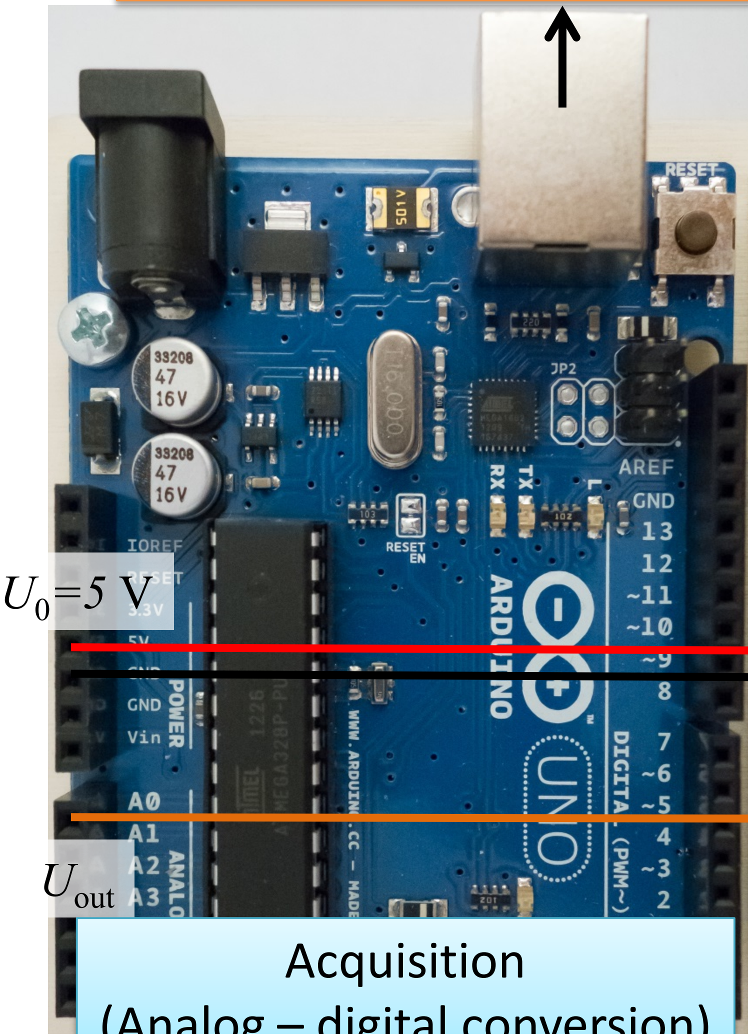
Measurement systems

Lecturer: Andras Kis

Chapter 4: Data Acquisition

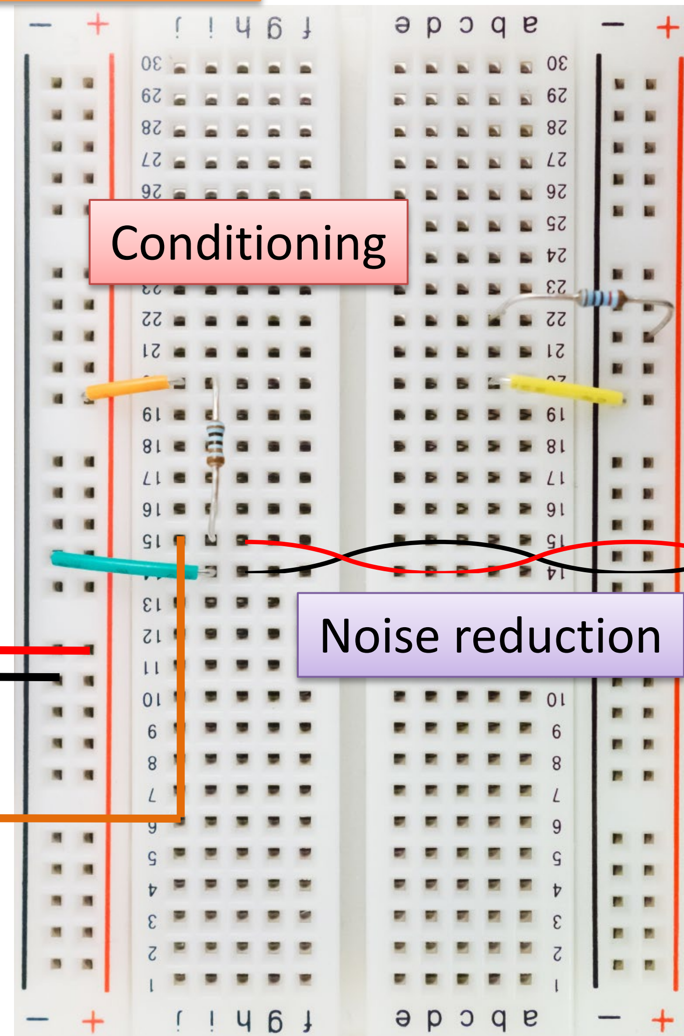
Measurement chain

Data analysis (recording, averaging, etc.)



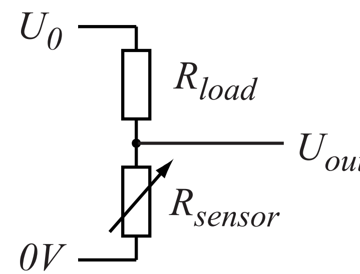
Acquisition
(Analog – digital conversion)

Arduino UNO board



Conditioning

Noise reduction

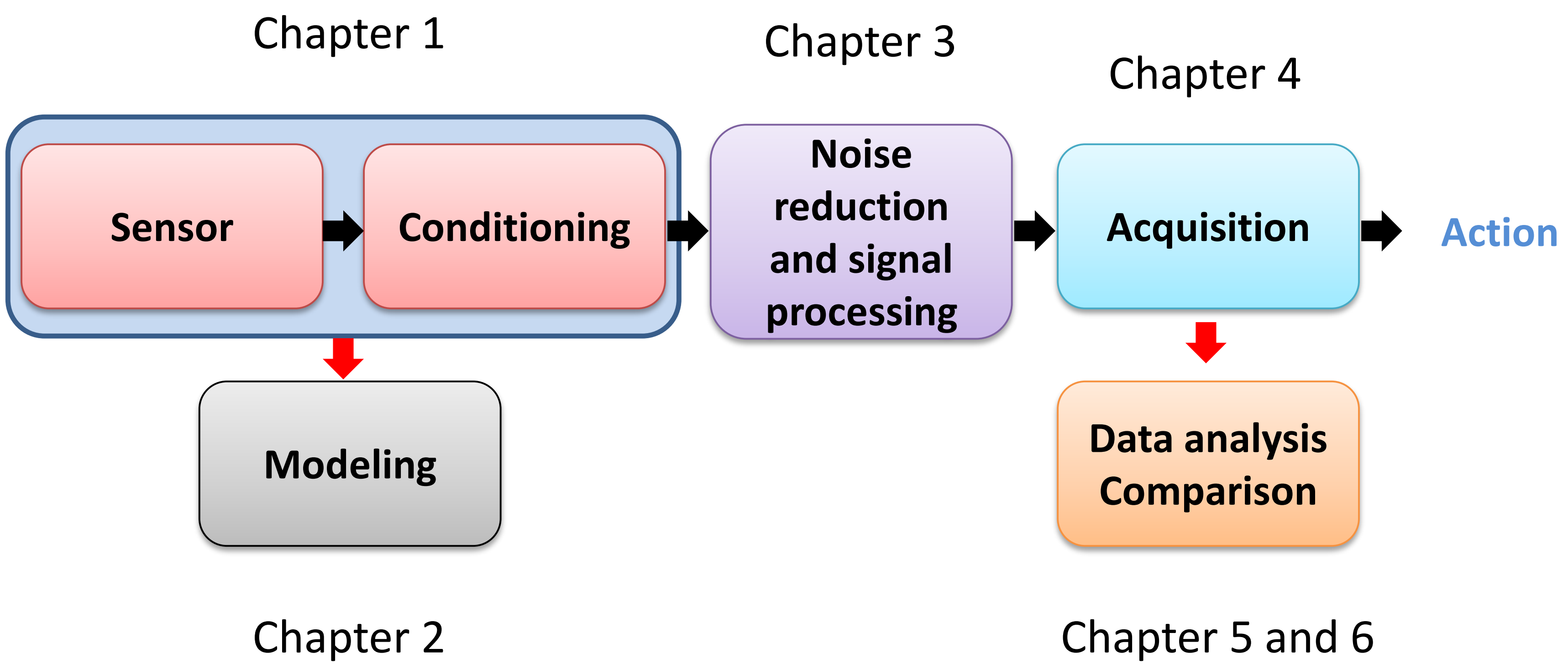


Sensor

Modeling

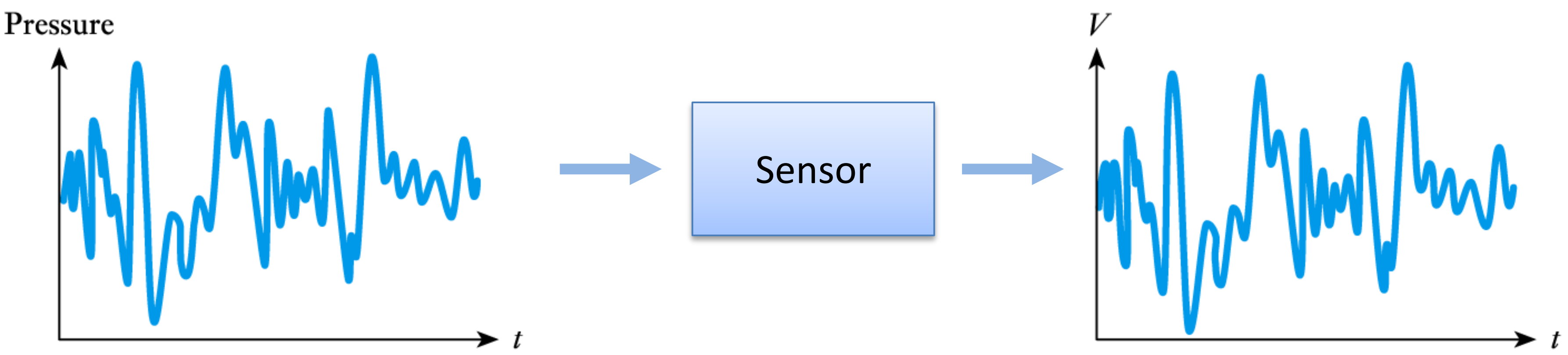
Conditioning circuit

Measurement chain



Analog signal

- Continuous both in amplitude and time and can assume an infinite number of different values – infinite resolution

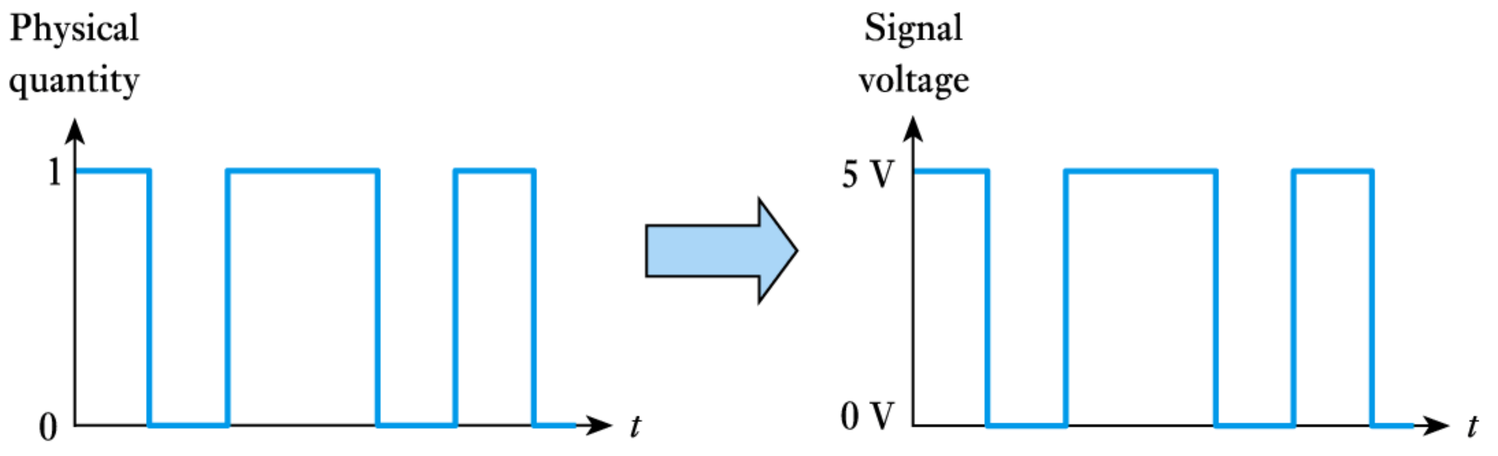


Analog music recording

- Late 1980's, early 90s

Digital signal

- Signal is represented as two values (“low” and “high”), with distinct voltage levels

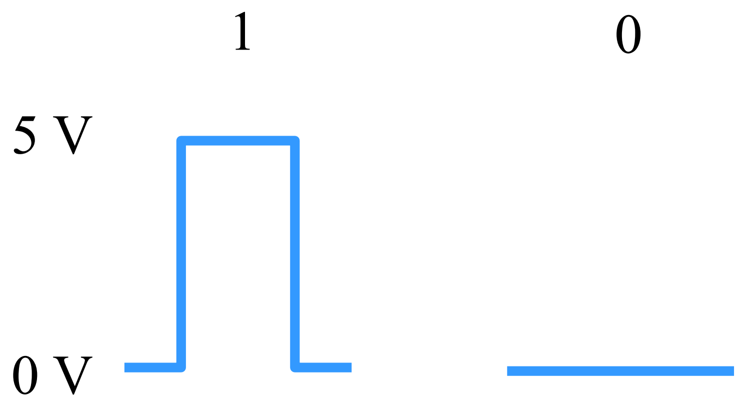


Bit and byte

- A digital signal can represent either a state of a quantity **(bit)** or be an element of a unit of information **(byte)**

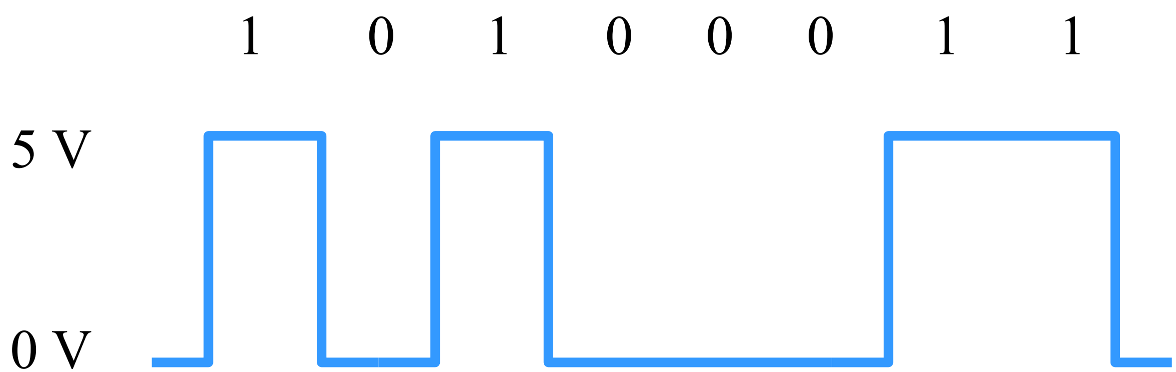
Possible values

Bit



2

Byte (composed of N bits)

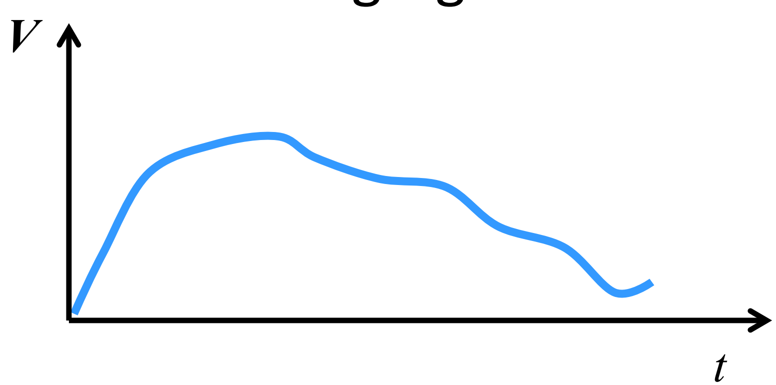


2^N

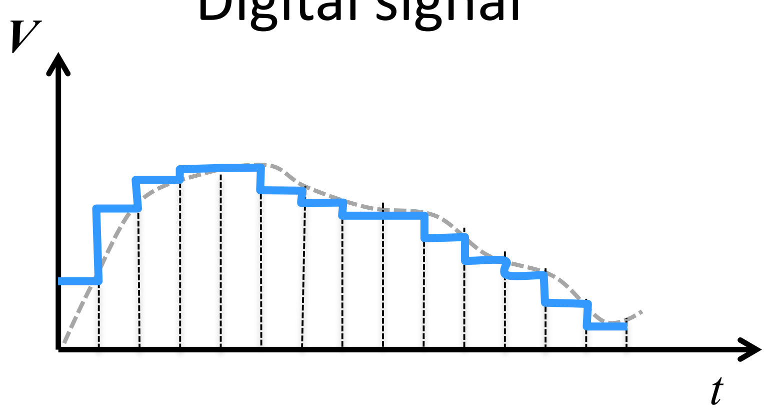
Analog – digital conversion

- The digital signal is:
  - Less perturbed by noise
  - Easier to process, transmit or store
- Signal is often converted between analog – digital forms
  - Music playback, generation of analog voltages using computer-controlled instruments etc.
- AD and DA converters

Analog signal

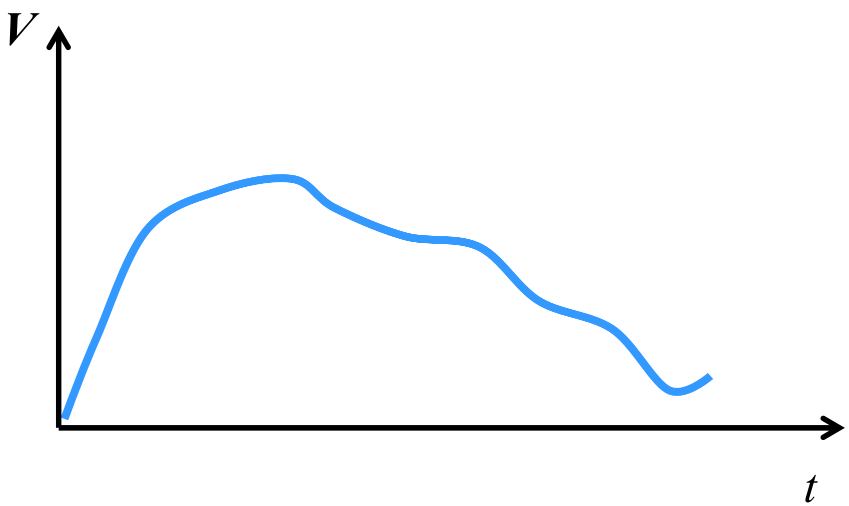


Digital signal



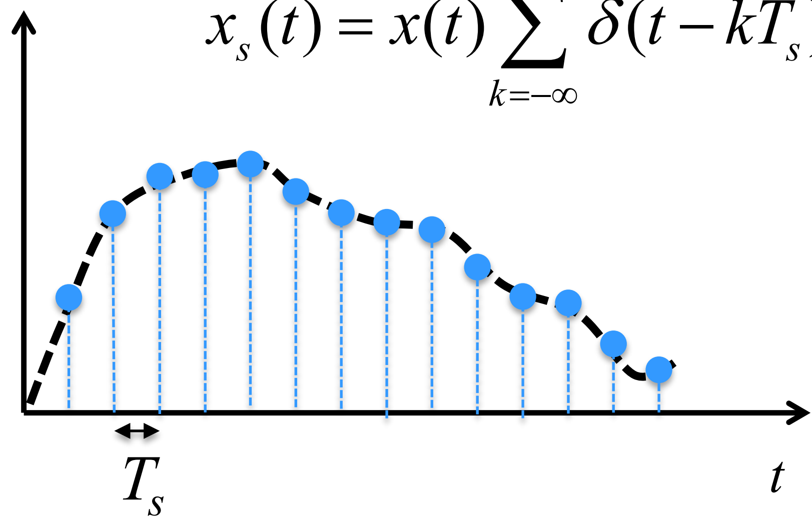
Sampling

Analog signal

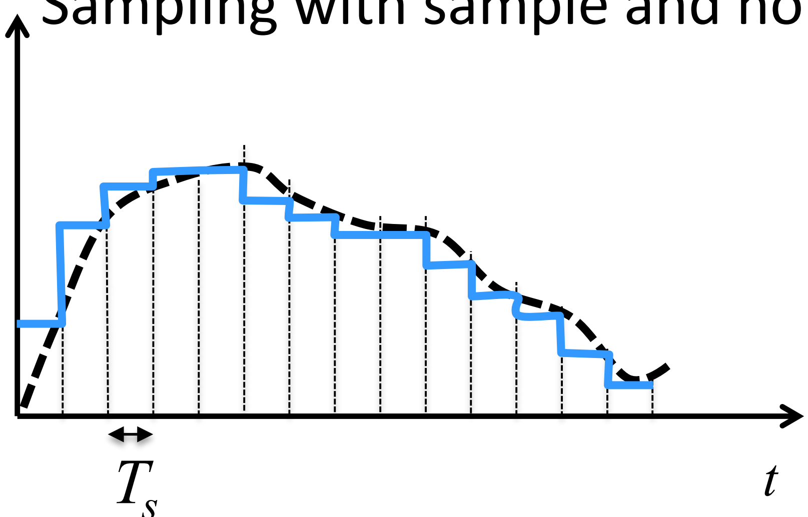


Ideal sampling

$$x_s(t) = x(t) \sum_{k=-\infty}^{+\infty} \delta(t - kT_s)$$



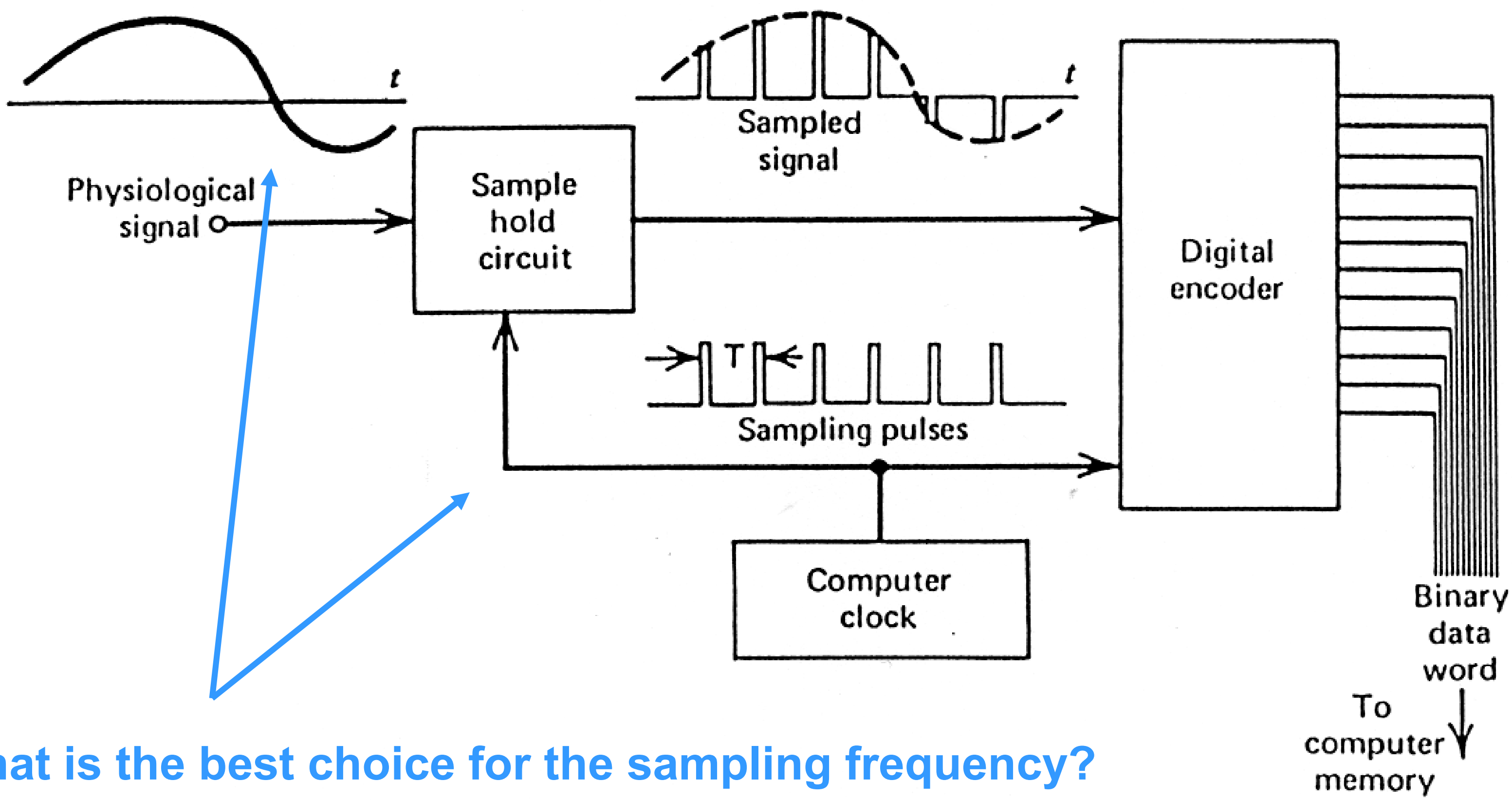
Sampling with sample and hold



$$f_s = \frac{1}{T_s}$$ sampling frequency

Sampling

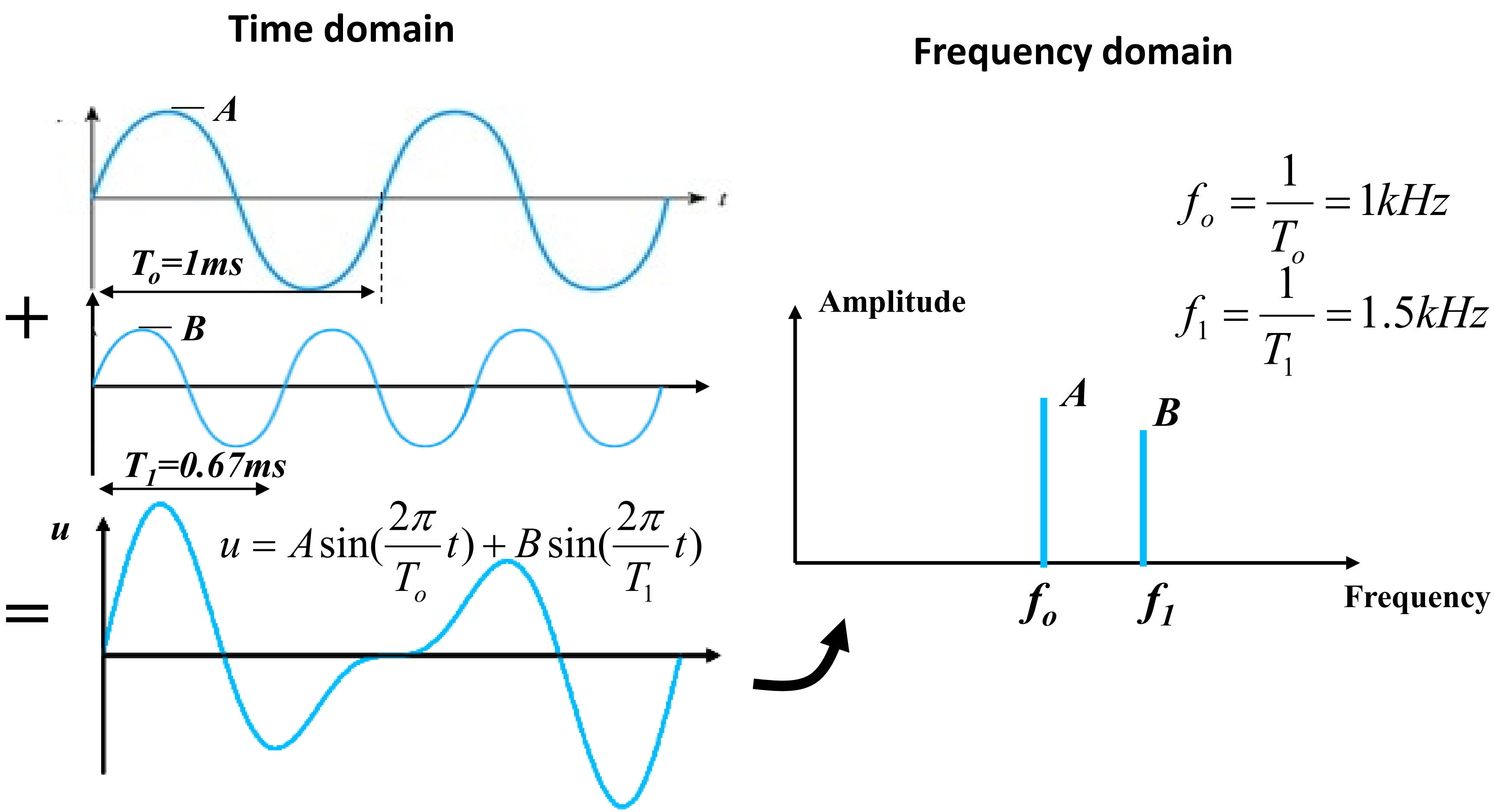
- Before the conversion, the analog signal is sampled
- The signal to be sampled is multiplied with a pulse train signal



What is the best choice for the sampling frequency?

Reminder: frequency spectrum

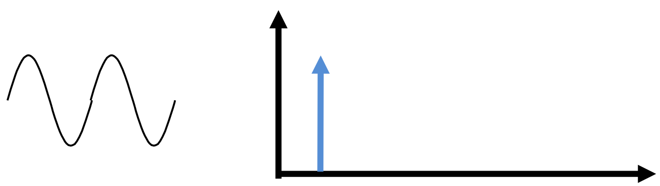
- Sinusoidal signals

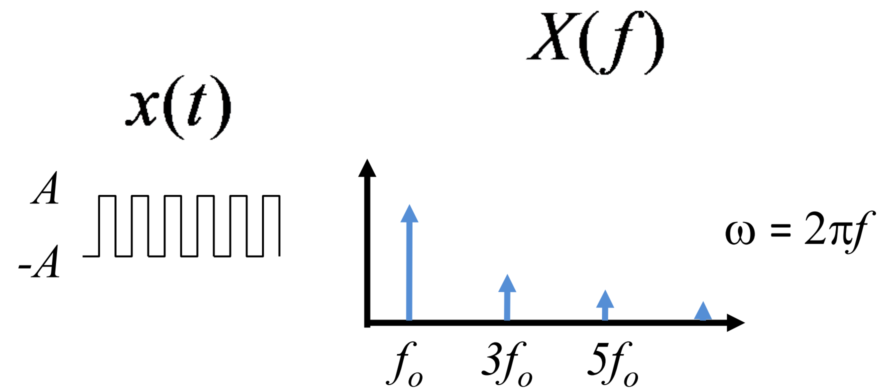


Representation

- Periodic signal

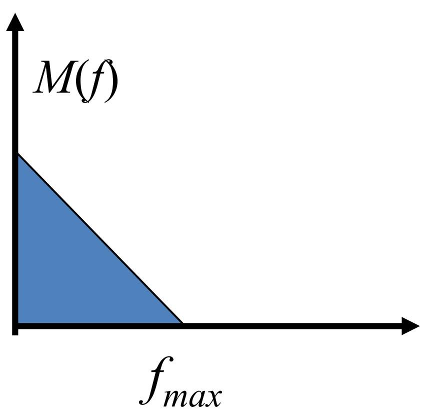
Signal FFT



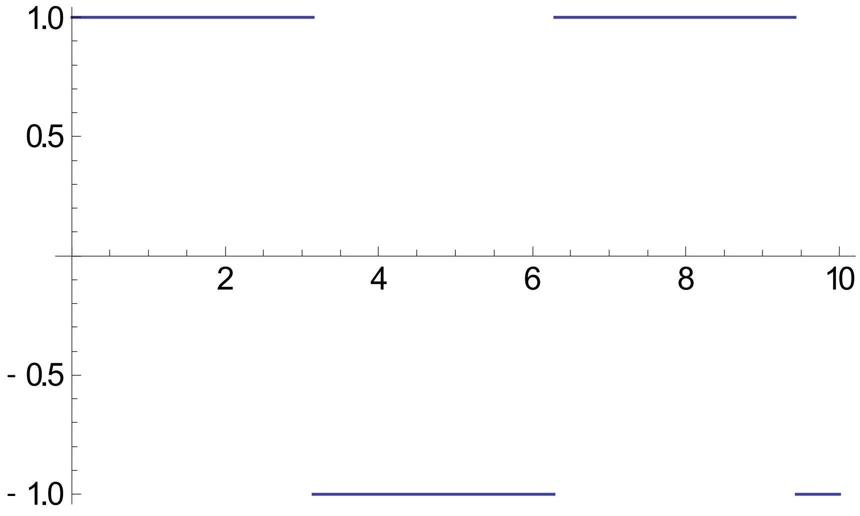


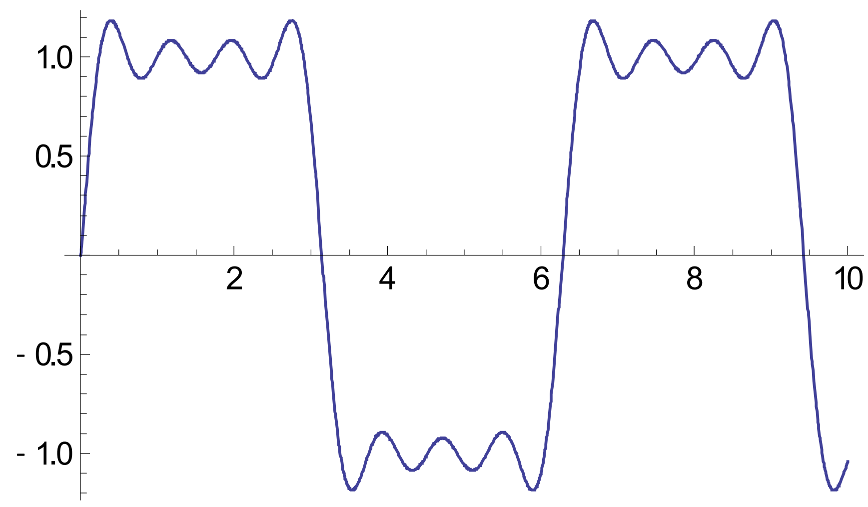
$$x(t) = A\frac{4}{\pi}\left[\sin\omega_o t + \frac{\sin 3\omega_o t}{3} + \frac{\sin 5\omega_o t}{5} + \ldots\right]$$

- Non-periodic signal



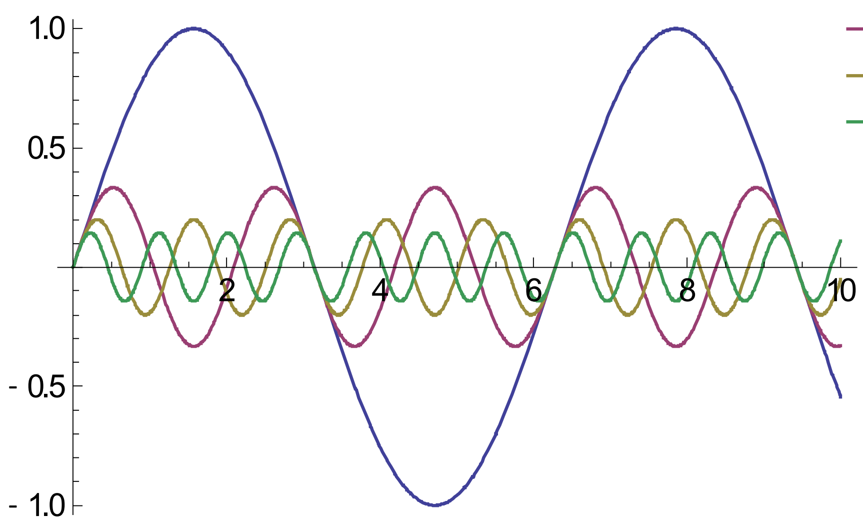
Reconstruction of a square signal





decomposition

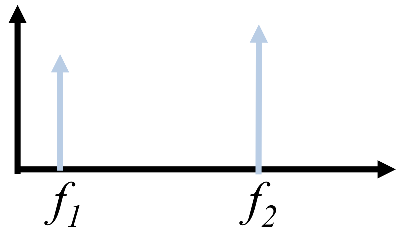
reconstruction

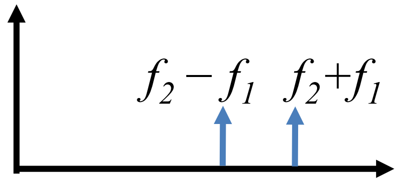


Multiplication operation

- Product:

$$\cos\left(2\pi f_1 t\right)\cdot\cos\left(2\pi f_2 t\right)=\left[\cos(2\pi(f_2-f_1)t)+\cos 2\pi(f_2+f_1)t\right]/2$$

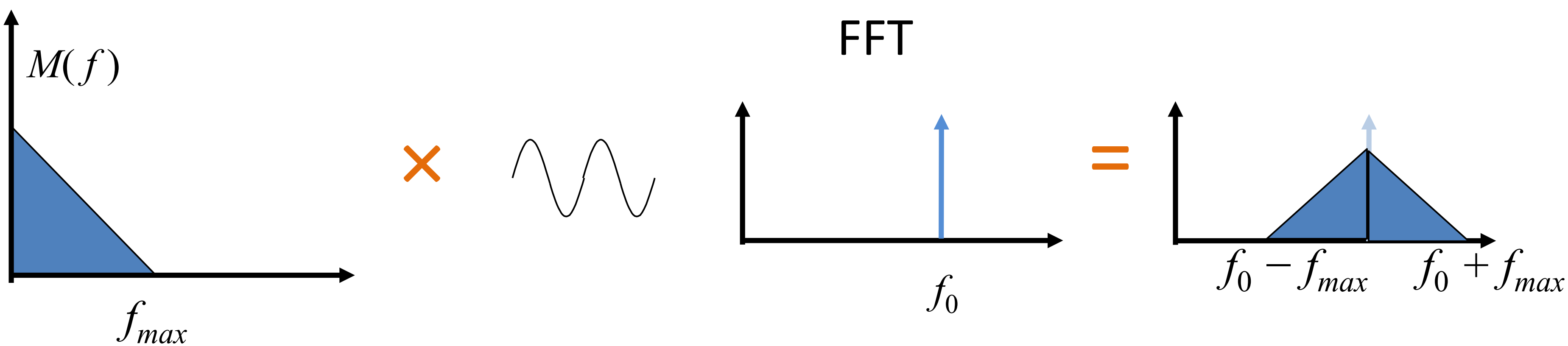




Signal A

Signal B

Signal A × B



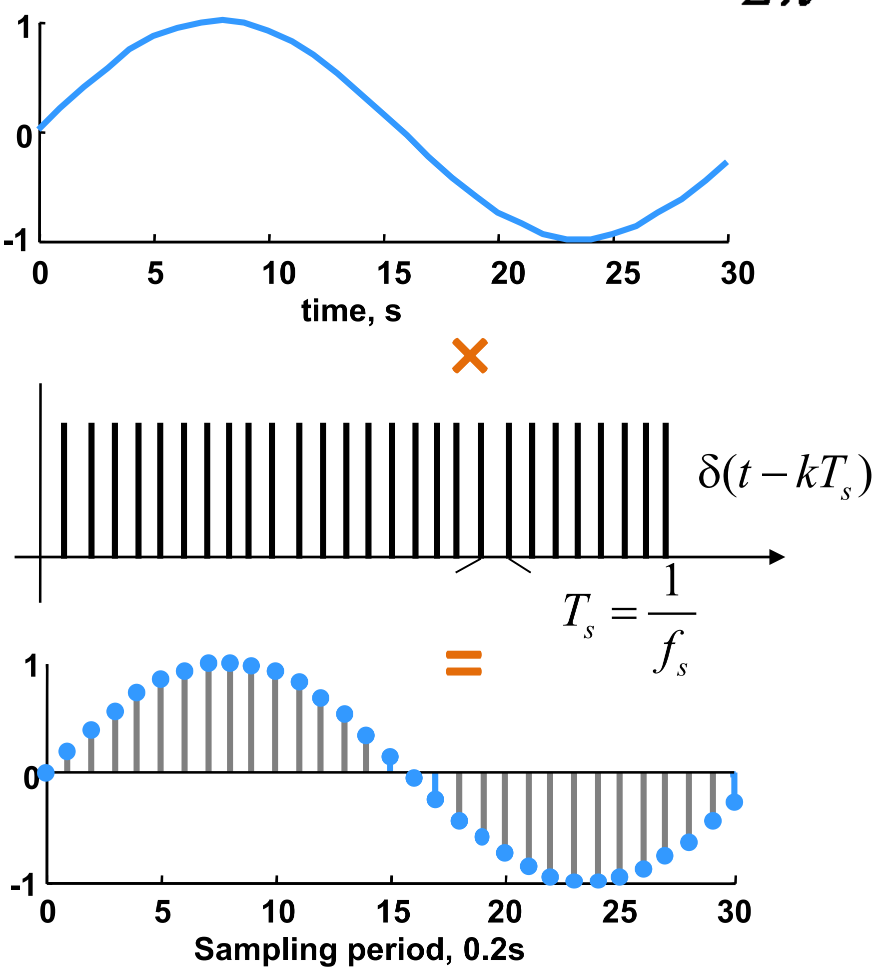
$\Delta f = f_{max}$ bandwidth,
continuous signal $m(t)$

periodic signal

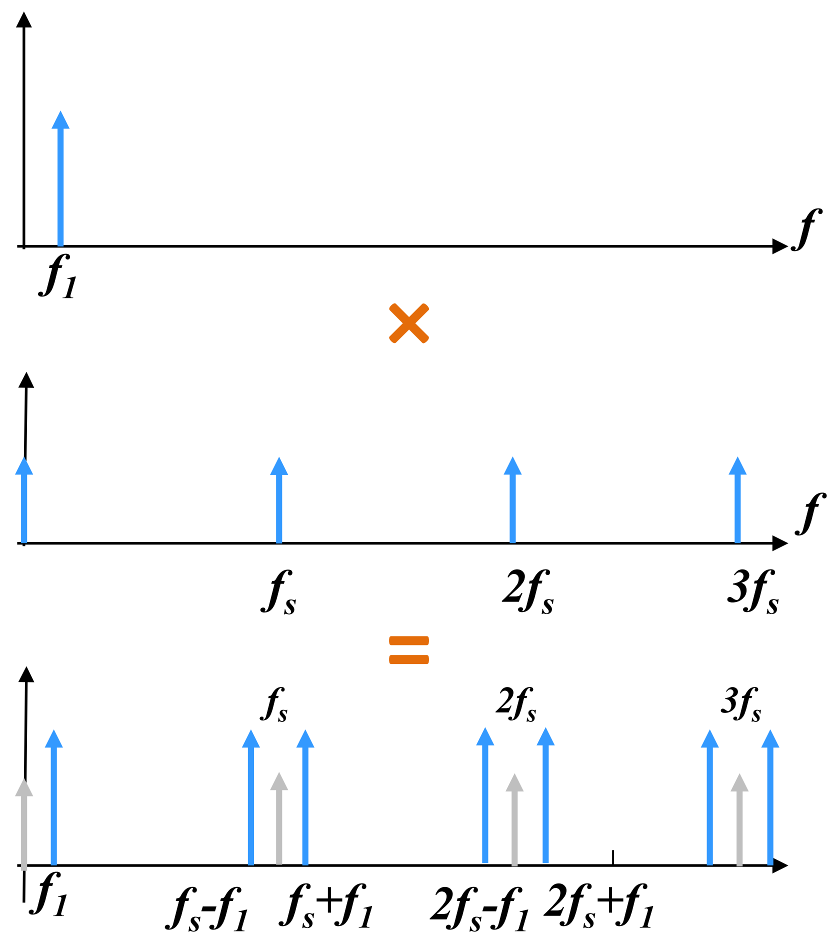
Sampling of a periodic signal

Time domain

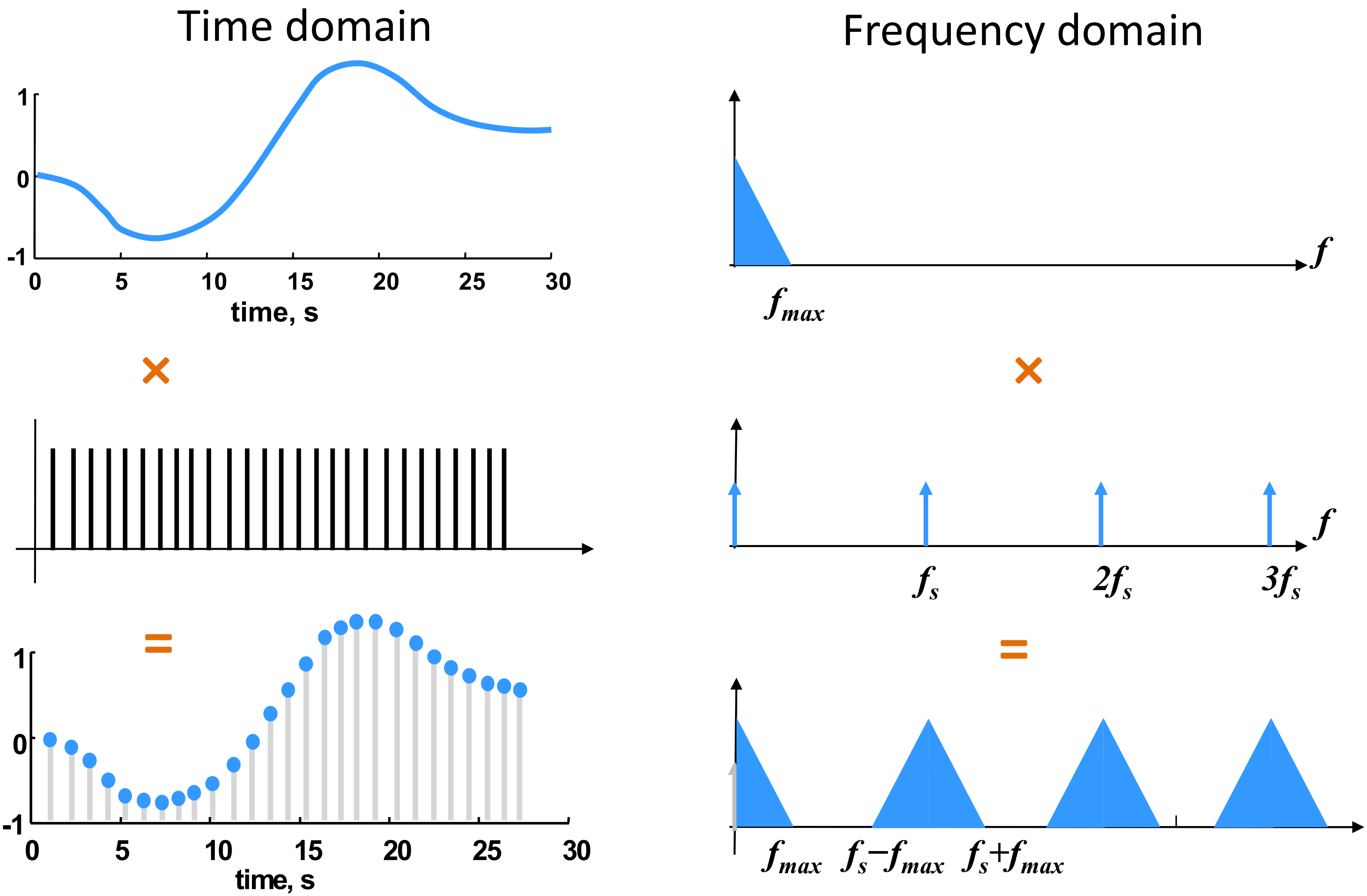
$$x(t) = A \cdot \sin \omega_1 t \qquad f_1 = \frac{\omega_1}{2\pi}$$



Frequency domain



Sampling of an arbitrary signal

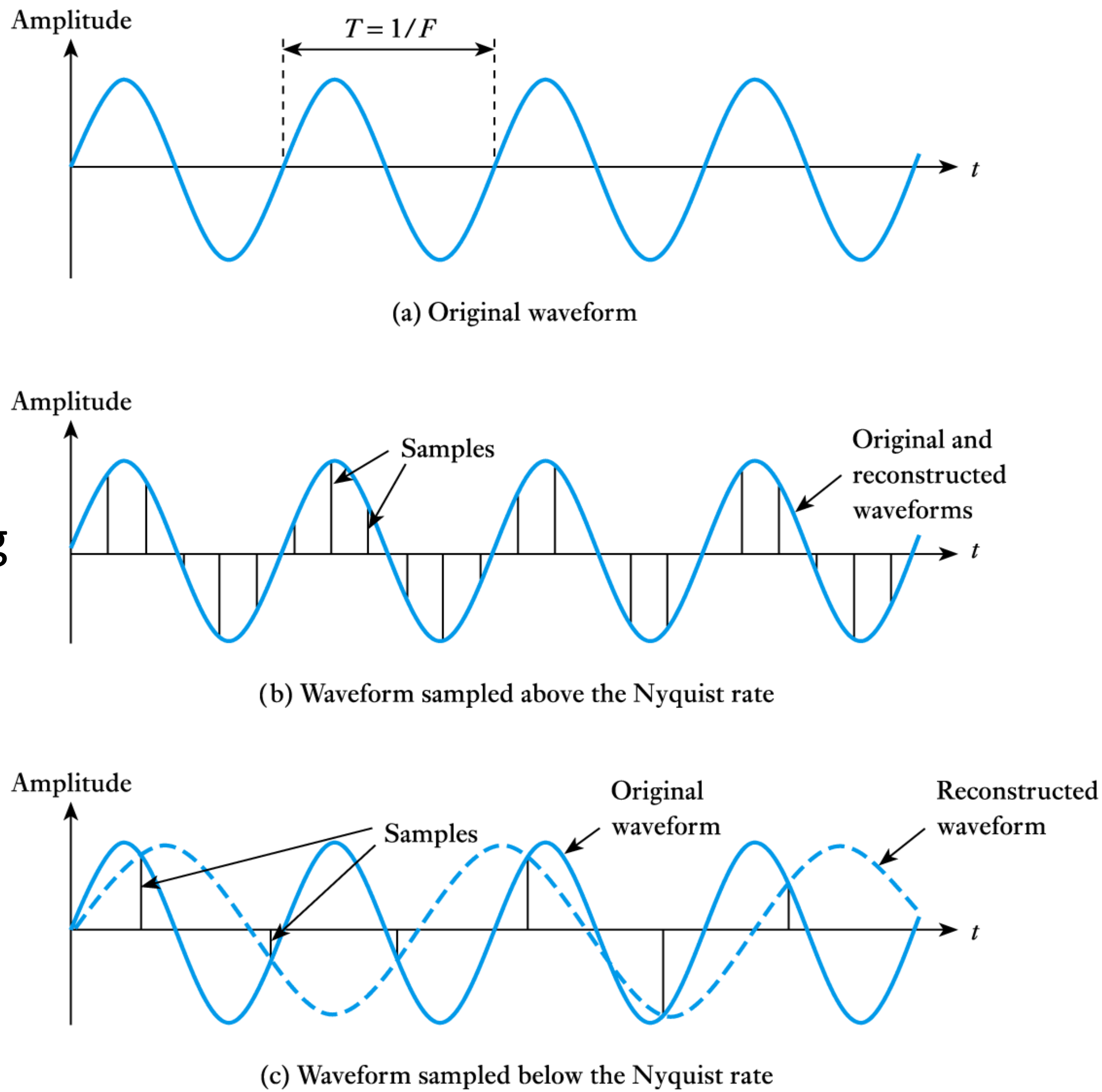


Choice of the sampling frequency

Analog signal

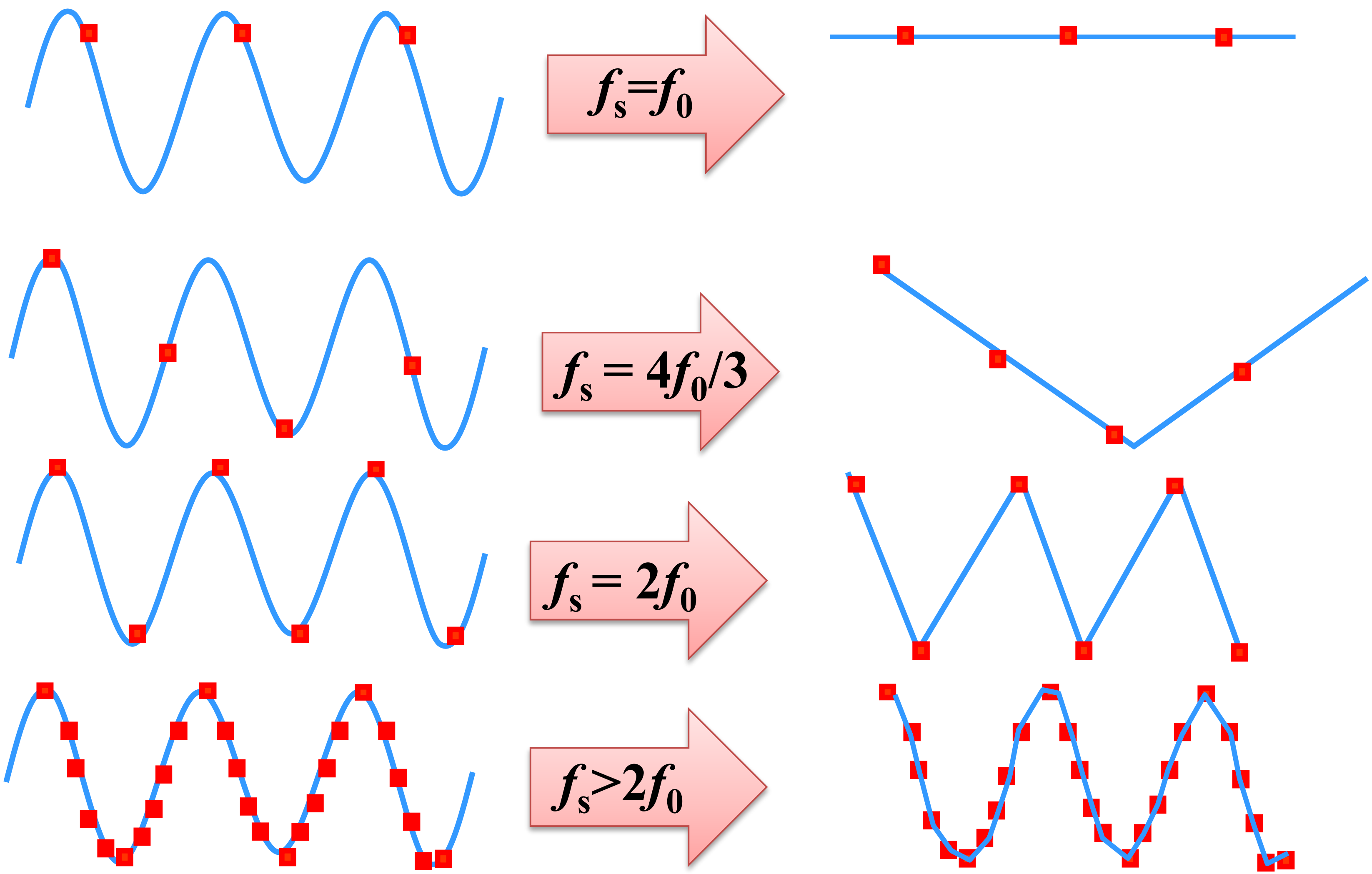
Good sampling

Bad sampling

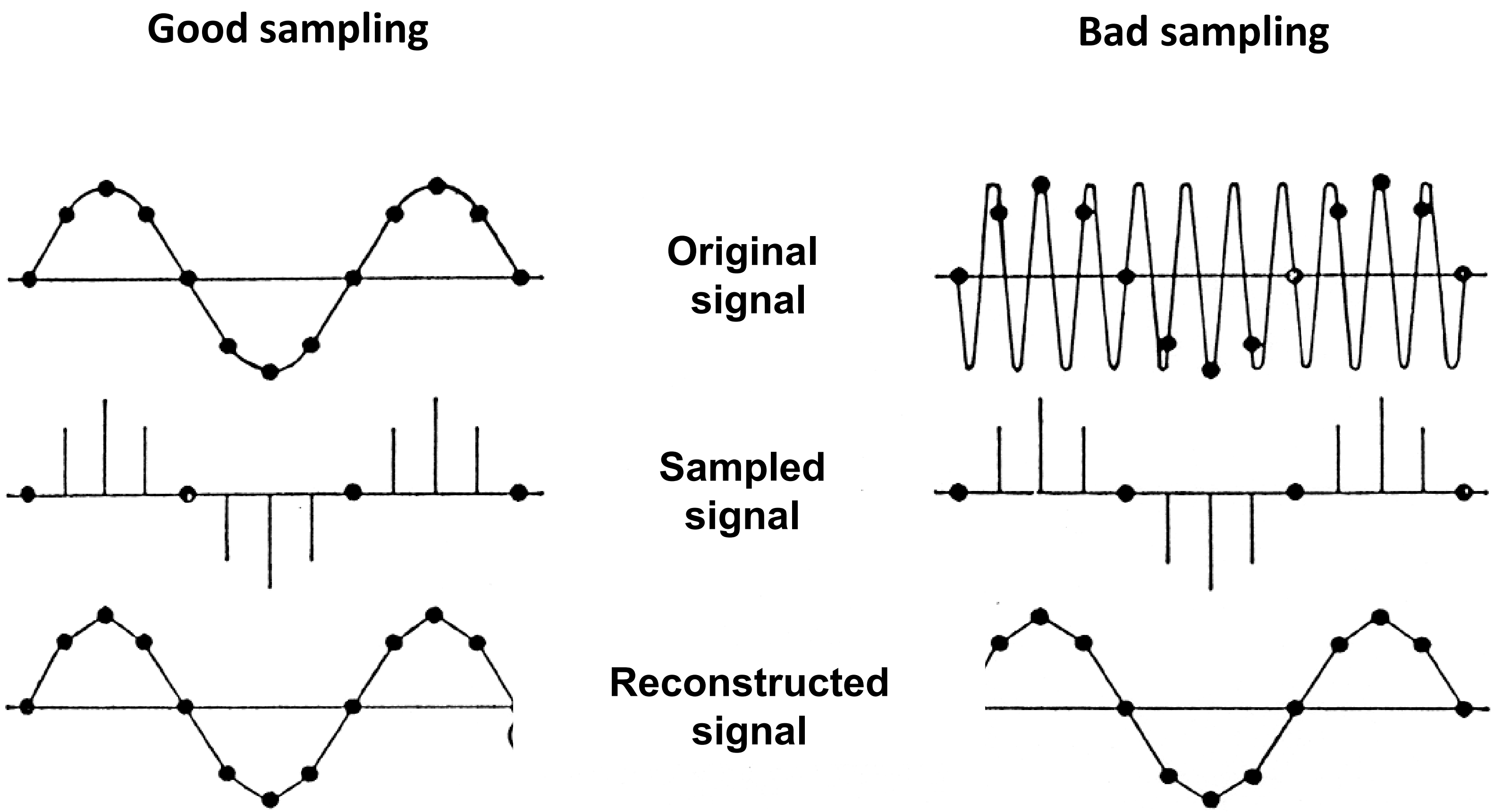


Example: sinusoidal signal, frequency f_0

Original signal + sampling points   Recovered signal

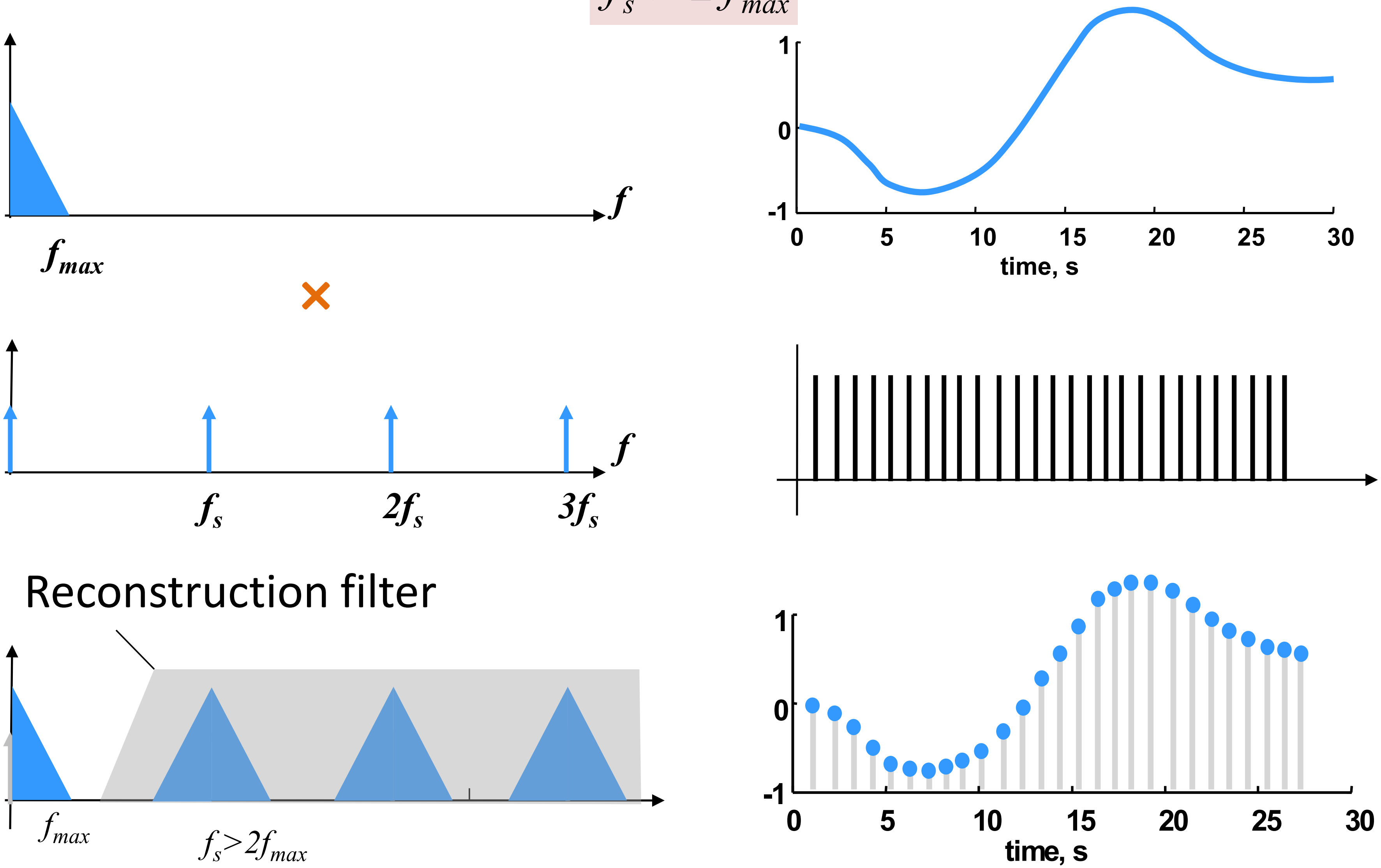


Example: fixed sampling frequency



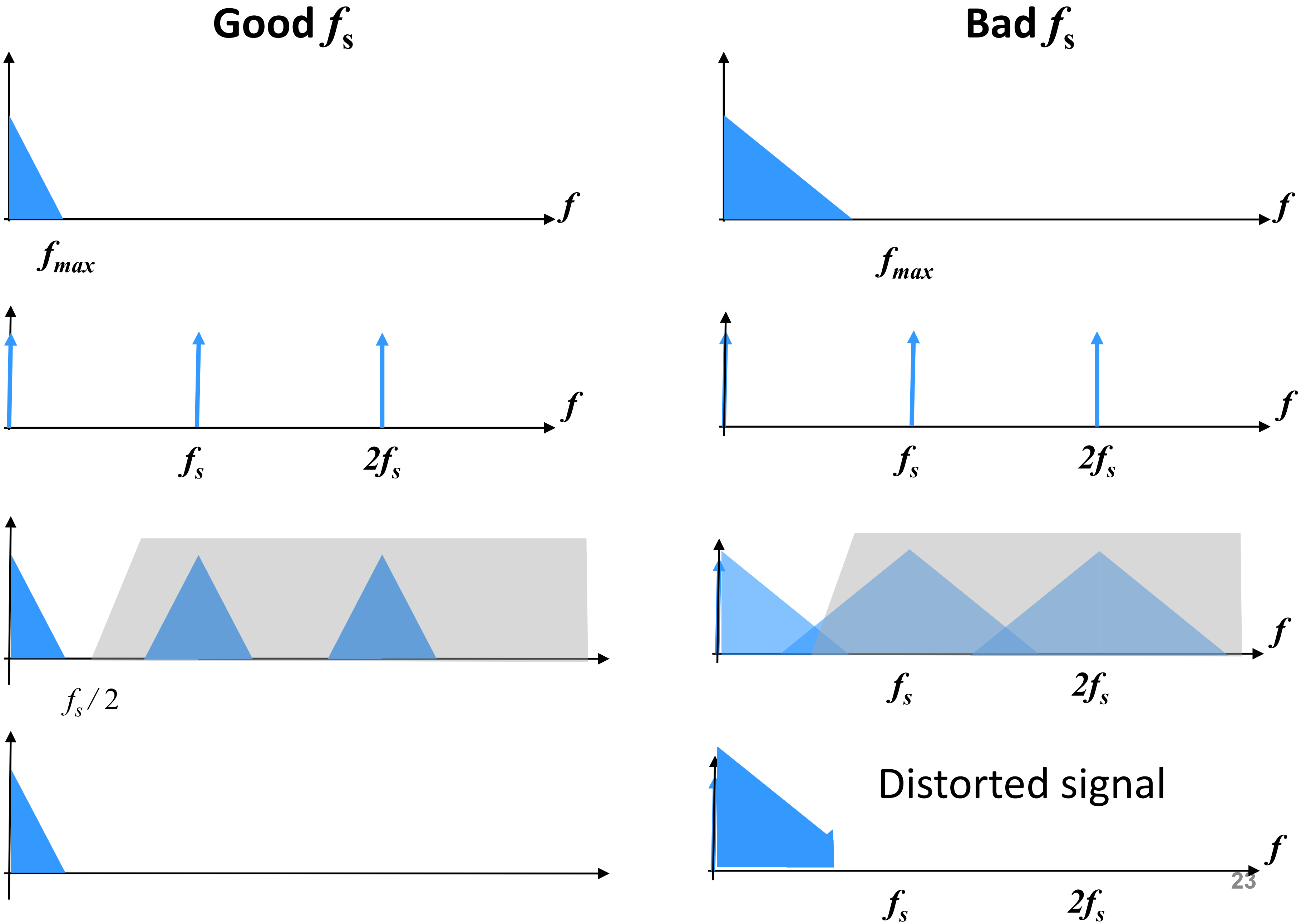
Nyquist - Shannon theorem of sampling

$$f_s > 2f_{max}$$

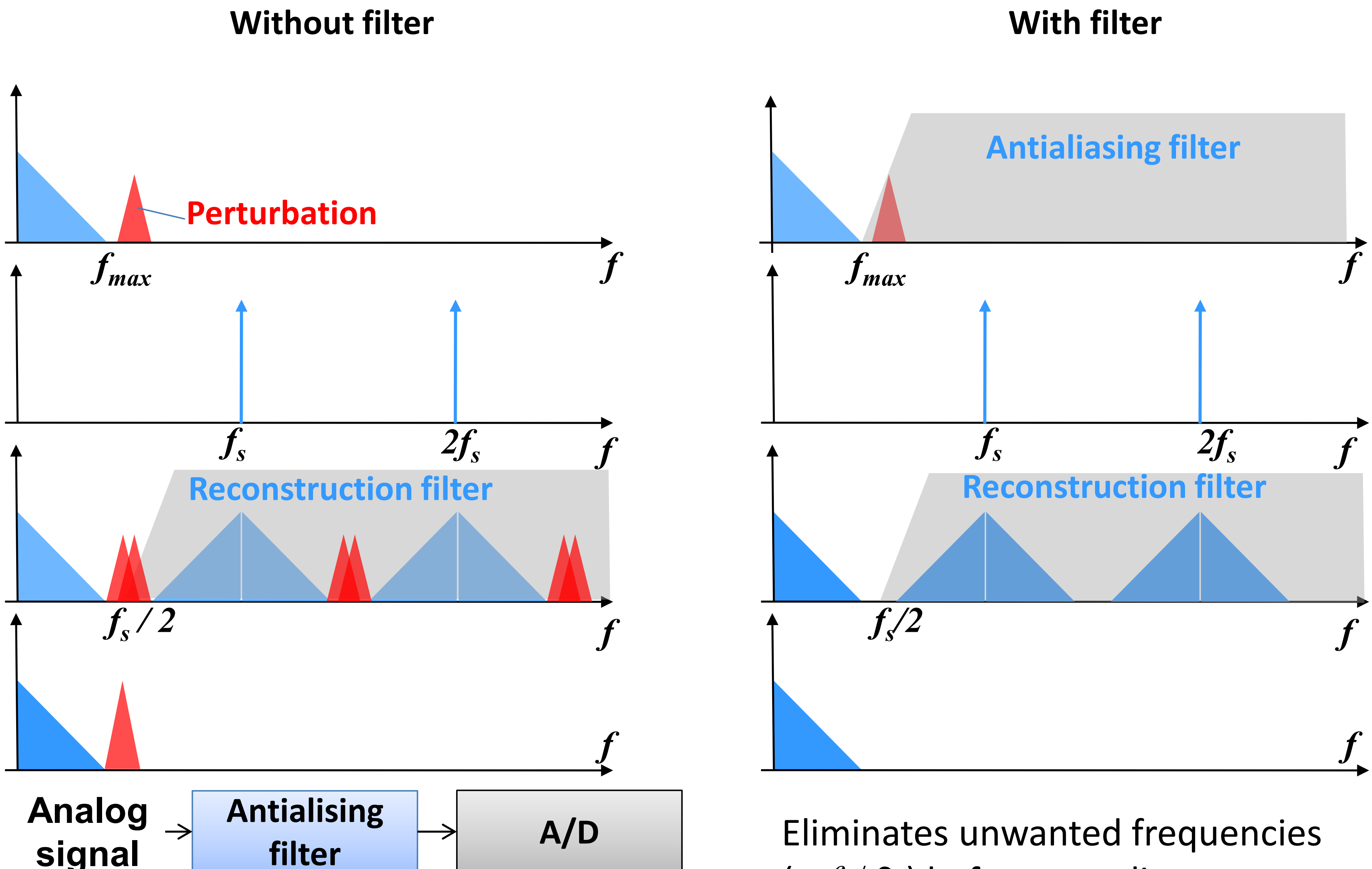


In practice f_s several times larger then f_{max}

Spectral folding

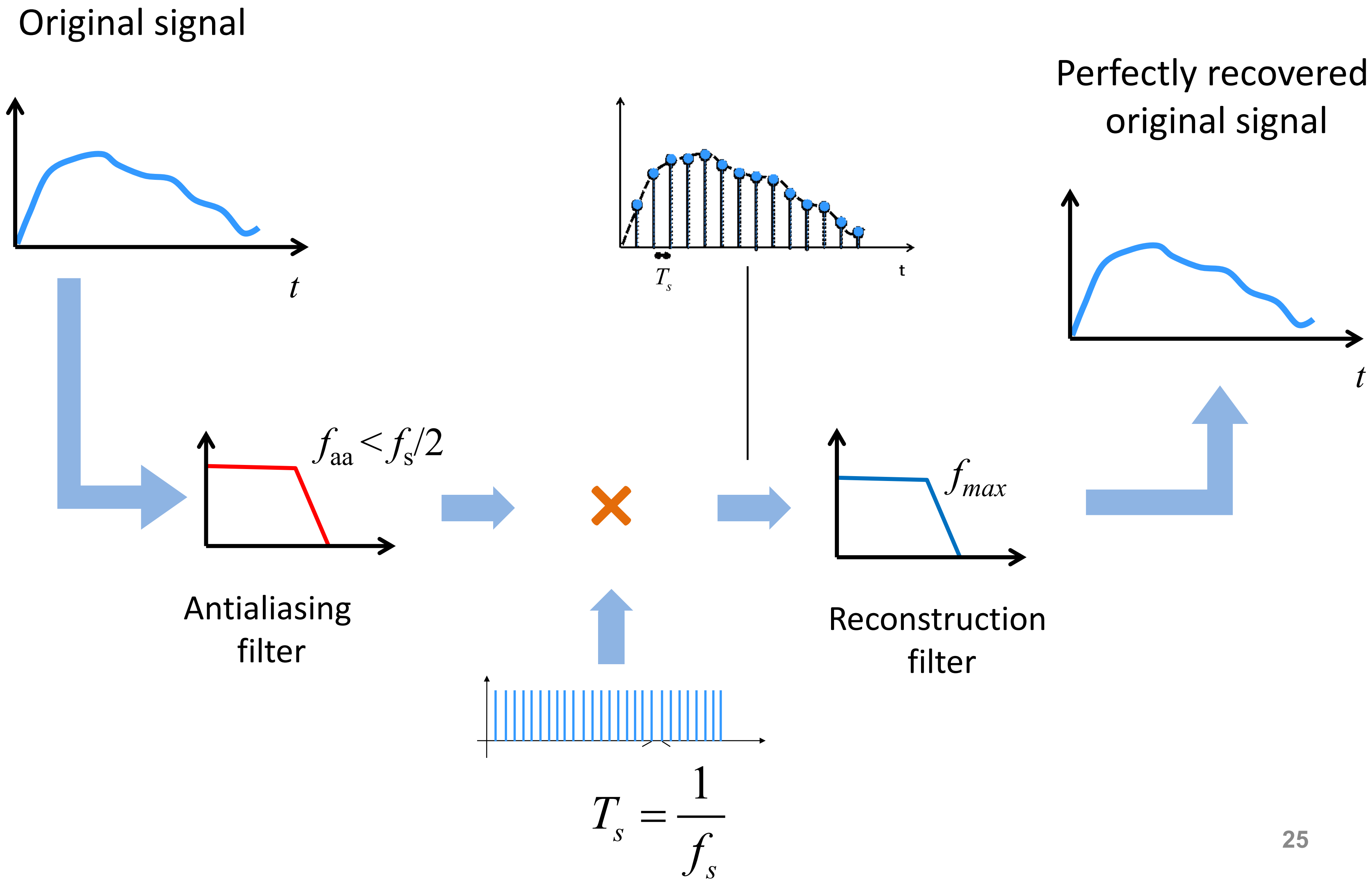


Antialiasing filter



Eliminates unwanted frequencies ($< f_s / 2$) before sampling

Bloc diagram for sampling



Decimal – binary number conversion

- Decimal system

$$1234_{10} = (1 \times 10^3) + (2 \times 10^2) + (3 \times 10^1) + (4 \times 10^0)$$

- Binary system

$$1101_2 = (1 \times 2^3) + (1 \times 2^2) + (0 \times 2^1) + (1 \times 2^0)$$

Conversion binary -> decimal

$$\begin{aligned} 11010_2 &= (1 \times 2^4) + (1 \times 2^3) + (0 \times 2^2) + (1 \times 2^1) + (0 \times 2^0) \\ &= 16 + 8 + 0 + 2 + 0 \\ &= 26_{10} \end{aligned}$$

Conversion decimal -> binary number

- Decimal to binary number

26_{10}

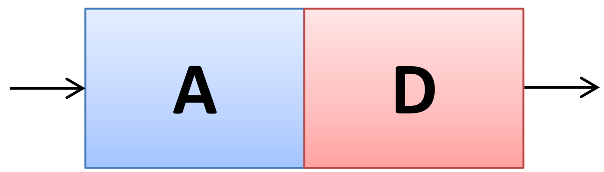
	quotient	remainder
	26	
÷ 2	13	0
÷ 2	6	1
÷ 2	3	0
÷ 2	1	1
÷ 2	0	1

read the number from starting from the last digit

=11010

Encoding

Continuously changing variable



Digital form (binary code)

3 → $0011_2 = 0 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 =$
$= 2 + 1 = 3_{10}$

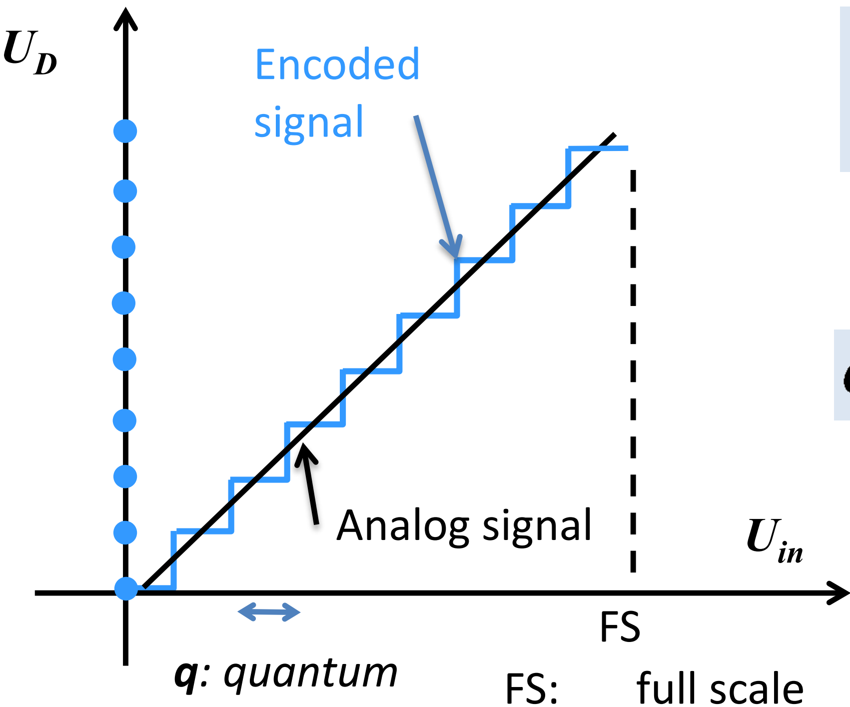
Conversion of a decimal number N_{dec} into a binary code

$\boldsymbol{N_{dec}}$ → $a_1 a_2 a_3 \ldots a_{\mathrm{n\text{-}1}} a_{\mathrm{n}}$

$$N_{dec} = \sum_{i=1}^{n} a_i 2^{n-i} = a_1 2^{n-1} + a_2 2^{n-2} + \cdots + a_{n-1} 2^1 + a_n 2^0$$
$$= 2^n \sum_{i=1}^{n} a_i 2^{-i} = 2^n \left(a_1 2^{-1} + a_2 2^{-2} + \cdots + a_n 2^{-n} \right)$$

a_1 MSB – most significant bit
a_{n} LSB – least significant bit

Quantisation



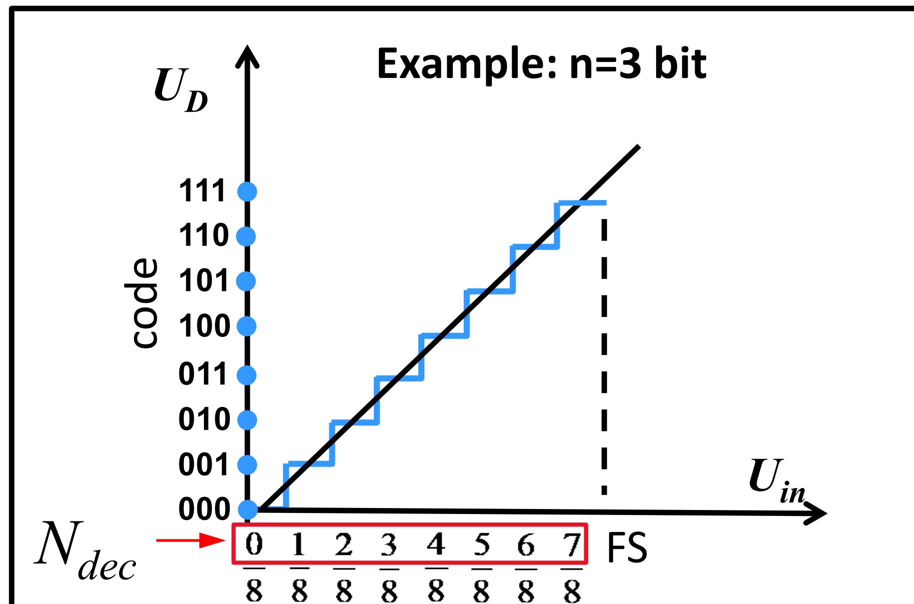
$$q = \frac{FS}{2^n}$$

$$N_{dec} = 2^n \cdot \frac{U_D}{FS}$$

ADC value (analog to digital converter output value) – ranges from 0 to 2^n

n: number of bits

$$code(N_{dec}) : a_1 a_2 a_3 \ldots a_{n-1} a_n$$



Example

- Convert 4.5V with an 8-bit AD converter with a FS = 5V

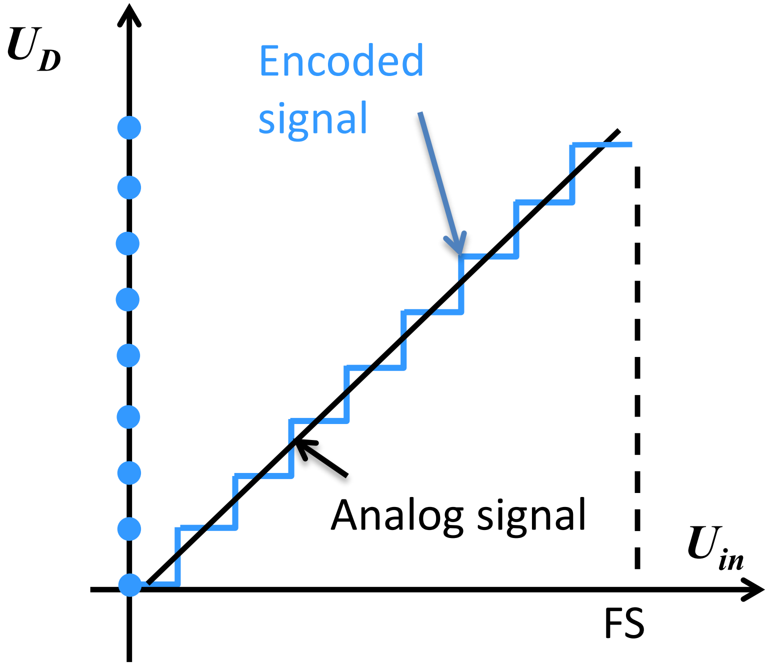
$$N_{dec} = 256 \times \frac{4.5}{5} = 230 = (11100110)_2$$

$$Resolution = \frac{5}{256} = 0.019V \ \left(0.01953V\right)$$

- Convert an ADC value of 156 to volts (8 bit converter and FS = 5V)

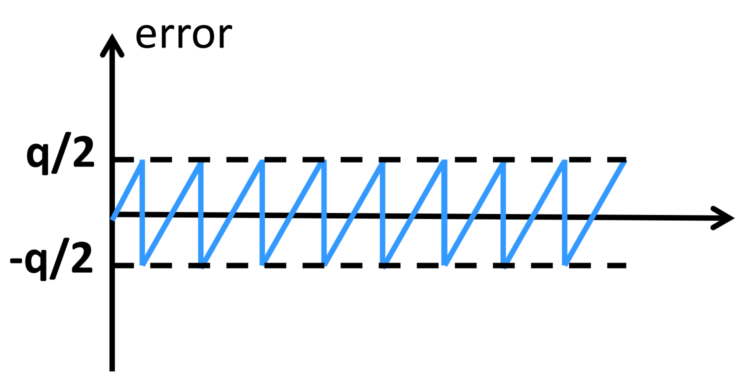
$$U_D = \frac{N_{dec}}{2^n} FS = \frac{156}{256} 5 = 3.0469V \qquad U_{in} = 3.0469 \pm 0.0098V$$

Quantisation error



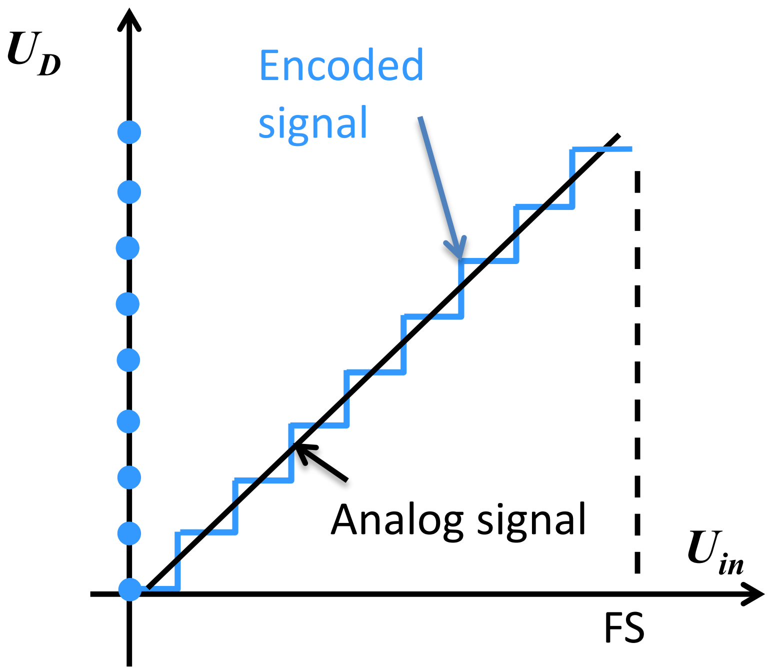
$$\textit{Quantisation error} = \left|U_D - U_{in}\right|$$

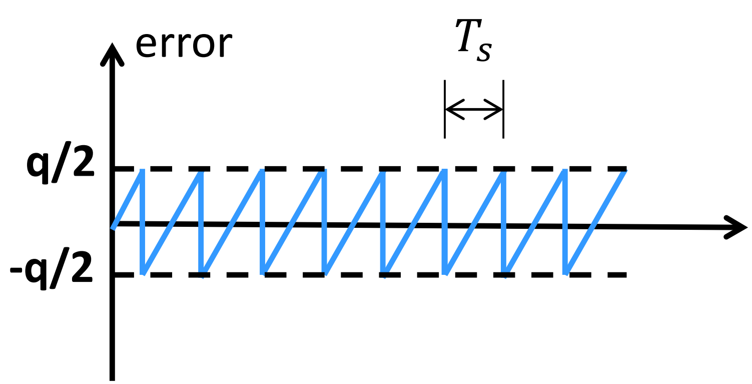
$$= \left|\frac{N_{dec}}{2^n} FS - U_{in}\right|$$



$$\textit{Max quantisation error} = \pm\frac{0.5 \cdot FS}{2^n} = \pm\frac{q}{2}$$

Quantisation error as noise





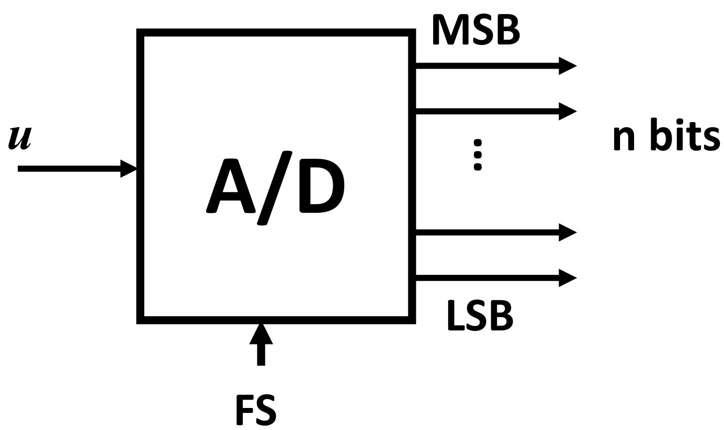
Power of the noise associated with the quantisation error (R = 1Ω)

$$P_n = \frac{1}{T}\int_0^T u^2{}_n(t)dt = \frac{2}{T_s}\int_0^{T_s/2}\left(\frac{q/2}{T_s/2}t\right)^2 dt =$$

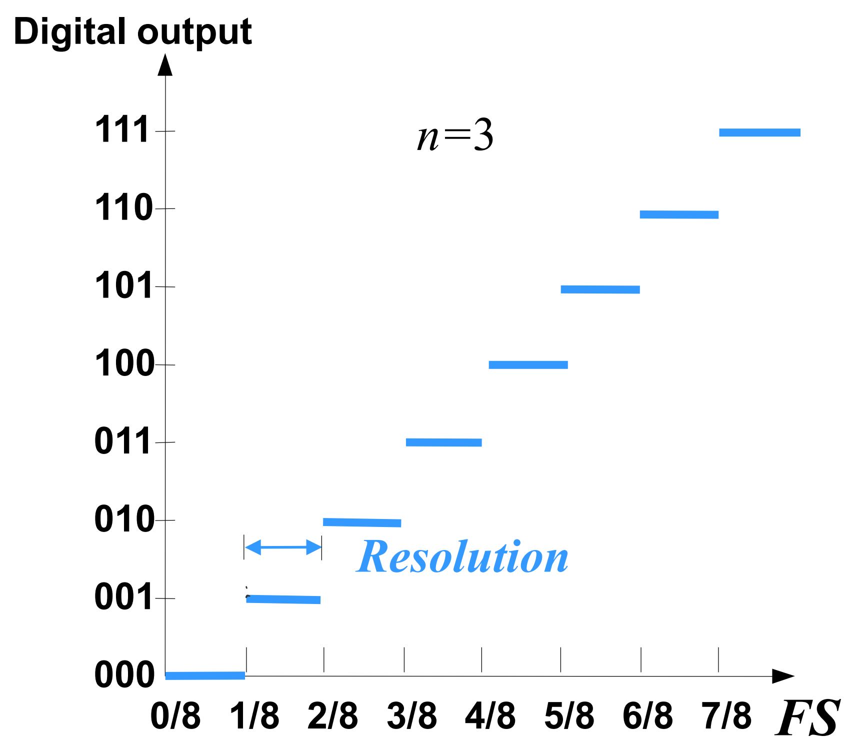
$$= \frac{2}{T_s}\frac{q^2}{T_s^2}\left[\frac{t^3}{3}\right]_0^{T_s/2} = \frac{2q^2}{T_s^3}\frac{T_s^3}{24} = \frac{q^2}{12}$$

Resolution

- The smallest detectable variation of the input



$$\text{Resolution} = \frac{1}{2^n} FS = q$$



Example:

FS = 5V, n = 4

resolution = 5V/2^4 = 0.31V

Example

- Convert 4.5V with an 8-bit AD converter with a FS = 5V

$$N_{dec} = 256 \times \frac{4.5}{5} = 230 = (11100110)_2$$

$$Error = \left| \frac{230}{256} 5 - 4.5 \right| = \left| 4.4922 - 4.5 \right| = 0.0078V$$

$$Max\ error = \frac{0.5 \times 5}{256} = 0.0098V$$

$$Resolution = \frac{5}{256} = 0.0195V \quad \left(0.01953V \right)$$

- Convert an ADC value of 156 to volts (8 bit converter and FS = 5V)

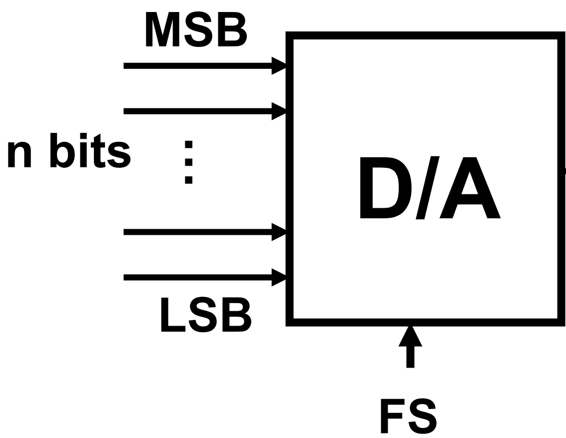
$$U_D = \frac{N_{dec}}{2^n} FS = \frac{156}{256} 5 = 3.0469V \qquad U_{in} = 3.0469 \pm 0.0098V$$

Example: 12 bit converter

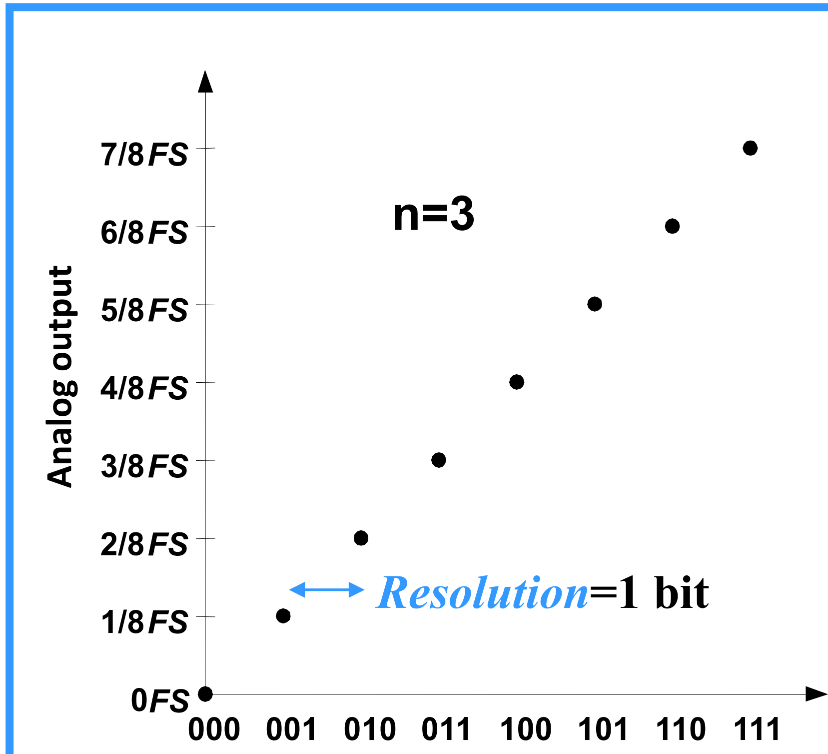
$$\text{Resolution} = \frac{1}{2^{12}} FS = \frac{FS}{4096}$$

FS	Resolution
0 à 10 V	2,44 mV
0 à 5 V	1,22 mV
0 à 2,5 V	610 µV
0 à 1,25 V	305 µV
0 à 1 V	244 µV
0 à 0,1 V	24,4 µV
0mV à 20 mV	4,88 µV
–5 à 5V	2,44 mV
–2,5 à 2,5 V	1,22 mV
–1,25 à 1,25 V	610 µV
–0,625 à 0,625 V	305 µV
–0,5 à 0,5 V	244 µV
–50mV à 50 mV	24,4 µV
–10mV à 10 mV	4,88 µV
–10 à 10 V	4,88 mV
–5 à 5 V	2,44 mV
–2,5 à 2,5 V	1,22 mV
–1,25 à 1,25 V	610 µV
–1 à 1 V	488 µV
–0,1 à 0,1 V	48,8 µV
–20mV à 20 mV	9,76 µV

Digital/Analog (D/A) Converter



$$U_{D/A} = \frac{N_{dec}}{2^n} FS$$

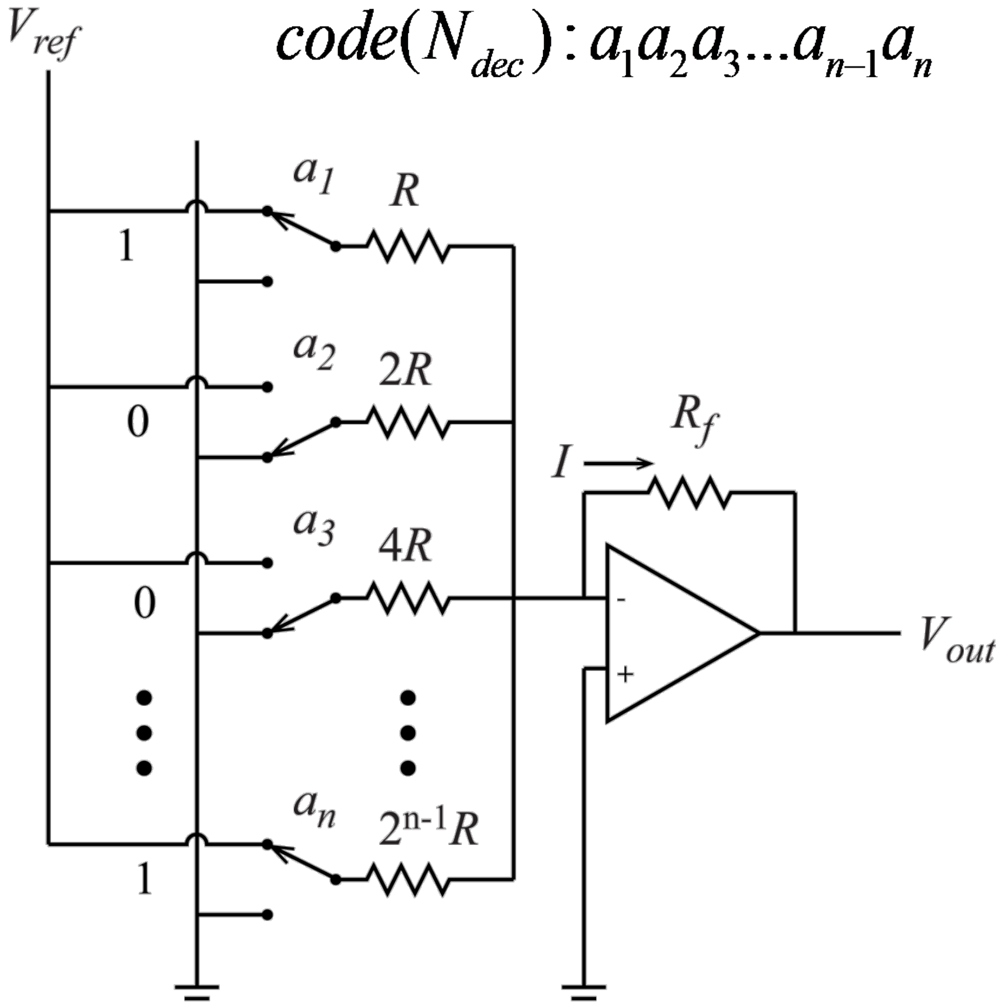


Ex. FS = 5V, n =4 , code= 1111:

$$N_{dec} = 15$$

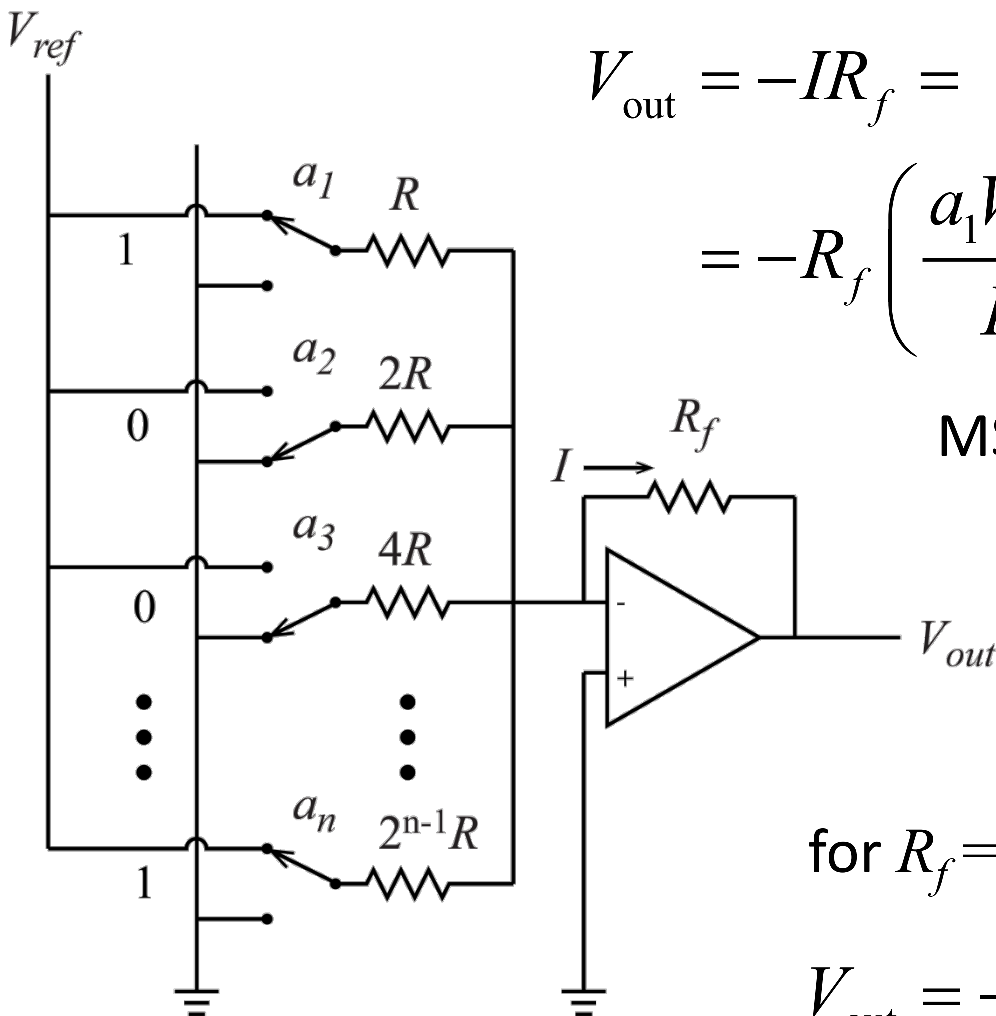
$$U_{D/A} = \frac{15}{2^4} \times 5 = 4.69V$$

D/A converter: binary weighted ladder



- Each input resistor is twice the value of the previous one
- Inputs are weighted according to their resistors

D/A converter: binary weighted ladder



$$V_{\text{out}} = -IR_f =$$

$$= -R_f\left(\frac{a_1 V_{ref}}{R} + \frac{a_2 V_{ref}}{2R} + \frac{a_3 V_{ref}}{4R} + \cdots \frac{a_n V_{ref}}{2^{\text{n-1}} R}\right)$$

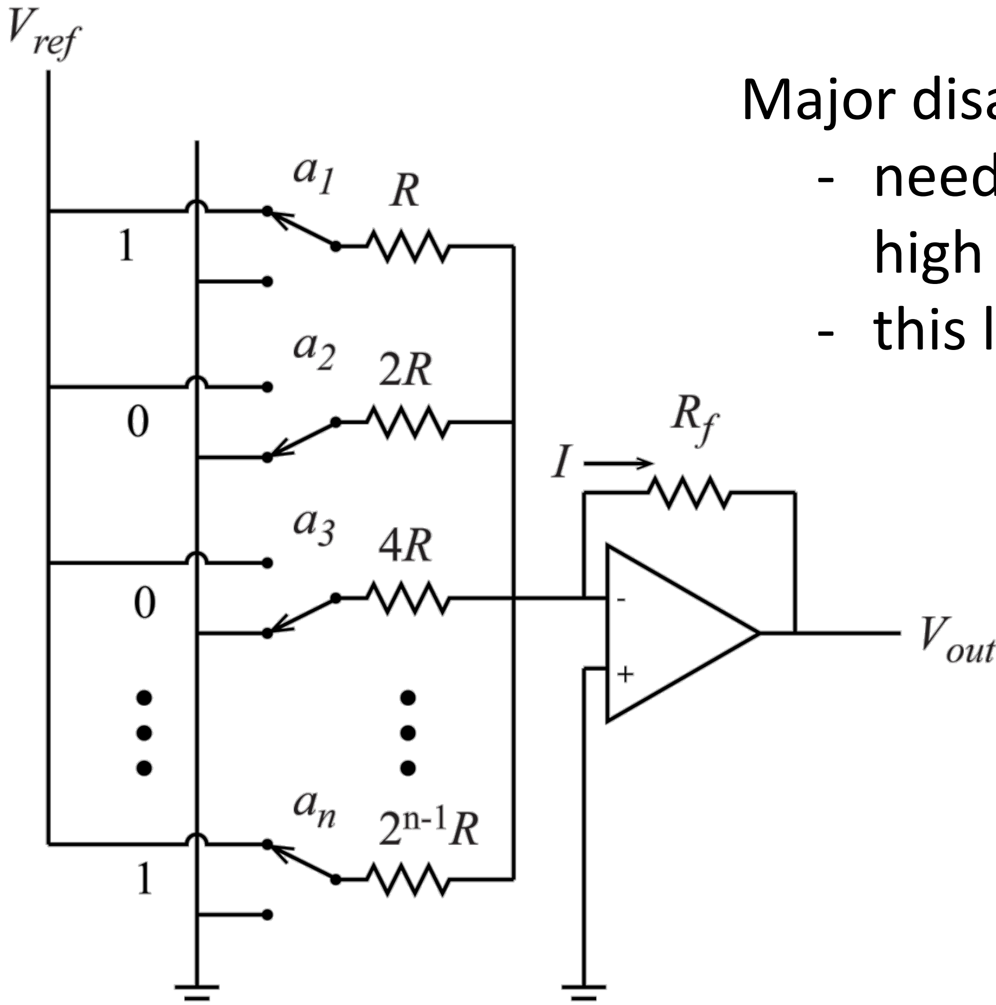
MSB LSB

for $R_f = R/2$:

$$V_{\text{out}} = -V_{ref}\left(\frac{a_1}{2} + \frac{a_2}{4} + \frac{a_3}{8} + \cdots \frac{a_n}{2^{\text{n}}}\right)$$

$$code(N_{dec}) : a_1 a_2 a_3 \ldots a_{n-1} a_n$$

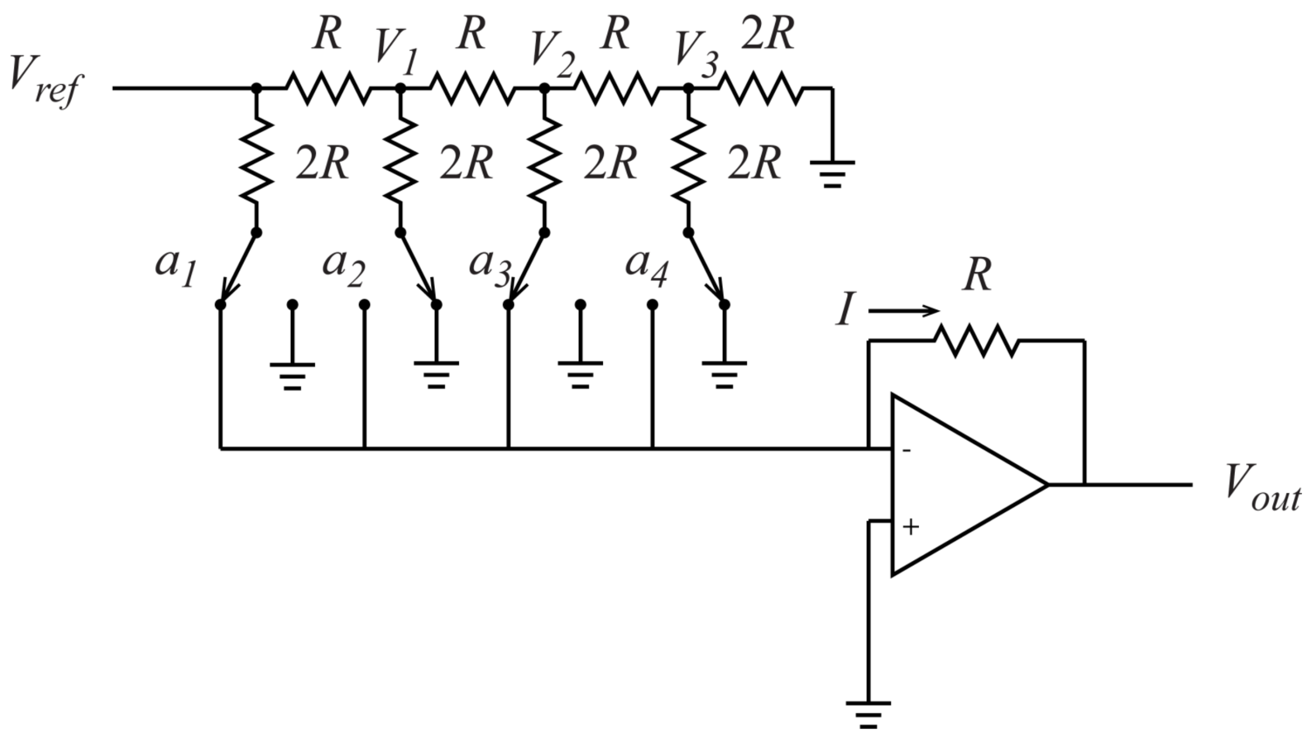
D/A converter: binary weighted ladder



Major disadvantage:

- needs a large range of resistors with high precision (2048:1 for 12-bit DAC)
- this limits it to 4-8 bit in practice

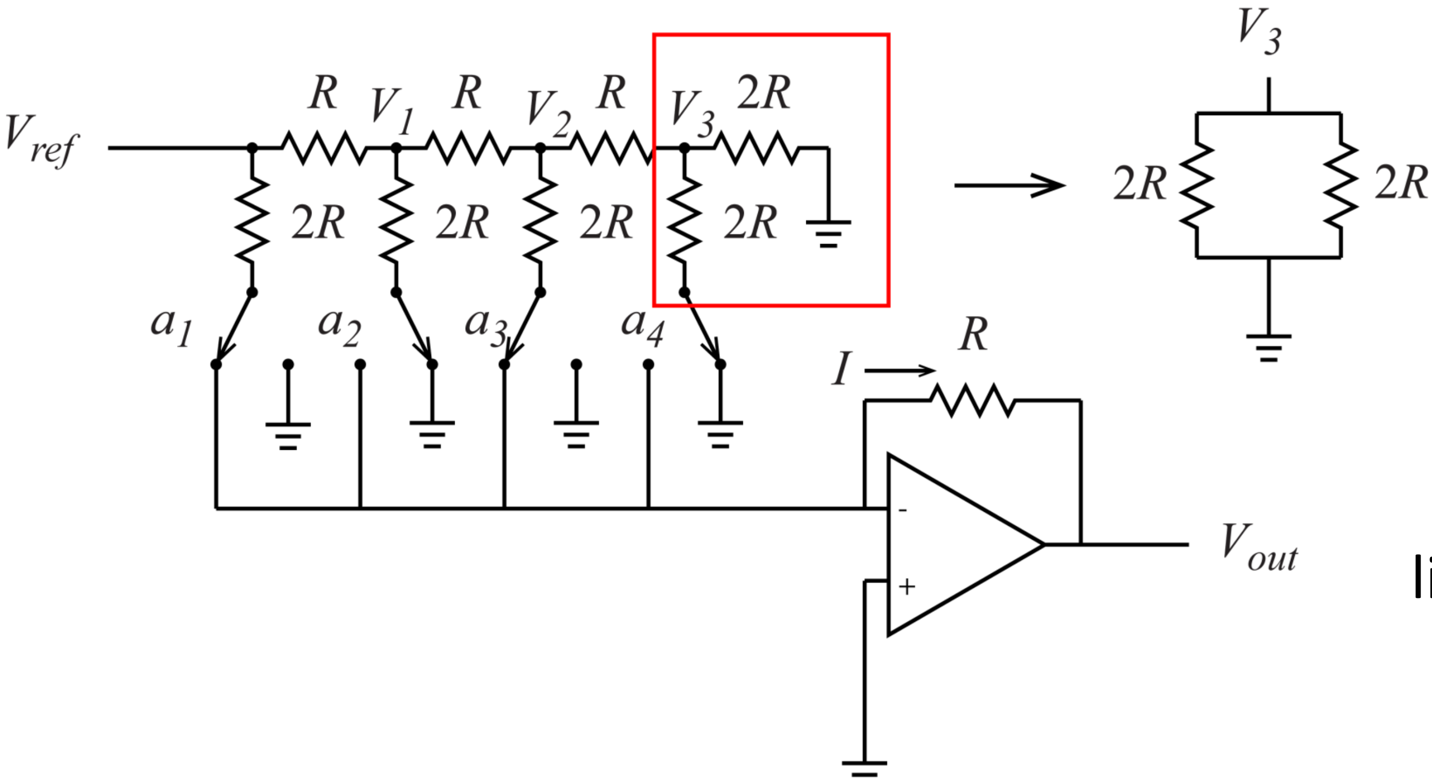
R-2R resistor ladder



-only two resistor values (R and 2R)

-does not require high precision resistors

R-2R resistor ladder



$$V_3 = \frac{1}{2} V_2$$

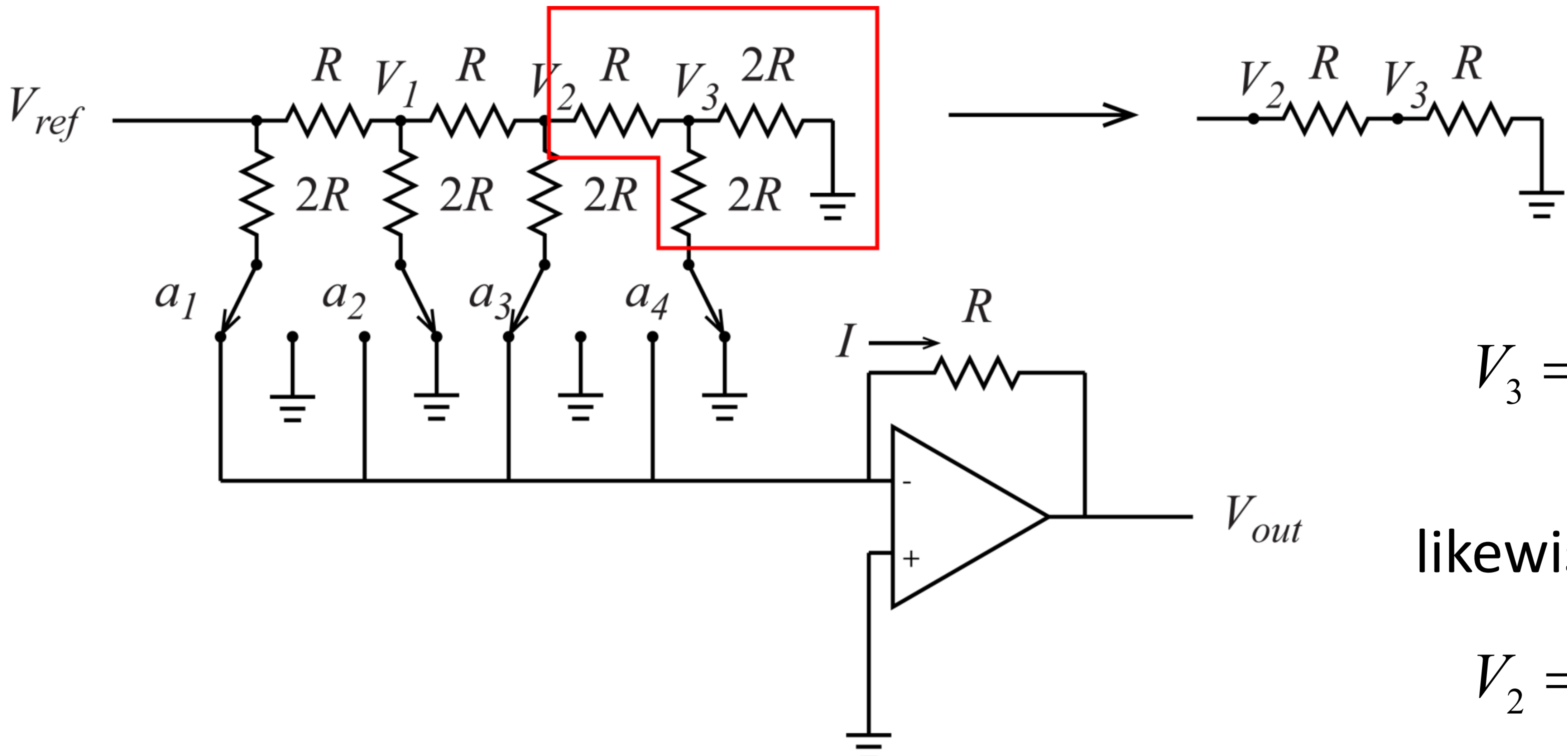
likewise:

$$V_2 = \frac{1}{2} V_1$$

$$V_1 = \frac{1}{2} V_{ref}$$

$$V_{out} = -IR$$

R-2R resistor ladder



$$V_3 = \frac{1}{2} V_2$$

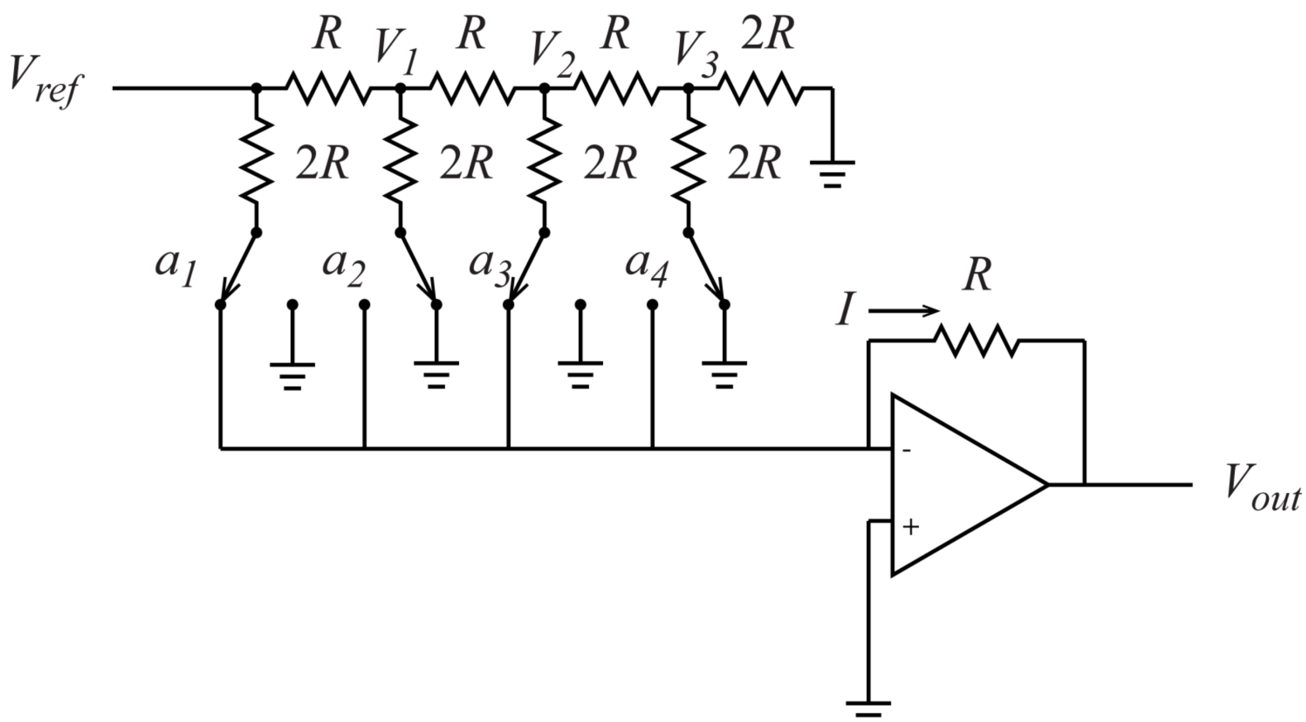
likewise:

$$V_2 = \frac{1}{2} V_1$$

$$V_1 = \frac{1}{2} V_{ref}$$

$$V_{out} = -IR$$

R-2R resistor ladder



likewise:

$$V_2 = \frac{1}{2} V_1$$

$$V_3 = \frac{1}{8} V_{ref}, V_2 = \frac{1}{4} V_{ref}, V_1 = \frac{1}{2} V_{ref}$$

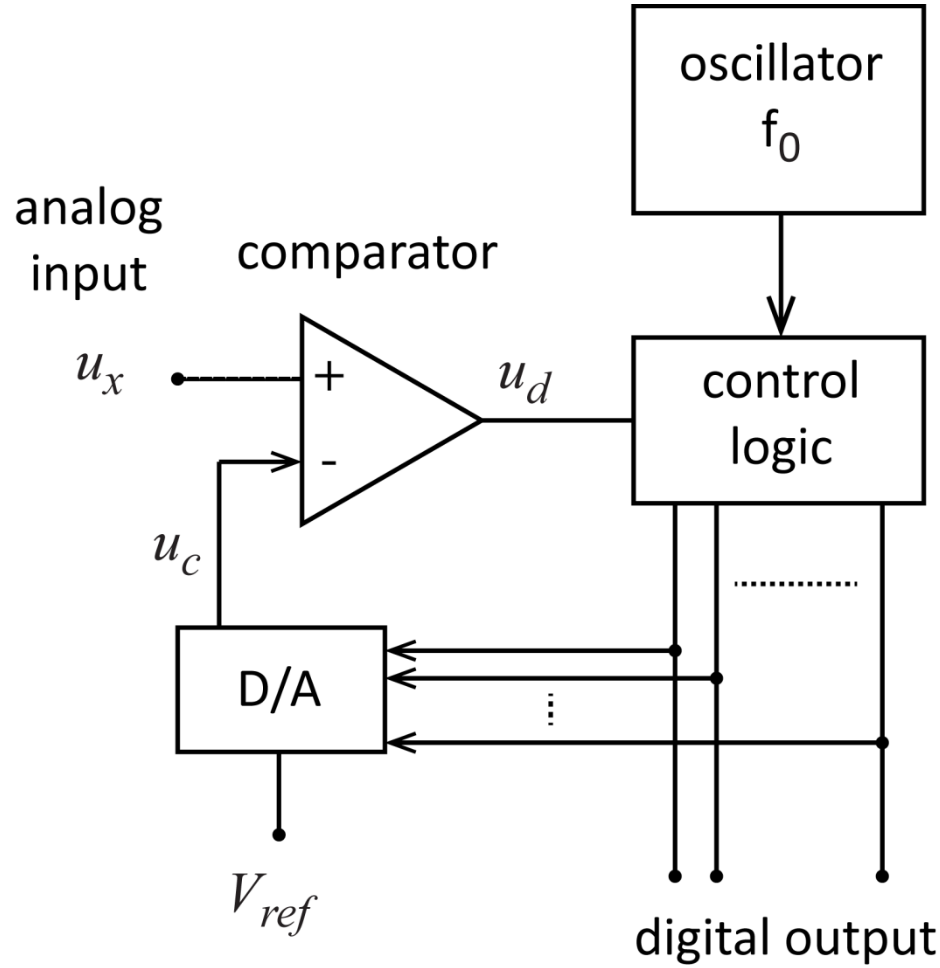
$$V_1 = \frac{1}{2} V_{ref}$$

$$V_{out} = -IR$$

$$V_{out} = -V_{ref} \left(\frac{a_1}{2} + \frac{a_2}{4} + \frac{a_3}{8} + \frac{a_4}{16} \right)$$

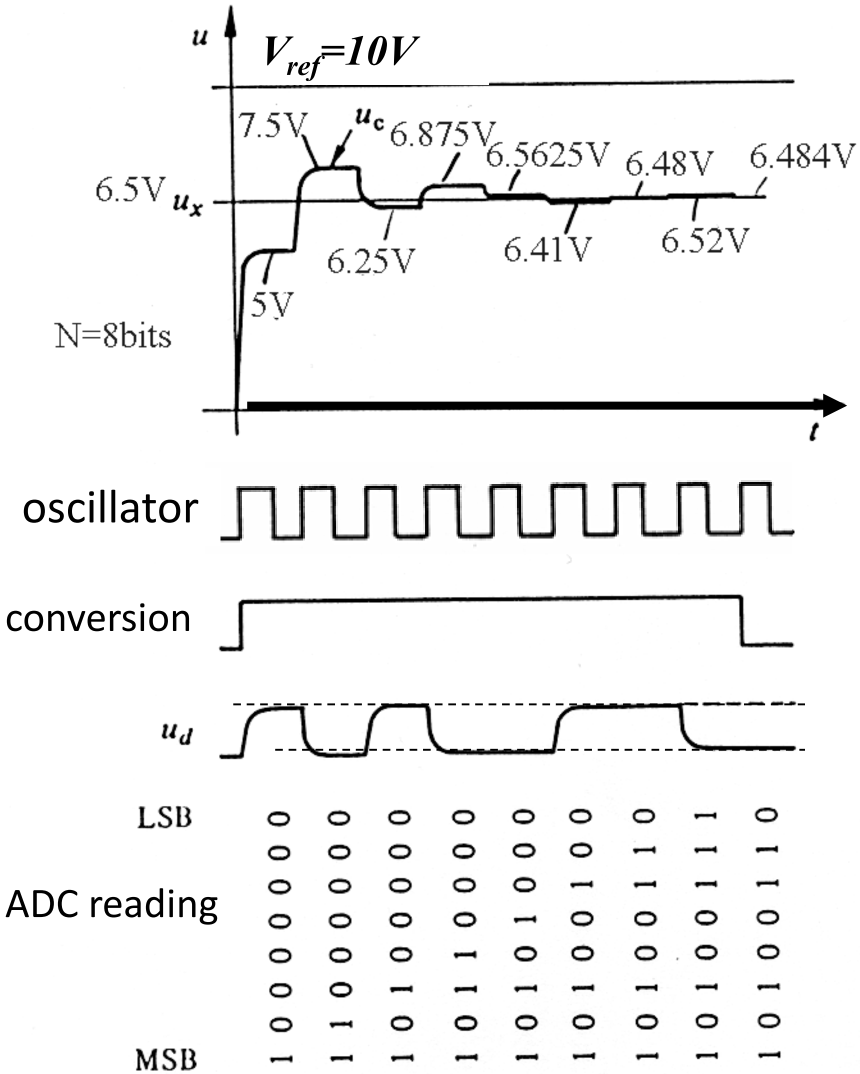
Successive approximation ADC

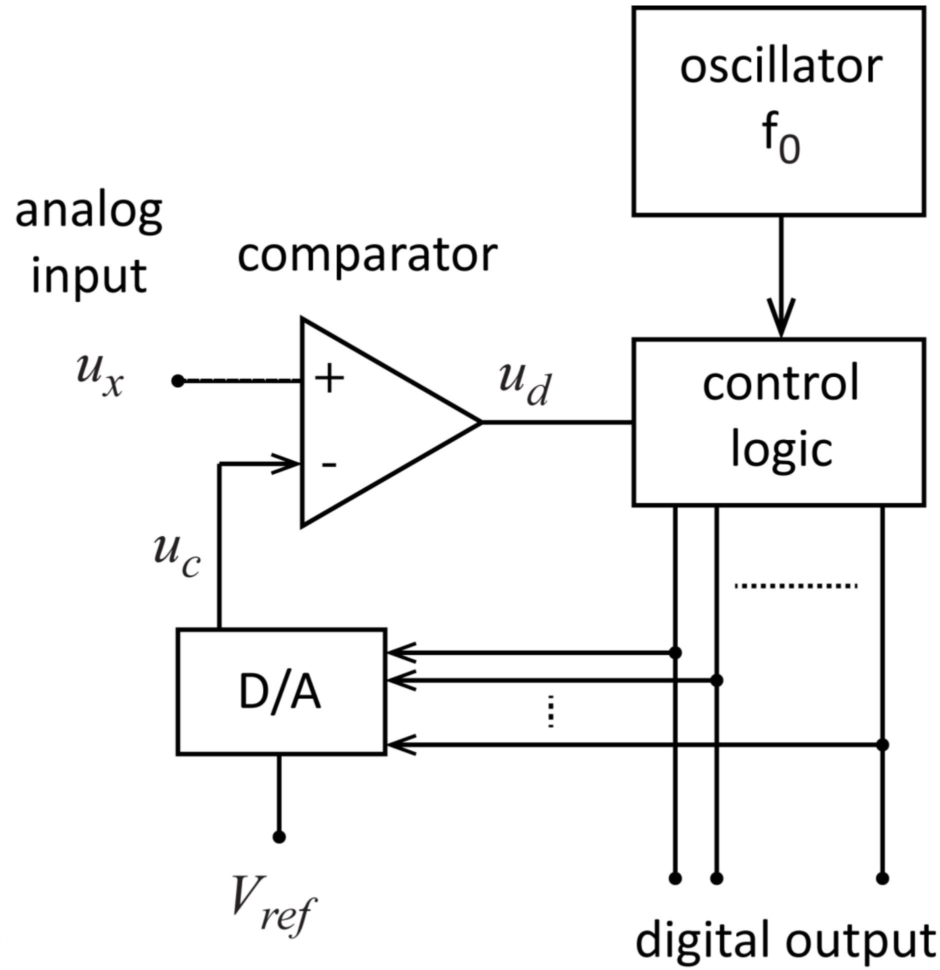
- Basic elements
  - digital to analog converter
  - analog comparator
  - control logic module
  - register



conversion time = n/f_0

Successive approximation ADC

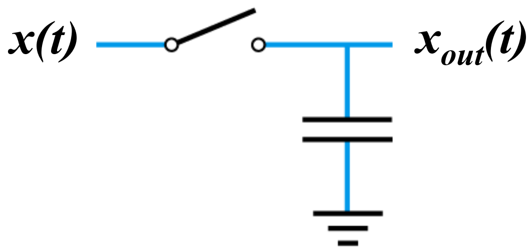




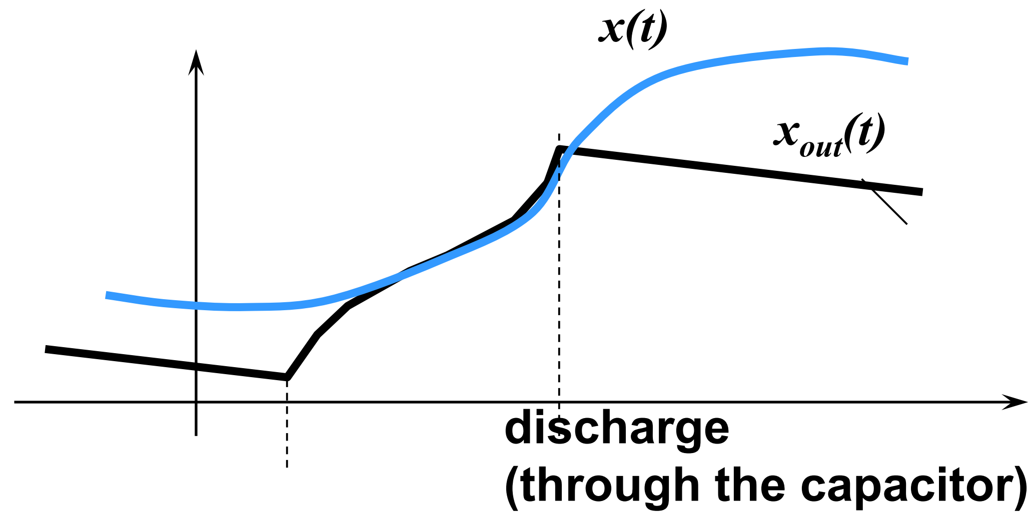
conversion time = n/f_0

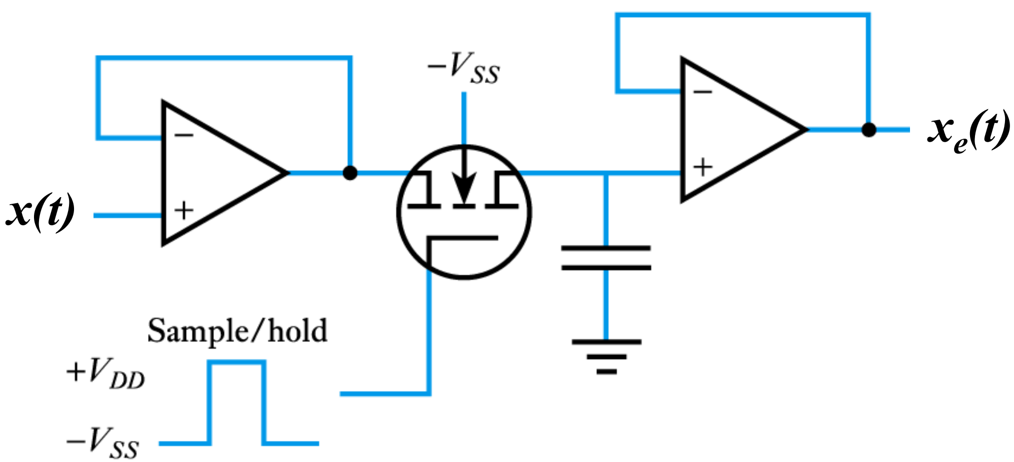
Sample and hold (SH) circuits

- used in the input stage of A/D converters
- captures the voltage of a varying analog signal and keeps it at a constant level during the sampling time



(a) Basic arrangement





(b) A typical circuit

Example

- We would like to convert a sinusoidal signal with the frequency *f* using a successive approximation converter with n bits and clock frequency f_o. Calculate a frequency above which we need to use a S/H circuit (n = 12, f_o = 1MHz)
  - *Conversion time* $t_c = n/f_o$
  - $u(t)=\hat{U}\cos(2\pi f t)$

Condition : change of u(t) during t_c ≤ less than the quantization error

a smaller change of signal would not change the outcome of digitization

$$\Delta u(t)_{\max} \leq \frac{1}{2}\frac{FS}{2^n}$$

$$\Delta u(t)_{\max} = \frac{du(t)}{dt}\Delta t = \frac{du(t)}{dt}t_c = 2\pi f \hat{U}_{\max} t_c = 2\pi f \frac{FS}{2} t_c \leq \frac{1}{2}\frac{FS}{2^n}$$

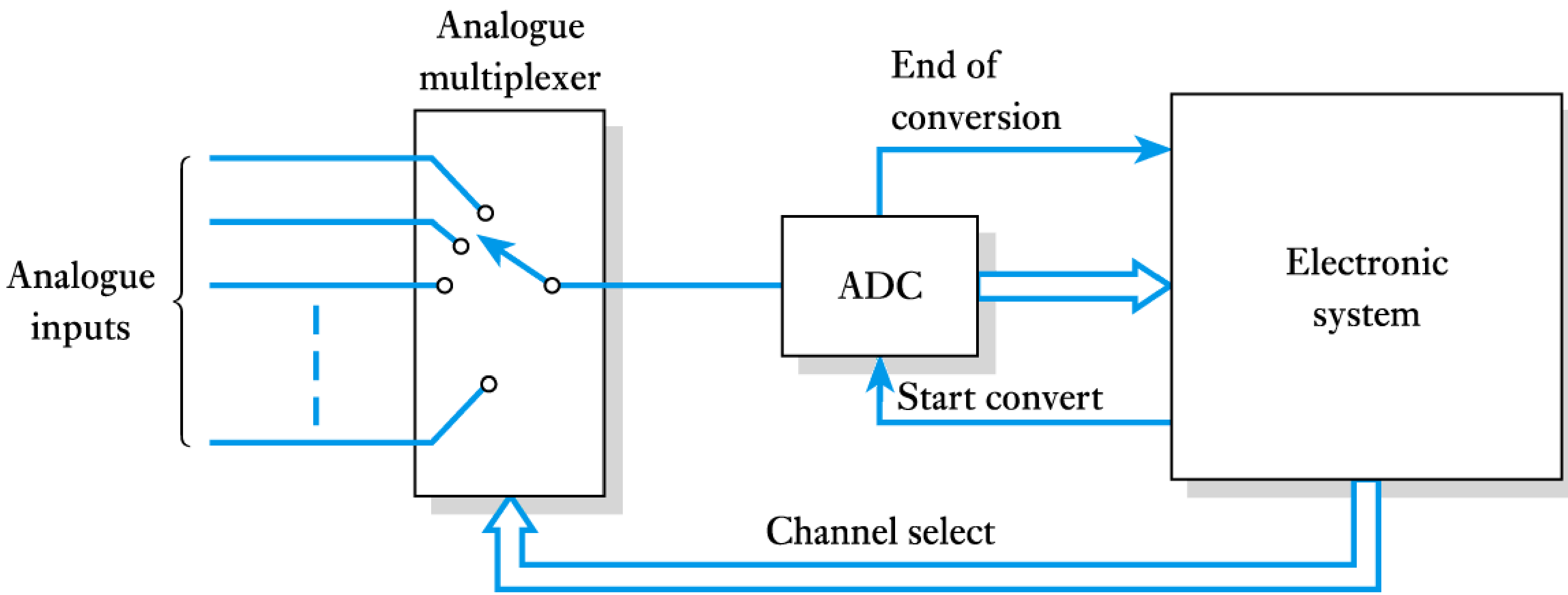
- Answer : t_c=12μs, f_{limit}=3.2 Hz

$$f \leq \frac{1}{2\pi 2^n t_c} = f_{\text{limit}}$$

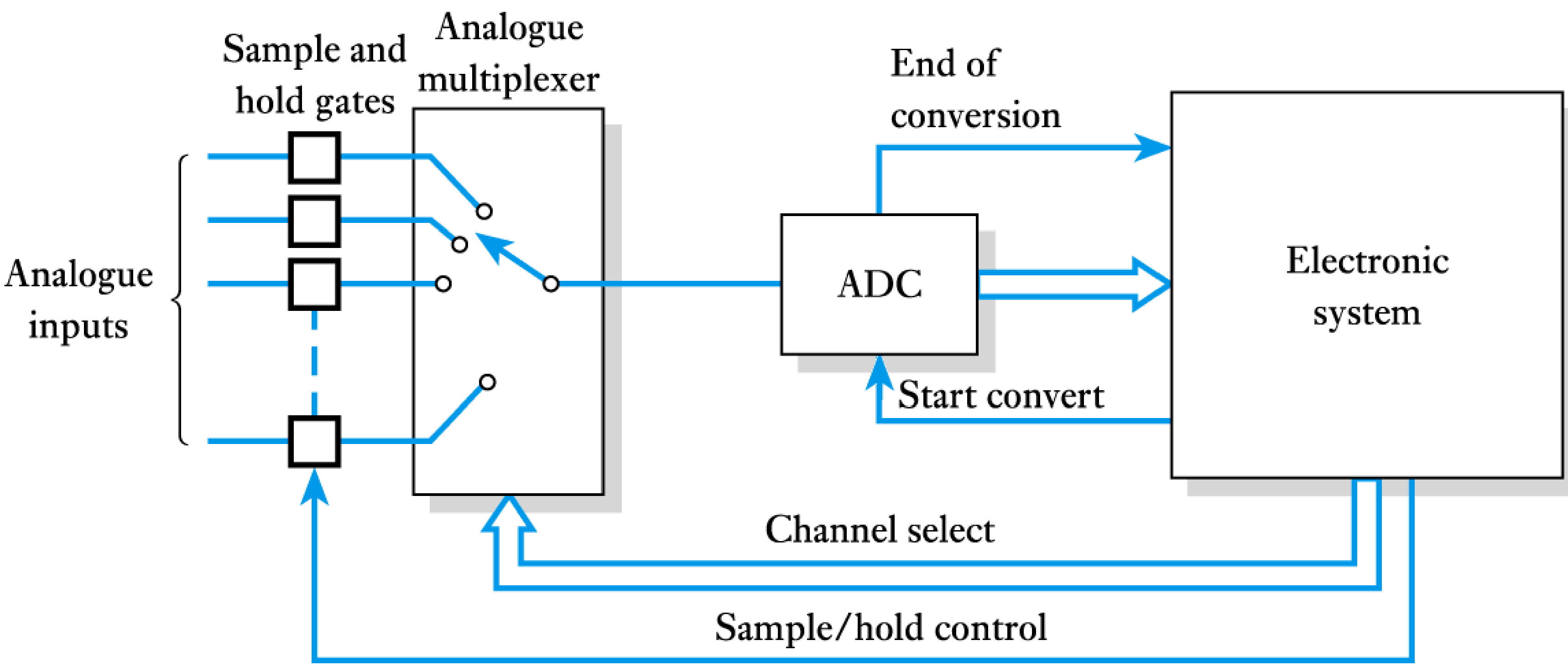
Multiplexing

- Measurement instruments often have multiple inputs and outputs
- Instead of putting an A/D or D/A converter for every input/output, we can use multiplexing:
  - use an electronic switch for selecting input/output
  - antialiasing and reconstruction filters for each input/output

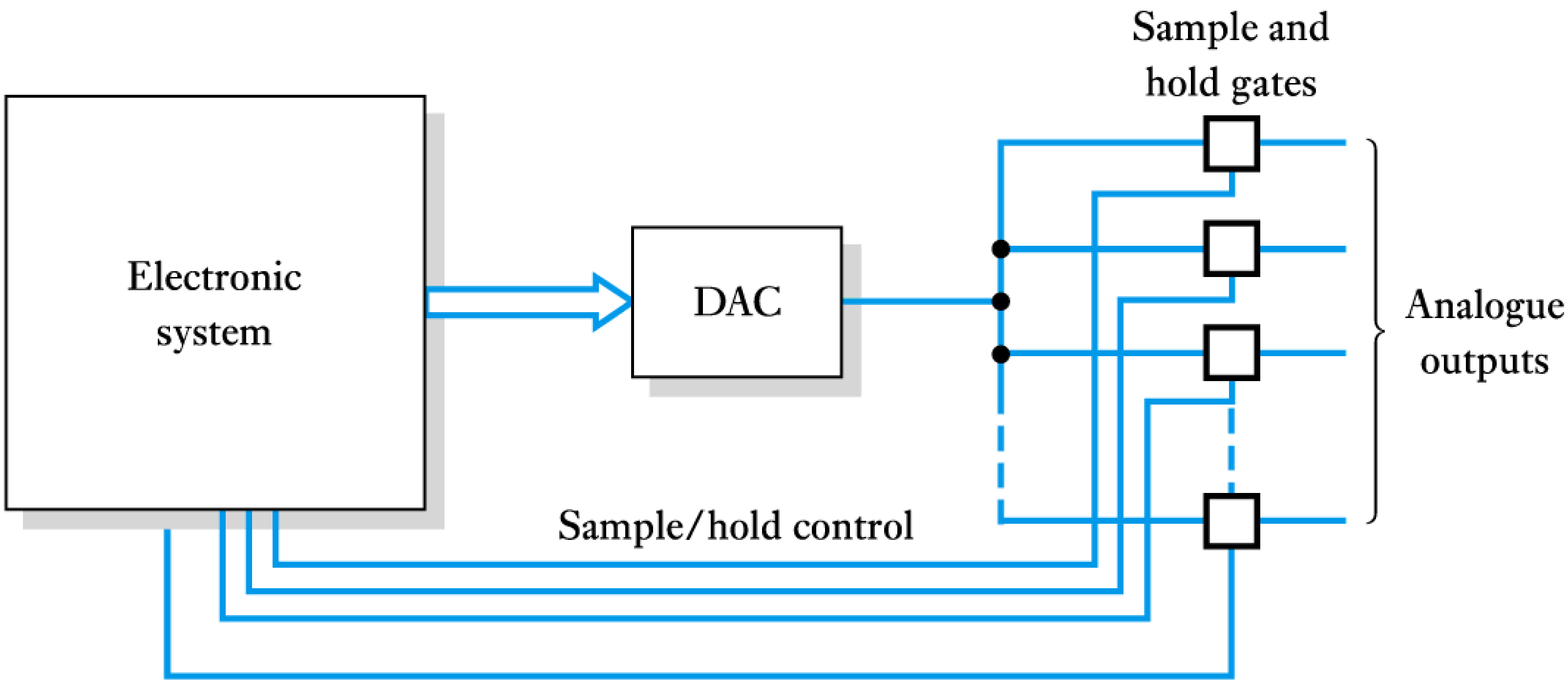
Input multiplexing



Input multiplexing with SH



Output multiplexing



Key points

- The conversion from analog to digital forms requires sampling
- Sampling frequency $f_s > 2f_{max}$
- In order to eliminate components with undesired frequencies, the signal can be filtered using a low-pass filter (antialiasing filter) with a cut-off frequency $f_c < f_s/2$
- Another low-pass filter allows us to reconstruct the signal by removing the high-frequency components due to sampling
- AD/DA converters
- SH circuits reduce conversion errors
- Multiplexing reduces the number of A/D and D/A converters and saves money