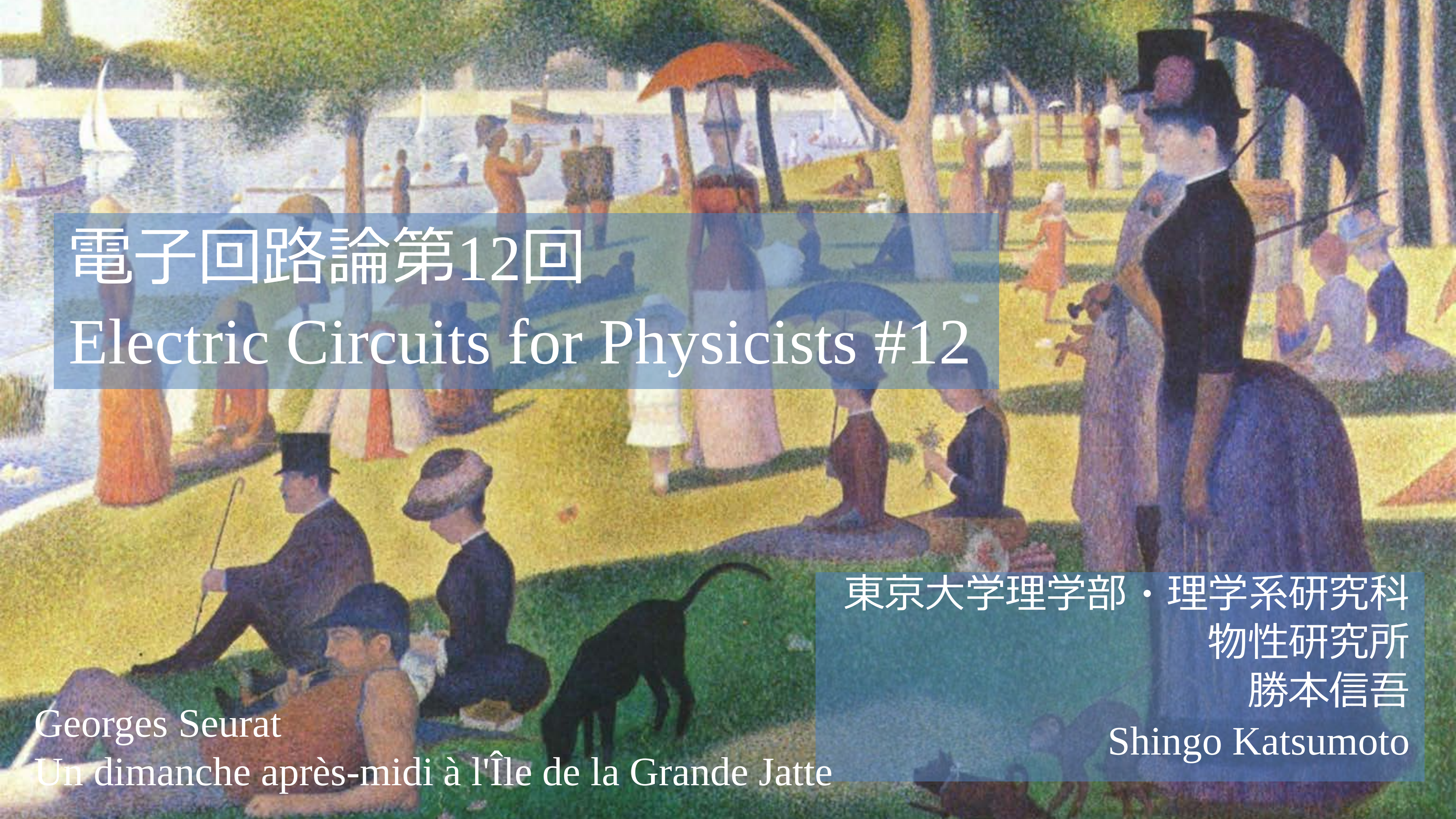
電子回路論第12回
Electric Circuits for Physicists #12

東京大学理学部・理学系研究科
物性研究所
勝本信吾
Shingo Katsumoto



Georges Seurat
Un dimanche après-midi à l'Île de la Grande Jatte

Outline

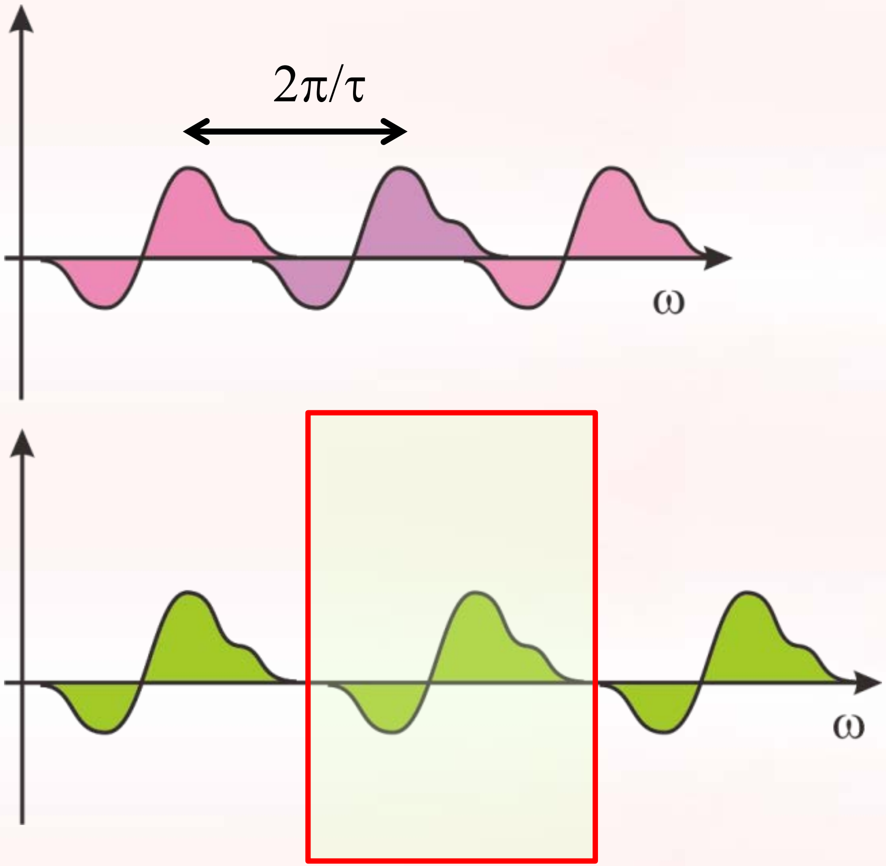
6.4 Discrete signal

- 6.4.1 Sampling theorem
- 6.4.2 Pulse amplitude modulation (PAM)
- 6.4.3 Discrete Fourier transform
- 6.4.4 z-transform
- 6.4.5 Transfer function of discrete time signal

Ch.7 Digital signals and circuits

- 7.2 Logic gates
- 7.3 Implementation of logic gates
- 7.4 Circuit implementation and simplification of logic operation

6.4.1 Sampling theorem



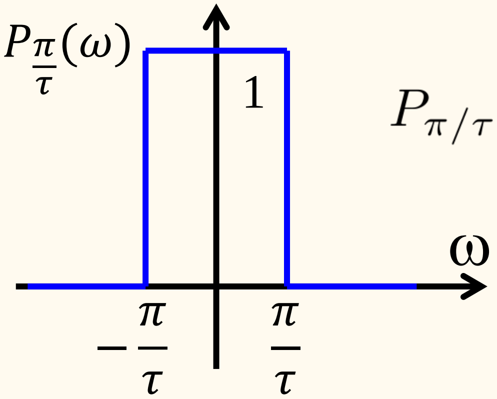
"Cutting out" the frequency spectrum

ω_h: Highest frequency in $\tilde{X}_\tau(\omega)$

$$\frac{2\pi}{\tau} > 2\omega_h, \quad \tau < \frac{\pi}{\omega_h}$$

$$\frac{1}{\tau} > \frac{\omega_h}{\pi} = 2f_h \; : \; \text{Nyquist frequency}$$

6.4.1 Sampling theorem: reconstructing signal



$$P_{\pi/\tau}(\omega) = \begin{cases} 1 & |\omega| \le \dfrac{\pi}{\tau}, \\ 0 & |\omega| > \dfrac{\pi}{\tau} \end{cases}$$

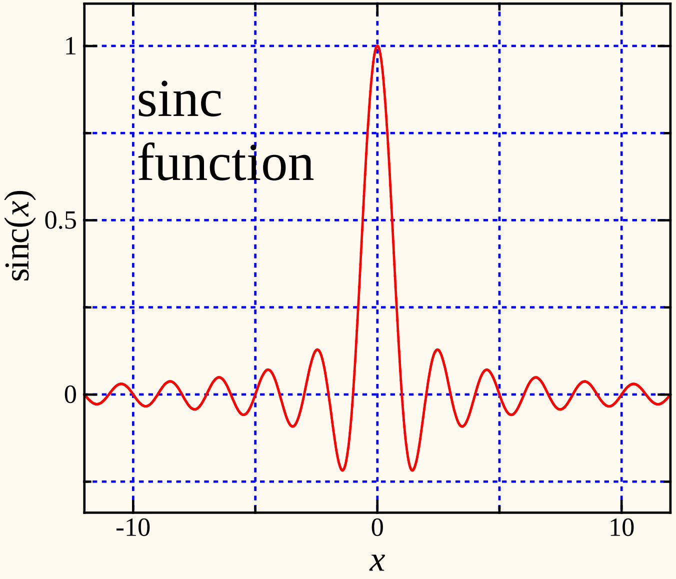
$$x(t) = \mathscr{F}^{-1}\{\tau P_{\pi/\tau}(\omega)\tilde{X}_\tau(\omega)\}$$

$$x(t) = \tau\frac{1}{\tau}\mathrm{sinc}\left(\frac{t}{\tau}\right) * \tilde{x}_\tau(t) \qquad \mathrm{sinc}(x) \equiv \frac{\sin(\pi x)}{\pi x}$$

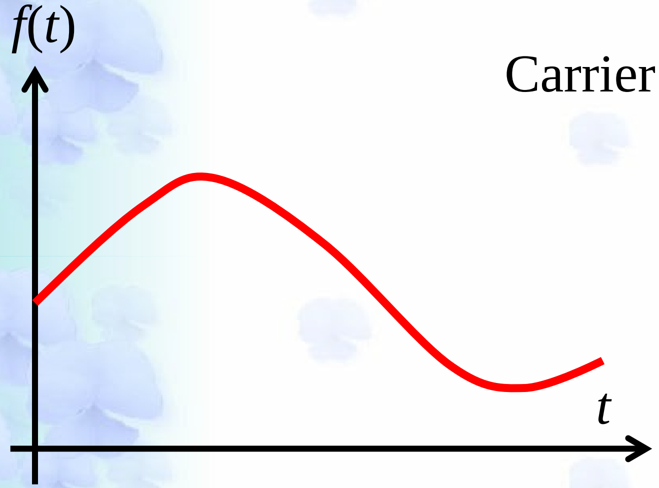
$$= \mathrm{sinc}\left(\frac{t}{\tau}\right) * \sum_{n=-\infty}^{\infty} x(t)\delta(t - n\tau)$$

$$= \int_{-\infty}^{\infty} \mathrm{sinc}\left(\frac{s}{\tau}\right) \sum_{n=-\infty}^{\infty} x(t-s)\delta(t - n\tau - s)ds$$

$$= \sum_{n=-\infty}^{\infty} \mathrm{sinc}\left(\frac{t - n\tau}{\tau}\right) x(n\tau)$$



6.4.2 Pulse amplitude modulation (PAM)



Carrier: $\delta_\tau(t)$ $\quad s(t) = f(t)\delta_\tau(t) = \sum_{n=-\infty}^{\infty} f(t)\delta(t - n\tau)$

Demodulation = Reconstruction of continuous signal from sampled data.

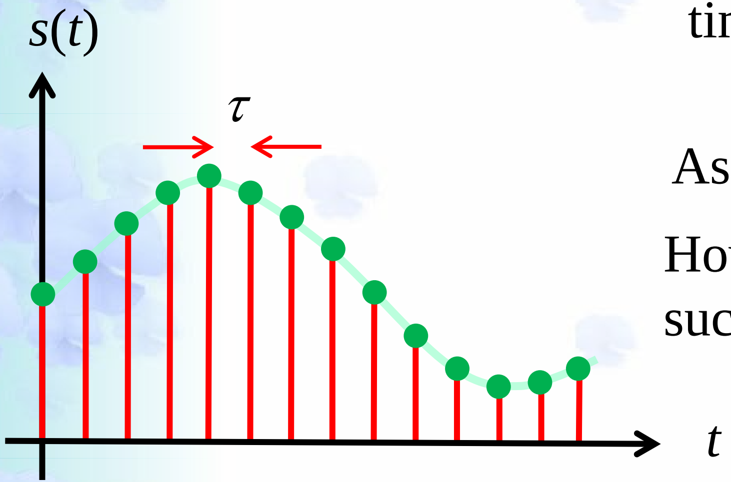
$$f(t) = \mathscr{F}^{-1}\{\tau P_{\pi/\tau}(\omega)\mathscr{F}\{s(t)\}\}$$

In the sampling theorem, though we only have discrete-time data, we can reconstruct complete original signal.

↑

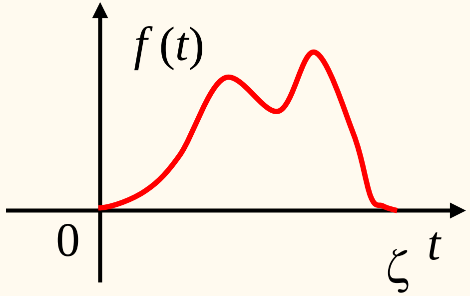
Assumption: we have data in infinite period $[-\infty, +\infty]$.

However in actual situations we can never have such data.



Need to consider handling data in a finite period.

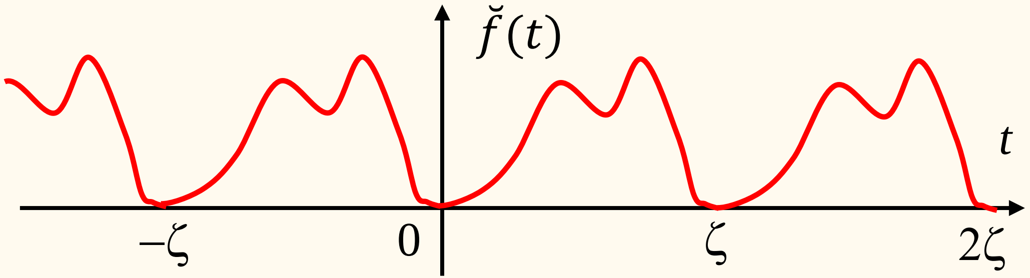
6.4.3 Discrete Fourier transform



Assumption: $F(\omega) = \mathscr{F}\{f(t)\}$, not zero in $\omega \in \left(-\frac{\pi}{\tau}, \frac{\pi}{\tau}\right)$

$N = \frac{\zeta}{\tau} \in \mathbb{N}$ can be assumed without loosing generality

Then we consider a periodic function $\breve{f}(t)$.



$$\breve{f}(t) = \sum_{n=-\infty}^{\infty} f(t - n\zeta),$$

$$\breve{F}(\omega) = \sum_{n=-\infty}^{\infty} F\left(\omega + n\frac{2\pi}{\zeta}\right)$$

$$\left(\breve{f}(t) = (f * \delta_\zeta)(t) = \sum_{n=-\infty}^{\infty} \int_{-\infty}^{\infty} f(\xi)\delta(t - n\zeta - \xi)d\xi\right)$$

6.4.3 Discrete Fourier transform

Fourier transform: $$\breve{f}(t) = \frac{1}{\zeta} \sum_{n=-\infty}^{\infty} F\left(n\frac{2\pi}{\zeta}\right) \exp\left(2n\pi i \frac{t}{\zeta}\right)$$

$$n = l + mN \qquad \sum_{n=-\infty}^{\infty} \to \sum_{l=0}^{N-1} \sum_{m=-\infty}^{\infty}$$

Discreteness: $t = j\tau \quad j \in \mathbb{Z}$

$$\breve{f}(j\tau) = \frac{1}{\zeta} \sum_{l=0}^{N-1} \sum_{m=-\infty}^{\infty} F\left[(l+mN)\frac{2\pi}{\zeta}\right] \exp\left[(l+mN)2\pi i \frac{j\tau}{\zeta}\right]$$

$$= \frac{1}{N\tau} \sum_{l=0}^{N-1} \sum_{m=-\infty}^{\infty} F\left(\frac{2\pi l}{\zeta} + m\frac{2\pi}{\tau}\right) \exp\left(2\pi i \frac{lj}{N}\right)$$

$$= \frac{1}{N\tau} \sum_{l=0}^{N-1} \breve{F}\left(l\frac{2\pi}{\zeta}\right) \exp\left(2\pi i \frac{lj}{N}\right) = \frac{1}{N\tau} \sum_{l=0}^{N-1} \breve{F}(l\eta) W_N^{-lj}$$

$$\eta \equiv \frac{2\pi}{\zeta}$$

Twiddle factor:

$$W_N \equiv \exp\left(-i\frac{2\pi}{N}\right)$$

6.4.3 Discrete Fourier transform

$$\eta \equiv \frac{2\pi}{\zeta} \qquad \breve{f}(j\tau) = \frac{1}{N\tau}\sum_{l=0}^{N-1} \breve{F}(l\eta) W_N^{-lj}$$

Properties of twiddle factor:

$$W_N \equiv \exp\left(-i\frac{2\pi}{N}\right)$$

$$\forall n, m \in \mathbb{Z} \quad W_N^{n+mN} = W_N^n,$$

$$\frac{1}{N}\sum_{n=0}^{N-1} W_N^{nm} = \begin{cases} 1 & \text{for} \quad m = 0, \\ 0 & \text{for} \quad m \neq 0. \end{cases}$$

$$\tau \sum_{j=0}^{N-1} \breve{f}(j\tau) W_N^{mj} = \sum_{j=0}^{N-1} \left[\frac{1}{N} \sum_{l=0}^{N-1} \breve{F}(l\eta) W_N^{(m-l)j} \right] = \breve{F}(m\eta)$$

$$f_n \equiv \breve{f}(n\tau), \quad F_k \equiv \frac{1}{\tau}\breve{F}(k\eta)$$

6.4.3 Discrete Fourier transform

Discrete Fourier transform (DFT):

$$F_k = \sum_{n=0}^{N-1} f_n W_N^{kn}, \quad f_n = \frac{1}{N}\sum_{k=0}^{N-1} F_k W_N^{-kn}$$

$$\boldsymbol{F} = \begin{pmatrix} F_0 \\ F_1 \\ \vdots \\ F_{N-1} \end{pmatrix}, \quad \boldsymbol{W} = \begin{pmatrix} W_N^{00} & \cdots & W_N^{0N-1} \\ \vdots & \ddots & \vdots \\ W_N^{N-10} & \cdots & W_N^{N-1N-1} \end{pmatrix}, \quad \boldsymbol{f} = \begin{pmatrix} f_0 \\ f_1 \\ \vdots \\ f_{N-1} \end{pmatrix}$$

$$\boldsymbol{F} = \boldsymbol{W}\boldsymbol{f}, \quad \boldsymbol{f} = \frac{1}{N}\boldsymbol{W}^*\boldsymbol{F}$$

$${}^t\boldsymbol{W}^*\boldsymbol{W} = N\boldsymbol{I}_N \quad i.e., \frac{1}{\sqrt{N}}\boldsymbol{W}$$: unitary

6.4.4 z-transform

Discrete Laplace transform: z-transform

$$\tilde{f}_\tau(t) = \sum_{n=0}^{\infty} f(n\tau)\delta(t - n\tau) \quad (t \geq 0)$$

$$\mathscr{L}\{\tilde{f}_\tau(t)\}(s) = \mathscr{L}\left\{\sum_{n=0}^{\infty} f(n\tau)\delta(t - n\tau)\right\} = \sum_{n=0}^{\infty} f(n\tau)\mathscr{L}\{\delta(t - n\tau)\}$$

$$= \sum_{n=0}^{\infty} f(n\tau)\exp(-sn\tau)$$

$$z = \exp(s\tau),\ f_n = f(n\tau),\ F(z) = \mathscr{L}\{\tilde{f}_\tau(t)\}$$

$$F(z) = \sum_{n=0}^{\infty} f_n z^{-n} = \mathscr{Z}[\tilde{f}_\tau(t)]$$

one-sided z-transform

6.4.4 z-transform (typical examples)

f_n	$F(z)$	conversion area
$\delta(n)$	1	z-plane
1	$\dfrac{1}{1-z^{-1}}$	$\lvert z\rvert > 1$
n	$\dfrac{z^{-1}}{(1-z^{-1})^2}$	$\lvert z\rvert > 1$
n^k	$\left(-z\dfrac{d}{dz}\right)^k \dfrac{1}{1-z^{-1}}$	$\lvert z\rvert > 1$
a^n	$\dfrac{1}{1-az^{-1}}$	$\lvert z\rvert > \lvert a\rvert$
$\sin(n\omega\tau)$	$\dfrac{\sin(\omega\tau)z^{-1}}{1-2\cos(\omega\tau)z^{-1}+z^{-2}}$	$\lvert z\rvert > 1$
$e^{-n\alpha\tau}\cos(n\omega\tau)$	$\dfrac{1-e^{-\alpha\tau}\cos(\omega\tau)z^{-1}}{1-2e^{-\alpha\tau}\cos(\omega\tau)z^{-1}+e^{-2\alpha\tau}z^{-2}}$	$\lvert z\rvert > e^{-\alpha\tau}$

6.4.4 z-transform (properties)

Property	Signal	z-transform
linearity	$af_n + bg_n$	$aF(z) + bG(z)$
z-domain scaling	$f_{\alpha n}$	$F(z^{1/\alpha})$
time shift	f_{n+k}	$z^k \left[F(z) - \sum_{l=0}^{k-1} f(l) z^l \right]$
time shift II	f_{n-k}	$z^{-k} F(z)$
scaling	$e^{\mp \alpha n} f_n$	$F(e^{\pm \alpha} z)$
scaling II	$a^n x_n$	$F(a^{-1} z)$
product with index	$n f_n$	$-z \frac{d}{dz} F(z)$
differentiation	$n^k f_n$	$\left(-z \frac{d}{dz} \right)^n F(z)$
integration	$\frac{f_n}{n+a}$	$z^a \int_z^{\infty} \xi^{-a+1} F(\xi) d\xi$
convolution	$f_n * g_n$	$F(z) \cdot G(z)$
product	$f_n \cdot g_n$	$\frac{1}{2\pi i} \oint_c F(\xi) G\left(\frac{z}{\xi} \right) \xi^{-1} d\xi$

6.4.5 Transfer function for discrete time signal

h_n: (impulse) response to $\delta(n\tau)$, response to discrete signal $f_n = f\ (n\tau)$

$$g_n = \mathscr{R}\{\tilde{f}_\tau(n\tau)\} = \mathscr{R}\left\{\sum_{k'=-\infty}^{\infty} f(k'\tau)\delta[(n-k')\tau]\right\} = \sum_{k'=-\infty}^{\infty} f_{k'}h_{n-k'} = \sum_{k=-\infty}^{\infty} h_k f_{n-k}$$

$$G(z) = \mathscr{Z}[g_n] = \mathscr{Z}\left[\sum_{k=0}^{\infty} h_k f_{n-k}\right] = \sum_{n=0}^{\infty}\left(\sum_{k=0}^{\infty} h_k f_{n-k}\right) z^{-n} = \sum_{k=0}^{\infty} h_k \sum_{n=0}^{\infty} f_{n-k} z^{-n}$$

$$= \sum_{k=0}^{\infty} h_k z^{-k} F(z)$$

$$H(z) = \mathscr{Z}[h_n] = \sum_{k=0}^{\infty} h_k z^{-k}$$:Transfer function

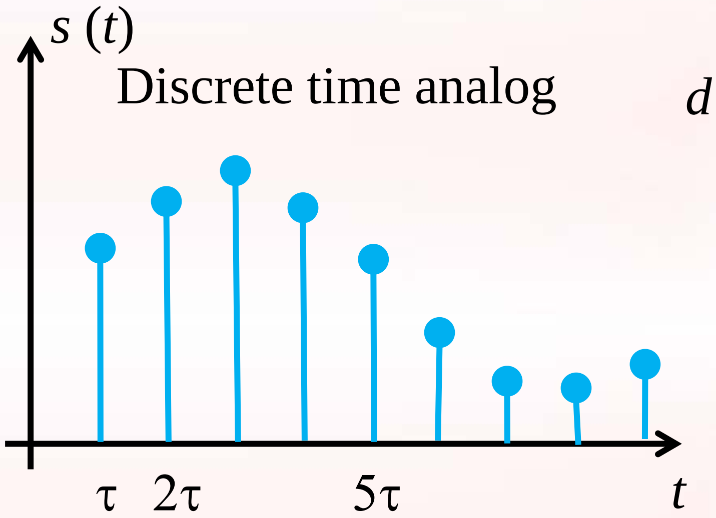
$$G(z) = H(z)F(z)$$

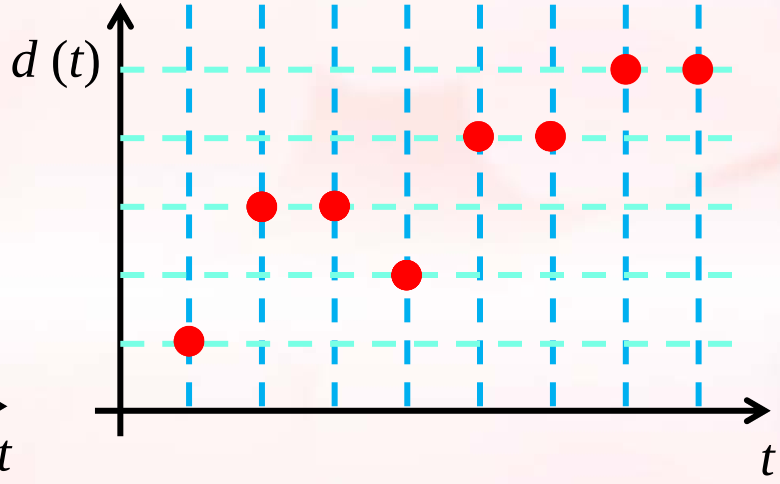
Byzantine mosaic

Chapter 7

Digital signal and circuits

Chartres Blue
(Stained glass)





Value discretized
↓
Digital signal

Signal unit : 0 xor 1 (bit)

Boolean algebra : F xor T

Voltage level : L xor H

Multiple bit → binary operation
→ parallel signal

7.2 Logic gates

Digital signal=logic value → Logic operation : logic gates → obeys Boolean algebra
(or → + (sum), and → · (product), not A → $\bar{A}$)

Logic variables: x, y $\quad x + x = x,\ (x + \bar{x}) \cdot y = y,\ \bar{\bar{x}} = x$

De Morgan's laws: $\overline{x + y} = \bar{x} \cdot \bar{y},\ \overline{x \cdot y} = \bar{x} + \bar{y}$

Truth table

		input				output				
t		t_1	t_2	$\cdots$	t_m		t_1	t_2	$\cdots$	t_m
Ch.	1	0	1	$\cdots$	f_{1m}	1	1	1	$\cdots$	q_{1m}
	2	1	0	$\cdots$	f_{2m}	2	0	1	$\cdots$	q_{2m}
	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
	n	0	1	$\cdots$	f_{nm}	l	0	1	$\cdots$	f_{lm}

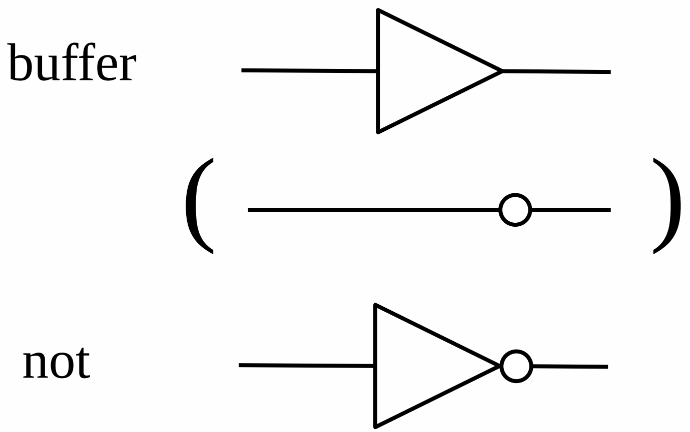
Combinational logic → Truth table (in practice, every gate has some time delay)
Sequential logic → Timing chart

7.2.1 Combinational logic: Single input gates

Truth table

Input	Buffer	not
0	0	1
1	1	0

Circuit symbol

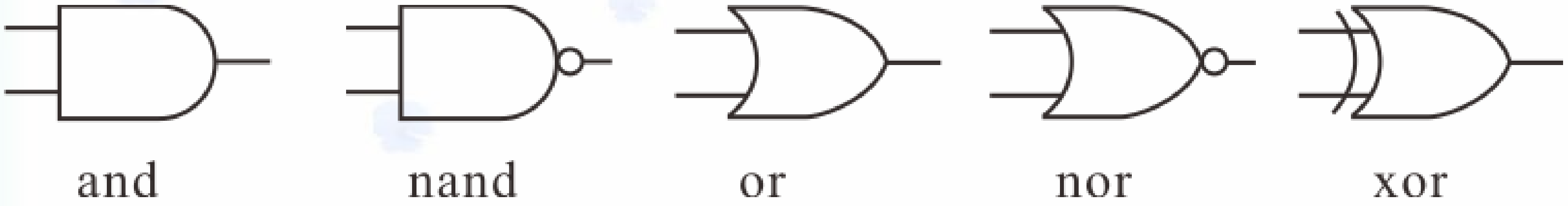


7.2.2 Combinational logic: Double input gates

Truth table

input1	input2	and	or	xor	Nand
0	0	0	0	0	1
1	0	0	1	1	1
0	1	0	1	1	1
1	1	1	1	0	0

Circuit symbols



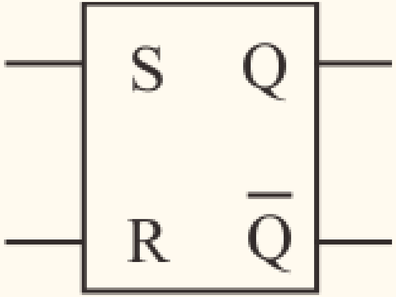
7.2.3 Sequential logic: Flip-Flop (FF)

RS (reset-set) Flip-Flop (FF)

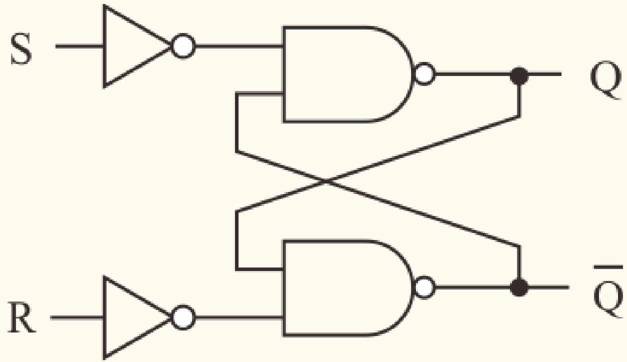
Truth table

S	R	Q	$\bar{Q}$	Response
0	0	Q	$\bar{Q}$	no change
0	1	0	1	reset
1	0	1	0	set
1	1	undetermined		

Circuit symbol



An equivalent circuit with discrete gates



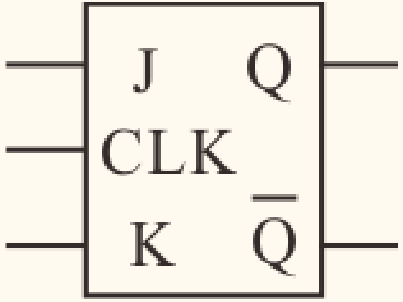
7.2.3 Sequential logic: Flip-Flop (with clock input)

JK Flip-Flop

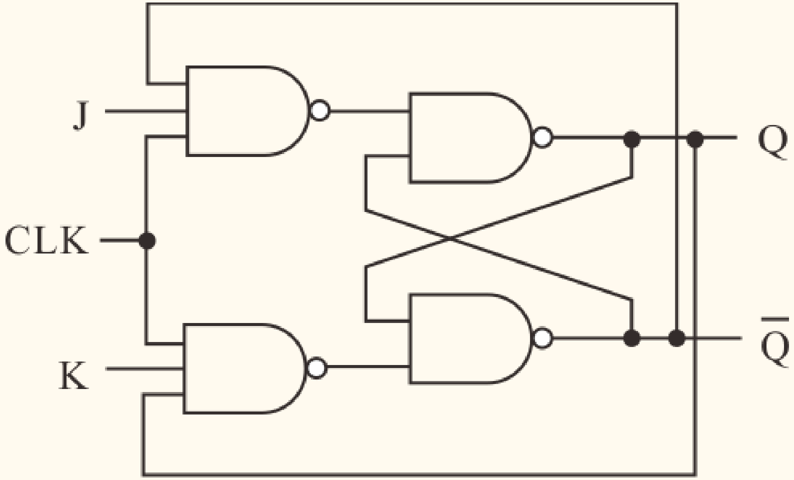
Truth table

J	K	Q	Q for the next CLK
0	0	0	0
0	0	1	1
0	1	–	0
1	0	–	1
1	1	0	1
1	1	1	0

Circuit symbol



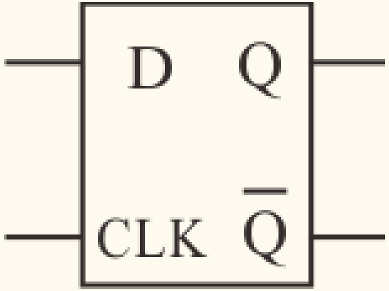
An equivalent circuit
with discrete gates



7.2.3 Sequential logic: D-FF, T-FF

D-FF

Symbol

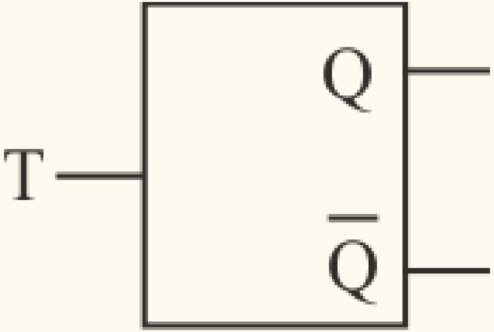


Truth table

D	CLK	Q
0	$\uparrow$	0
1	$\uparrow$	1
–	$\downarrow$	Q (hold)

T-FF

Symbol

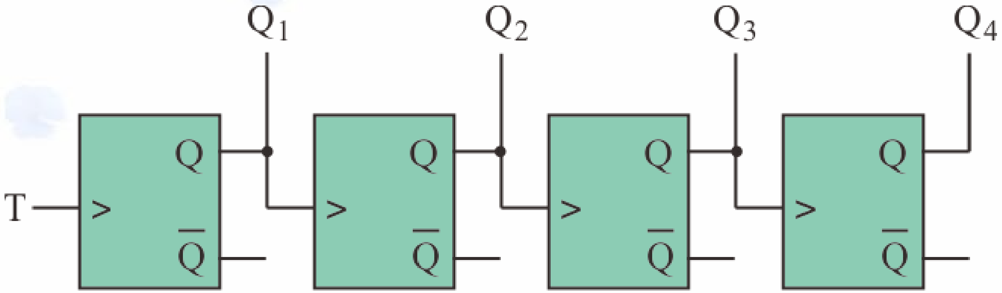


Truth table

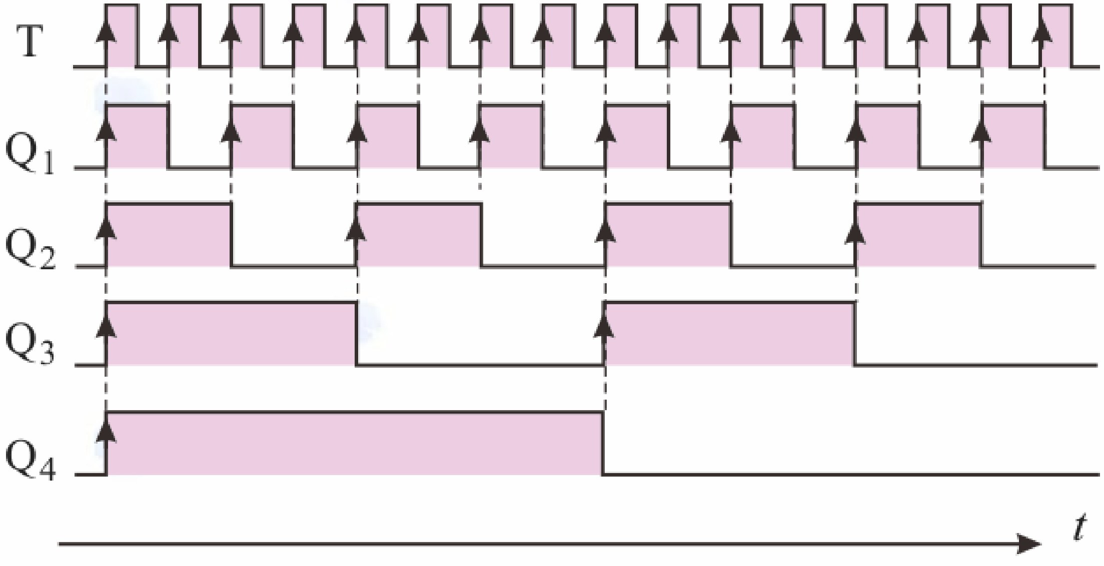
T	Q	Q
$\downarrow$	0	0
$\downarrow$	1	1
$\uparrow$	0	1
$\uparrow$	1	0

7.2.4 Sequential logic: Counters

Unsynchronized counter (ripple counter)



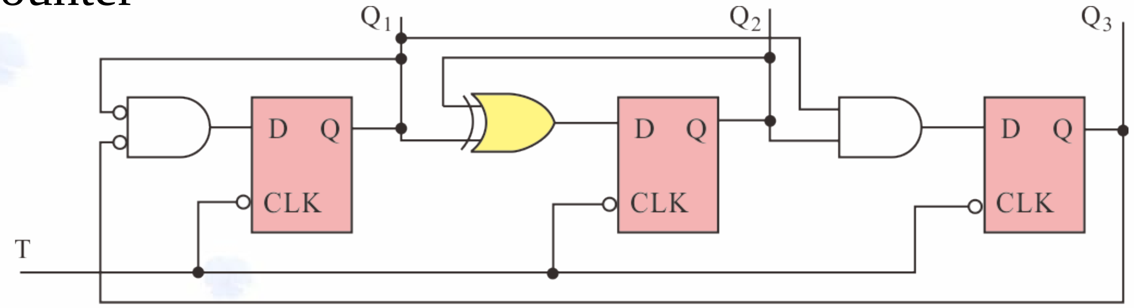
Timing chart



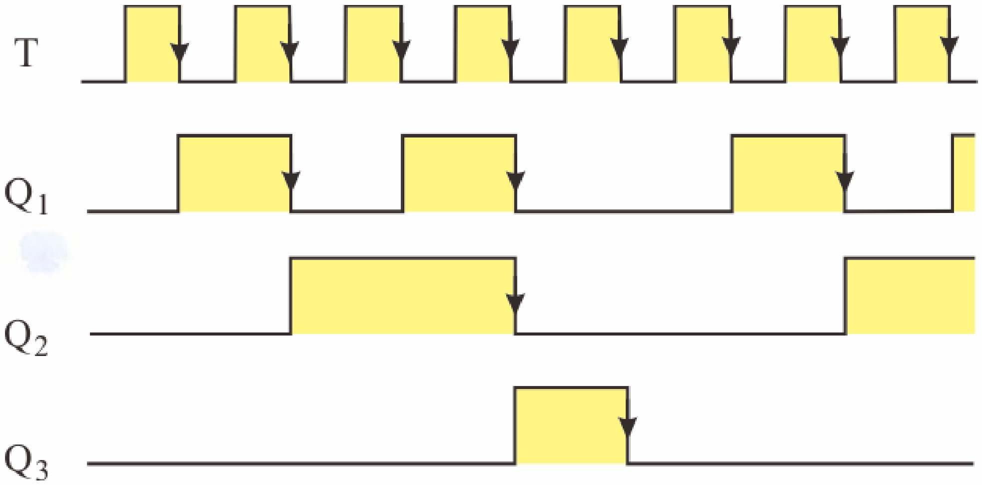
7.2.4 Sequential logic: Counters

Synchronized counter

Equivalent circuit with discrete gates



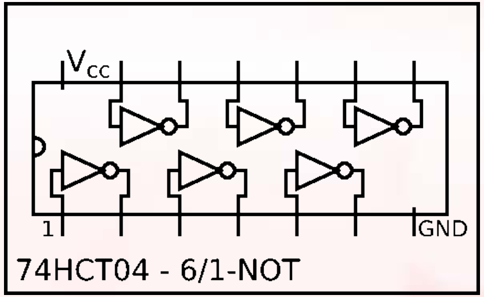
Timing chart

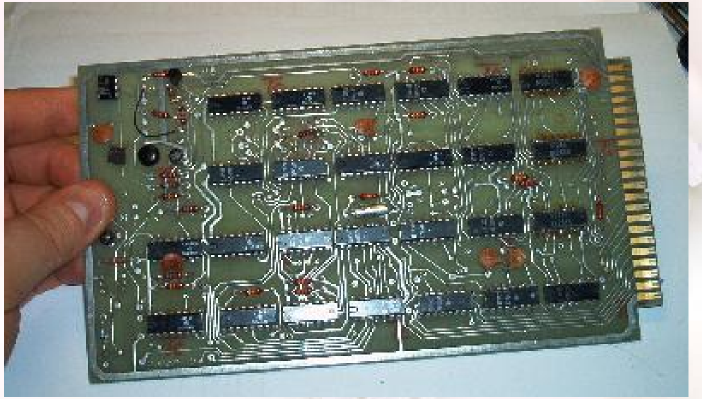


Standard gate logic packaging and wiring

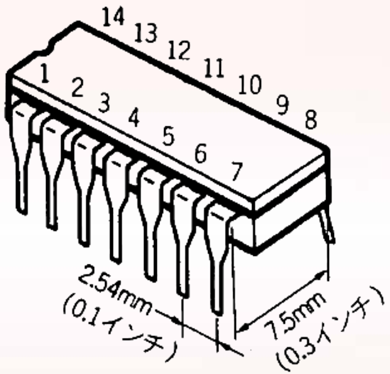
Full pitch

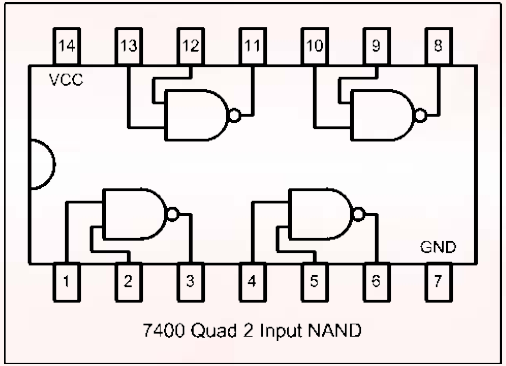
Half pitch surface mount





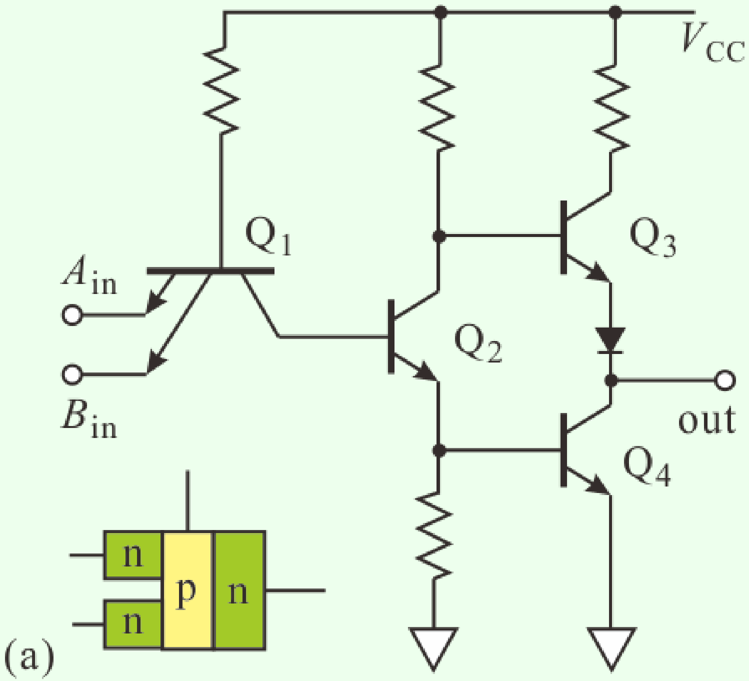
Printed board with soldering

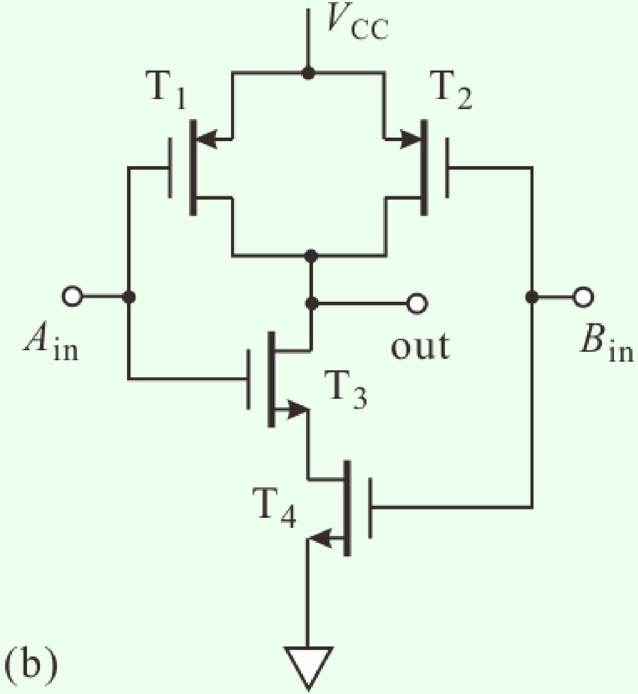




Surface mounting

NAND gates

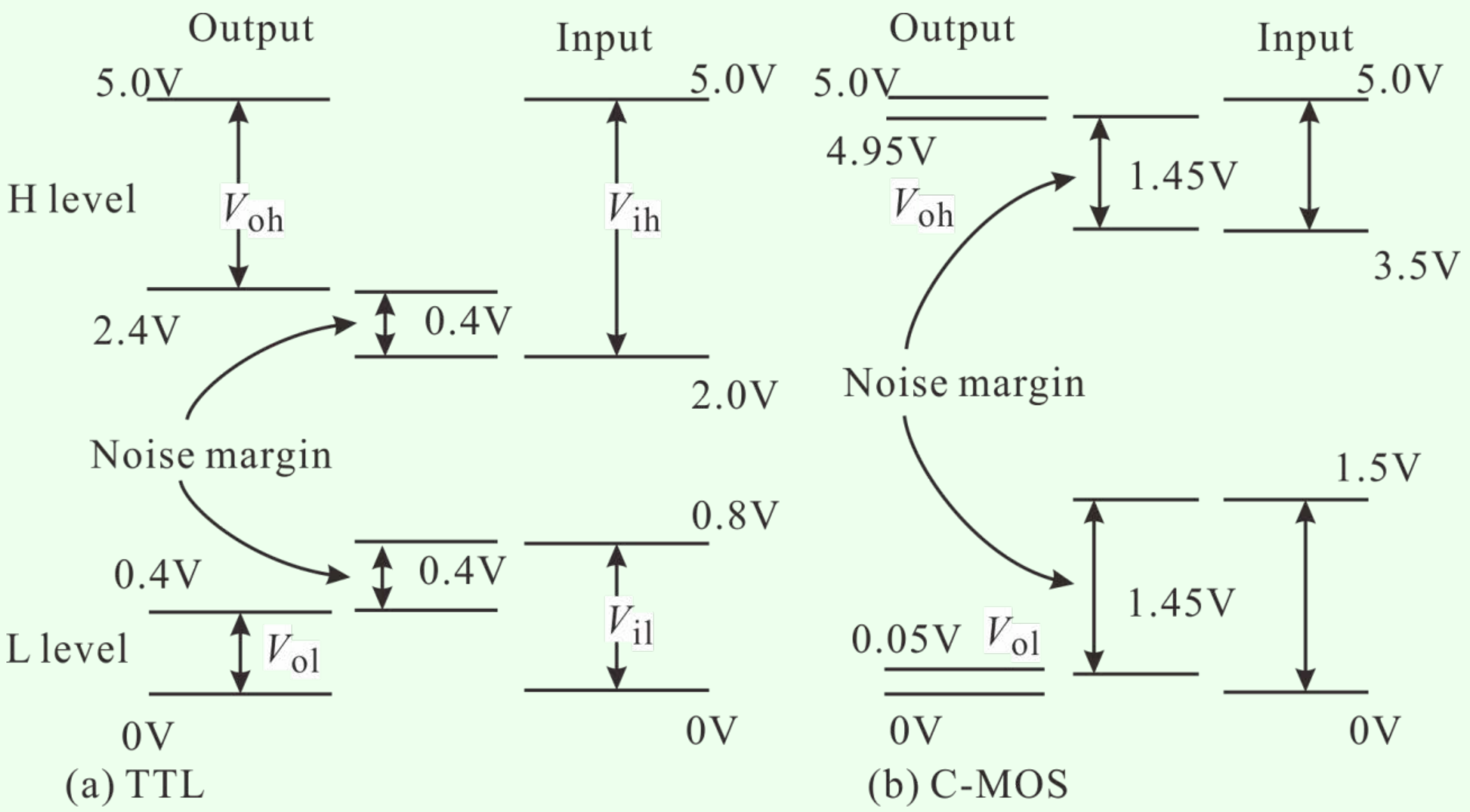




TTL (transistor-transistor logic)

CMOS (complimentary MOS)

Voltage levels diagram



7.4 Circuit implementation and simplification of logic operation

Design procedure:

Truth table → Simplification → Circuit diagram

Simplification
- Visual method: Karnaugh mapping
- Quine-McClusky algorithm

Example of simplification: $Y = A \cdot B + A \cdot \bar{B} + \bar{A} \cdot B$

$$A + A = A$$
$$A + \bar{A} = 1$$

$$Y = A \cdot B + A \cdot B + A \cdot \overline{B} + \overline{A} \cdot B$$
$$= A \cdot (B + \overline{B}) + B \cdot (A + \overline{A}) = A + B$$

	$\overline{A}$	A
$\overline{B}$	a	b
B	c	d

	$\overline{A}$	A
$\overline{B}$		1
B	1	1

Karnaugh mapping: Represent the logic on a two-dimensional table

(1) The raw and column indices are logical expressions. Neighboring expressions should have a single element with the value reversed.

(2) Put to neighboring 1.

(3) One means a reproducible pair.

Quein-McCluskey algorithm

Original form: $Y = f(A_1, A_2, \cdots, A_n)$

Logic variables

Transform to a standard starting form.

For that we define $g_i(0) = \overline{A_i}$, $g_i(1) = A_i$

Product of all the logic variables: minimum canonical term $\prod_{i=1}^{n} g_i(a_i), \quad a_i = 0, 1$

Y	1	2	$\cdots$	n	
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	
1	A_1	$\overline{A_2}$	$\cdots$	$\overline{A_n}$	$\{g_i(a_i)\}$
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	

This $\{a_i\} = \{1,0,\cdots,0\}$ corresponds to a row in the truth table.

Pick up all the $\{a_i\}$ for $Y = 1 \rightarrow \{a_{ij}\}$ with index j

$$Y = \sum_j \prod_{i=1}^{n} g_i(a_{ij})$$

principal disjunctive canonical expansion (主加法標準展開)

Quein-McCluskey algorithm

(Example) $Y = \bar{A}\cdot\bar{B}\cdot C\cdot D + B\cdot C\cdot D + A\cdot B\cdot\bar{C} + A\cdot\bar{B}\cdot C\cdot D$

$$Y = \bar{A}\cdot\bar{B}\cdot C\cdot D + (A+\bar{A})\cdot B\cdot C\cdot D + A\cdot B\cdot\bar{C}\cdot(D+\bar{D}) + A\cdot\bar{B}\cdot C\cdot D$$
$$= \bar{A}\cdot\bar{B}\cdot C\cdot D + A\cdot B\cdot C\cdot D + \bar{A}\cdot B\cdot C\cdot D + A\cdot B\cdot\bar{C}\cdot D$$
$$+ A\cdot B\cdot\bar{C}\cdot\bar{D} + A\cdot\bar{B}\cdot C\cdot D$$

Or in binary: Y = 0011+1111+0111+1101+1100+1011

Classification with the number of 1

Compression with $A + \bar{A} = 1$ occurs between different classes.

Num.of 1	smallest	compress1	compress2
2	0011	0_11	__11
	1100	_011	__11
3	0111	110_	
	1011	_111	
	1101	1_11	
4	1111	11_1	

Quein-McCluskey algorithm

Original terms →

Y = __11+110_+11_1

	smallest					
	0011	1100	0111	1011	1101	1111
__11	○		○	○		○
110_		○			○	
11_1					○	○

Search for redundant terms.
Put circles if the original term contains the expression.
Then indispensable ones should be marked with double circles.

	smallest					
	0011	1100	0111	1011	1101	1111
__11	◎		◎	◎		◎
110_		◎			◎	
11_1					○	○

Single circle in one column

Give priority to already indispensable terms.

Final form *Y* = __11+110_