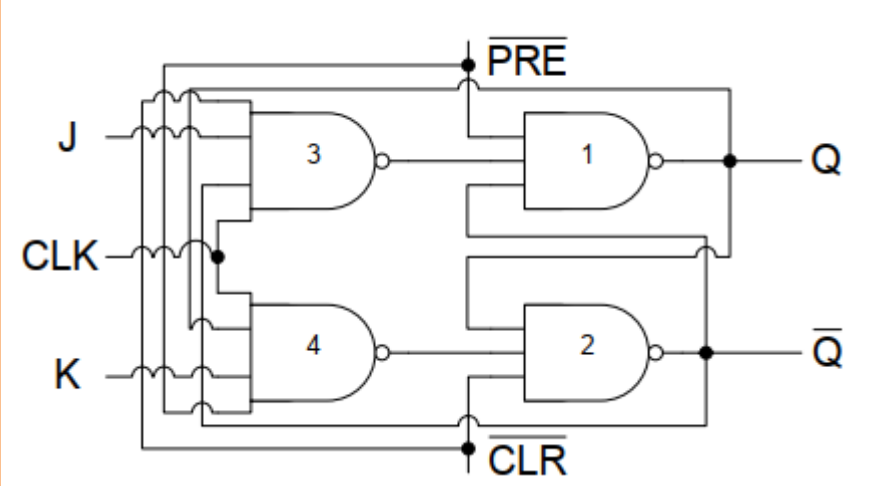
J-K flip-flop used as a shift register



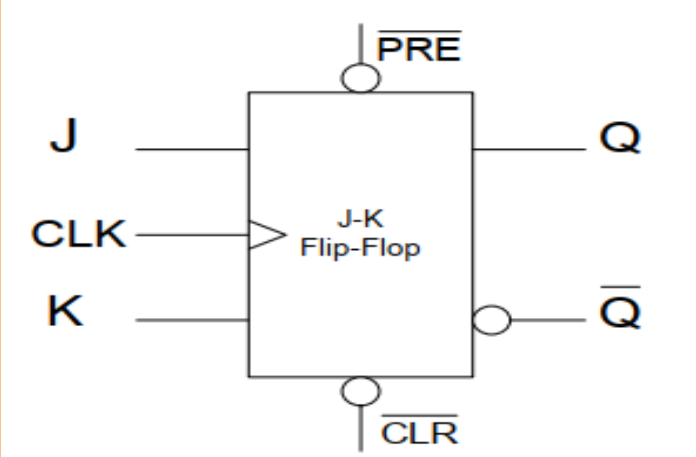
J-K flip-flop used as a up counter



ASYNCHRONOUS PRESET AND CLEAR INPUTS



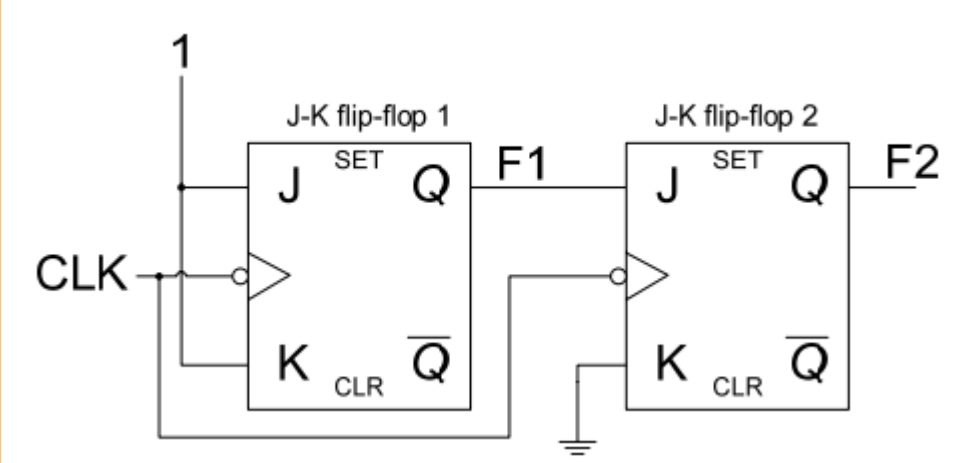
Logic Symbol of a J-K flip-flop with Asynchronous inputs



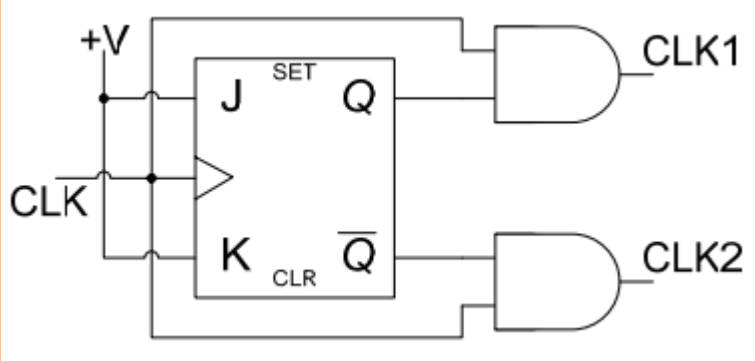
Truth table of J-K flip-flop with Asynchronous inputs

Input		Output
$\overline{PRE}$	$\overline{CLR}$	Q_{t+1}
0	0	Invalid
0	1	1
1	0	0
1	1	Clocked operation

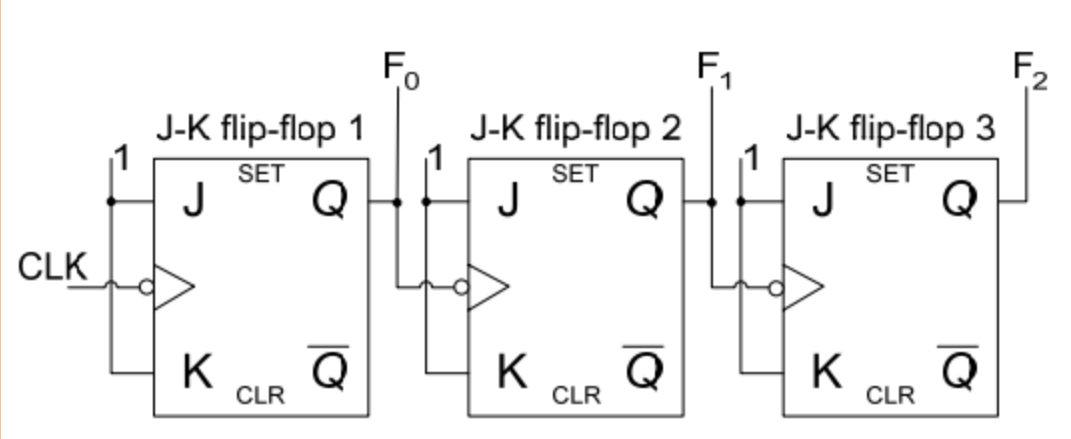
J-K flip-flop circuit with potential timing problem



J-K flip-flop circuit with potential race condition



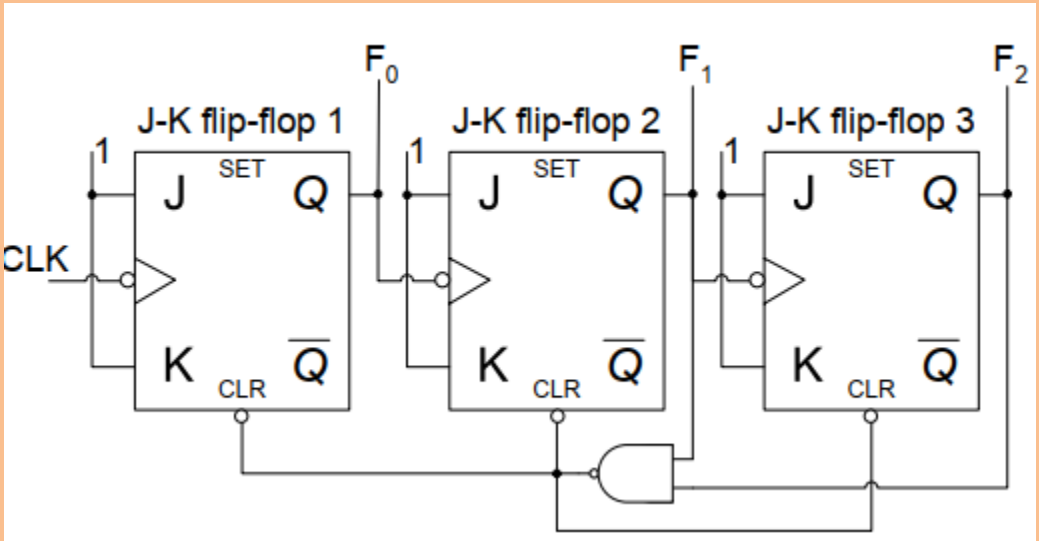
3-bit Asynchronous Up-Counter



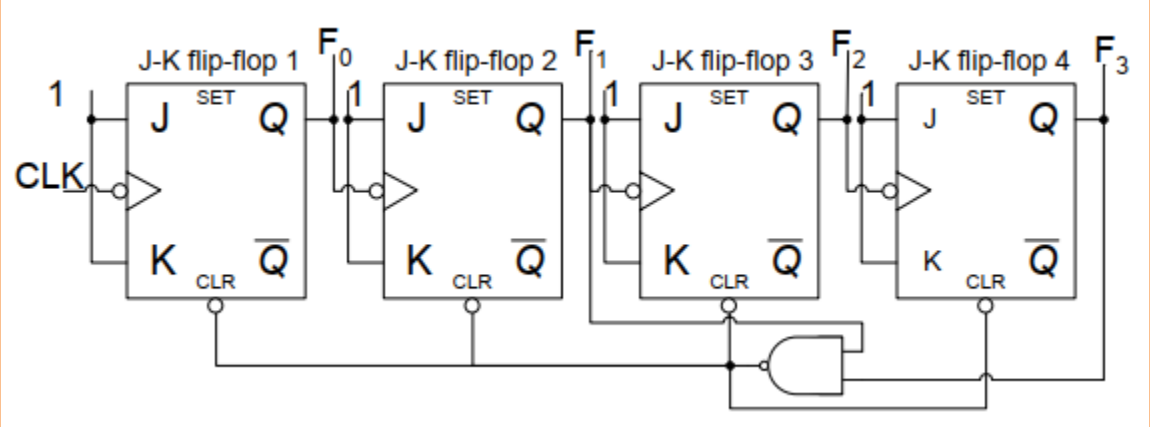
Output State of a 3-bit Asynchronous Up-Counter

Input Clock Pulses	Output F_2	F_1	F_0
1	0	0	0
2	0	0	1
3	0	1	0
4	0	1	1
5	1	0	0
6	1	0	1
7	1	1	0
8	1	1	1

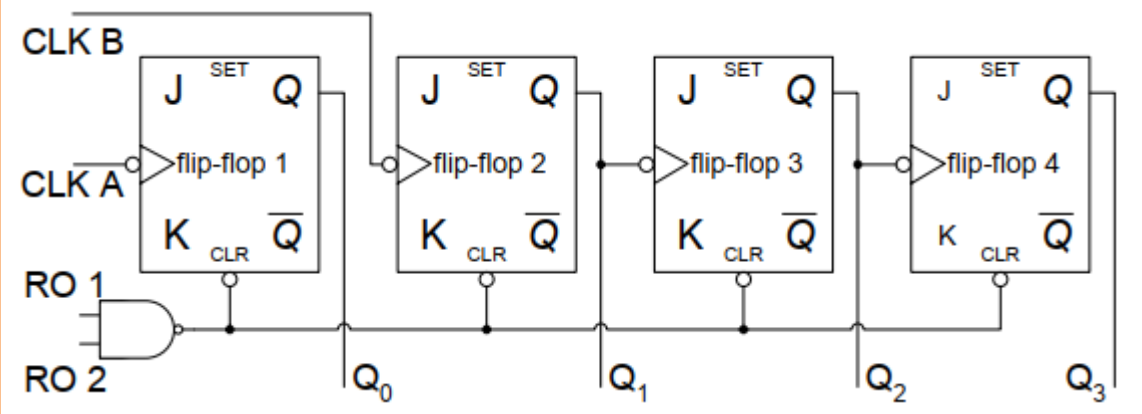
Mod-n Counters\ Mod-6 Counter



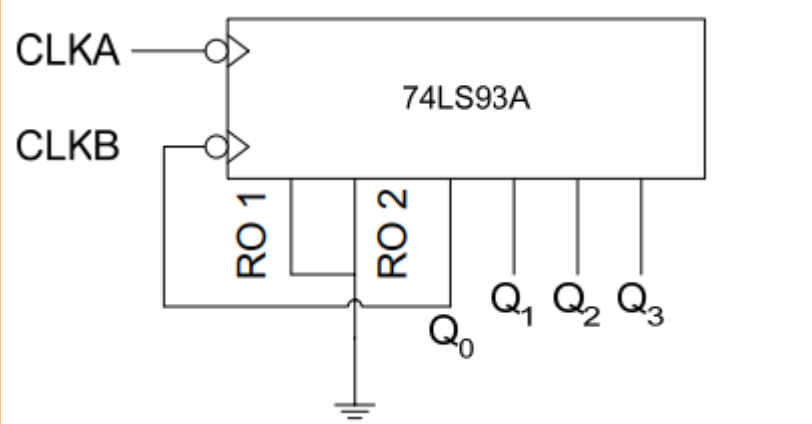
Mod-10 Counter (Decade Counter) Asynchronous Decade Counter



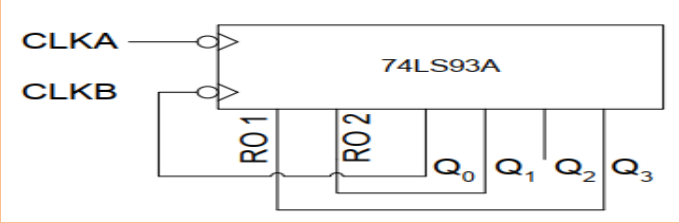
Internal circuit diagram of the 74LS93A Counter



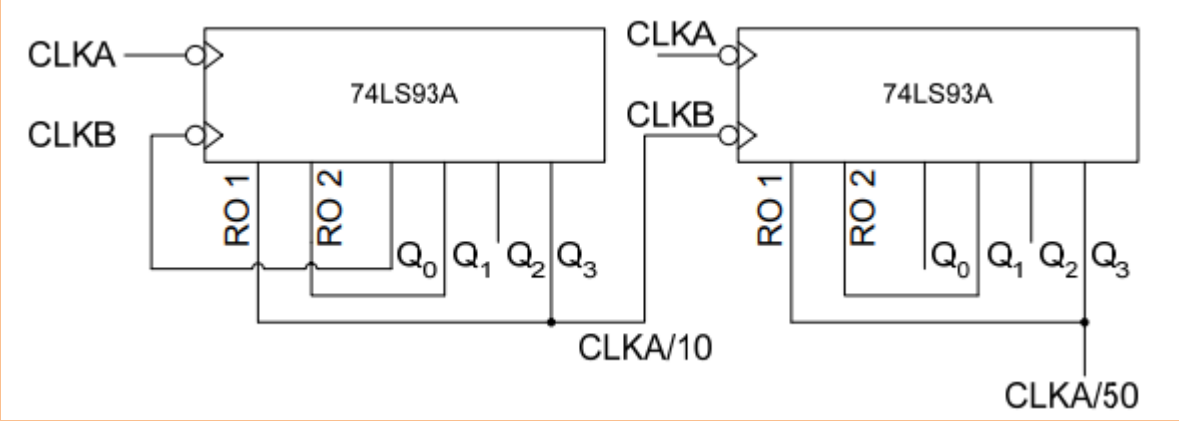
74LS93A connected as MOD-16 Counter



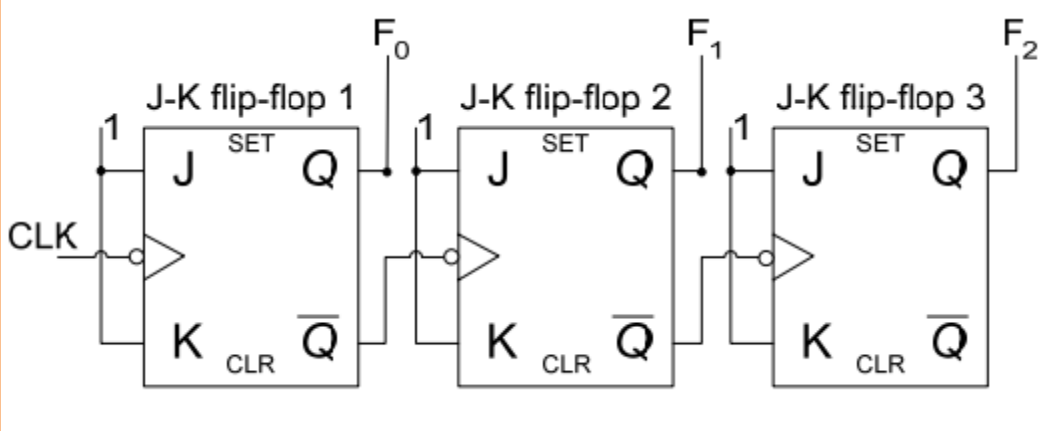
74LS93A connected as Decade Counter



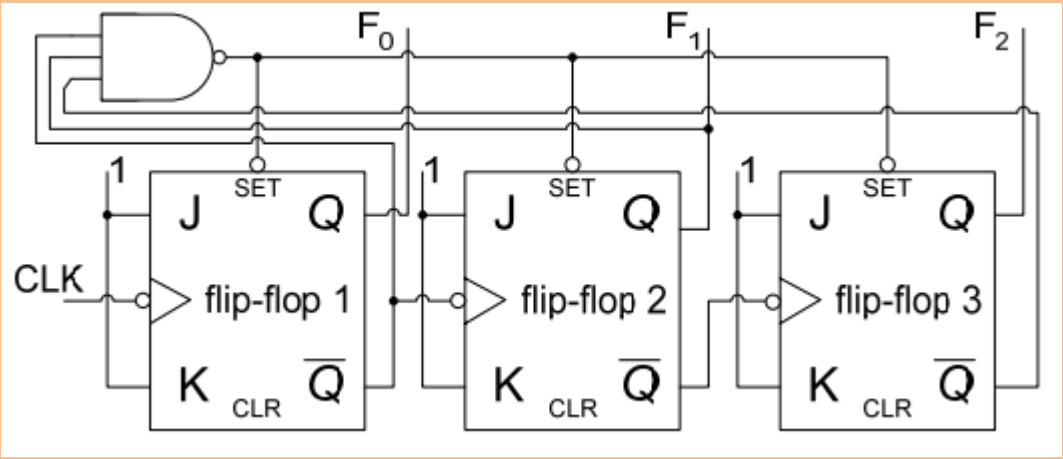
74LS93A Connected as a frequency divider (divide by 50)



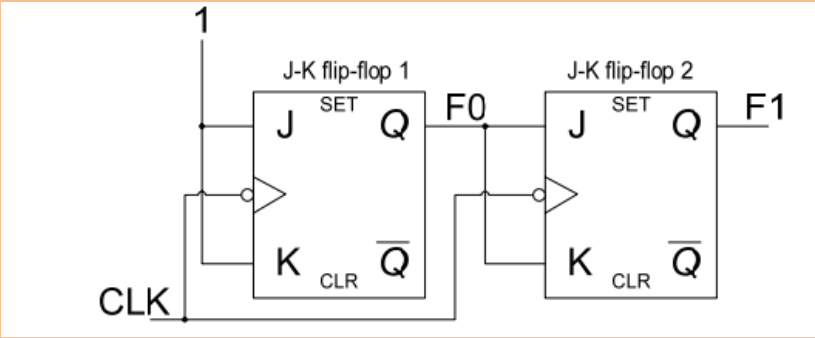
3-bit Asynchronous Down-Counter



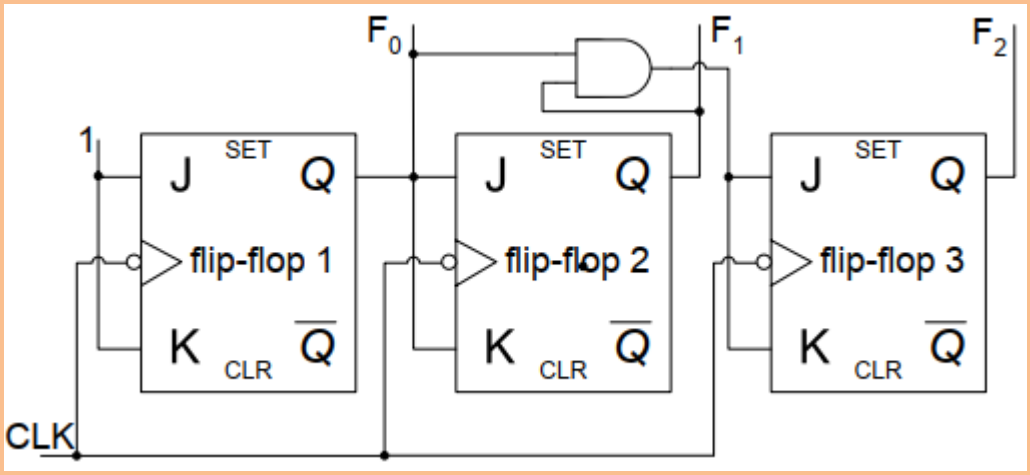
Down Counter with truncated sequence\ Down-counter configured to count a truncated sequence



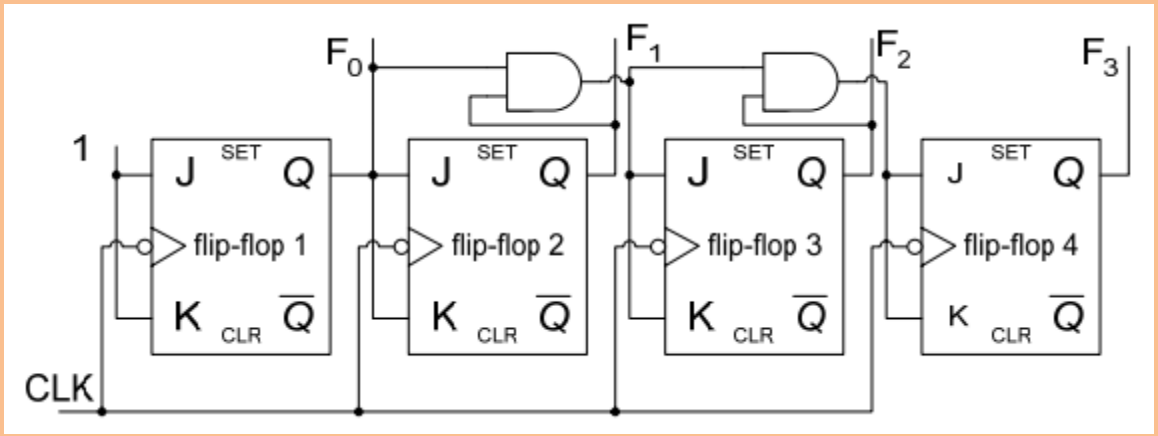
2-bit Synchronous Counter



A 3-bit Synchronous Counter



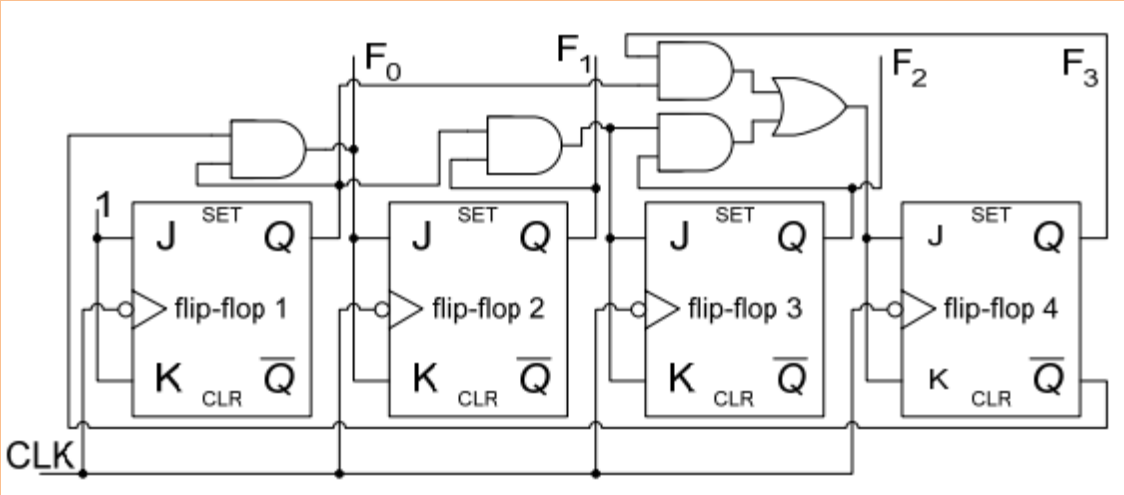
4-bit Synchronous Binary Counter



Output of a Synchronous Decade Counter

Input	Output			
Clock Pulses	F_3	F_2	F_1	F_0
1	0	0	0	0
2	0	0	0	1
3	0	0	1	0
4	0	0	1	1
5	0	1	0	0
6	0	1	0	1
7	0	1	1	0
8	0	1	1	1
9	1	0	0	0
10	1	0	0	1

Synchronous Decade Counter

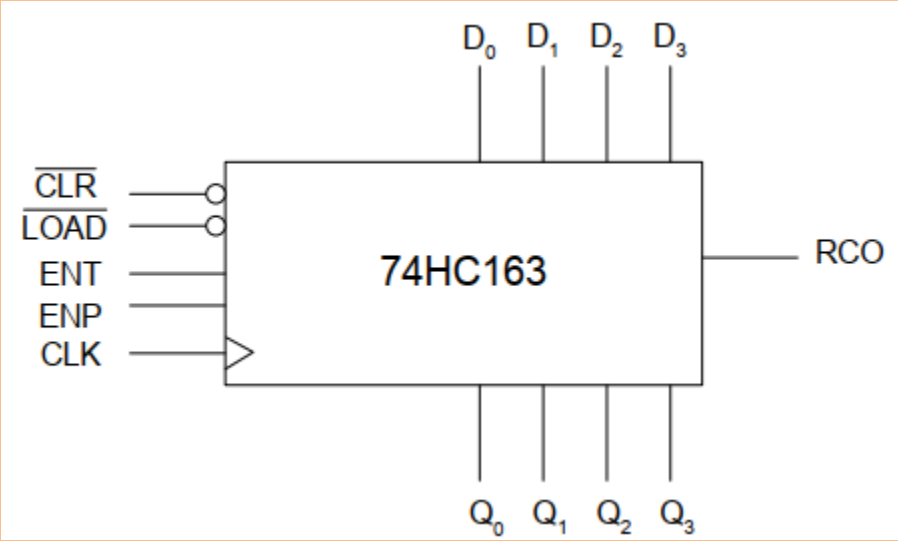


The 74HC163 is a 4-bit Synchronous Counter. Figure 28.2. The counter has the following pins.

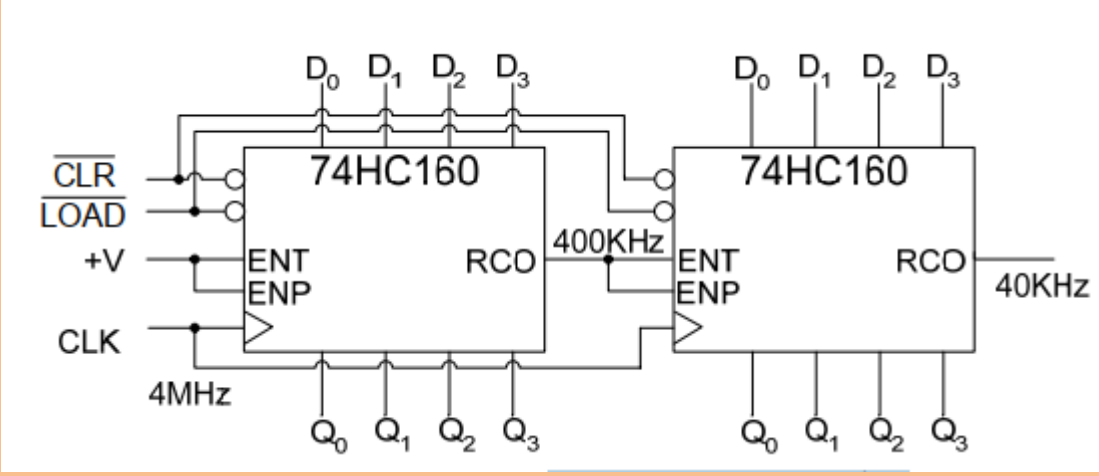
1. Parallel data inputs D0, D1, D2 and D3
2. Data outputs Q0, Q1, Q2 and Q3
3. Positive edge-triggered CLOCK signal
4. Active-low CLR input which resets the Counter output to 0000

5. Active-low LOAD input which loads the 4-bit data applied at the counter inputs
6. Active-high ENT and ENP enable inputs. For the counter to operate both the enable inputs
have to be high
7. The Ripple Clock Output RCO goes high when the Counter reaches the terminal count
1111. The RCO output along with ENT and ENP enable input pins are used to cascade
multiple counter ICs for implementing larger counters

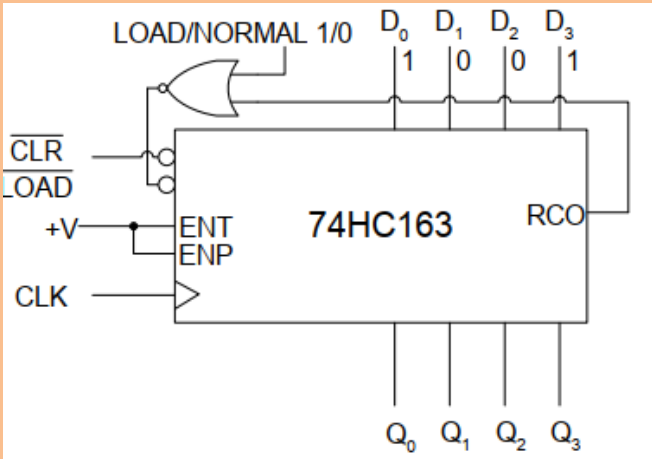
74HC163 4-bit Synchronous Counter



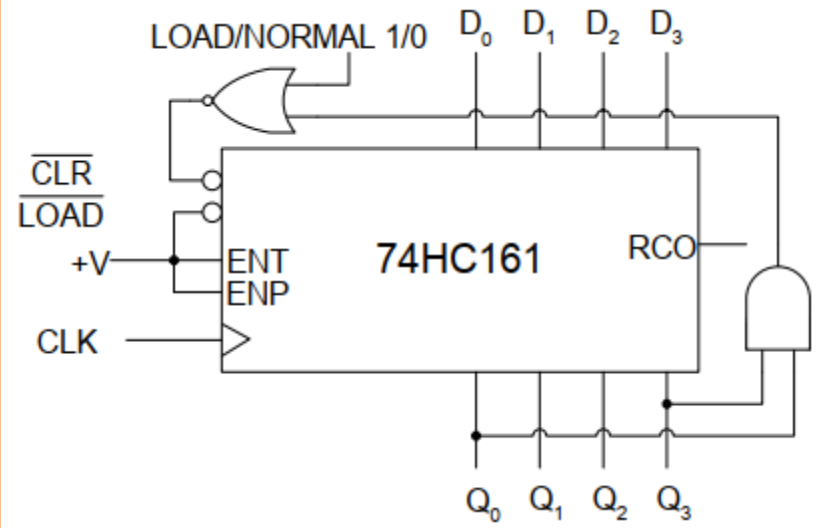
Cascaded Decade Counters



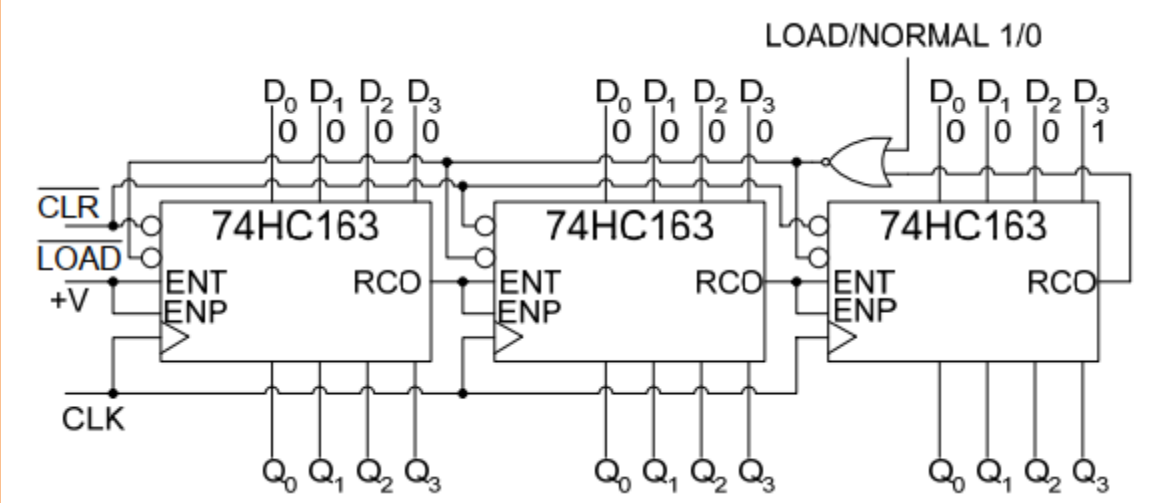
Integrated Circuit Counters with Truncated Sequences 74HC163 configured as Mod-7 counter



74HC161 configured as Mod-9 counter



74HC163 counters connected for cascaded truncated count sequence



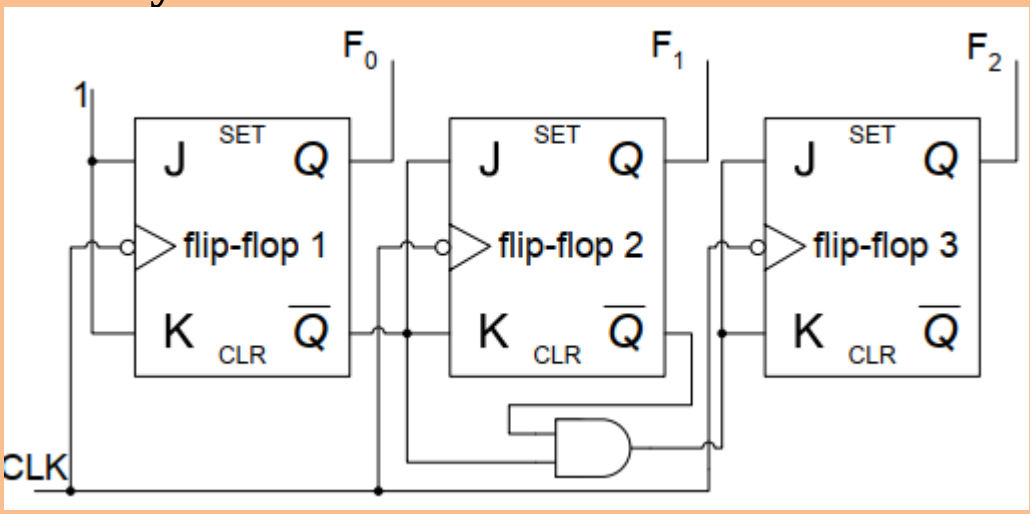
Up-counting sequence of a 3-bit Synchronous Counter

Clock Pulse	Q_2	Q_1	Q_0
0	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
6	1	1	0
7	1	1	1

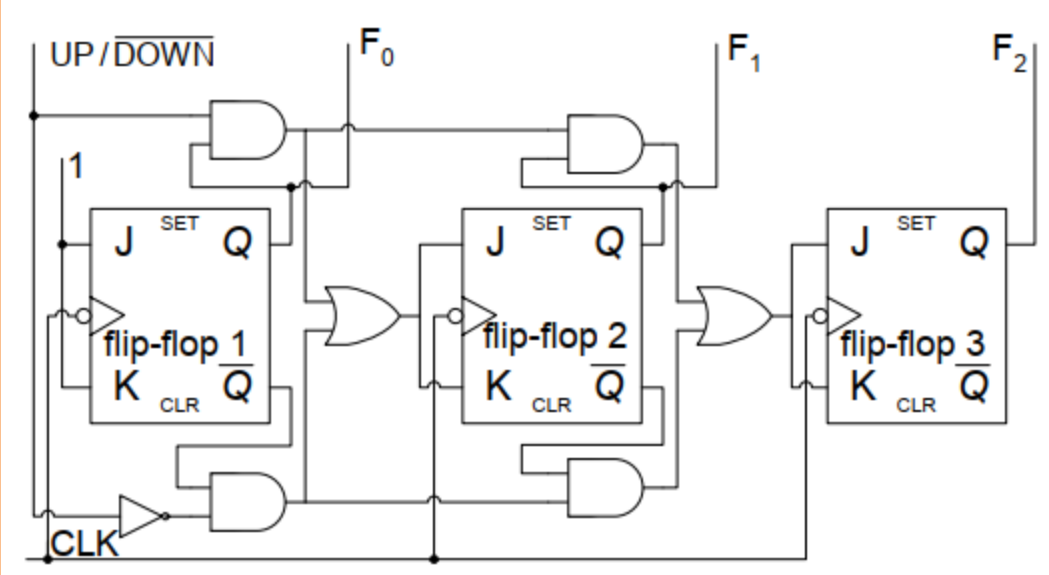
Down-counting sequence of a 3-bit Synchronous Counter

Clock Pulse	Q_2	Q_1	Q_0
0	1	1	1
1	1	1	0
2	1	0	1
3	1	0	0
4	0	1	1
5	0	1	0
6	0	0	1
7	0	0	0

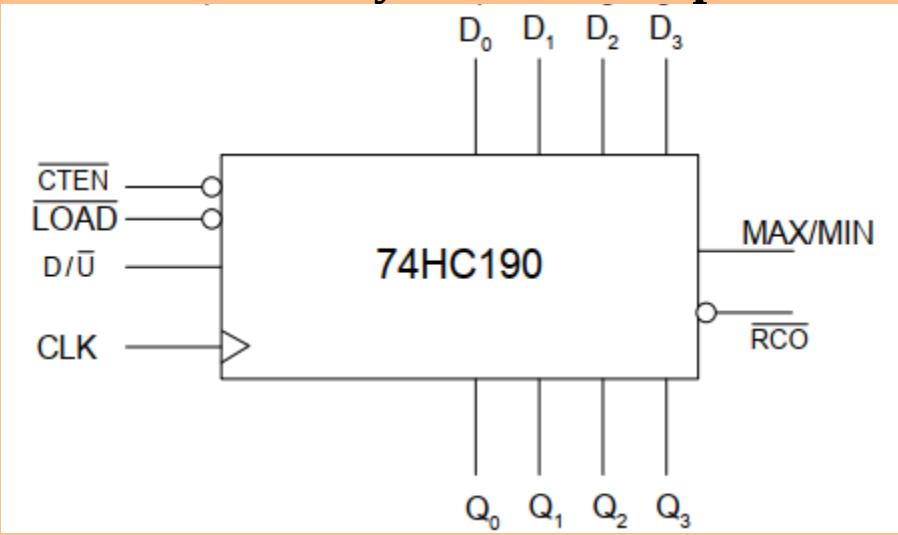
3-bit Synchronous Down-counter



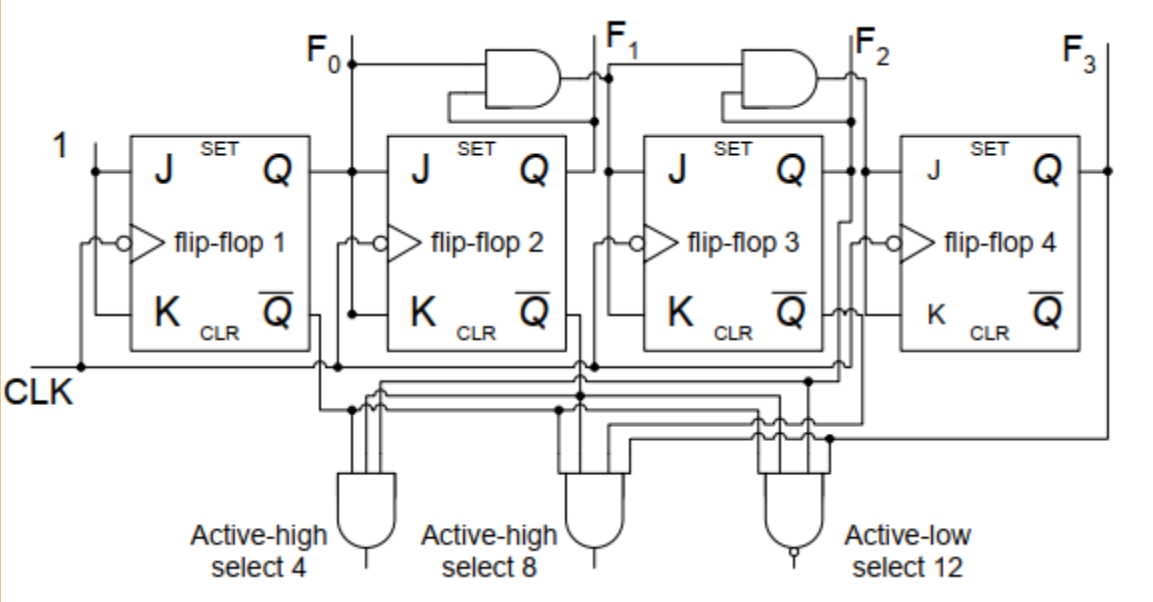
Up-Down Synchronous Counter



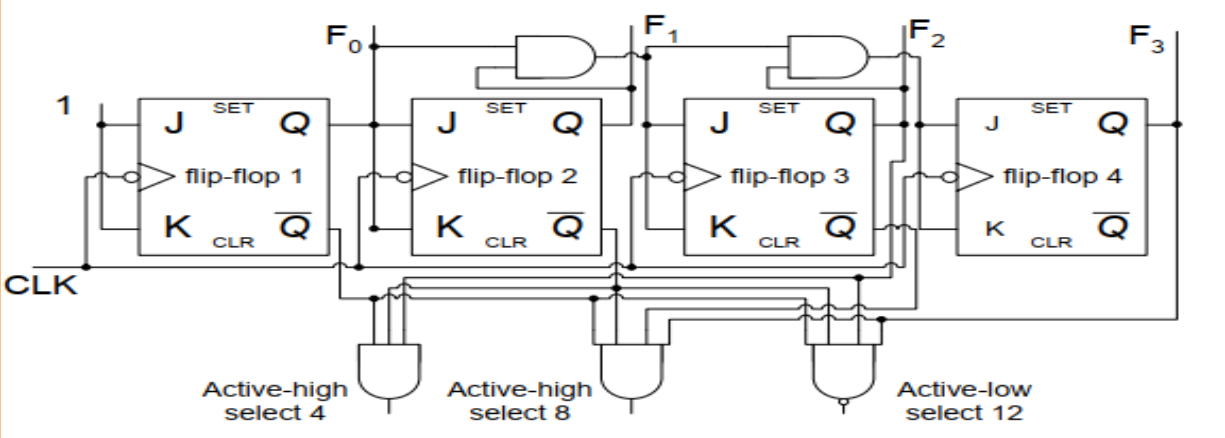
74HC190 4-bit Synchronous Up/Down Counter



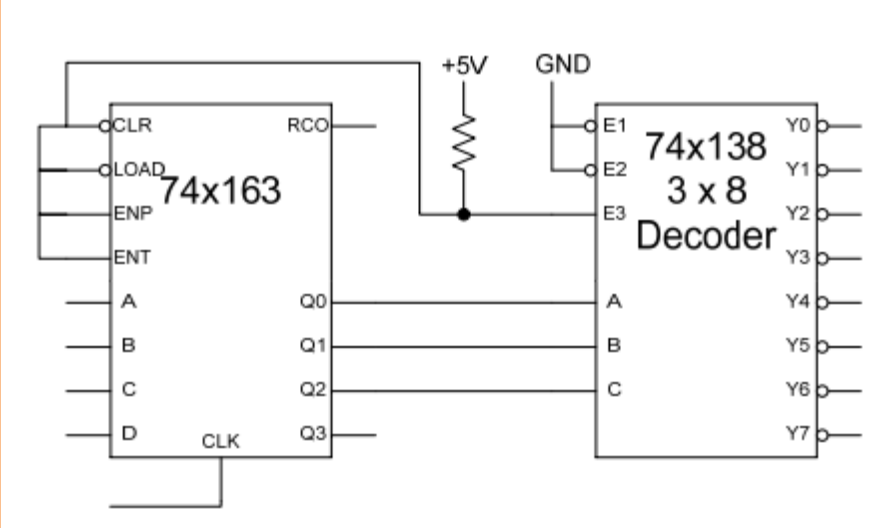
Decoder circuit decoding counter outputs 4, 8 and 12



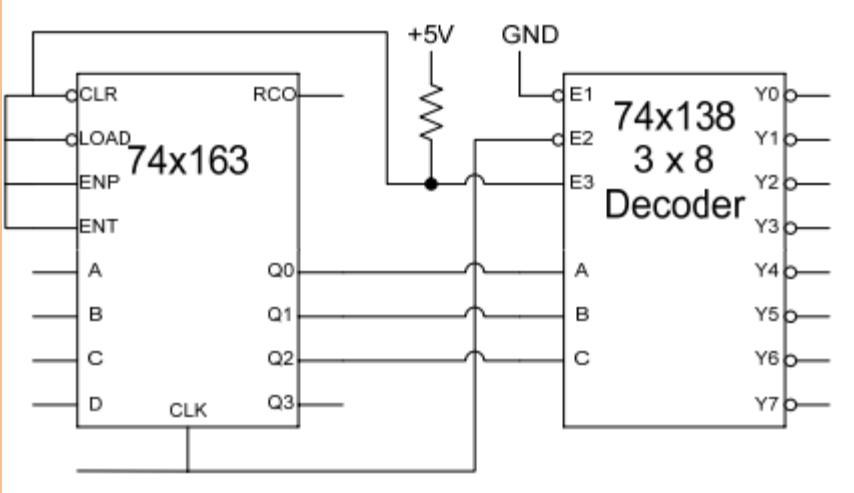
The Decoder circuit connected to remove glitches



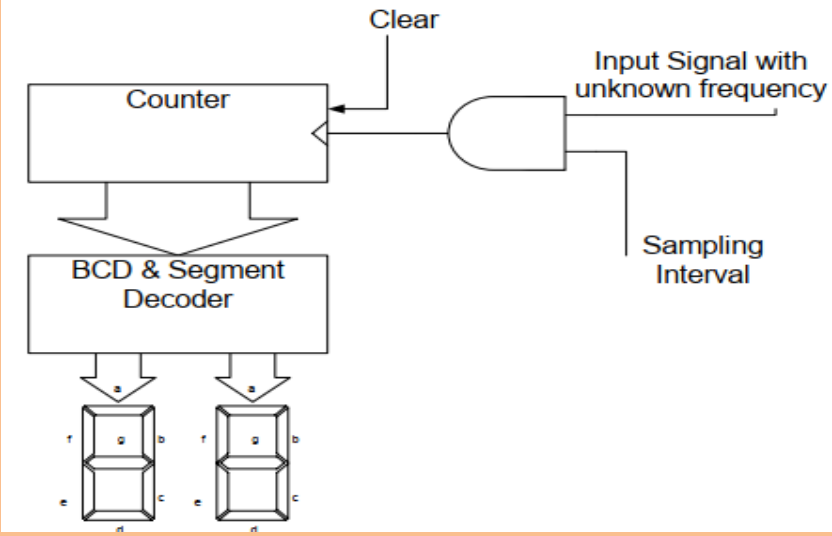
74x163 Counter output Decoded using a 3 x 8, 74x138 Decoder



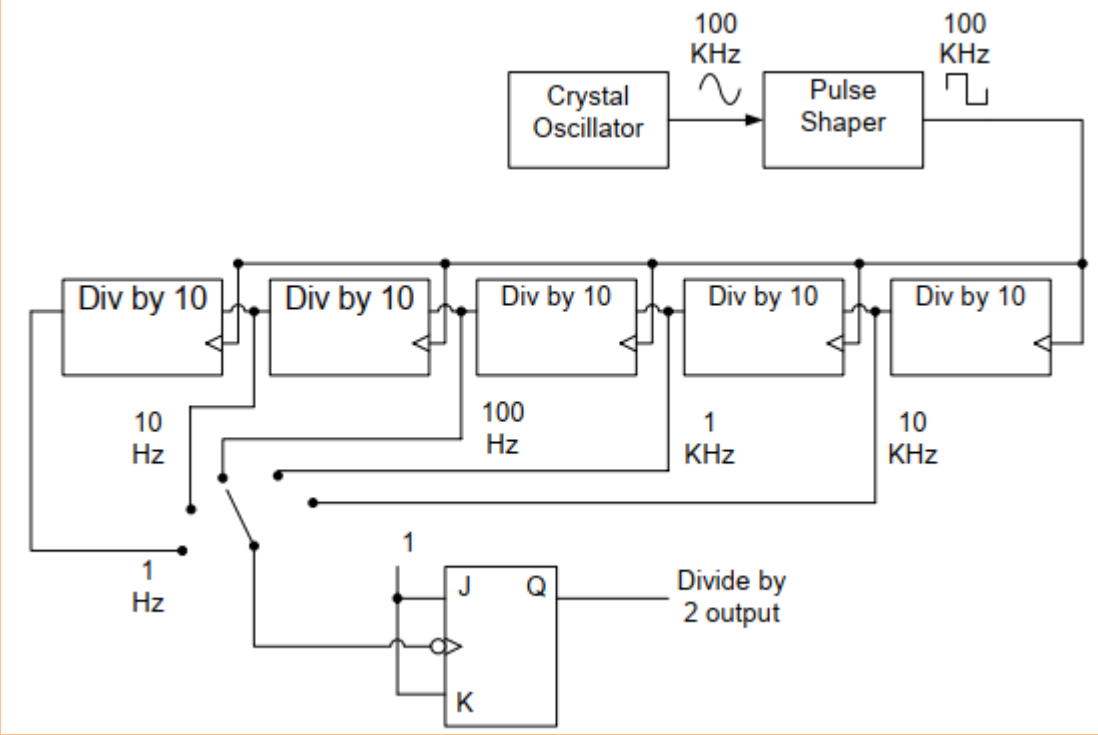
74x138 Decoder enabled by a clock signal



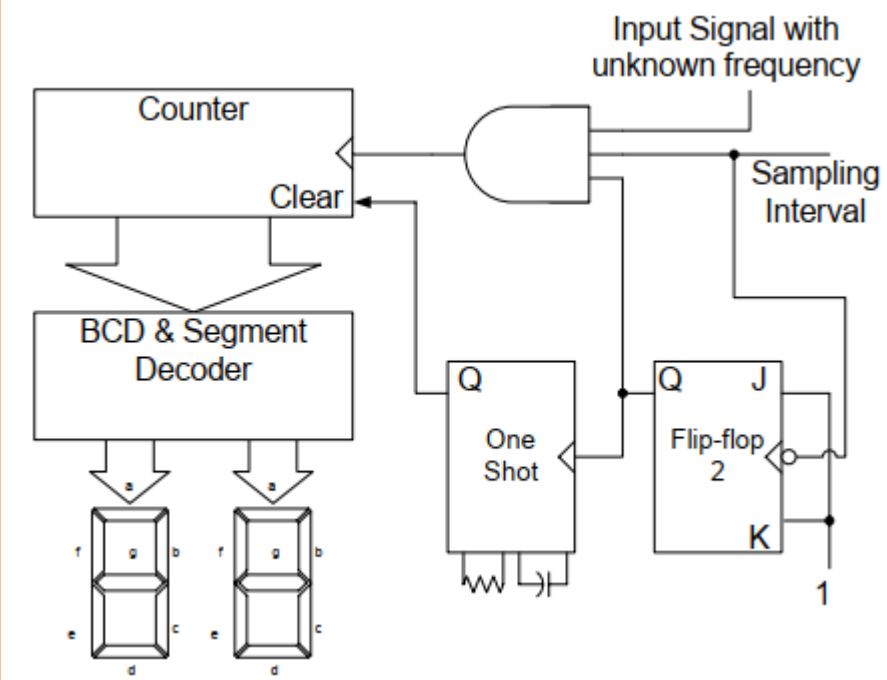
Frequency Counter Circuit



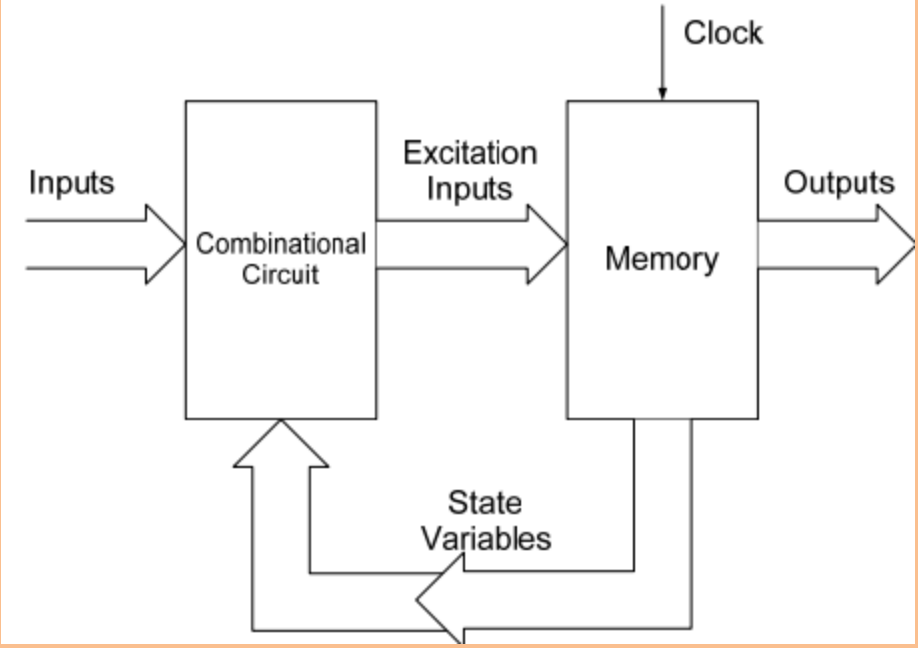
Cascaded Counter circuit for obtaining accurate sampling intervals



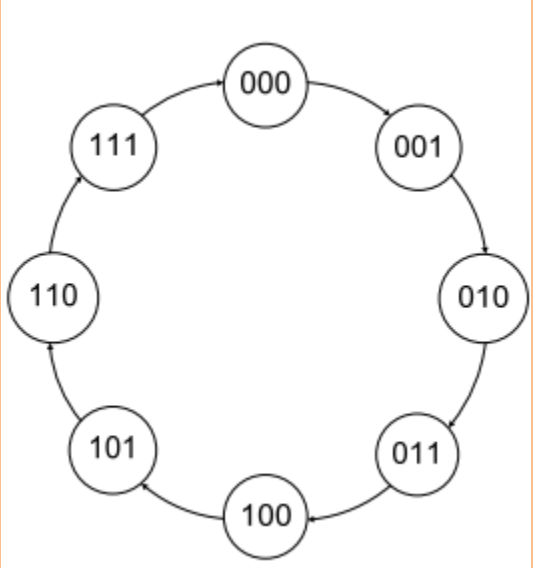
Detailed circuit diagram of a frequency counter



Sequential Circuit (State Machine) Clocked Sequential Circuit Block diagram



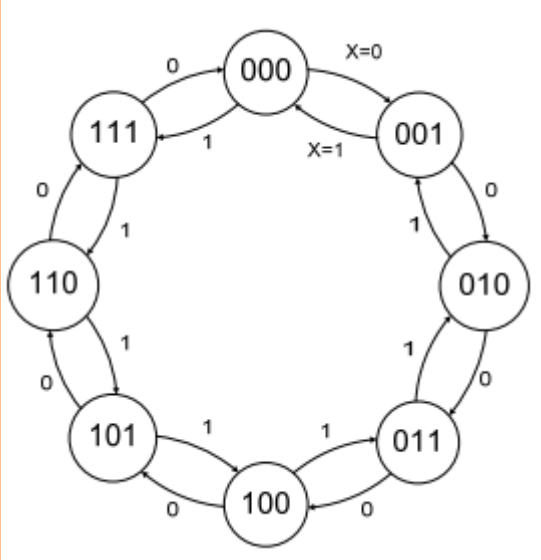
State diagram of a 3-bit Up-Counter



Next-State Table for a 3-bit Up-Counter

Present State			Next State		
Q_2	Q_1	Q_0	Q_2	Q_1	Q_0
0	0	0	0	0	1
0	0	1	0	1	0
0	1	0	0	1	1
0	1	1	1	0	0
1	0	0	1	0	1
1	0	1	1	1	0
1	1	0	1	1	1
1	1	1	0	0	0

State diagram of a 3-bit Up-Counter



Next-State Table for a 3-bit Up-Counter

Present State			Next State X=0			Next State X=1		
Q_2	Q_1	Q_0	Q_2	Q_1	Q_0	Q_2	Q_1	Q_0
0	0	0	0	0	1	1	1	1
0	0	1	0	1	0	0	0	0
0	1	0	0	1	1	0	0	1
0	1	1	1	0	0	0	1	0
1	0	0	1	0	1	0	1	1
1	0	1	1	1	0	1	0	0
1	1	0	1	1	1	1	0	1
1	1	1	0	0	0	1	1	0

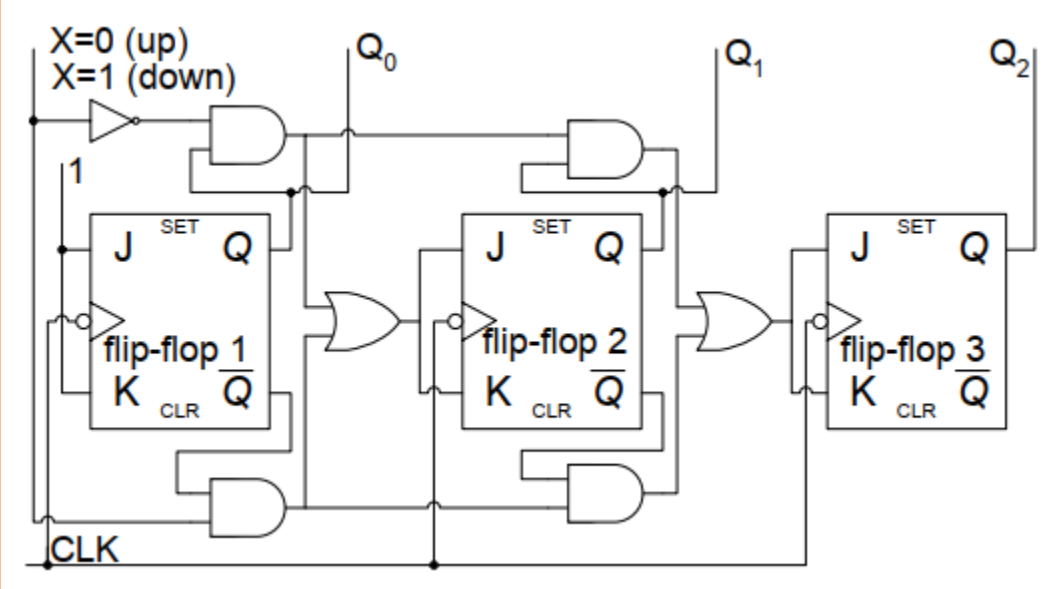
J-K flip-flop input table for X=0

Present State			Next State X=0			J-K flip-flop inputs					
Q_2	Q_1	Q_0	Q_2	Q_1	Q_0	J_2	K_2	J_1	K_1	J_0	K_0
0	0	0	0	0	1	0	x	0	x	1	x
0	0	1	0	1	0	0	x	1	x	x	1
0	1	0	0	1	1	0	x	x	0	1	x
0	1	1	1	0	0	1	x	x	1	x	1
1	0	0	1	0	1	x	0	0	x	1	x
1	0	1	1	1	0	x	0	1	x	x	1
1	1	0	1	1	1	x	0	x	0	1	x
1	1	1	0	0	0	x	1	x	1	x	1

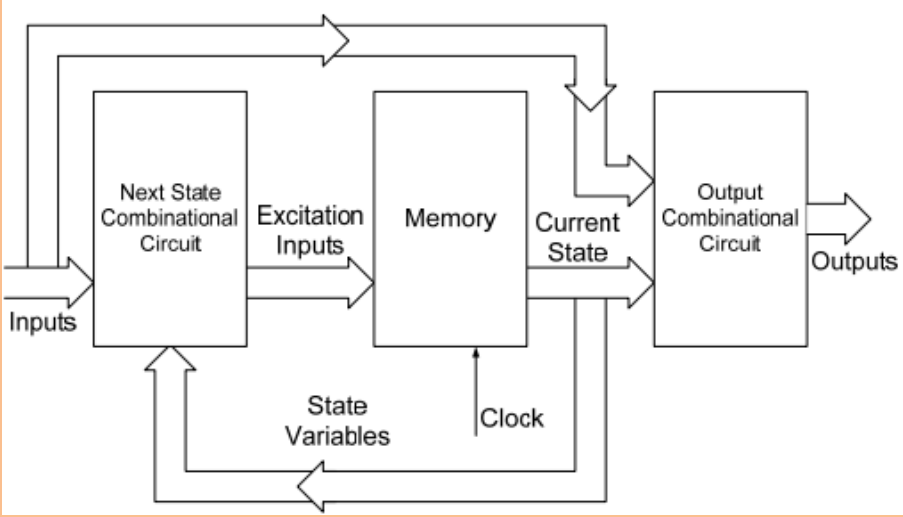
J-K flip-flop input table for X=1

Present State			Next State X=1			J-K flip-flop inputs					
Q_2	Q_1	Q_0	Q_2	Q_1	Q_0	J_2	K_2	J_1	K_1	J_0	K_0
0	0	0	1	1	1	1	x	1	x	1	x
0	0	1	0	0	0	0	x	0	x	x	1
0	1	0	0	0	1	0	x	x	1	1	x
0	1	1	0	1	0	0	x	x	0	x	1
1	0	0	0	1	1	x	1	1	x	1	x
1	0	1	1	0	0	x	0	0	x	x	1
1	1	0	1	0	1	x	0	x	1	1	x
1	1	1	1	1	0	x	0	x	0	x	1

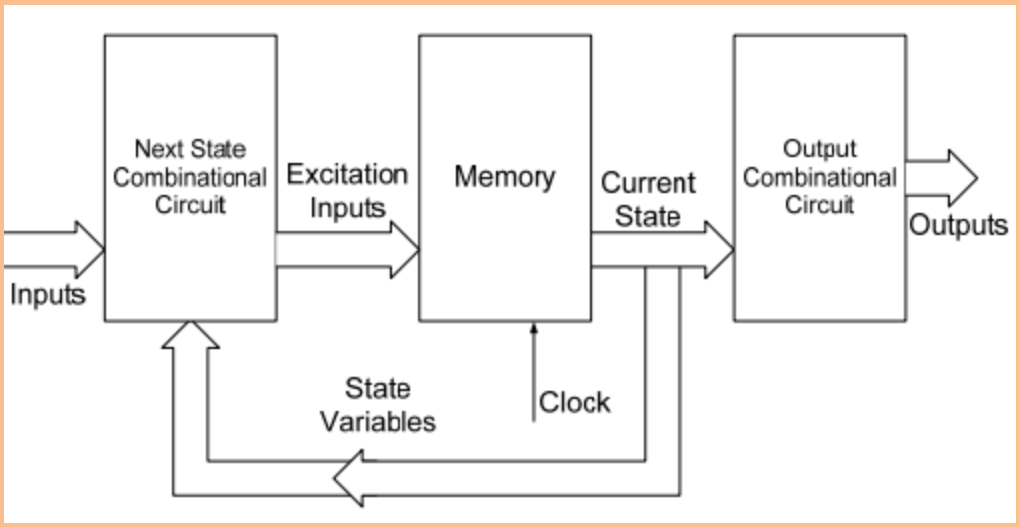
Implementation of the Sequential Circuit



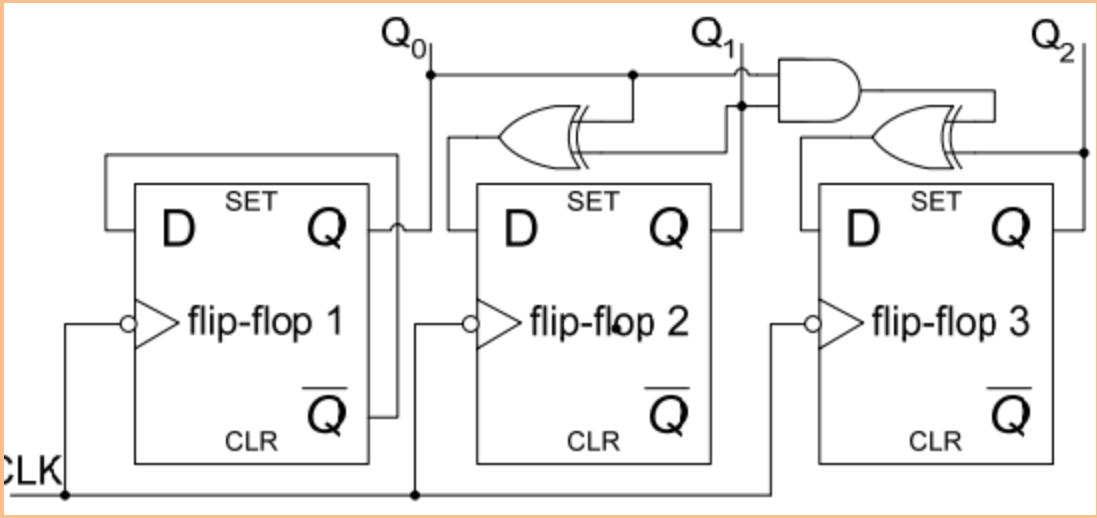
Clocked Sequential State Machine (Mealy Machine)



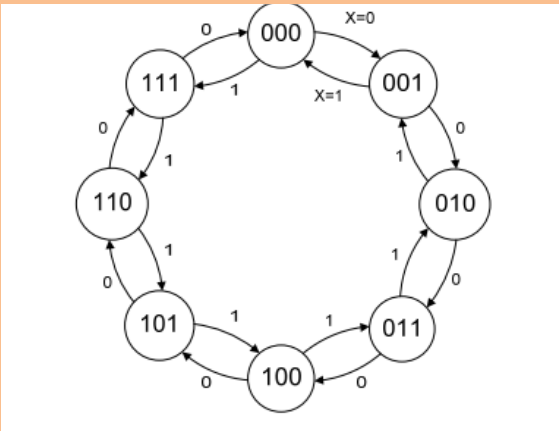
Clocked Sequential State Machine (Moore Machine)



D flip-flop based implementation of 3-bit Synchronous Counter



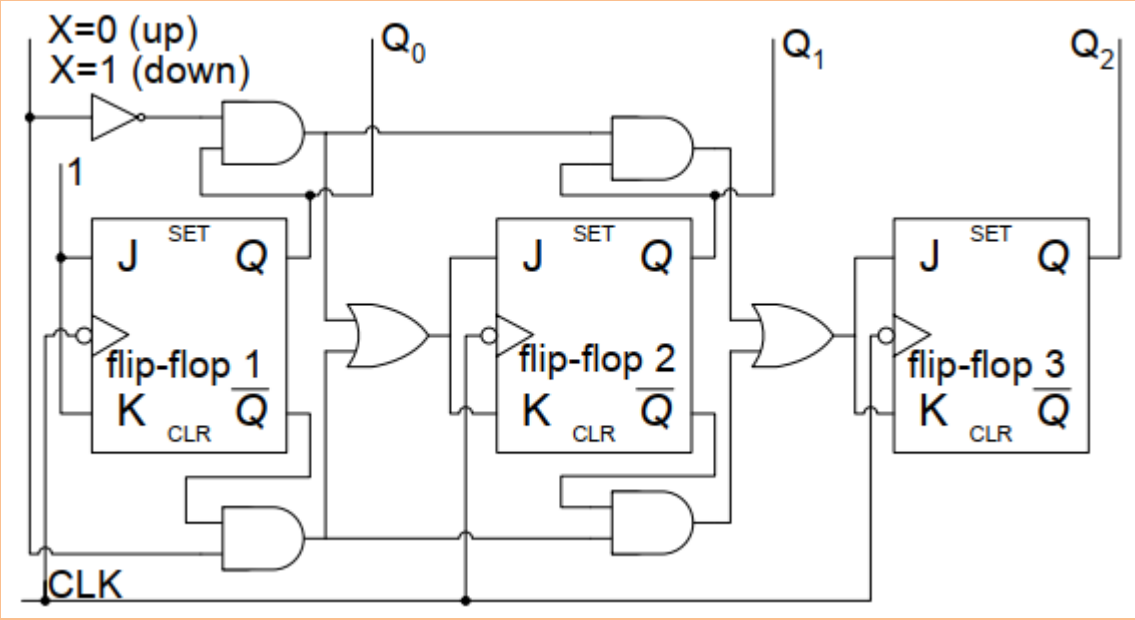
State diagram of a 3-bit Up-Counter



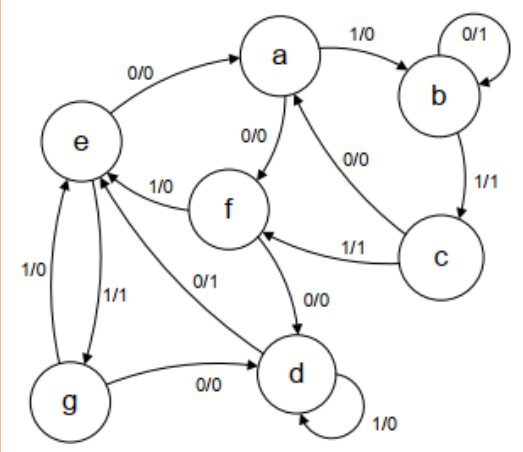
Next-State Table for a 3-bit Up-Counter

Present State			Next State X=0			Next State X=1		
Q_2	Q_1	Q_0	Q_2	Q_1	Q_0	Q_2	Q_1	Q_0
0	0	0	0	0	1	1	1	1
0	0	1	0	1	0	0	0	0
0	1	0	0	1	1	0	0	1
0	1	1	1	0	0	0	1	0
1	0	0	1	0	1	0	1	1
1	0	1	1	1	0	1	0	0
1	1	0	1	1	1	1	0	1
1	1	1	0	0	0	1	1	0

Implementation of the Sequential Circuit



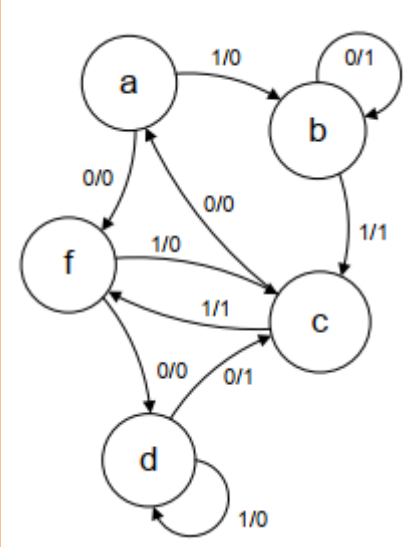
State Reduction \ State diagram



The input and output table of **State Reduction** sequence

state	a	b	c	f	d	d	e	g	d	e	a	f	d	e	a
Input	1	1	1	0	1	0	1	0	0	0	0	0	0	0	
Output	0	1	1	0	0	1	1	0	1	0	0	0	1	0	

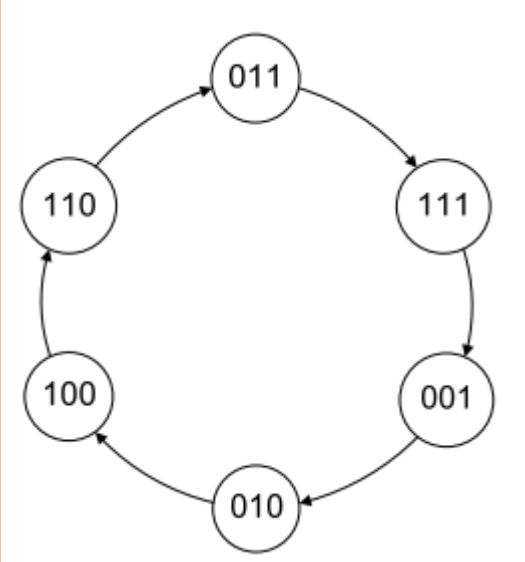
Simplified State diagram



The input and output sequence obtained from the simplified state diagram

state	a	b	c	f	d	d	c	f	d	c	a	f	d	c	a
Input	1	1	1	0	1	0	1	0	0	0	0	0	0	0	
Output	0	1	1	0	0	1	1	0	1	0	0	0	1	0	

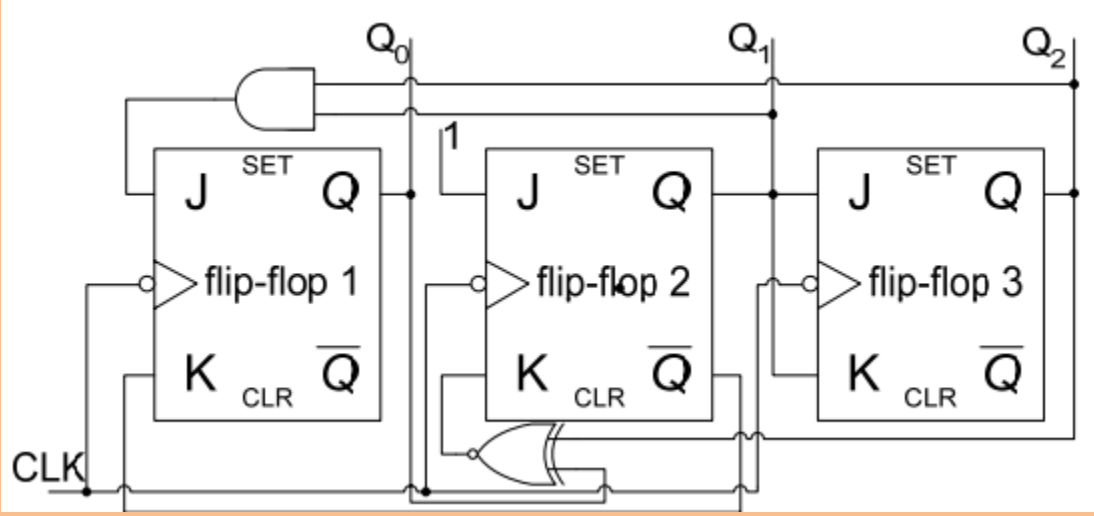
State diagram of a Moore Machine



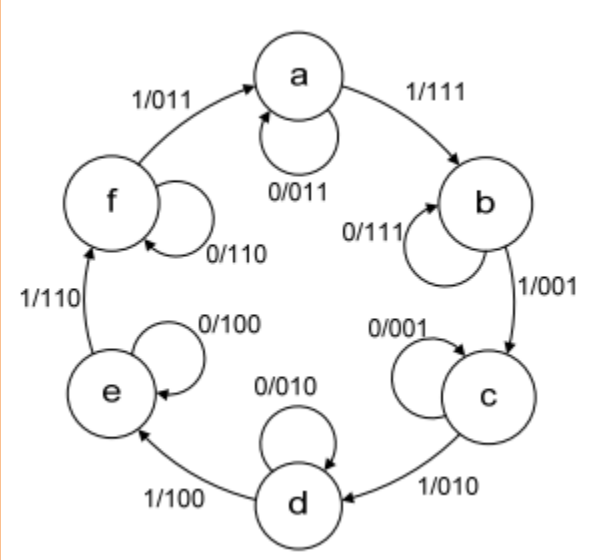
Next-State table of the Moore Machine

Present State			Next State		
Q_2	Q_1	Q_0	Q_2	Q_1	Q_0
0	1	1	1	1	1
1	1	1	0	0	1
0	0	1	0	1	0
0	1	0	1	0	0
1	0	0	1	1	0
1	1	0	0	1	1

Implementation of the Moore Machine



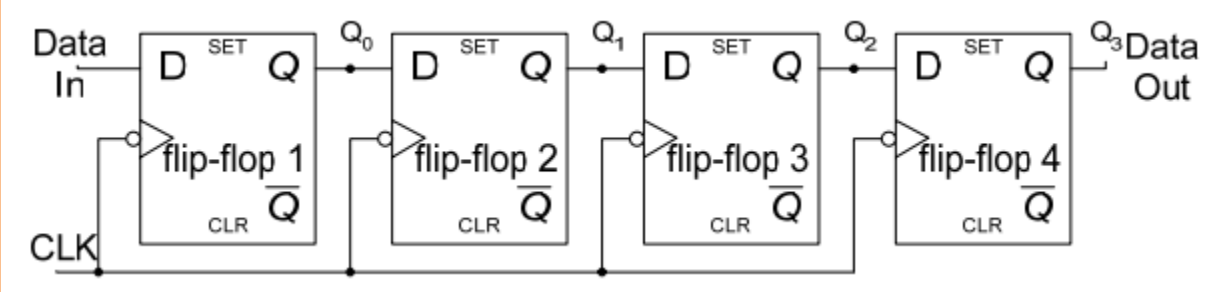
State diagram of a Mealy Machine



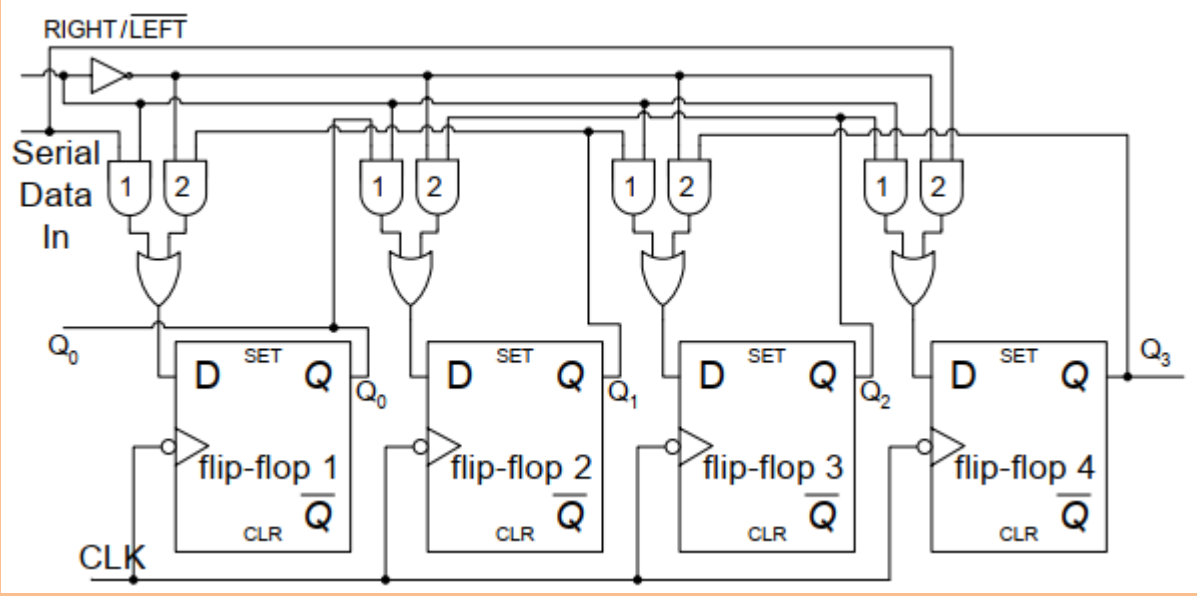
Next-State table of a Mealy Machine

Present State	Next State		Output	
	X=0	X=1	X=0	X=1
a	a	b	011	111
b	b	c	111	001
c	c	d	001	010
d	d	e	010	100
e	e	f	100	110
f	f	a	110	011

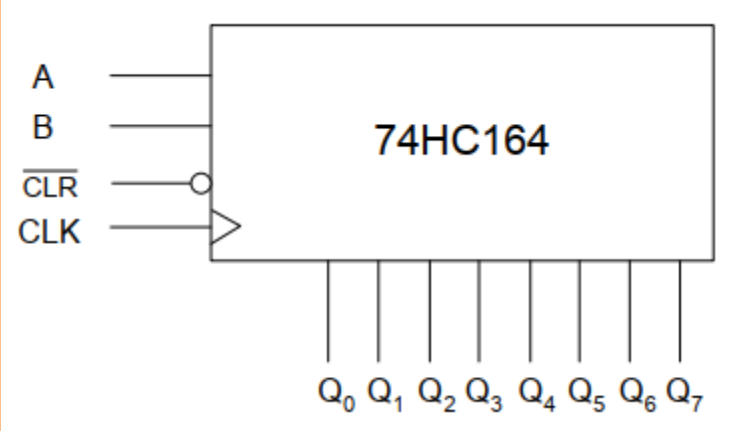
Serial In/Shift Right/Serial Out Register



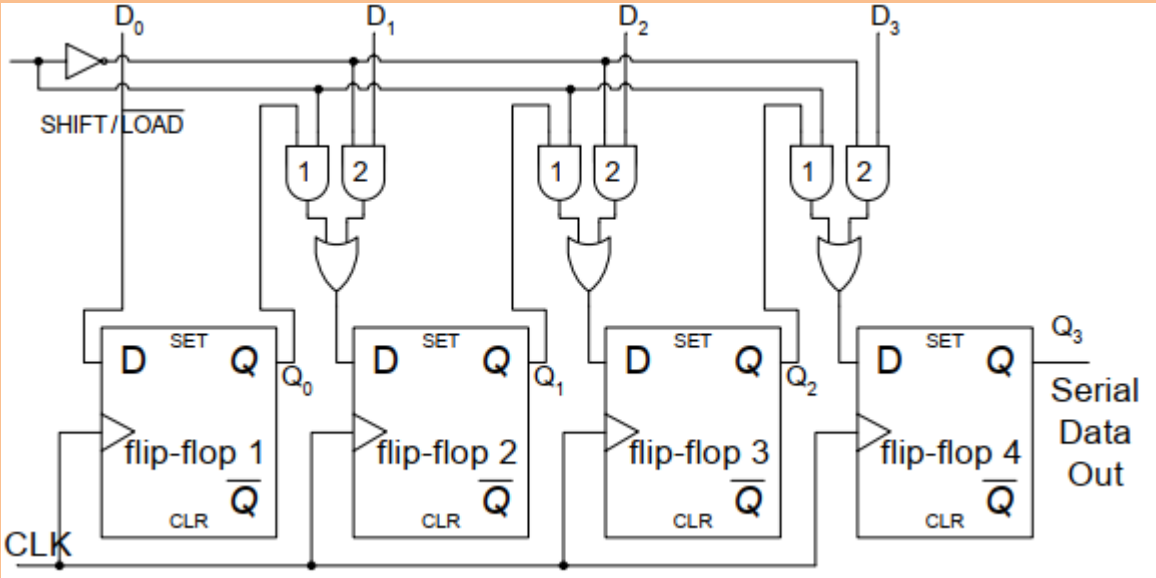
Bi-directional, 4-bit Shift register



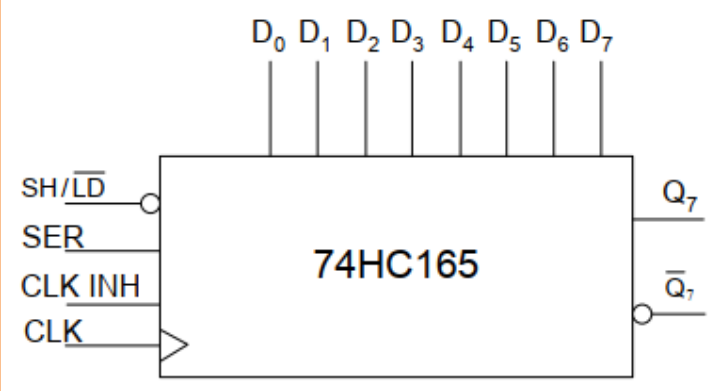
74HC164, 8-bit Serial In/Parallel Out Shift Register



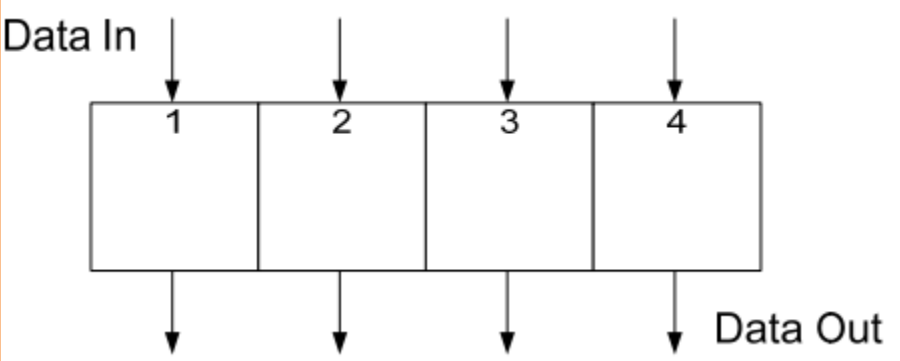
4-bit Parallel In/Serial Out Shift register



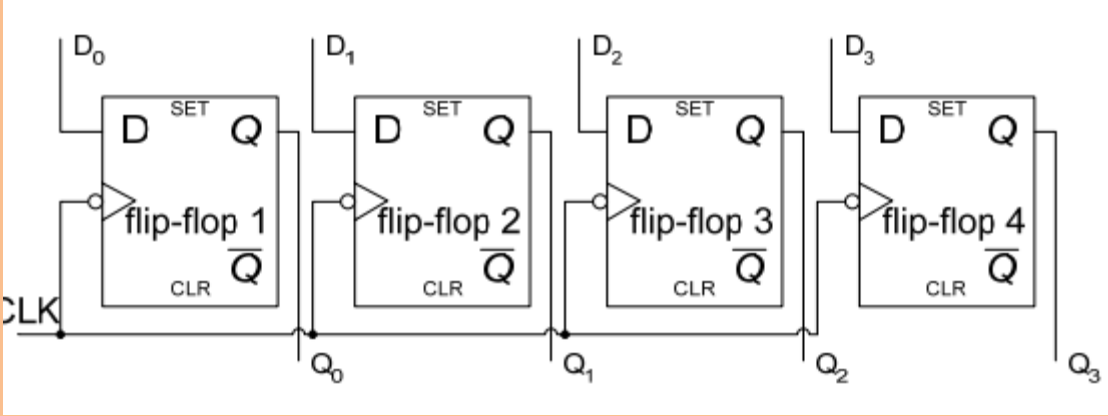
74HC165, 8-bit Parallel In/Serial Out Shift Register



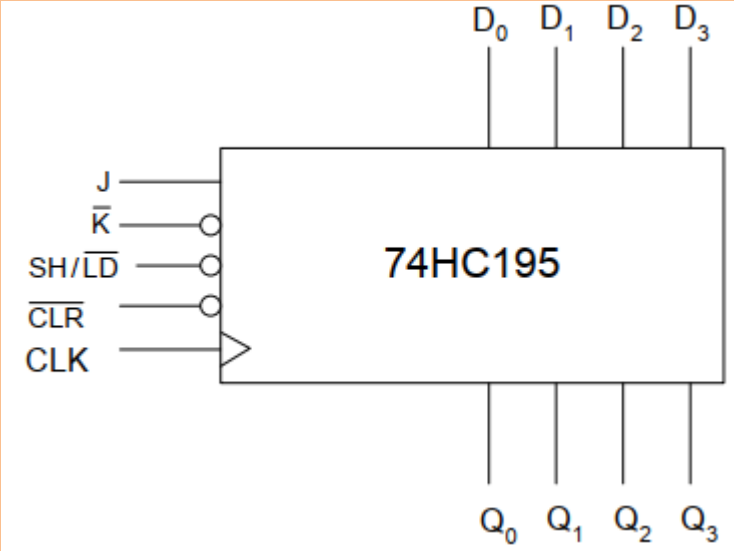
Parallel In/Parallel Out Operation



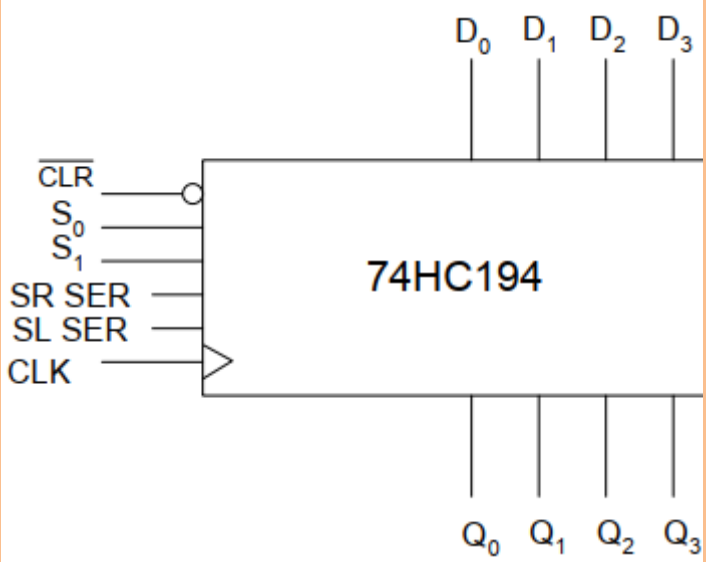
A D-flip-flop based 4-bit Parallel In/Parallel Out Register



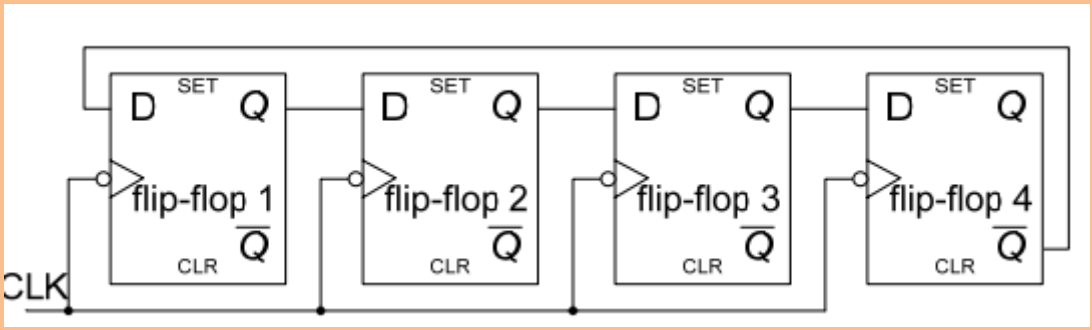
74HC195, 4-bit Parallel In/Parallel Out Shift Register



Bi-directional 4-bit Universal Shift Register



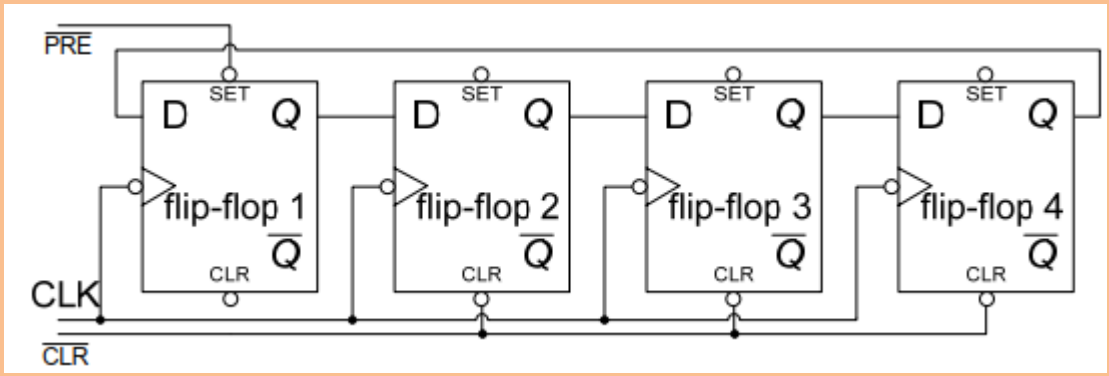
4-bit Johnson Counter



Sequence of states of a 4-bit Johnson Counter

Clock Pulse	Q_0	Q_1	Q_2	Q_3
0	0	0	0	0
1	1	0	0	0
2	1	1	0	0
3	1	1	1	0
4	1	1	1	1
5	0	1	1	1
6	0	0	1	1
7	0	0	0	1

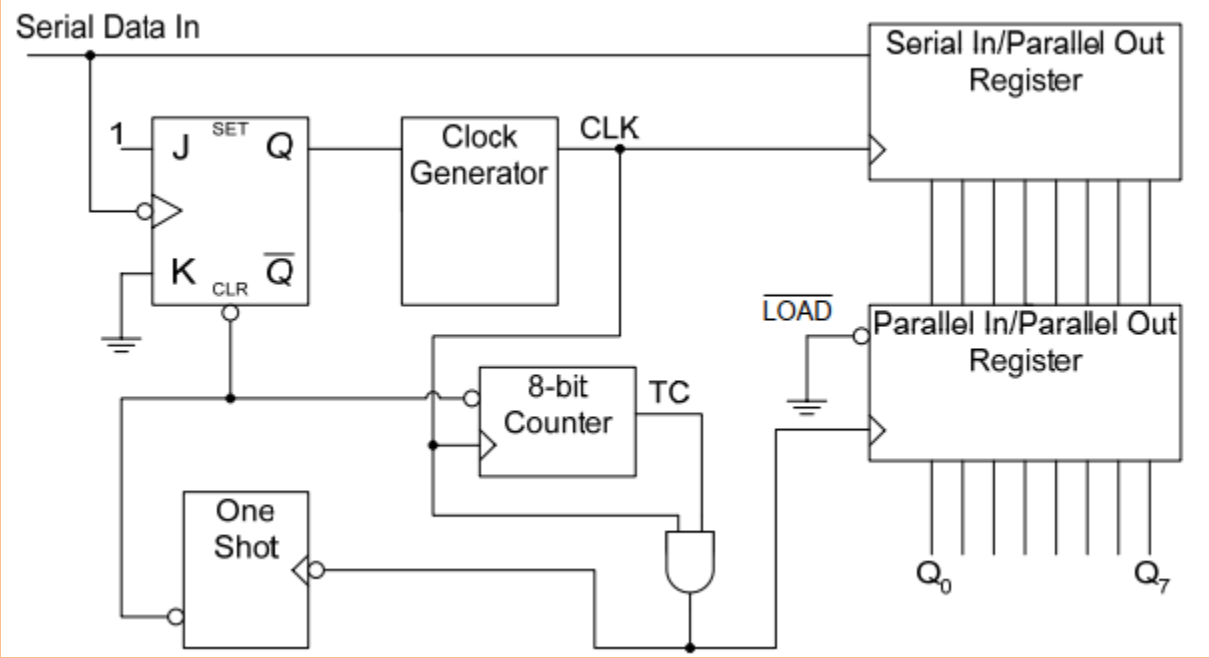
4-bit Ring Counter



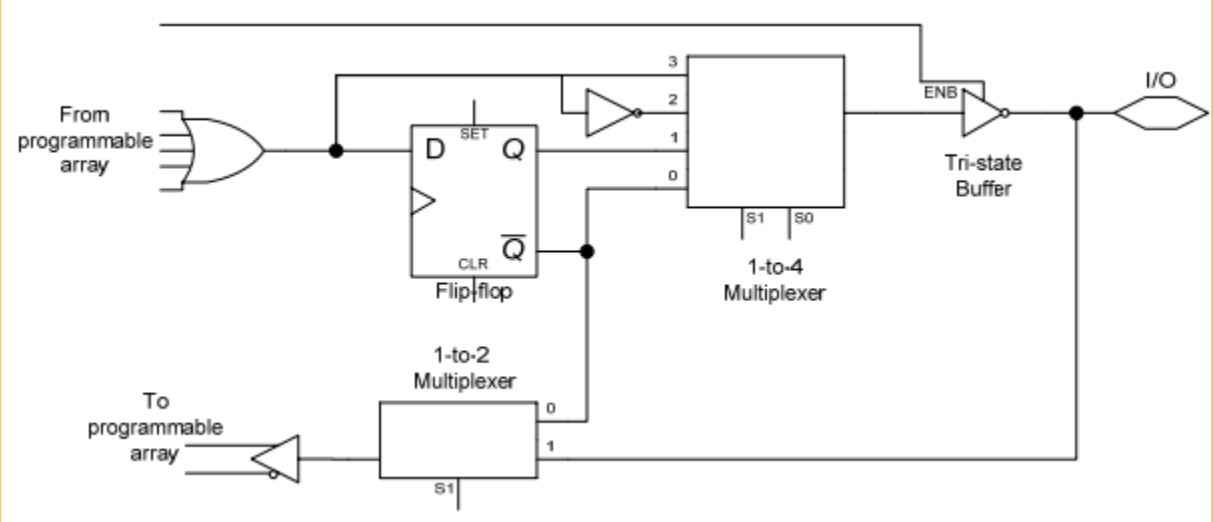
Sequence of states of a 4-bit Ring Counter

Clock Pulse	Q_0	Q_1	Q_2	Q_3
0	1	0	0	0
1	0	1	0	0
2	0	0	1	0
3	0	0	0	1

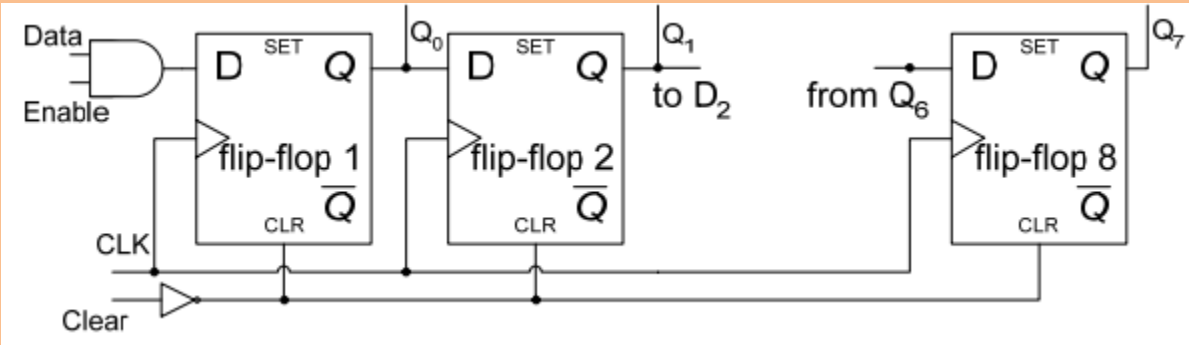
Series-to-Parallel Converter



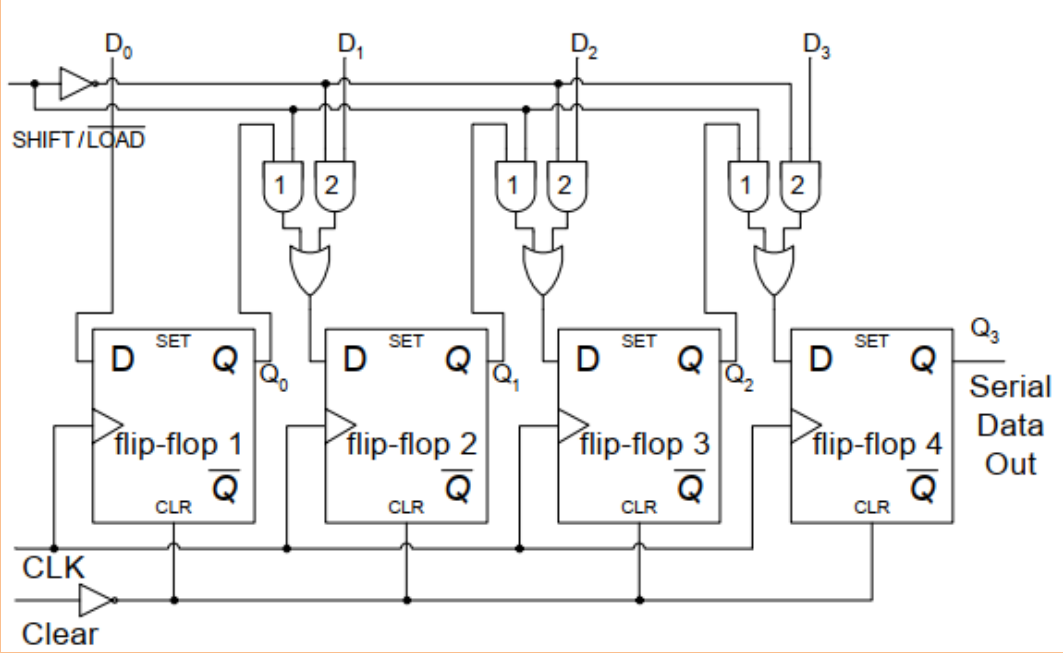
OLMC of the GAL22V10 device



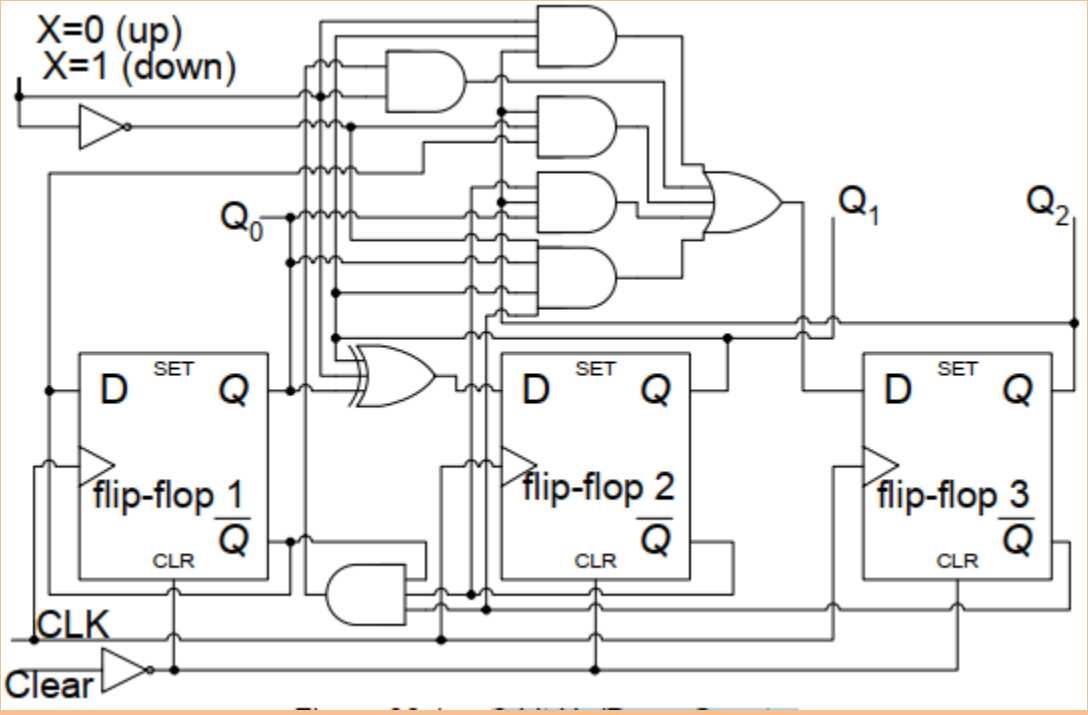
8-bit Serial In/Parallel Out Shift Register



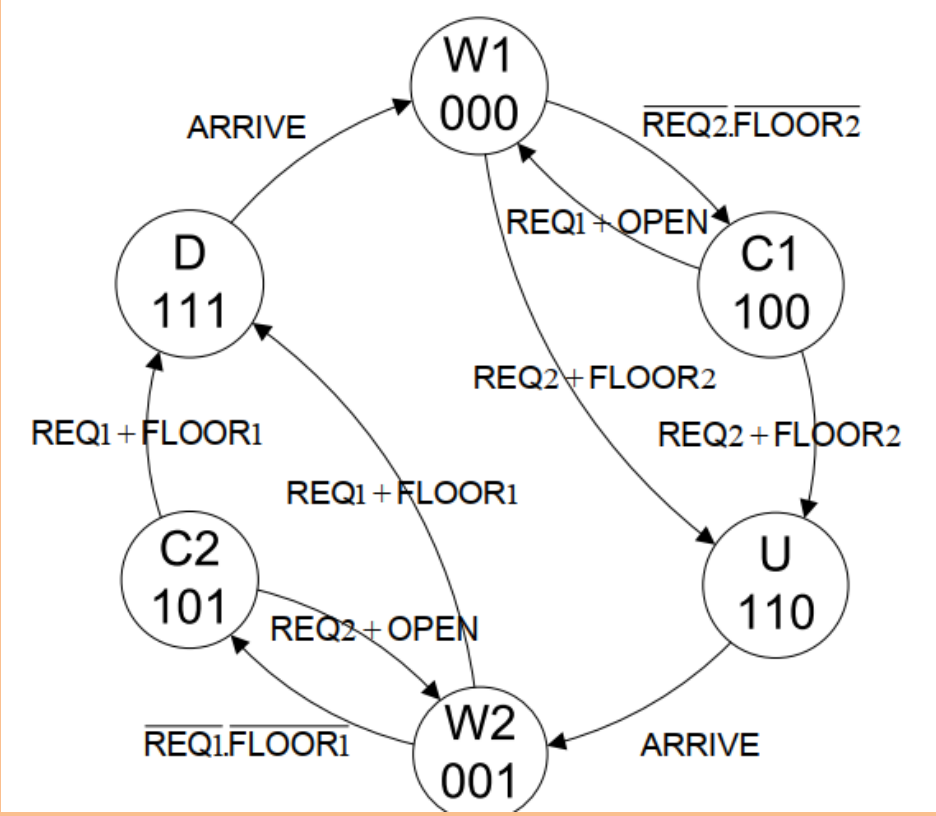
4-bit Parallel In/Serial Out Shift Register



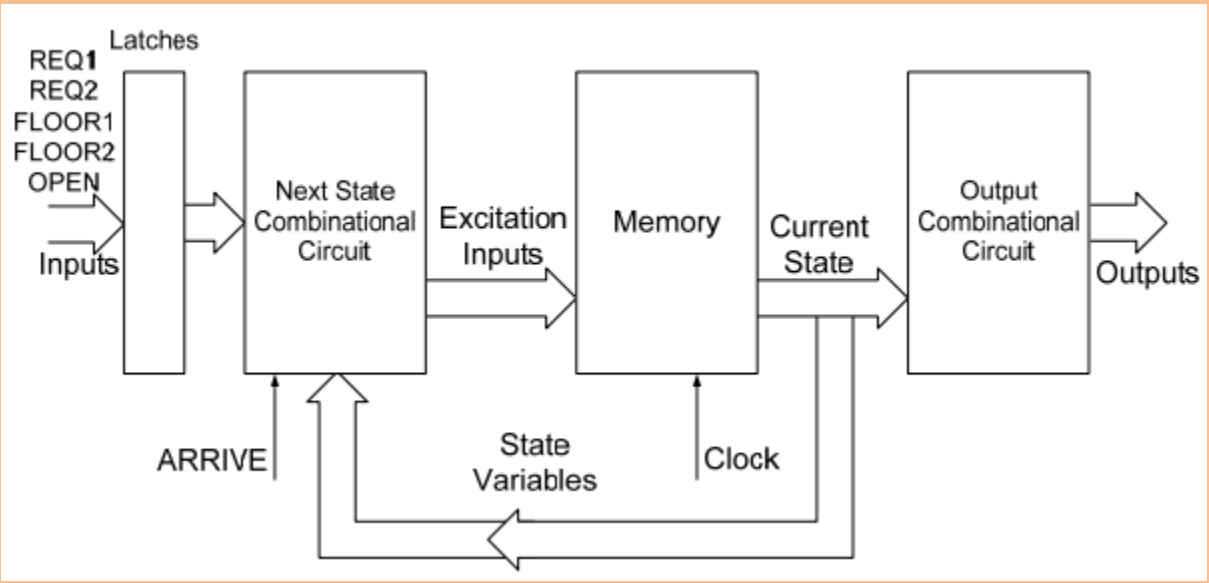
3-bit Up/Down Counter



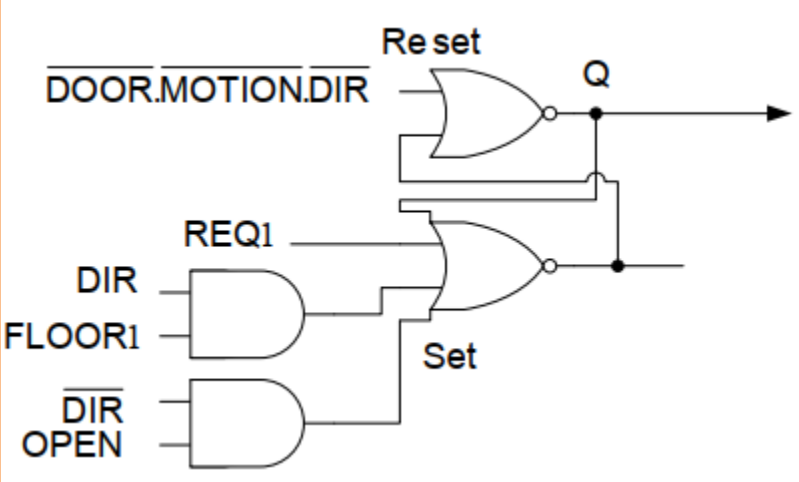
State Diagram of Elevator



Block diagram of the Elevator State Machine



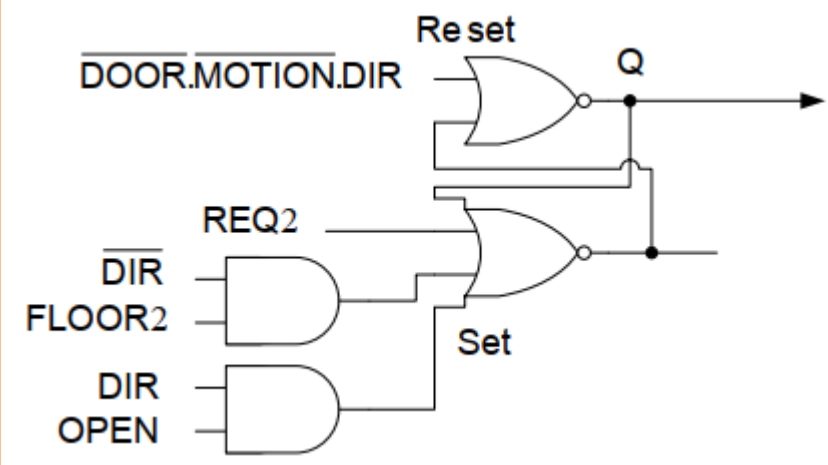
The circuit diagram of the SR1 latch \ SR1 latch which stores the status of the REQ1, FLOOR1 and OPEN buttons



Simplified State table for Elevator Control for REQ1, FLOOR1 and OPEN inputs

Present State	Next State	Next State
	SR1=0	SR1=1
W1(000)	x	x
C1(100)	C1	W1
UP(110)	x	x
W2(001)	C2	DO
C2(101)	C2	DO
DO(111)	x	x

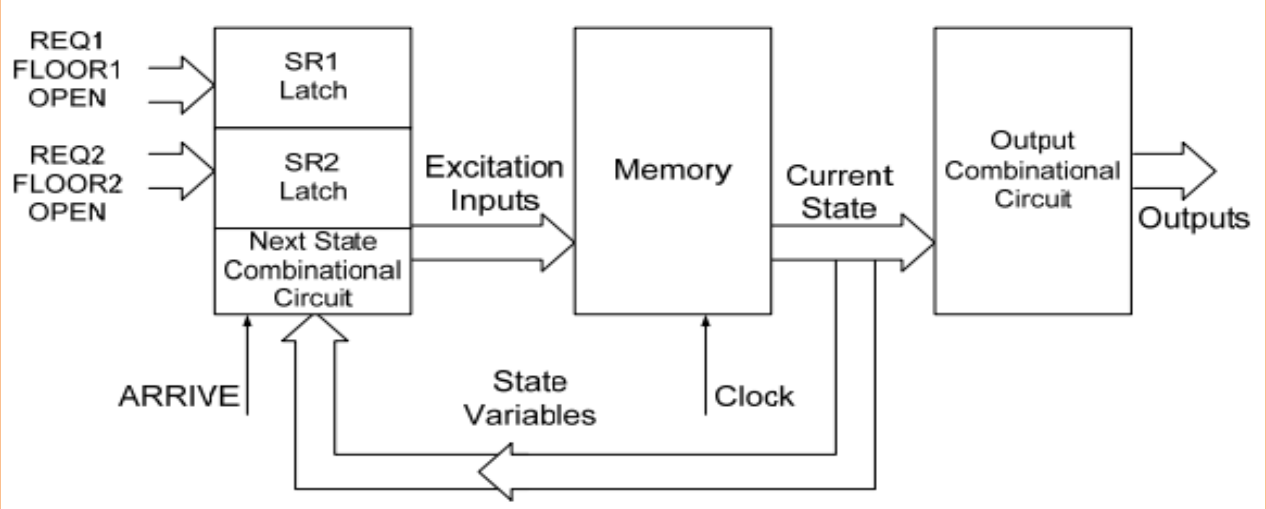
The circuit diagram and the Boolean expression can similarly be represented \ SR2 latch which stores the status of the REQ2, FLOOR2 and OPEN buttons



Simplified State table for Elevator Control for REQ2, FLOOR2 and OPEN inputs

Present State	Next State	Next State
	SR2=0	SR2=1
W1(000)	C1	UP
C1(100)	C1	UP
UP(110)	x	x
W2(001)	x	x
C2(101)	C2	W2
DO(111)	x	x

Modified Block diagram of the Elevator State Machine \ The Next State Table for the Elevator State Machine based on the inputs SR1, SR2 and ARRIVAL



Traffic Signal Controller Inputs and Outputs

The State Machine which controls the Traffic Signal has several inputs and outputs.
The inputs are the

• NSSR: The NSSR is activated when a car is over either of the four sensors on the NorthSouth section of the road

• EWSR: The EWSR is activated when a car is over either of the four sensors on the EastWest section of the road

A Timer is used to count the 5 minute and 1 minute traffic signal cycle during the day and
night. Two signals LTIME and STIME provide the timing inputs to the State Machine.

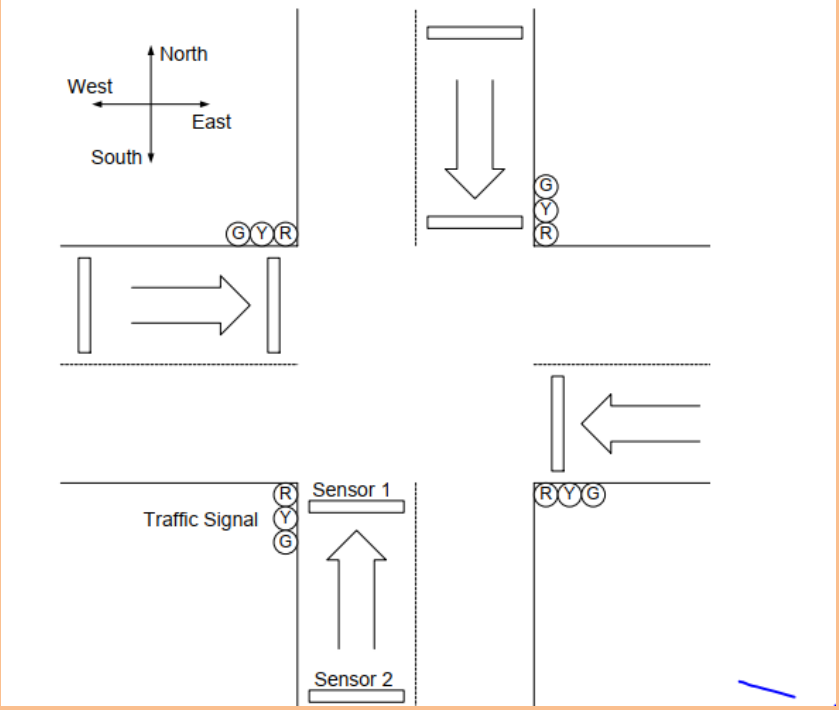
• LTIME: The LTIME signal is activated if 5 minutes have elapsed; the signal remains
active unless the timer is reset.

• STIME: The STIME signal is activated if 1 minute has

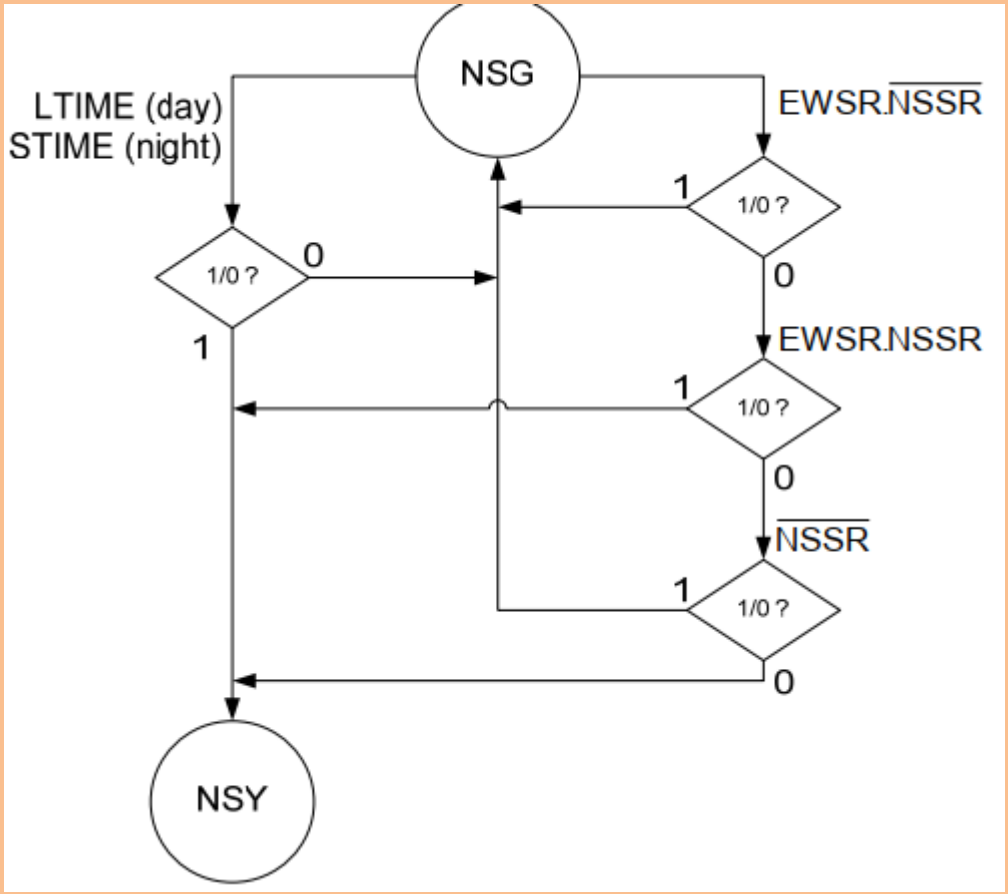
elapsed; the signal remains
active unless the timer is reset.
The outputs of the State Machine are
• NSGrn: The Green signal controlling the traffic on the North-South section
• NSYel: The Yellow signal controlling the traffic on the North-South section

NSRed: The Red signal controlling the traffic on the North-South section
• EWGrn: The Green signal controlling the traffic on the East-West section
• EWYel: The Yellow signal controlling the traffic on the East-West section
• EWRed: The Red signal controlling the traffic on the East-West section
• TMRST: The Reset signal which resets the timer after the LTIME or the STIME signals
are activated to indicate a time interval of 5 and 1 minutes respectively.

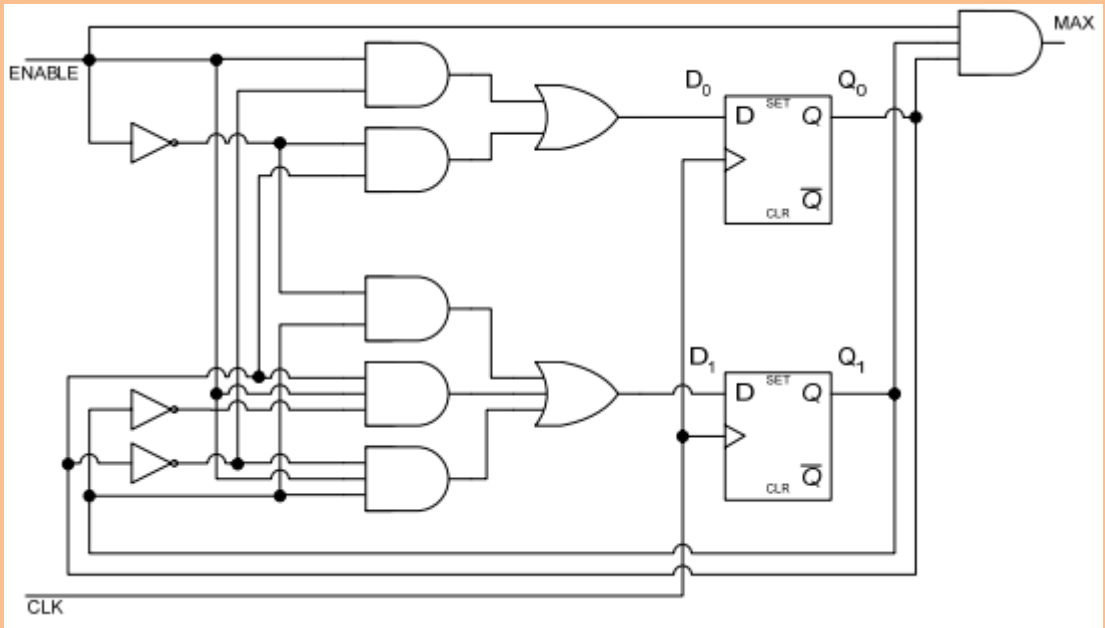
The Traffic signals and sensors at a Traffic Intersection



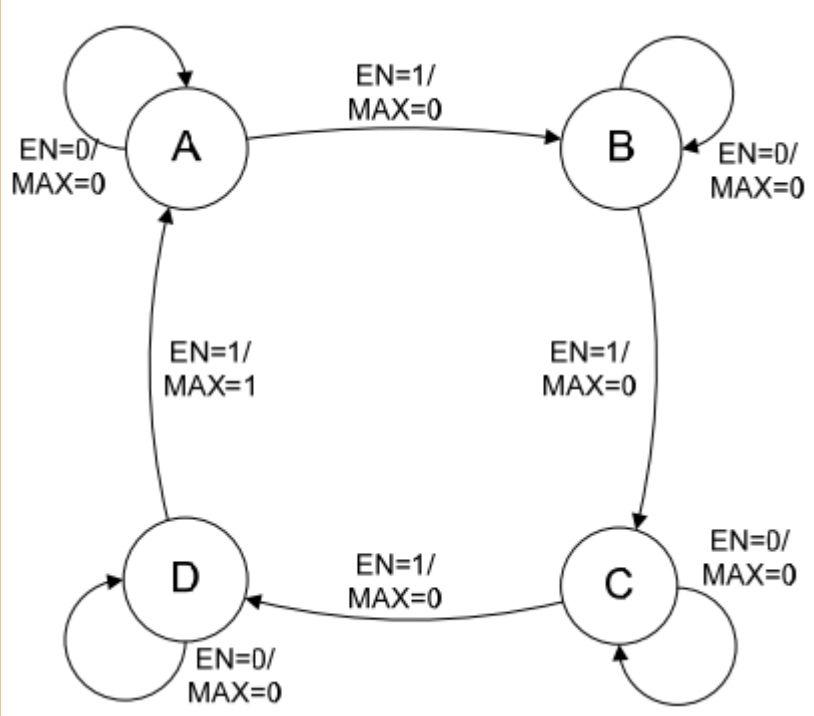
The State Machine can be implemented using a GAL16V8 device. \ Flow chart of conditions which switch the state from NSG to NSY



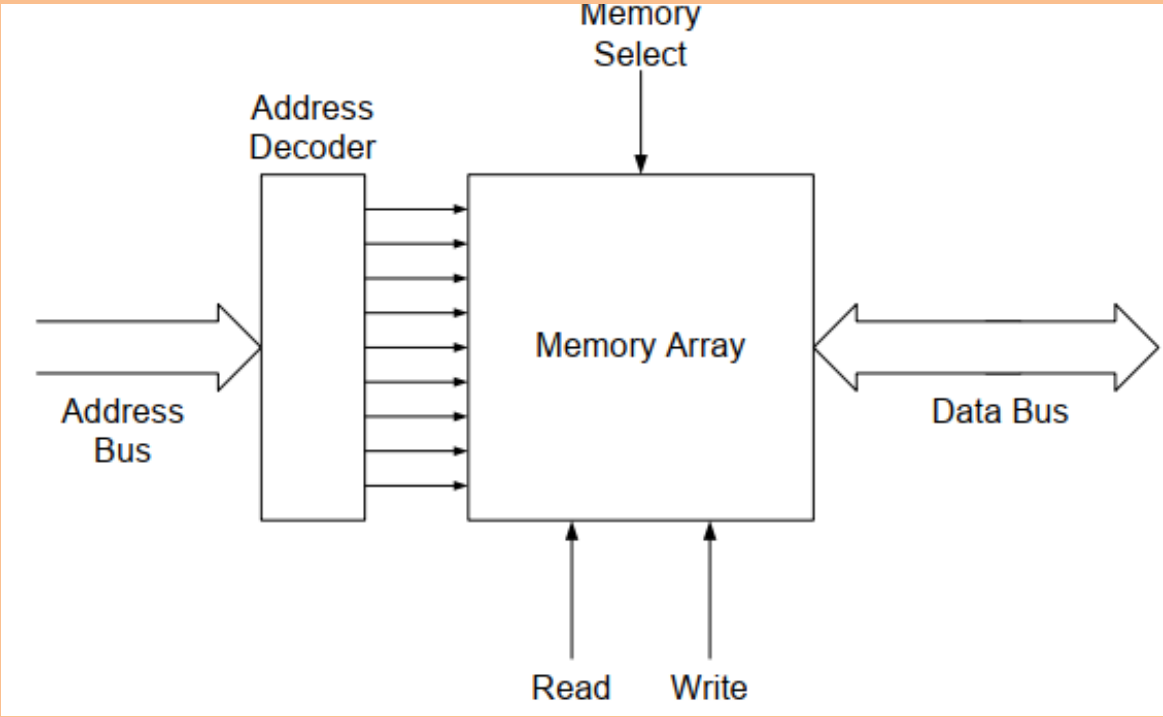
A State Machine with two positive-edge triggered D flip-flops



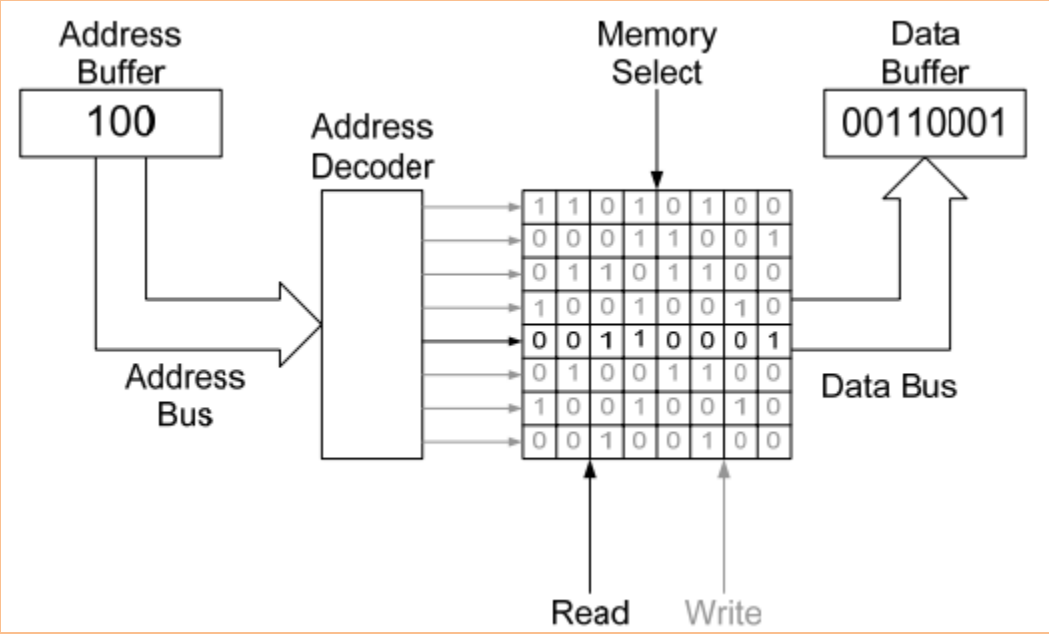
The State Diagrams for the Mealy State machine derived from the State



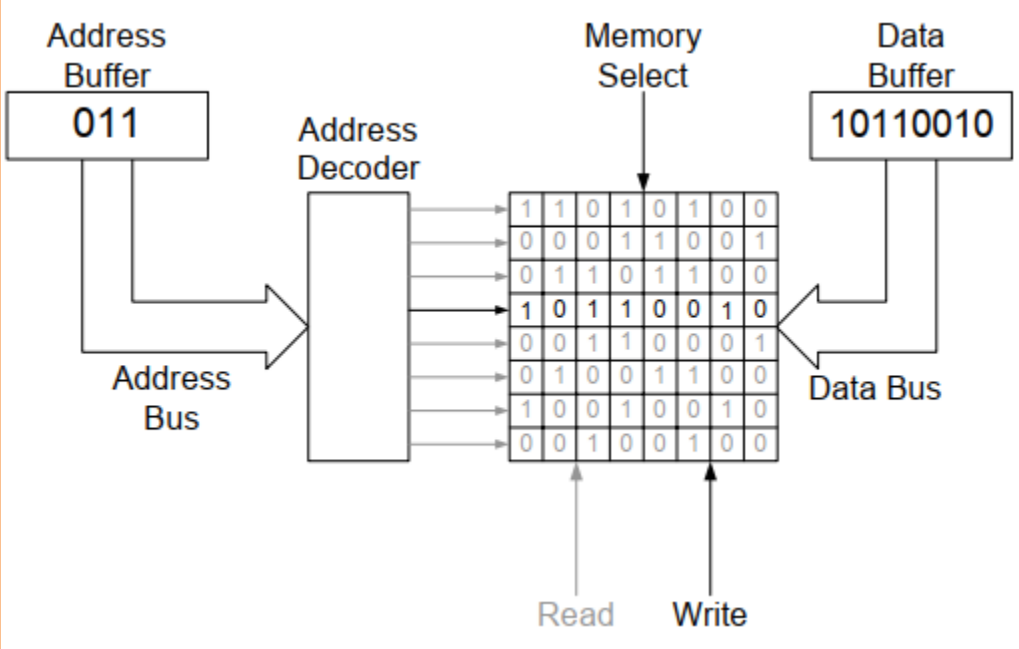
To support the two read and write operations memories provide several signals\ Block Diagram of a Read-write Memory



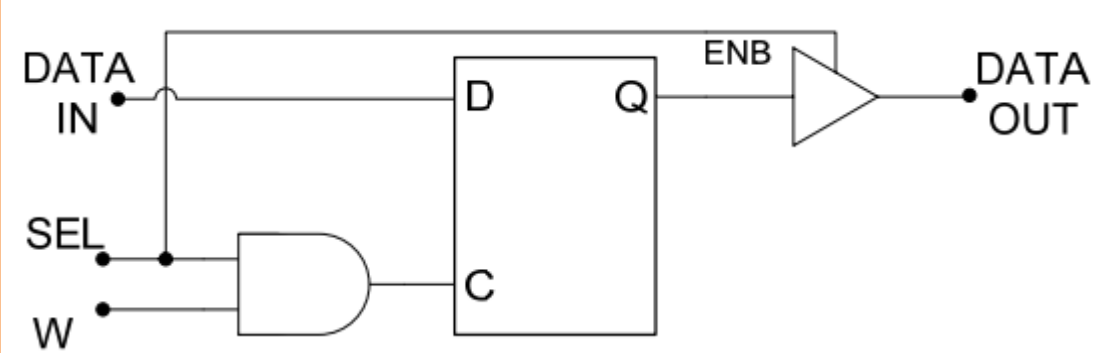
Memory Read Operation diagram



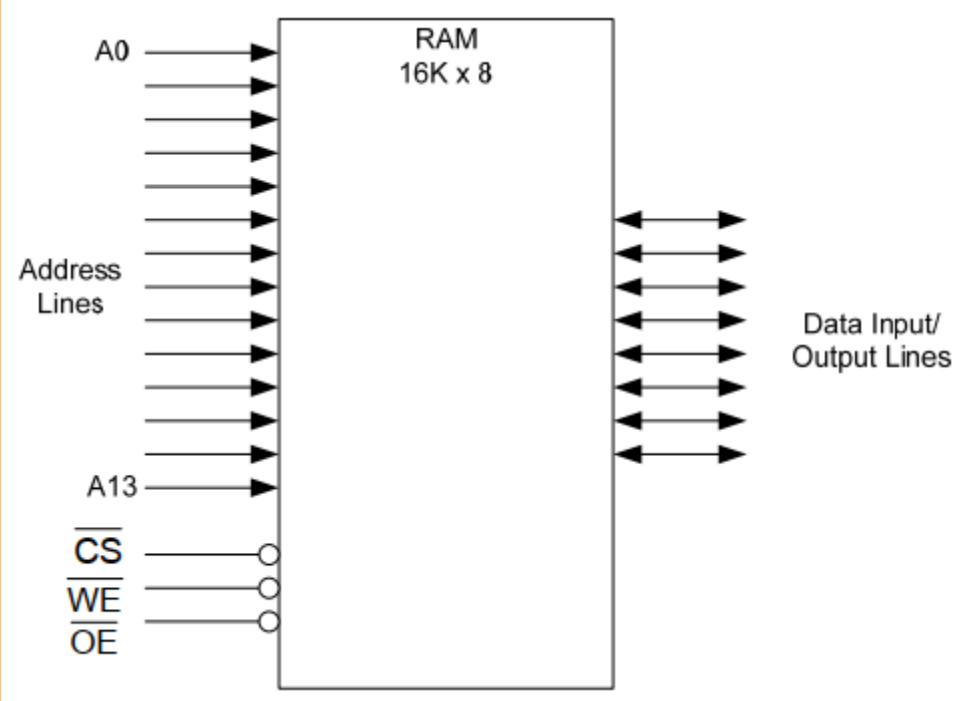
Memory write Operation diagram



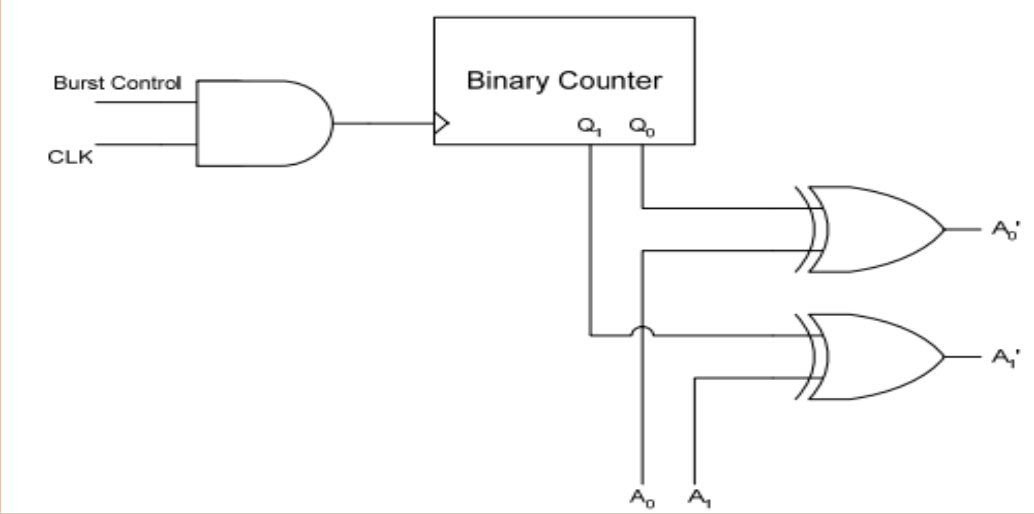
The circuit of a single flipflop based cell which can store a binary 0 or 1



16K x 8 Static RAM



Burst Logic Circuit

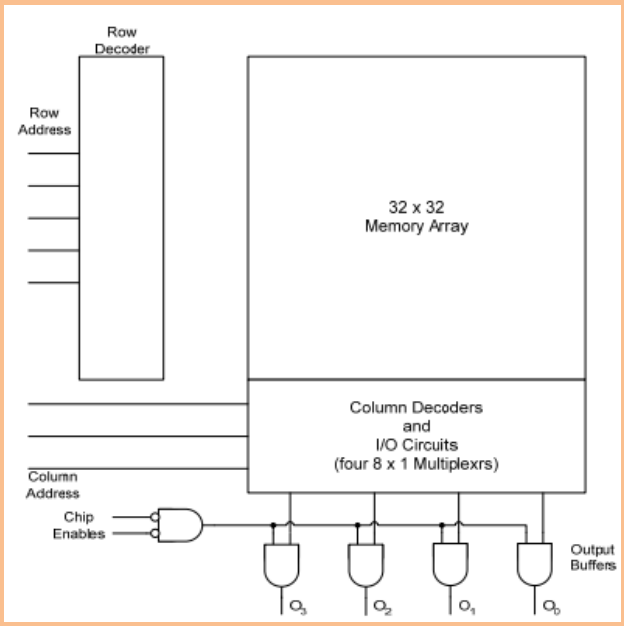


ROM Read-Only Memory

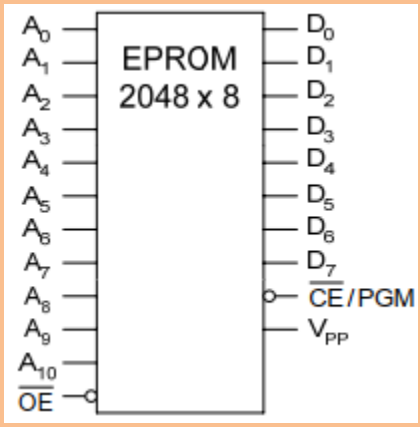
• Mask ROM: Data is permanently stored during the manufacturing process.

• PROM: Programmable ROM allows storage of data by the user using a PROM
programmer. The PROM once programmed stores the data permanently.
• EPROM: Erasable PROM allows erasing of stored data and reprogramming.
• UV EPROM: Is a programmable ROM. Data is erased by exposing the PROM to Ultraviolet
light.
• EEPROM: Electrically Erasable PROM is erased electrically. EEPROM allows in-circuit
programming and doesn’t need to be removed from the circuit for erasure or programming.

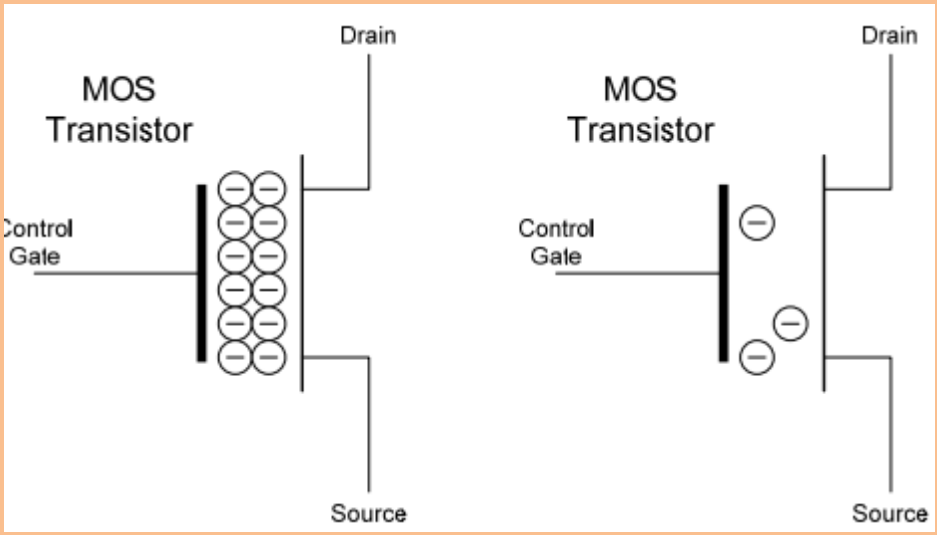
1 out of 8 column lines by four 8 x 1 Multiplexers
Internal structure of a 264 x 4 ROM



A typical UV EPROM memory chip\ A 2 KB EPROM



MOS transistor with charge (logic 0) and no charge (logic 1)

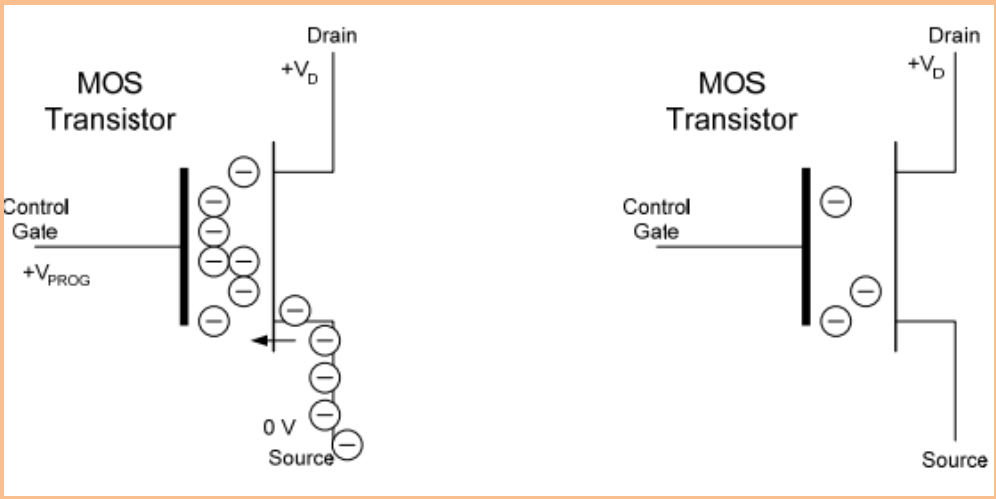


FLASH Memory Operations

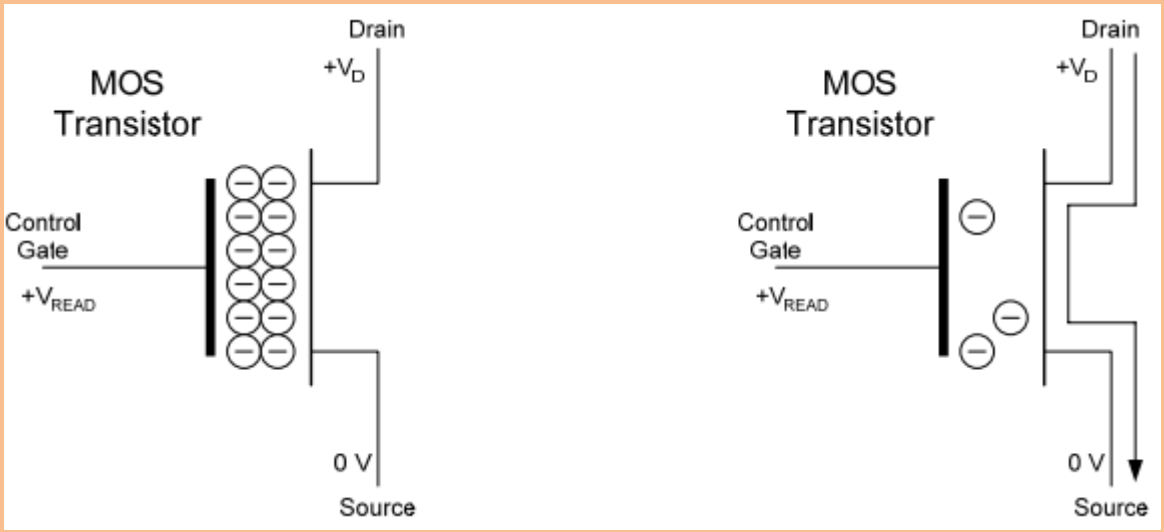
FLASH Memory operations are classified into

- Programming Operation
- Read Operation
- Erase Operation

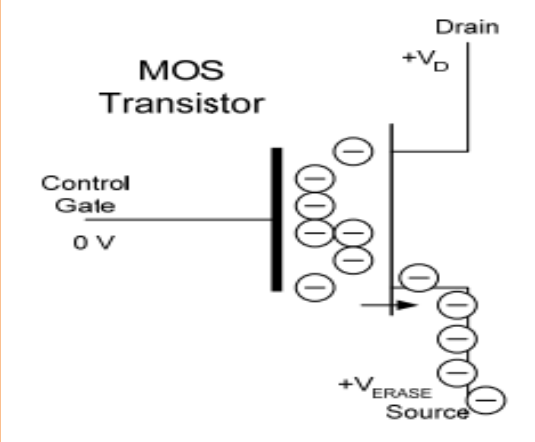
FLASH Memory Cell programmed with logic 0 and logic 1



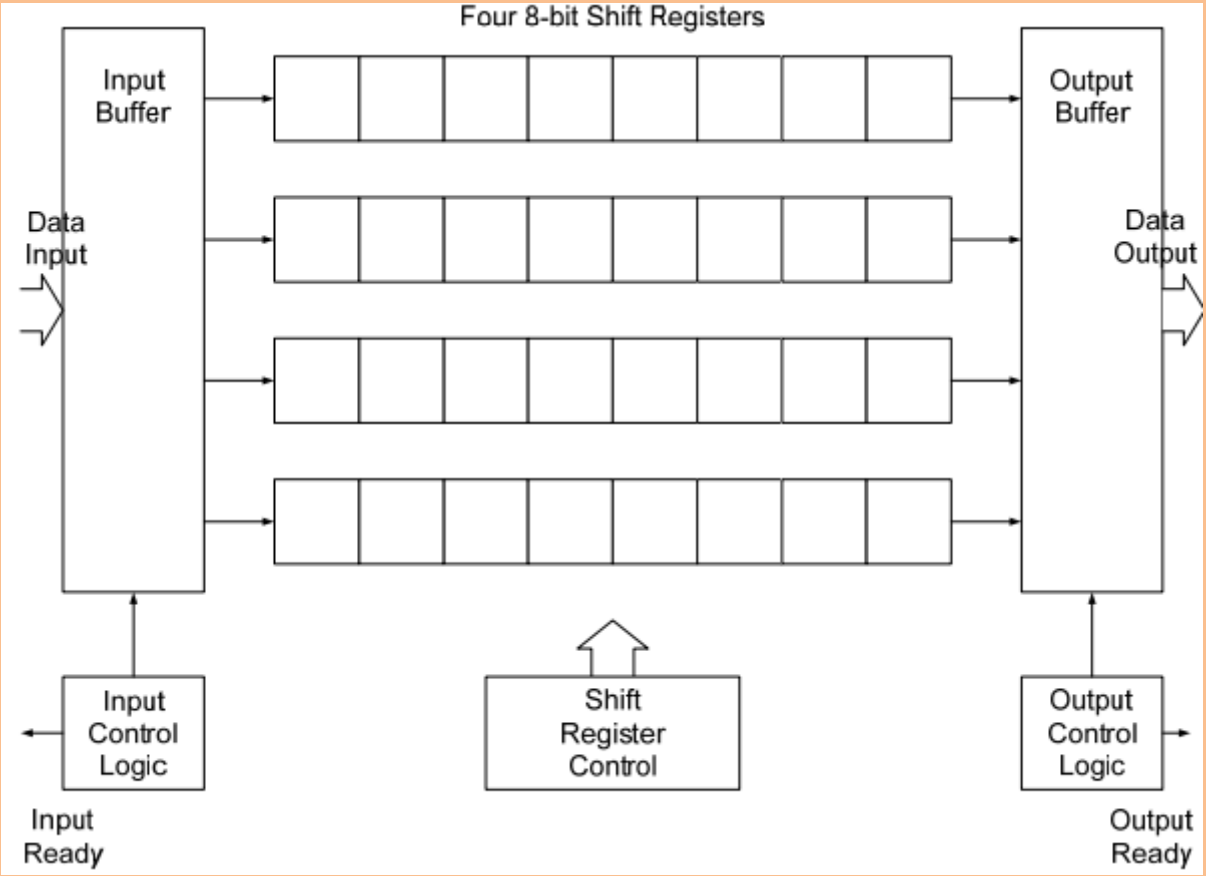
Read operation to read a logic 0 and a logic 1



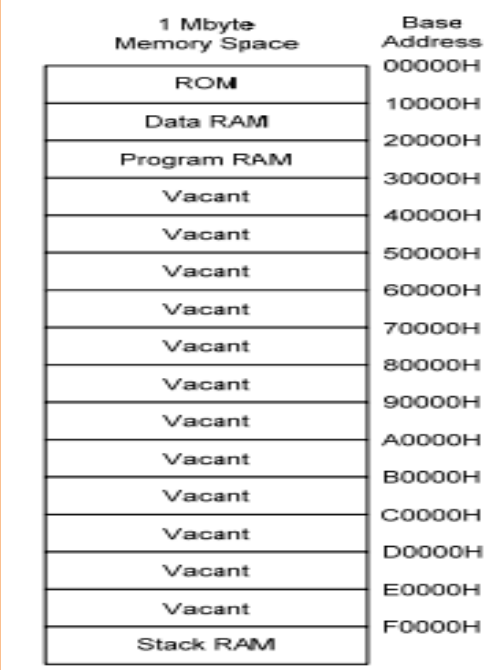
The Erase Operation of the FLASH Memory Cell



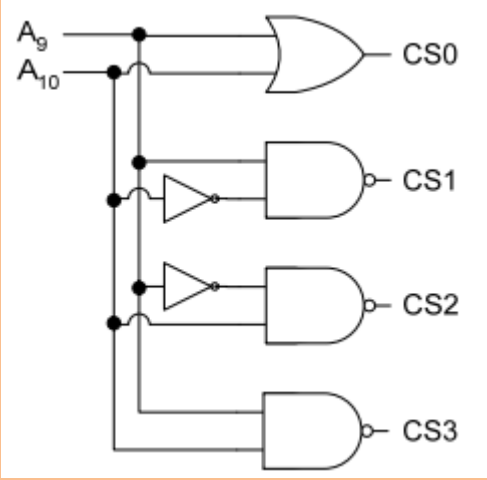
FIFO Implementation using four 8-bit Shift Registers



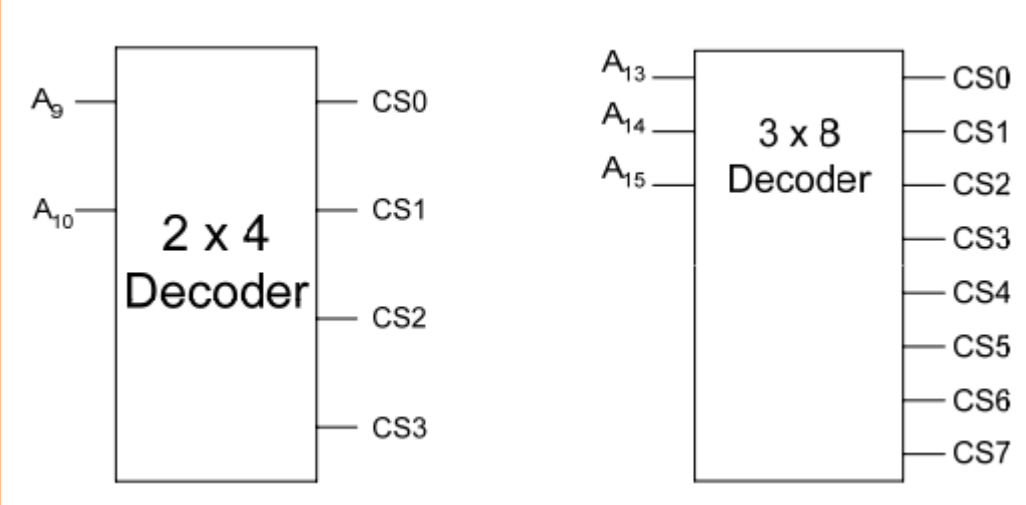
1 MByte Memory Map



Memory Decoders can be implemented using Logic Gate based Address Decoder



2 x 4 and 3 x 8 Decoder based Address Decoders



A 4-bit Binary-Weighted-Input D/A Converter