3.24pt

Chapter 2: Analog-to-Digital Conversion

(Digital signal processing)

Nguyen Linh Trung, Tran Duc Tan, Huynh Huu Tue
University of engineering and technology
Vietnam national university,Hanoi

31/8/2015

Content

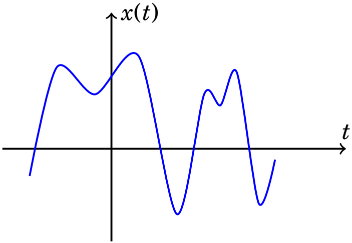
3.24pt

Overview

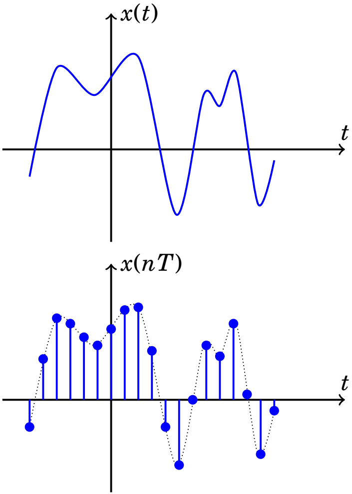
- As is well known, a physical quantity represented by a function that varies with time is known as a continuous signal.
- In order to process continuous signals by computers, the first thing to do is digitization, that is to present the continuous signal by a sequence of numbers which computers can understand and process.
- Digitization process has three steps in order, as below: sampling, quantization và coding.

- Sampling: gets values of the signal at discrete times. Therefore, sampling is also called discretization.
- Quantization: approximates values of the signal samples according to some quantum levels (discrete values). Quantization is determined by the accuracy of computers.
- Coding: represents a number by binary system so that computers can read. Therefore, this is the most important operation in digitalization.
- Three steps above are combined in an analog-to-digital converter (ADC).

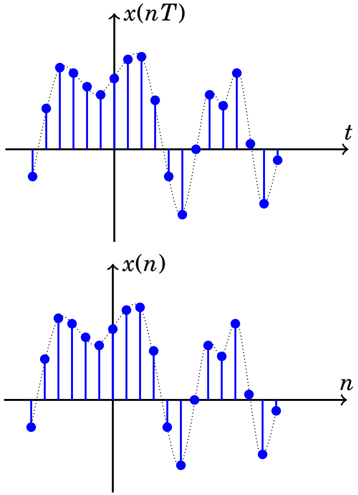
- Let us illustrate the digitization process. Consider an analog signal $x(t)$



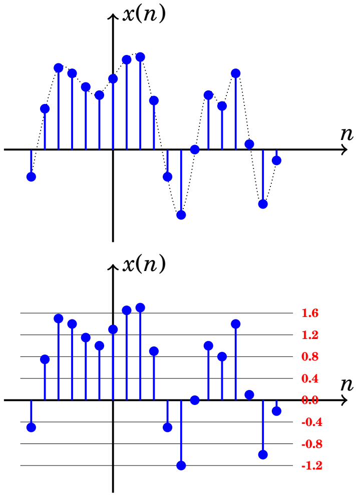
- Now, sample $x(t)$ at the equidistant points every T seconds to get the signal $x(nT)$; n is integer. T is called the sampling period. This method is called uniform sampling.


$x(t)$
t
$x(nT)$
t

- Next, instead of using $x(nT)$ with time nT, we can simply use $x(n)$ with sample n and call it a discrete-time signal.

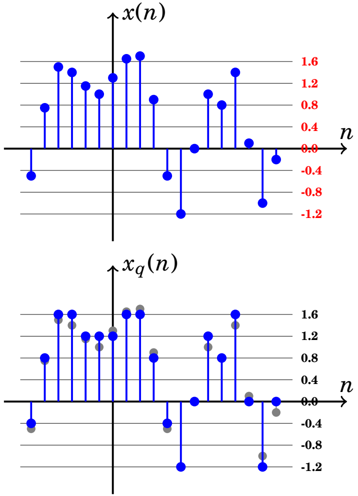


- Because samples of $x(n)$ are real-valued, we need to approximate them by finite numbers corresponding to different pre-defined quantum levels. For example, we choose eight levels $\{-1,2;-0,8;\ldots;1,6\}$,

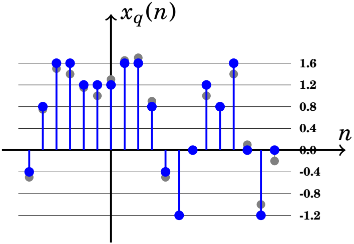


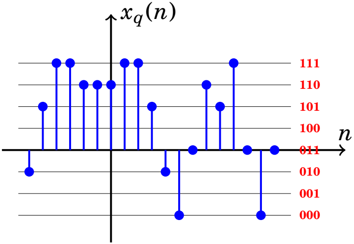
$x(n)$
n
$x(n)$
n
1.6
1.2
0.8
0.4
0.0
-0.4
-0.8
-1.2

- Then, we round $x(n)$ to get the quantized signal $\tilde{x}(n)$. This step produces errors, which is called quantization errors.

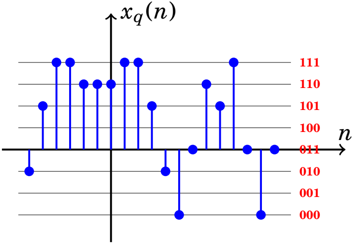


- With eight selected levels of quantization, we can use 3-bit binary numbers in order to represent the levels.





- Therefore, quantized signal $\tilde{x}(n)$ is represented in a binary sequence $\mathbf{x} = \{010101\cdots\}$.



$\mathbf{x} =$ 010 101 111 111 110 110 110 111 111 101 010 000 011 110 101 111 011 000 011

- To reduce the uantization error, we can use more quantum levels. However, that means we need more binary bits to represent the levels and thus the required memory storage increases.

Content

3.24pt

Sampling method

- There are many sampling methods, depending on the nature of the signal or the information that we need to process. Here we only consider the simplest method, but is the most fundamental, which is uniform sampling.
- Uniform sampling (for short, it is now referred to as sampling) a continuous signal $x(t)$ is to records a sequence of signal values at time points $t = nT$; n is a integer varies from $-\infty$ to $+\infty$. T is a constant whose unit is second (s), and is called the sampling period.
- Discretizing signal $x(t)$ to get the sequence $x(nT)$ is only valid/useful if later $x(t)$ can be perfectly reconstructed. The set of conditions to ensure perfect reconstruction is called the sampling conditions. To understand the sampling conditions, we now examine the spectrum of the sampled signal.

Spectrum of sampling signal

- Let $X(\Omega)$ is spectrum of $x(t)$, using the definition of $X(\Omega)$ with Fourier transform of $x(t)$ as below:

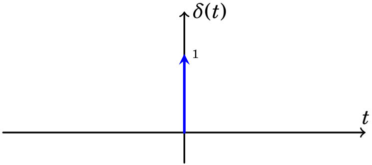
$$X\left(\Omega\right) = \int_{-\infty}^{\infty} x\left(t\right) e^{-j\Omega t} dt.$$

Note, in this definition, the unit of Ω (read as "big omega") is radian/second.

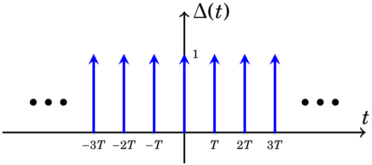
- Consider the following signal:

$$\Delta(t) = \sum_{n=-\infty}^{\infty} \delta(t - nT), \tag{1}$$

- $\delta(t)$ is Dirac pulse



- $\Delta(t)$, periodic with period T, contains a train of Dirac impulses at time points nT.



- Expand $\Delta(t)$ into Fourier series

$$\Delta(t) = \sum_{n=-\infty}^{\infty} C_n e^{jn\Omega_0 t}.$$

in which $\Omega_0 = 2\pi/T$ and

$$C_n = \frac{1}{T}\int_T \Delta(t)\, e^{-jn\Omega_0 t} dt.$$

The integral range of T takes on any interval of length T; we usually use the interval $[-T/2, T/2]$.

- We have

$$\begin{aligned}
C_n &= \frac{1}{T}\int_{-\frac{T}{2}}^{\frac{T}{2}} \Delta(t)\, e^{-jn\Omega_0 t} dt \\
&= \frac{1}{T}\int_{-\frac{T}{2}}^{\frac{T}{2}} \left[\sum_{k=-\infty}^{\infty} \delta(t-kT)\right] e^{-jn\Omega_0 t} dt \\
&= \frac{1}{T}\sum_{k=-\infty}^{\infty}\int_{-\frac{T}{2}}^{\frac{T}{2}} \delta(t-kT) e^{-jn\Omega_0 t} dt \\
&= \frac{1}{T}\int_{-\frac{T}{2}}^{\frac{T}{2}} \delta(t)\, e^{-j0\Omega_0 t} dt \\
&= \frac{1}{T}
\end{aligned}$$

- Hence

$$\Delta(t) = \frac{1}{T}\sum_{n=-\infty}^{\infty} e^{jn\Omega_0 t}. \tag{2}$$

- So, $\Delta(t)$ can be represented by two different ways in the time domain:

$$\Delta(t) = \sum_{n=-\infty}^{\infty} \delta(t - nT),$$

$$\Delta(t) = \frac{1}{T} \sum_{n=-\infty}^{\infty} e^{jn\Omega_0 t}.$$

- Next, using $\Delta(t)$ to sample the signal $x(t)$ to obtain the sampled signal

$$x_\Delta(t) = x(t)\Delta(t).$$

  This signal is considered as sampling $x(t)$ with period T by the Dirac pulses.

- We have two expressions for $x_\Delta(t)$ in the time domain

$$\begin{aligned} x_\Delta(t) &= \sum_{n=-\infty}^{\infty} x(t)\delta\left(t - nT\right) \\ &= \sum_{n=-\infty}^{\infty} x(nT)\delta\left(t - nT\right), \end{aligned} \tag{3}$$

$$x_\Delta(t) = \frac{1}{T}\sum_{n=-\infty}^{\infty} e^{jn\Omega_0 t} x(t). \tag{4}$$

- Taking the Fourier transform of $x_\Delta(t)$ to obtain two expressions of its spectrum $X_\Delta(\Omega)$:
- We then have two expressions for $X_\Delta(\Omega)$:

$$X_\Delta(\Omega) = \int_{-\infty}^{\infty} \left[\sum_{n=-\infty}^{\infty} x(nT)\delta(t-nT) \right] e^{-j\Omega t} dt$$

$$= \sum_{n=-\infty}^{\infty} x(nT) e^{-jn\Omega T} \tag{5}$$

$$X_\Delta(\Omega) = \int_{-\infty}^{\infty} \left[\frac{1}{T} \sum_{n=-\infty}^{\infty} e^{jn\Omega_0 t} x(t) \right] e^{-j\Omega t} dt$$

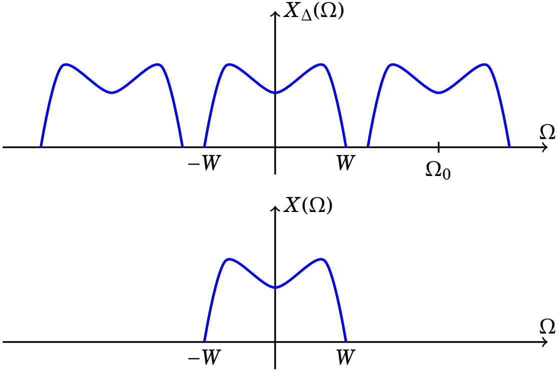
$$= \frac{1}{T} \sum_{n=-\infty}^{\infty} \int_{-\infty}^{\infty} x(t) e^{-j(\Omega - n\Omega_0)t} dt$$

$$= \frac{1}{T} \sum_{n=-\infty}^{\infty} X(\Omega - n\Omega_0) \tag{6}$$

- Two expression (5) và (6) show that the spectrum $X_{\Delta}(\Omega)$ can be represented either directly by the sampled signal $x(nT)$ or by the spectrum $X(\Omega)$ of the analog signal $x(t)$. And therefore we see the relation between the spectrum $X(\Omega)$ and the sampled signal $x(nT)$.

Signal reconstruction

- Equation (6) shows that from $X_\Delta(\Omega)$ we can derive $X(\Omega)$ with perfect accuracy if the components $X\left(\Omega+\Omega_0\right)$, $X\left(\Omega\right)$, and $X\left(\Omega-\Omega_0\right)$ are not overlapped in the frequency domain. This condition is called là non-aliasing.



- To satisfy the non-aliasing, there are two conditions:
  - Spectrum $X(\Omega)$ of original signal $x(t)$ must be band-limited; with bandwidth of W,
  - For $X\left(\Omega+\Omega_0\right)$, $X\left(\Omega\right)$, $X\left(\Omega-\Omega_0\right)$ not to have any overlap, we must have: $\Omega_0 > 2W$.
- Nyquist sampling theory: *A signal $x(t)$ and its sampled signal $x(nT)$ are completely equivalent if the spectrum of $x(t)$ has a finite bandwidth W and the sampling frequency must be larger than twice the signal bandwidth.*
- Hence, processing an analog signal or its equivalent discrete-time signal will yield the same result if the two conditions are met.

- If true, from spectrum $X_\Delta(\Omega)$, we only use a ideal lowpass filter $H_r(\Omega)$ to obtain $X(\Omega)$

$$X_\Delta(\Omega) \longrightarrow \boxed{H_r(\Omega)} \longrightarrow X(\Omega)$$

- $H_r(\Omega)$ is given by

$$H_r(\Omega) = \begin{cases} 1, & \text{if } |\Omega| \leq W_0 \\ 0, & \text{otherwise} \end{cases}$$

  in which W_0 is satisfy $W < W_0 < \Omega_0 - W$.
- Usually, we choose $W_0 = \Omega_0/2$.
- Frequency $B_0 = W_0/2\pi$ in unit of Hz and is called the Nyquist frequency.

- In the time domain, we have

$$x(t) = h_r(t) \star x_\Delta(t),$$

- $h_r(t)$ is the impulse response of the filter $H_r(\Omega)$ and is obtained as the inverse Fourier transform of $H_r(\Omega)$

$$h_r(t) = 2\pi B_0 \operatorname{sinc}(B_0 t),$$

where $\operatorname{sinc}(x) = \sin(\pi x)/\pi x$.

- Therefore,

$$x(t) = 2\pi B_0 \sum_{n=-\infty}^{\infty} x(nT) \operatorname{sinc} B_0 \left(t - nT\right). \tag{7}$$

- We see that, when all sampling conditions are satisfy, equation (7) ensure that $x(t)$ is perfectly reconstructed from its samples $x(nT)$.
- (7) is usually called interpolation formula of $x(t)$.

Practical sampling

- Above,

$$\Delta(t) = \sum_{n=-\infty}^{\infty} \delta(t - nT),$$

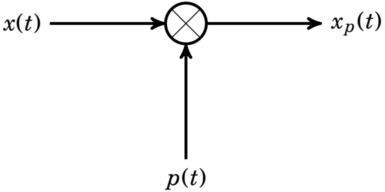
is used for sampling $x(t)$.
- The mathematical Dirac impulse $\delta(t)$ cannot be built in practice.
- In practice, we replace $\Delta(t)$ by

$$p(t) = \sum_{n=-\infty}^{\infty} w(t - nT), \quad (8)$$

in which T is sampling period and

$$w(t) = \begin{cases} 1, & \text{if } -\frac{T_0}{2} \leq t \leq \frac{T_0}{2} \\ 0, & \text{otherwise} \end{cases} \quad (9)$$

- Using signal $p(t)$ to sample $x(t)$ in the circuit as below:



Therefore

$$x_p(t) = x(t)p(t). \tag{10}$$

- Fourier series expansion of $p(t)$:

$$p(t) = \sum_{n=-\infty}^{\infty} C_n e^{jn\omega_0 t},$$

in which

$$C_n = \frac{1}{T}\int_{-\frac{T_0}{2}}^{\frac{T_0}{2}} e^{jn\omega_0 t} dt = \frac{T_0}{T}\,\mathrm{sinc}\left(n\frac{T_0}{T}\right).$$

- Hence, $x_p(t)$ is given by

$$x_p(t) = \sum_{n=-\infty}^{\infty} C_n e^{jn\omega_0 t} x(t). \tag{11}$$

- Take the Fourier transform of both sides of equation (11):

$$X_p(\Omega) = \sum_{n=-\infty}^{\infty} C_n X\left(\Omega - n\Omega_0\right). \tag{12}$$

- This result shows that spectrum $X(\Omega)$ can be derived from spectrum $X_p(\Omega)$ if two sampling conditions are satisfy.
- In practice, we choose $T_0 = T$ for simplicity; in this case, this method is called sample-and-hold.

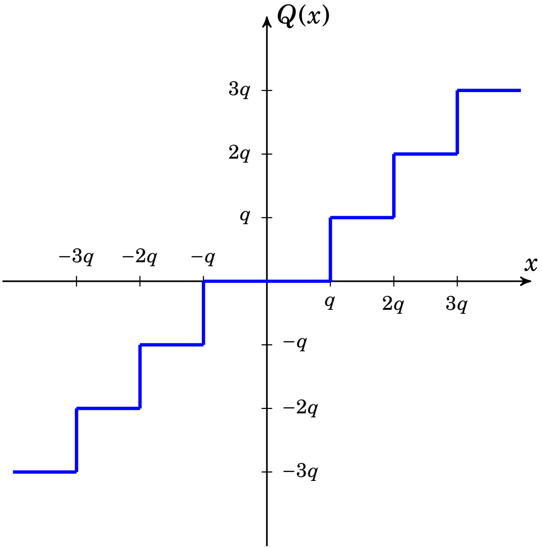
Content

3.24pt

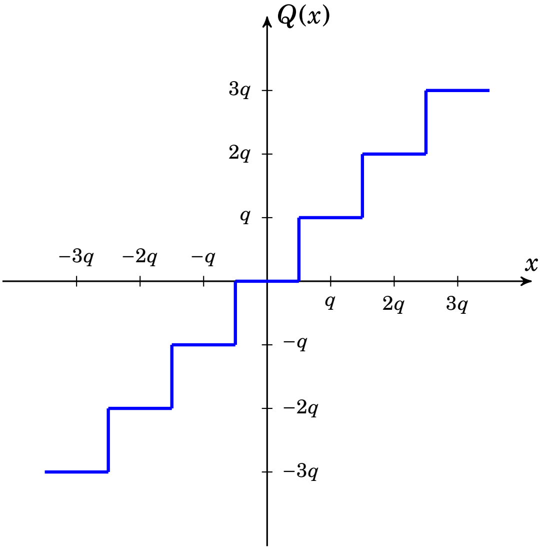
Quantization

- After sampling, the next step of digitization operation is quantization. At sampling time, each sample value can change continuously from the lowest to the highest level of signal.Therefore, the corresponding binary representation must need a new infinite length with absolute precision.
- In reality, the computer only certain accuracy, we must accept approximately the samples with some determined level by the accuracy of the computer. The thickness of each level is called quantum level and this device is call quantum device quantum device.
- The levels of the same quantum device may be different each other in order to operate for many different situations that always ensure the quality of the system. In the case of sampling uniform, the quantum device is called the uniform quantum.

- Tail cutting approximation



- Rounding approximation



- Quantization approximate a signal with errors

$$x(n) = x_q(n) + e_q(n). \tag{13}$$

  Error $e_q(n)$ has absolute value smaller than $q/2$, in which q is quantum level.
- We see that, quantization and sampling can exchange together without changing the results. Typically, the sampling is performed before quantization. However, if the sampling is performed after quantization,sampling velocity need higher than Nyquist sampling velocity of original signal. Because this case cause sample error signal, with this signal can has bandwidth greater than origin signal.Thus, if not careful, after sampled signals can be affected by spectrum folding phenomenon.

Content

Coding and binary presentation

- After quantization, we need presenting $x_q(n)$ such that computer can understand; computer use binary system to present a value. Thus, we only present $x_q(n)$ in binary system.
- Note: two presenting methods fixed-point and floating point, because those method are equivalent to the different level of accuracy.
- With fixed-point calculations, multiplication always lead to round errors while the addition does not generate errors.
- With floating-point calculations, all addition and multiplication creates this kind of error.
- Fixed-point calculations may give rise to over-floating phenomena.

- The over-floating phenomenon does not appear in floating point calculations.