CMOS Logic Circuit Design

http://www.rcns.hiroshima-u.ac.jp

Link（リンク）: センター教官講義ノート の下 CMOS論理回路設計

Special-Purpose Digital Circuits

- Buffer Circuits
- Path-Selector Circuits
- Information-Storing Circuits
- Trigger Circuits
- Multi-Vibrator Circuits
- Voltage-Generator Circuits

Necessary Functions other than Logic Operations

1) Transmission of signals over long interconnection lines
   or to many receivers
   - Buffer (inverting, non-inverting, tri-state)
2) Selection of an interconnection for a Signal according to a condition
   - Selector (multiplexer, demultiplexer)
3) Storing an information for some time
   - Flip-flop, latch
4) Removing Noise from a Signal
   - Trigger circuits
5) Generation of Synchronous or Asynchronous Control Signal
   - Multi-vibrator circuits (a-stable, bi-stable, mono-stable)
6) Generation of other Voltages than VDD or VSS
   - Voltage generator circuits

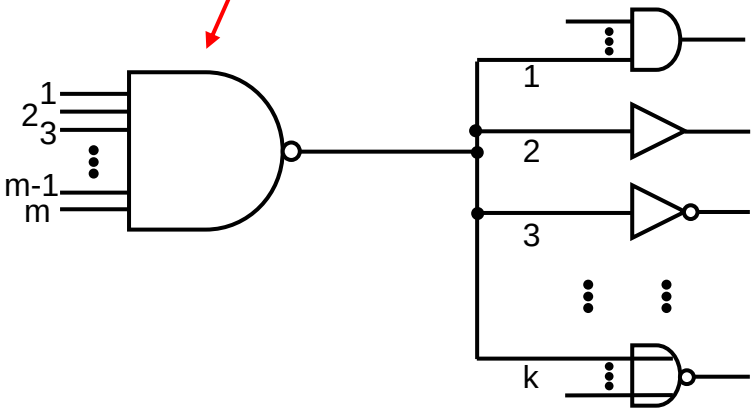
CMOS logic circuits do contain more than only logic gates.

Buffer Circuits

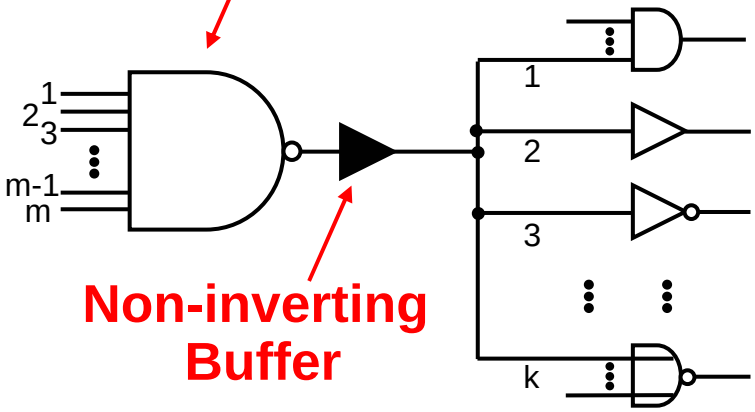
- Increasing the driving capability of a logic signal for large load capacities
- Conventional non-inverting buffers
- Inverting buffers
- Tri-state buffers

Reduction of Logic-Gate Fan-Out with a Buffer

NAND-gate with
fan-out = k, fan-in = m



NAND-gate with
fan-out = 1, fan-in = m



Delay without buffer

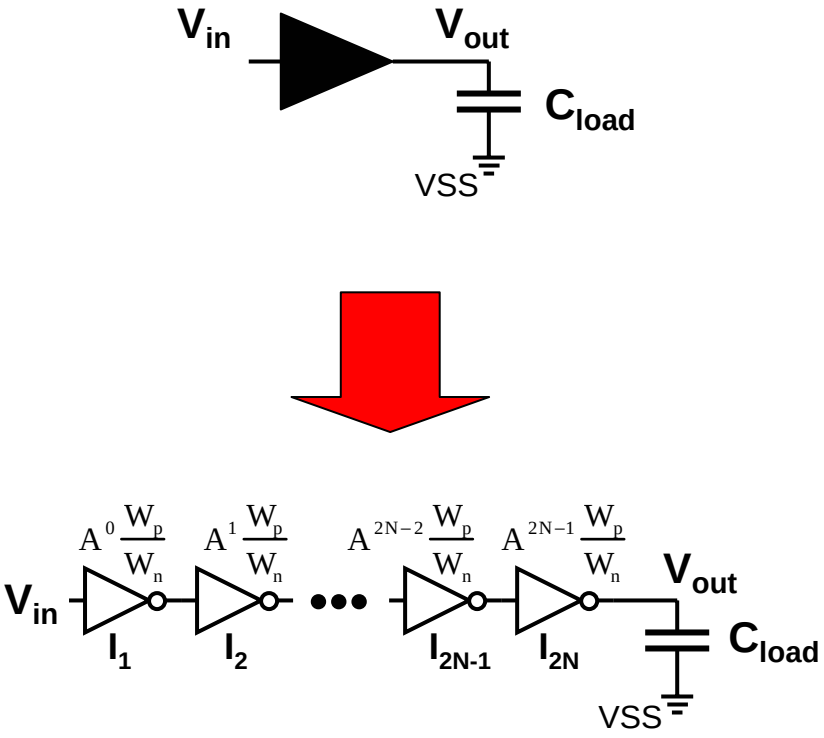
$$t_{df,NAND} = \mathrm{m}\cdot(\mathrm{m}\cdot t_{fin} + \mathrm{k}\cdot t_{fex})$$
$$t_{dr,NAND} = \mathrm{m}\cdot t_{rin} + \mathrm{k}\cdot t_{rex}$$

Delay with buffer

$$t_{df,NAND} = \mathrm{m}\cdot(\mathrm{m}\cdot t_{fin} + t_{fex}) + t_{buffer}$$
$$t_{dr,NAND} = \mathrm{m}\cdot t_{rin} + t_{rex} + t_{buffer}$$

The delay of a circuit with large fan-out (i. e. large output load) can be reduced with a buffer, if (k-1)·t_{rex} > t_{buffer} is valid.

Construction of Non-Inverting CMOS Buffers



Optimum choice of A and N

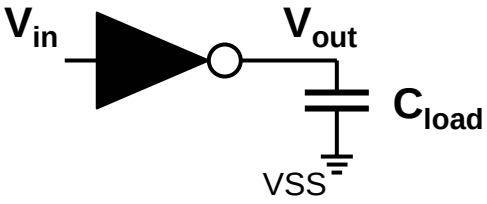
$$A_{ni-buffer} = \left[\frac{C_{load}}{C_{in1}}\right]^{\frac{1}{2N}}$$

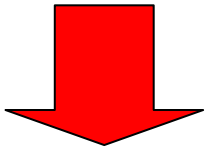
$$N_{ni-buffer} = int\left[\frac{1}{2}\ln\frac{C_{load}}{C_{in1}}\right]$$

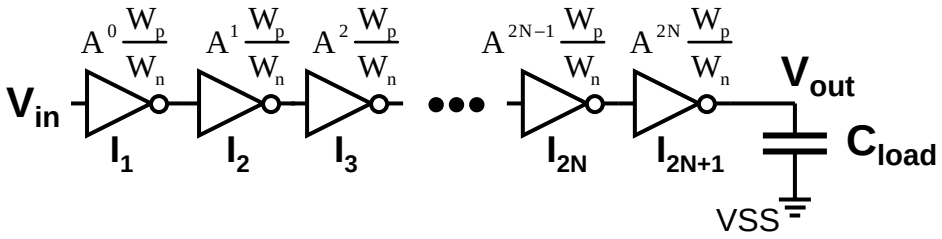
(C_{in1} is the input capacity of the 1st inverter)

Non-inverting buffers have even number of inverters. Each stage has a factor $A_{ni\text{-}buffer}$ (C_{load},C_{in}) larger driving capability.

Construction of Inverting CMOS Buffers







Optimum Choice of A and N

$$A_{i-buffer} = \left[\frac{C_{load}}{C_{in1}}\right]^{\frac{1}{2N+1}}$$

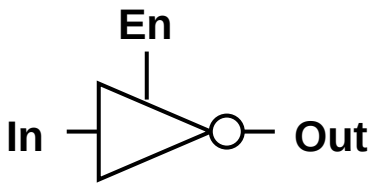
$$N_{i-buffer} = int\left[\frac{1}{2}\ln\left(\frac{C_{load}}{C_{in1}}\right) - \frac{1}{2}\right]$$

(C_{in1} is the input capacity of the 1st inverter)

Inverting buffers use an odd number of cascaded inverters. Each stage has again $A_{i\text{-}buffer}(C_{load},C_{in})$ larger driving capability.

Tri-State Inverter

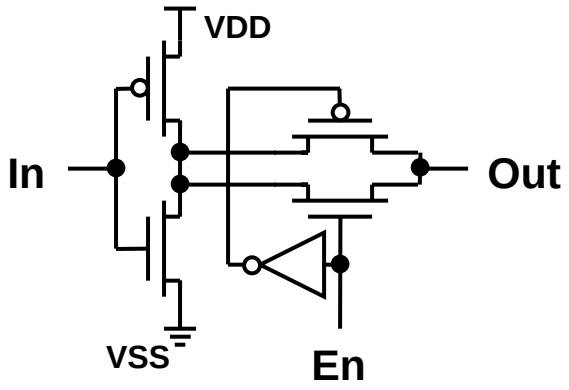
Symbol



Truth Table

En	In	Out
0	0	floating
0	1	floating
1	0	1
1	1	0

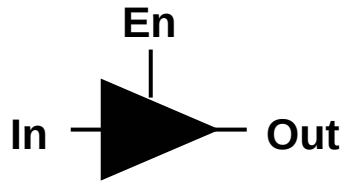
CMOS-Circuit Implementation



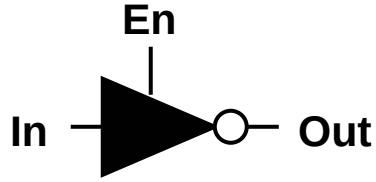
A tri-state inverter has an additional high-impedance or floating output state selected with an enable signal. It can be built with a conventional inverter and a transmission gate.

Tri-State Buffers

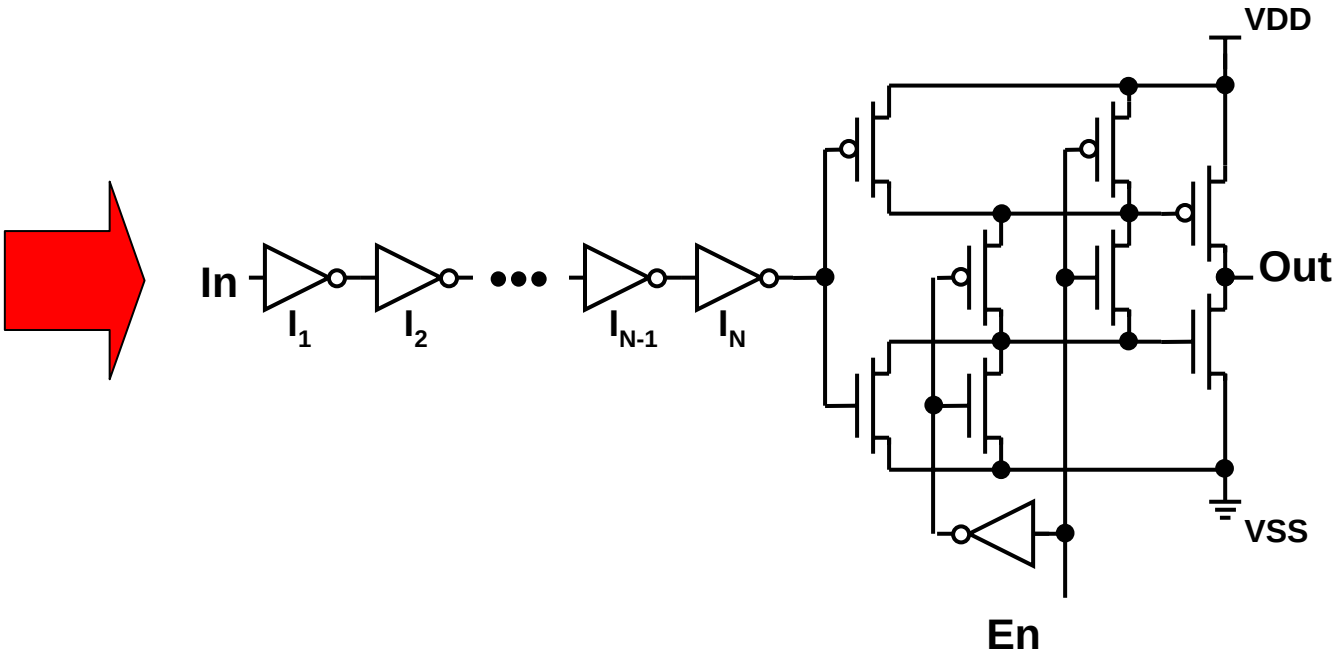
non-inverting tri-state buffer



inverting tri-state buffer



CMOS-Circuit Implementation

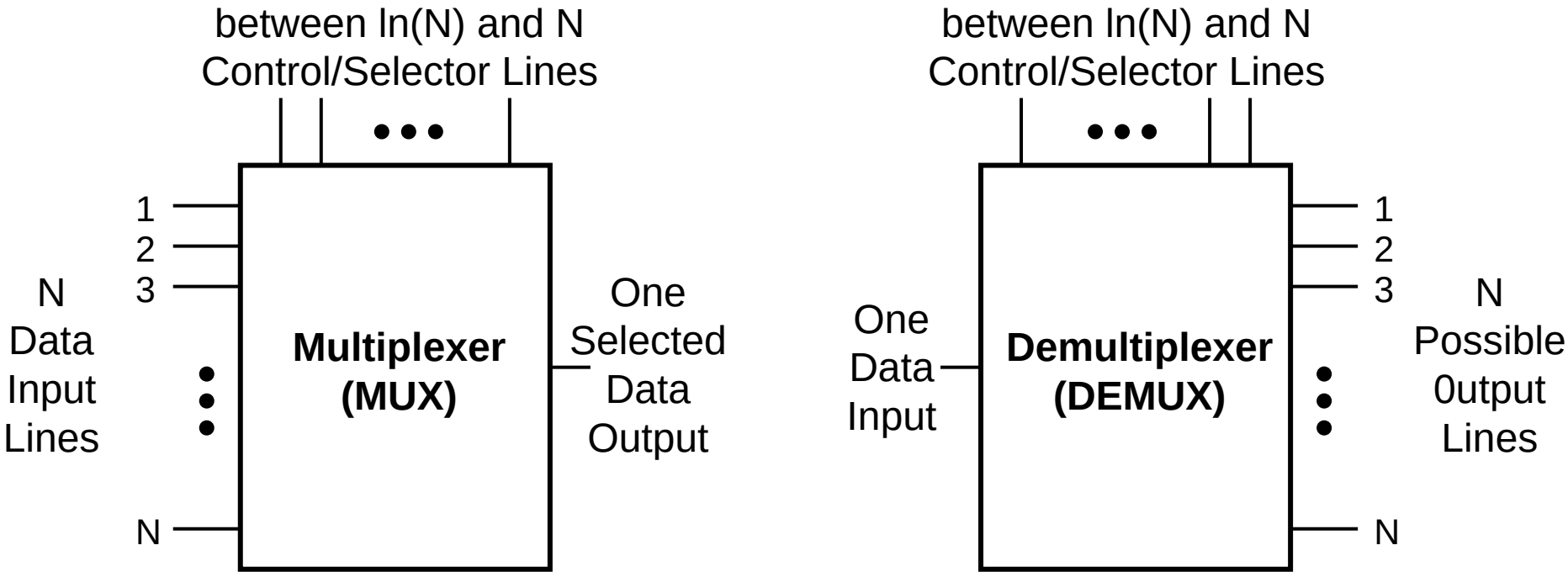


A tri-state buffer combines high driving capability for a large load capacity C_{load} and the possibility of a floating output.

Path-Selector Circuits

- Multiplexer- and Demultiplexer Principles
- Implementation with Transmission Gates
- Series Connection of Transmission Gates
- Implementation with Tri-State Inverters or Tri-State Buffers

Multiplexer and Demultiplexer Principles



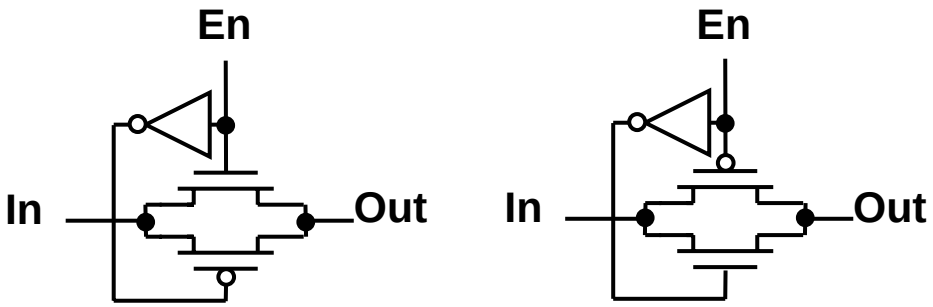
Conditional signal-path selection is performed with multiplexer- or demultiplexer circuits.

Multiplexer Realization with Transmission Gates

Transmission Gates

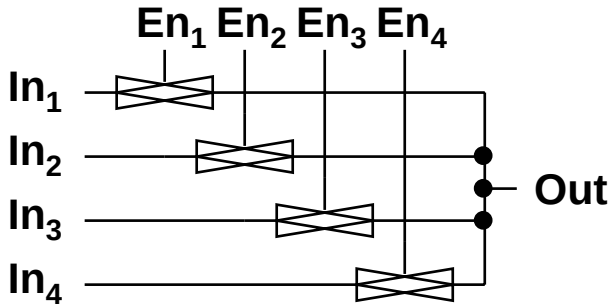
Symbols

Circuits

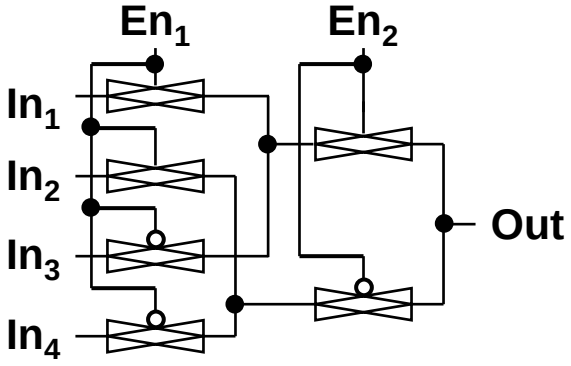


4-Input Multiplexer

Minimum Transmission Gates



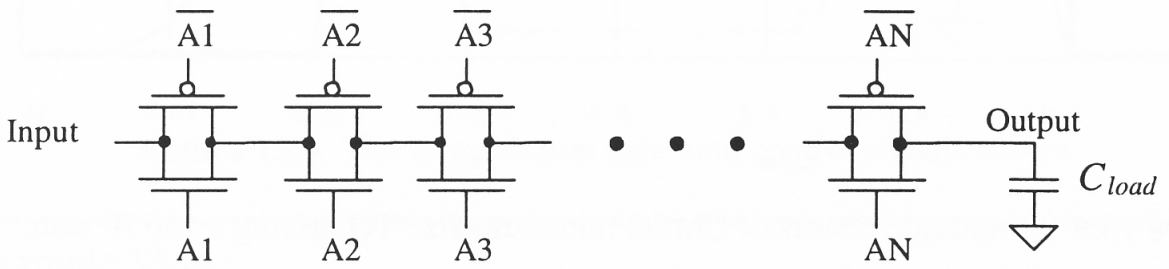
Minimum Select Signals



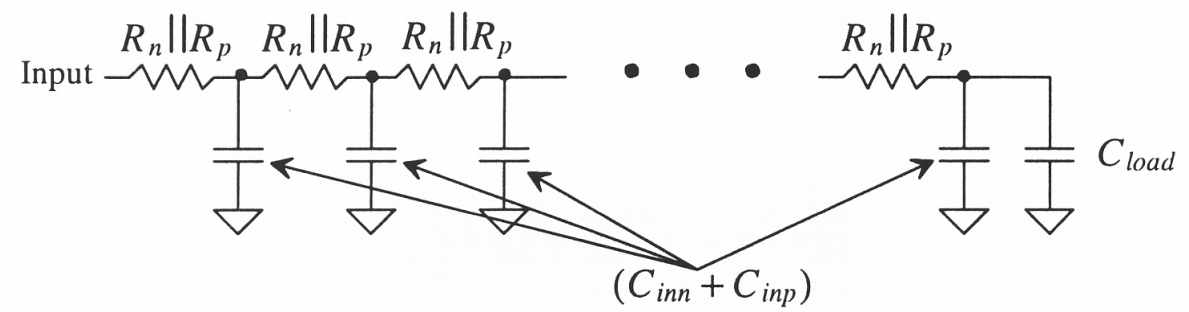
Path-selector realization is easiest by transmission gates.

Series Connection of Transmission Gates

Series of N transmission gates driving a load



Delay model for a series of N transmission gates



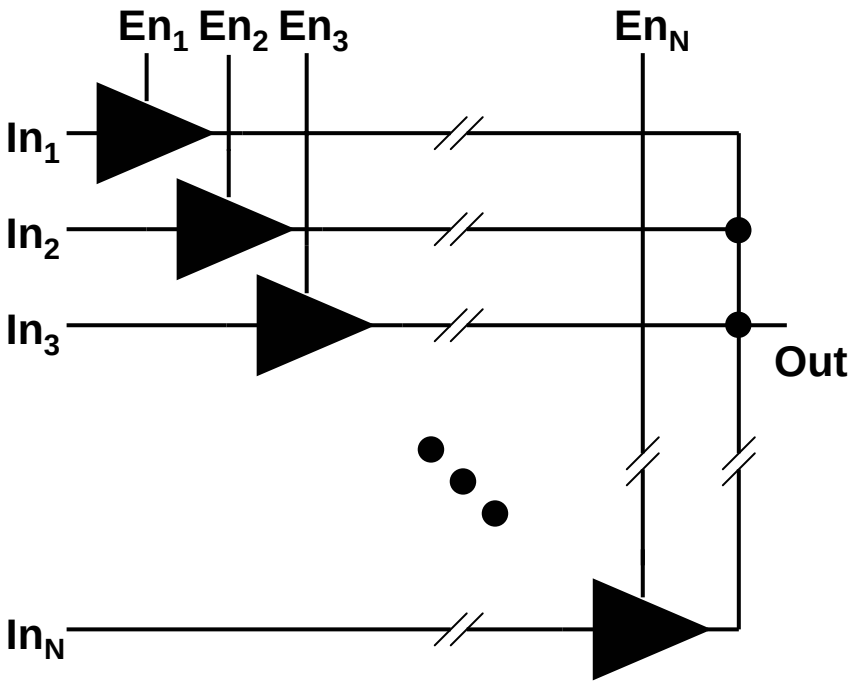
Delay equation as a function of N transmission gates

$$t_{PS,hl} \approx t_{PS,lh} \approx (R_n \parallel R_p)(C_{load}) \cdot \mathrm{N} + 0.35 \cdot (R_n \parallel R_p)(C_{inn} + C_{inp}) \cdot \mathrm{N}^2$$

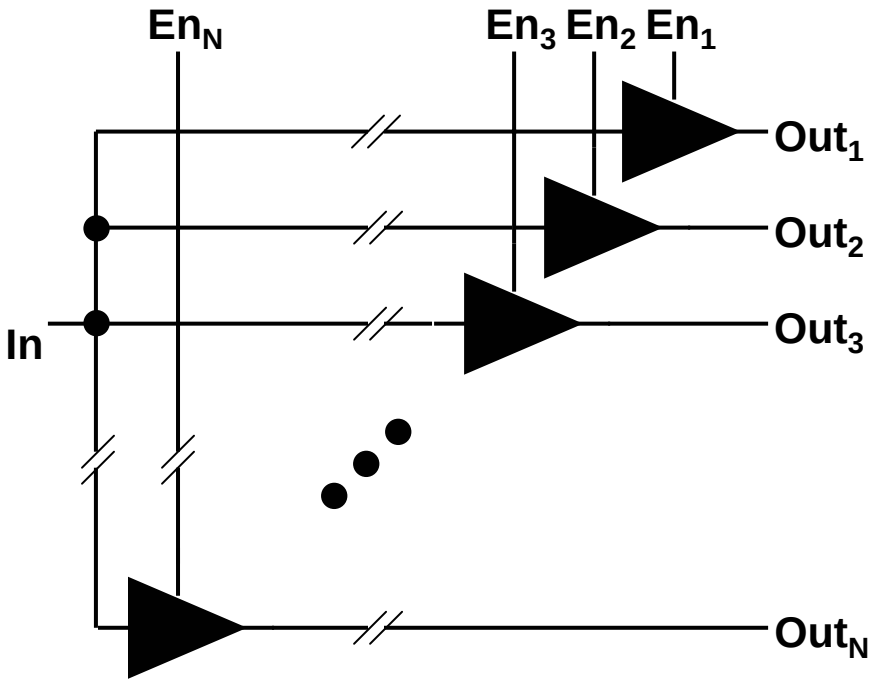
A series connection of N transmission gates represents an RC-chain. Therefore, its delay time increases with N^2.

MUX/DEMUX Realization with Tri-State Buffers

Multiplexer



Demultiplexer



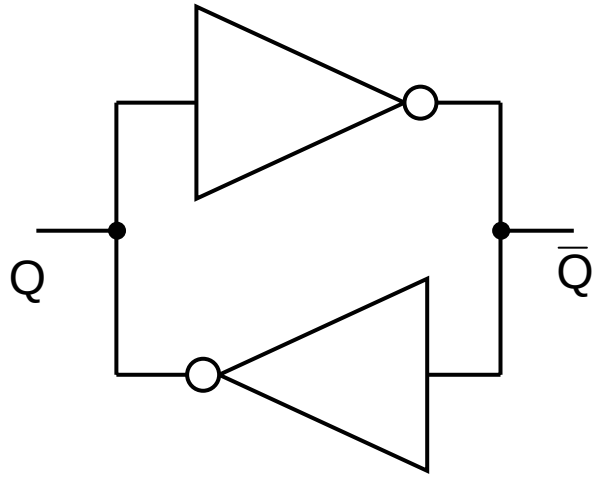
With tri-state buffers the delay problem of signal-path selectors is solved at the cost of larger integration-area.

Information-Storing Circuits

- Stabilizing-Feedback Principle
- Set-Reset Flip-Flop
- Clocked Flip-Flops
  - Level Sensitive Flip-Flops
  - Edge-Triggered Flip-Flops
  - Flip-Flop Timing

Stabilizing-Feedback Principle of Data Storage

Stabilizing inverter-feedback coupling



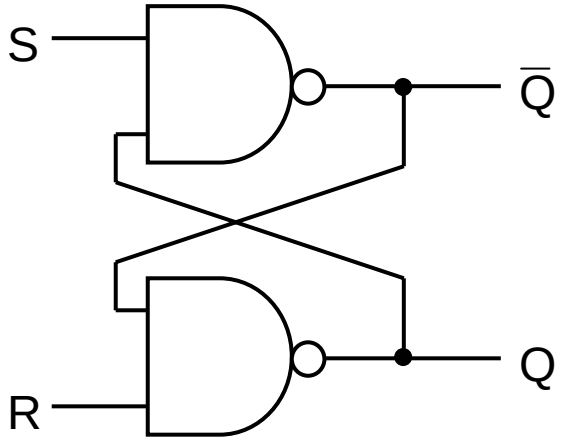
Resulting stable circuit states

Stable States	Q	$\overline{Q}$
“one”	1	0
“zero”	0	1

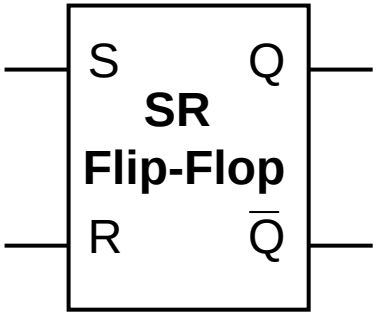
By feeding back the identical signal to a circuit node, stable circuit states result, which are usable for data storage.

Set-Reset (SR) Flip-Flop

Circuit diagram
(constructed with NAND gates)



Logic Symbol



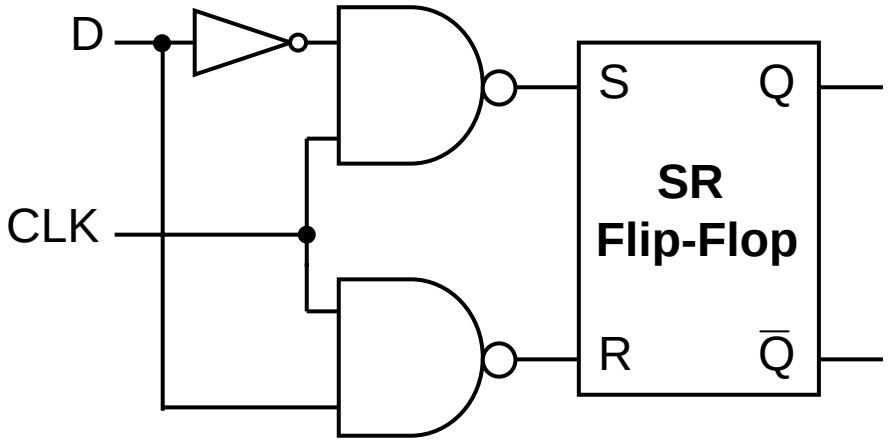
Truth table

S	R	Q	$\overline{Q}$
0	0	1	1
1	0	1	0
0	1	0	1
1	1	Q	$\overline{Q}$

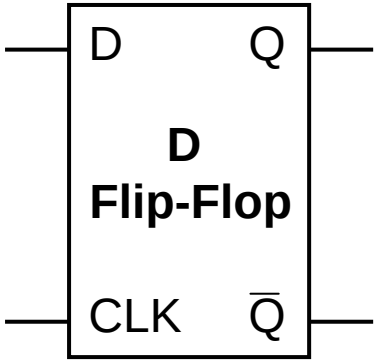
Set-reset flip-flops extend the stabilizing feedback principle by a method for external modification of the stored data.

Level-Sensitive Data (D) Flip-Flop

Circuit diagram
(constructed with NAND gates)



Logic Symbol

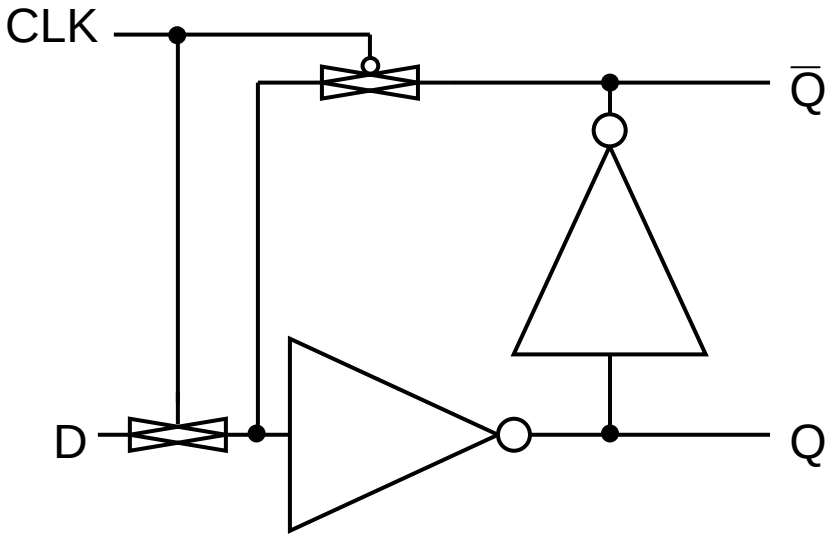


The level-sensitive data (D) flip-flop extends the SR flip-flop with additional circuitry for clock-controlled writing of data.

Latch: Transmission-Gate Version of D Flip-Flop

Circuit diagram of a latch

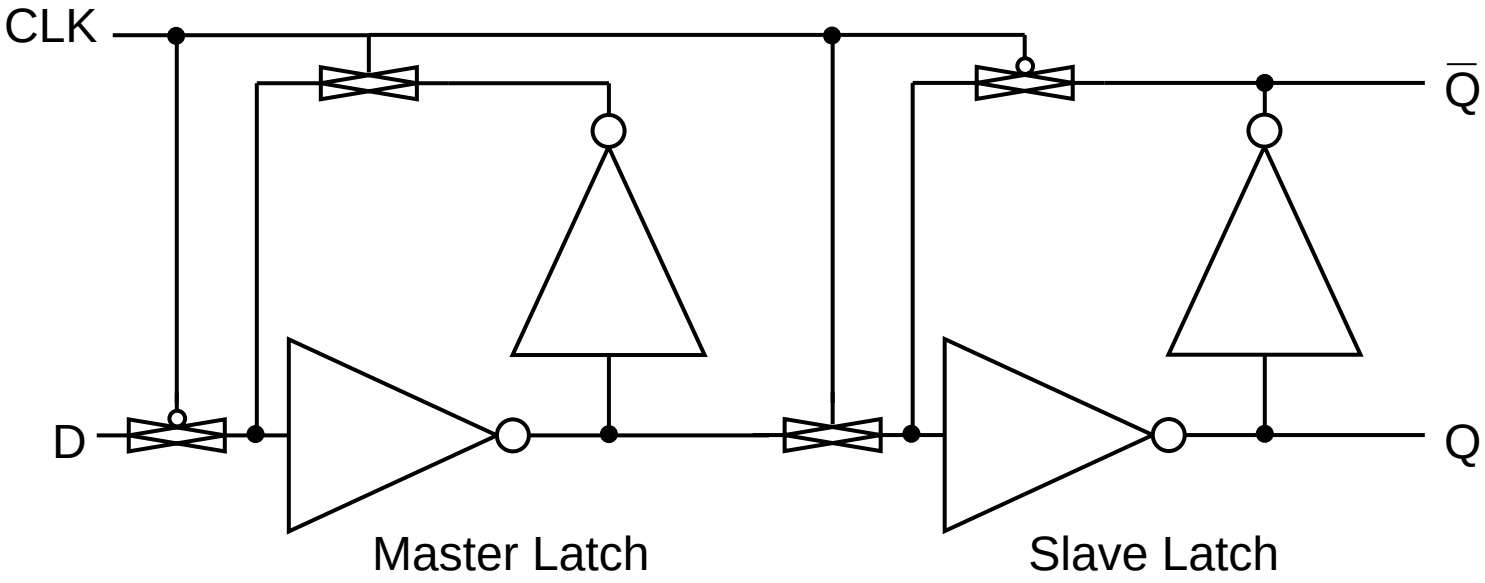
(data flip-flop constructed with inverters and transmission gates)



The simplest construction of level-sensitive data (D) flip-flops has 2 inverters and 2 transmission gates and is called “latch”.

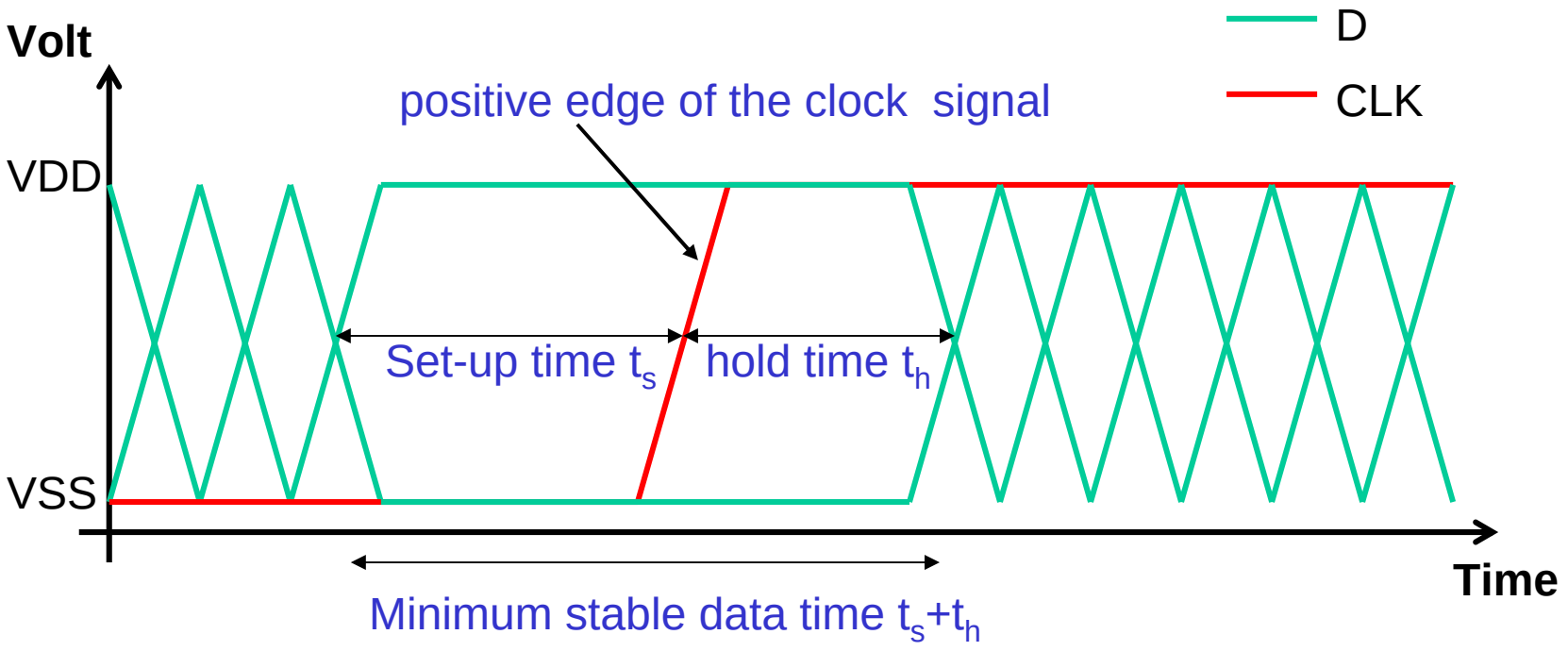
Edge-Triggered data (D) Flip-Flop

Circuit diagram of a D flip-flop into which data is written at the positive edge (low-high) change of the clock
(constructed with 2 latches)



The edge-triggered D flip-flop has 2 latches. Data transfer to the slave latch occurs only at transition edges of the clock.

Timing of Flip-Flops for Safe Data Writing

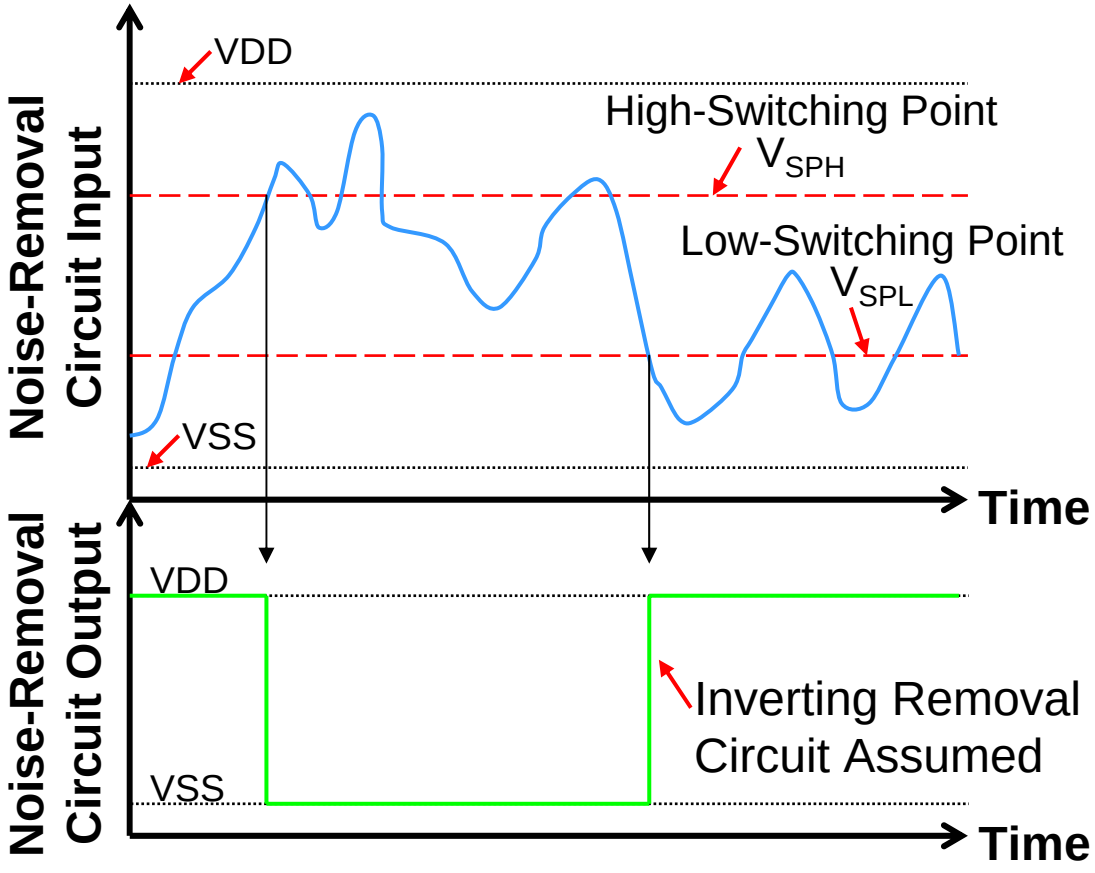


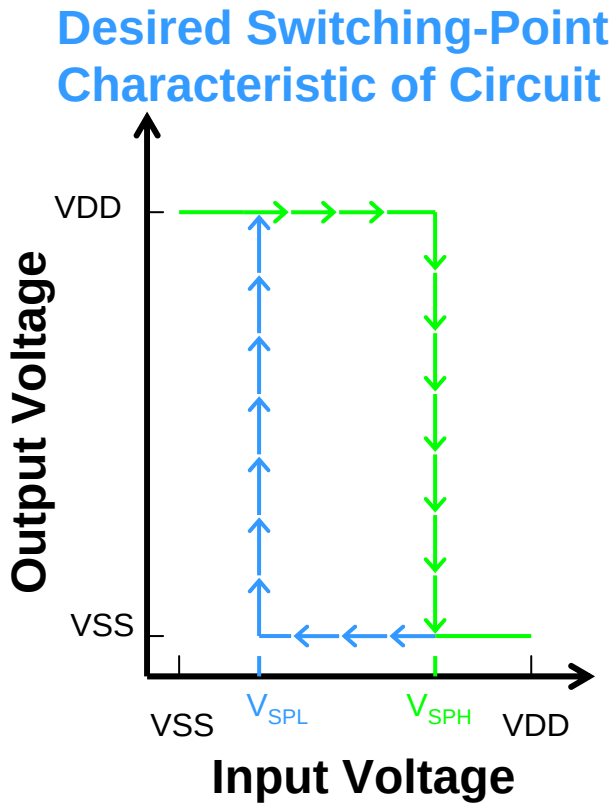
The safe operation of a flip-flop requires stable data signals for a minimum time around the clock edge, which determines data transfer into the storage part of the flip-flop.

Trigger Circuits

- Removal Possibilities of Signal Noise
- Schmitt-Trigger Circuit

Signal Noise and Removal Possibilities

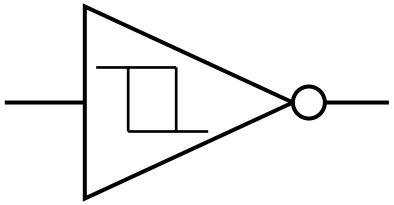




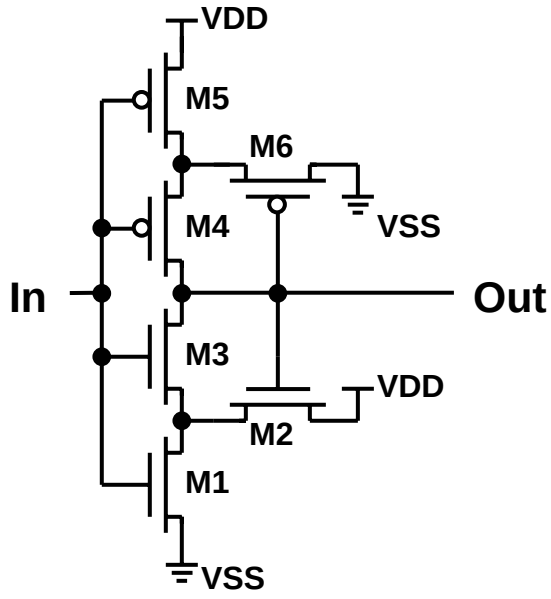
Noise can be removed from a signal with a circuit who has different switching points for low-high and high-low transition.

Schmitt-Trigger Circuit

Schmitt-Trigger Symbol



CMOS Circuit



Design of n-MOS Transistors M1 and M2 determines the High-Switching Point

$$\frac{\beta_1}{\beta_2} \approx \left[\frac{\mathrm{VDD} - \mathrm{V_{SPH}}}{\mathrm{VSS} + \mathrm{V_{SPH}} - \mathrm{V_{TH,n}}} \right]^2$$

Design of p-MOS Transistors M5 and M6 determines the Low-Switching Point

$$\frac{\beta_5}{\beta_6} \approx \left[\frac{\mathrm{VSS} + \mathrm{V_{SPL}}}{\mathrm{VDD} - \mathrm{V_{SPL}} - \mathrm{V_{TH,p}}} \right]^2$$

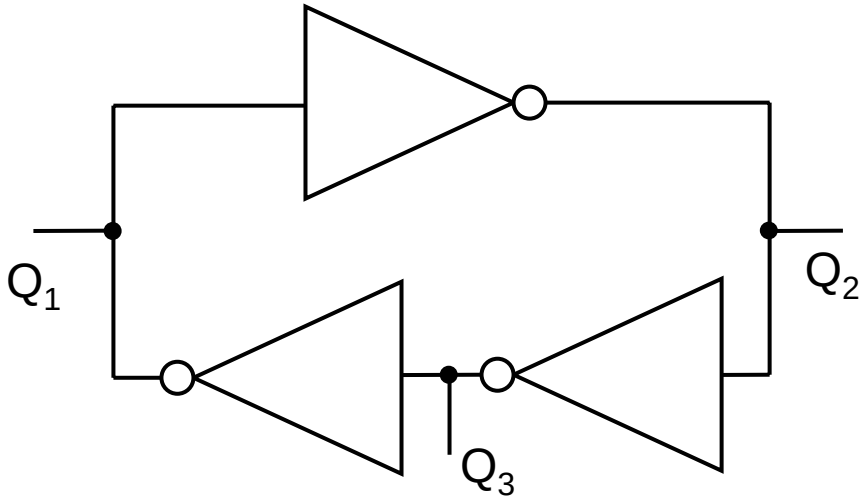
The CMOS inverter circuit can be easily modified to obtain an inverting Schmitt-trigger circuit to reduce input-signal noise.

Multi-Vibrator Circuits

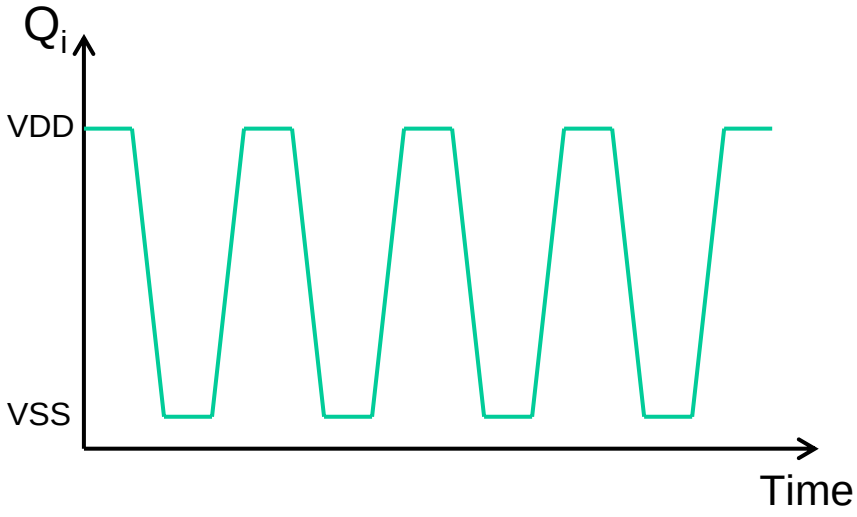
- Destabilizing-Feedback Principle
- A-Stable Multi-Vibrator or Oscillator
- Bi-Stable Multi-Vibrator or Flip-Flop
  (see Part on Information-Storing Circuits)
- Mono-Stable Multi-Vibrator

Destabilizing Feedback: Oscillator Circuits

Destabilizing inverter-feedback coupling



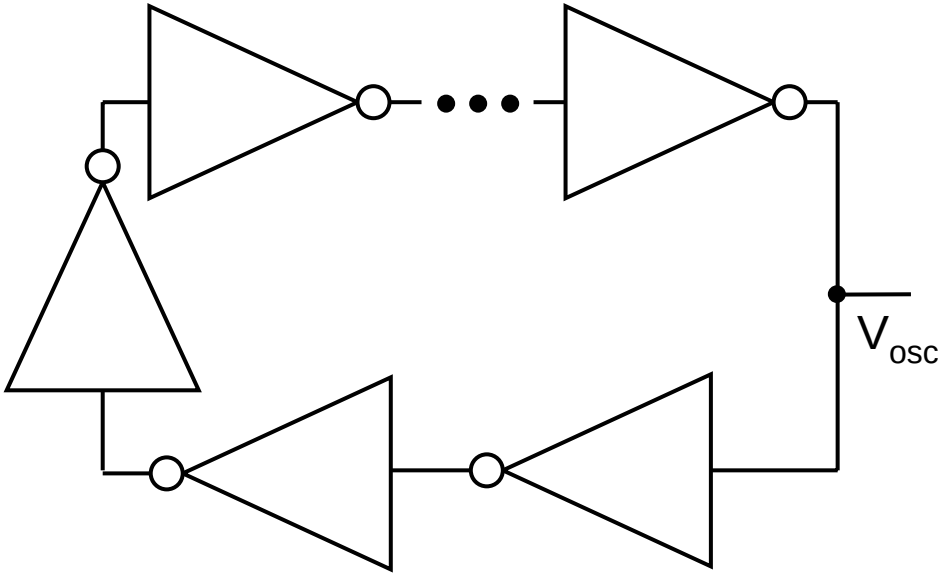
Resulting unstable (oscillating) signals at circuit nodes



By feeding back the inverted signal to a circuit node, an unstable state is occurs, which is used for oscillator circuits.

Ring-Oscillator Circuit with N Stages

Ring-Oscillator constructed with an odd number N of inverters



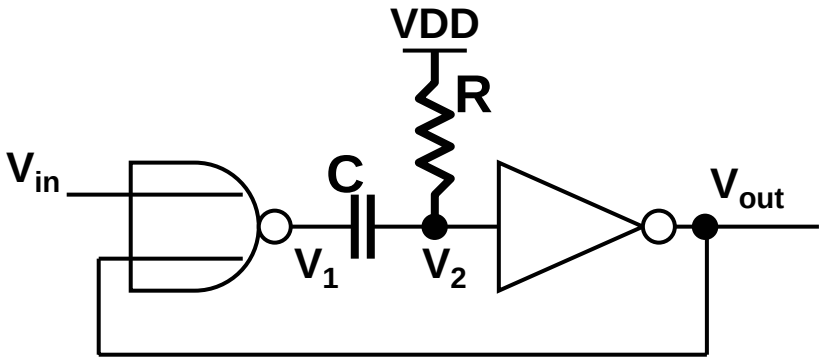
Obtained oscillator frequency

$$f_{osc} \approx \frac{1}{N \cdot (t_{IHL} + t_{ILH})}$$

CMOS oscillators can be constructed with an odd number of inverters. The oscillator frequency f_{osc} is determined by inverter low-high/high-low transitions and inverter number.

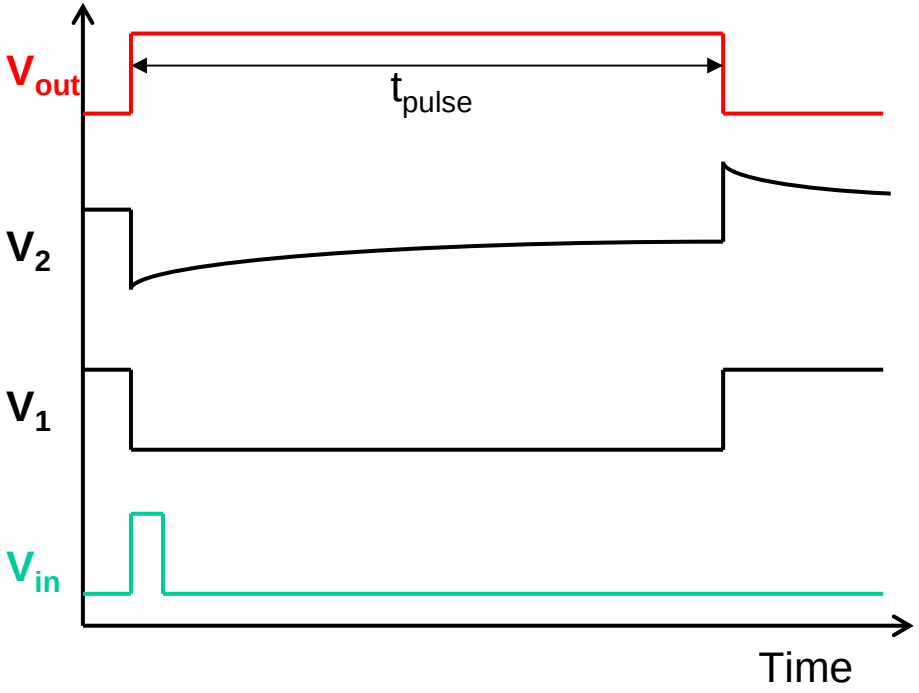
Mono-Stable Multi-Vibrator

Mono-stable multi-vibrator example constructed with NOR and inverter



$$t_{pulse} \approx RC \cdot \ln \frac{VDD}{VDD - V_{SP,I}}$$

Generation of long pulse with fixed length by short trigger pulse at input



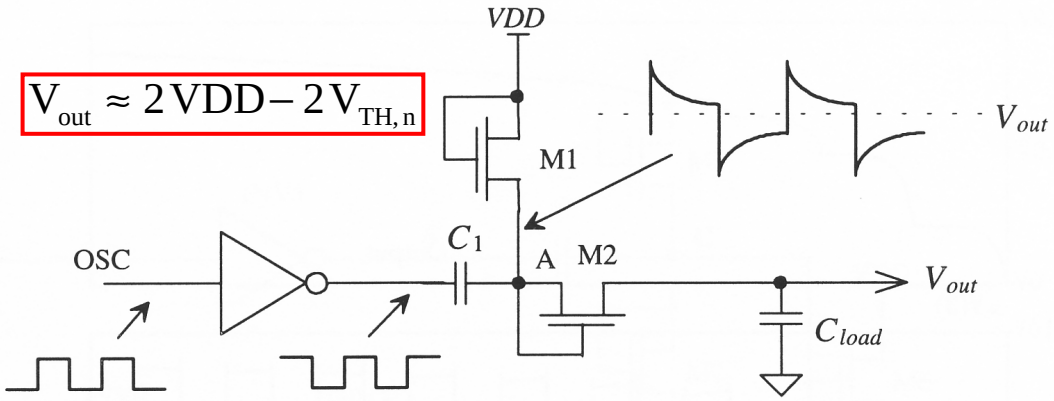
A mono-stable multi-vibrator is a circuit with delayed stable feedback. Thus pulses with fixed length can be generated.

Voltage-Generator Circuits

Simple Generator for Voltages >VDD and <VSS

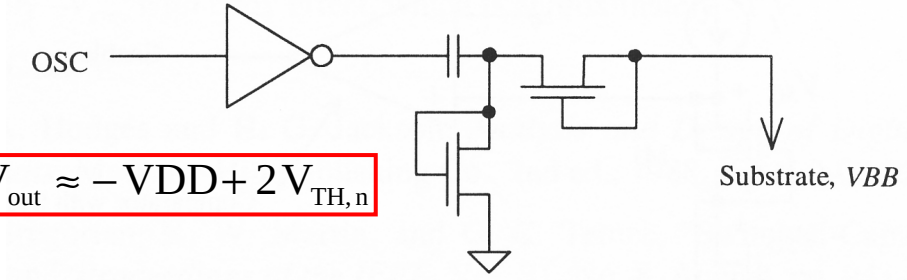
High-voltage generator

$V_{out} \approx 2VDD - 2V_{TH,n}$

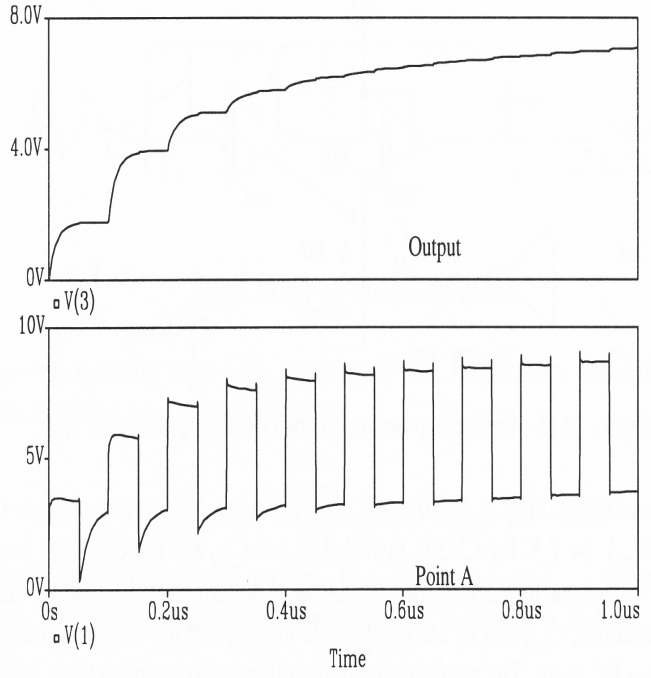


Low-voltage generator

$V_{out} \approx -VDD + 2V_{TH,n}$



Transient output of the high voltage generator



Voltage-generator circuits are applied, if the circuits in the CMOS chip need other supply voltages than VDD and VSS.