EBU5375 Signals and systems: Sampling Theory

Dr Jesús Requena Carrión

Queen Mary University of London

Agenda

Quick review

Introduction to sampling

AD conversion

DA conversion

Agenda

Quick review

Introduction to sampling

AD conversion

DA conversion

Strange world

Look at the following videos:

Bombardier video (click on the text)
Aston Martin (click on the text) → Go to 02:55

The real movement of the Bombardier's propeller and Aston Martins'wheels are:

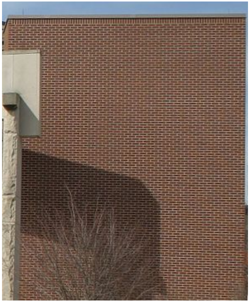
(a) Exactly as shown in both videos.

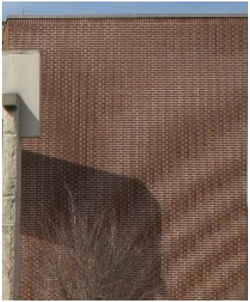
(b) Different, our visual limitations are deceiving us.

(c) Different, what we see is the result of filming technical limitations.

Strange world II

The following two digital photos have been taken almost simultaneously:





The ripples in the right-hand side photo are due to:

(a) Differences in illumnation.

(b) Differences in camera settings.

(c) Gravitational waves.

What are we learning this week?

1. Applications of Signals and Systems to IoT Engineering: Overview.
2. **Sampling Theory and Interpolation.**
3. Applications of Signal Theory to Telecommunication Systems.

Agenda

Quick review

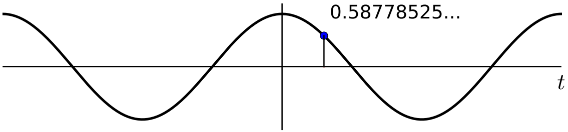
Introduction to sampling

AD conversion

DA conversion

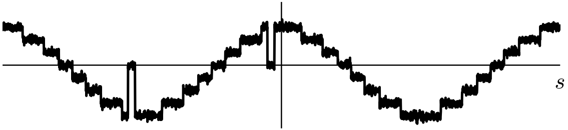
The storage problem: Storage media

How can we store a signal from the physical world? Storage media *copy* values that are contiguous in time into contiguous spatial locations.



Physical signals are **continuous both in time and in amplitude**, which means that in order for us to store them we need to store an **infinite number of values** described by an **infinite number of digits**!

The storage problem: Analog vs Digital storage media



- Analog approaches use physical media that are **continuous in space and in amplitude**. However, the quality of the stored signal is poor due to a limited precision and media imperfections.
- Digital approaches store physical signals as **binary sequences** (0's and 1's) and hence are **discrete in space and in amplitude**. But how? And how are binary sequences converted back into physical signals? What is the quality of the stored signal?

Format conversions

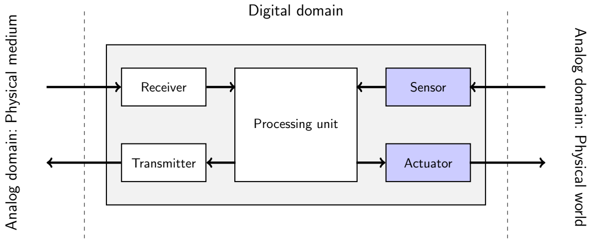
Nowadays, most of the systems that interface with the physical world (for instance, for storing, analysing and transmitting signals produced in the physical world) use digital approaches. This is made possible by two format-conversion techniques:

- **AD conversion** or **digitisation**, which converts signals in an analog format into signals in a digital format.
- **DA conversion**, which converts signals in a digital format into signals in an analog format

IoT devices and the physical world

IoT devices interact with the physical world through sensors and actuators.

- **Sensors** provide **inputs** from the physical world. Therefore, they must be associated to a **AD conversion** stage.
- **Actuators** provide **outputs** to the physical world. Therefore, they must be associated to a **DA conversion** stage.



Agenda

Quick review

Introduction to sampling

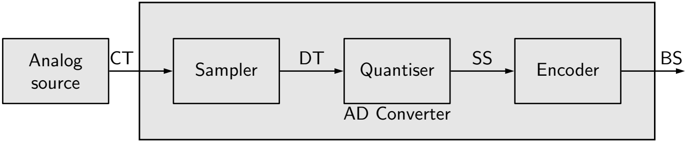
AD conversion

DA conversion

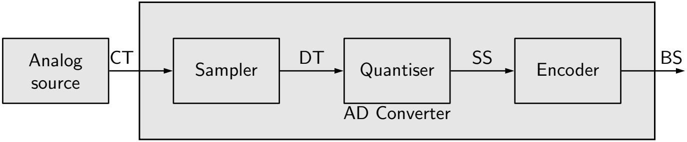
The digitisation pipeline

The process of digitisation or AD conversion converts a signal which is continuous in time and in amplitude into a binary sequence (discrete in time and in amplitude) and involves several steps:

- **Sampling** converts a CT signal into a DT signal.
- **Quantisation** converts each sample of a DT signal into one out of a finite number of values called symbols. The result is a symbol sequence (SS).
- **Encoding** converts a SS into a bit sequence (BS).



The digitisation pipeline

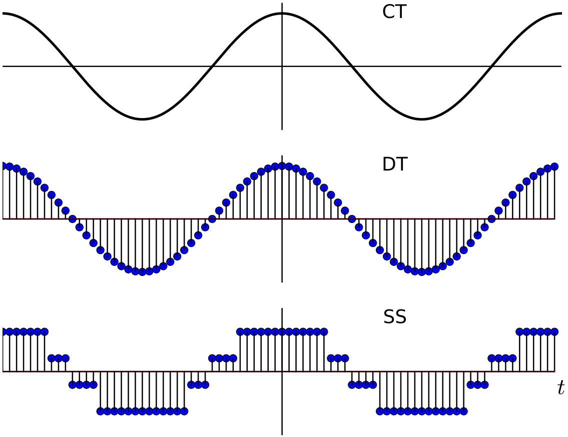


In this pipeline:

- Signals produced by the analog source are CT and their amplitude can take a continuous range of values.
- Sampled signals are DT and their amplitude can take a continuous range of values.
- Symbol sequences are DT an their amplitude can take a finite range of values.
- Binary sequences are DT an their amplitude can take either one of two values (0 and 1).

The digitisation pipeline: Sampling and quantisation

In this example, SS consists of only four values (**symbols**), namely -0.75, -0.25, 0.25 and 0.75.



The digitisation pipeline: Encoding

Encoding is the last step in the digitisation pipeline and consists of generating a binary sequence from a sequence of symbols. The easiest form of encoding translates each symbol into a binary sequence (binary code).

For instance, assigning the following binary codes to each symbol -0.75, -0.25, 0.25 and 0.75:

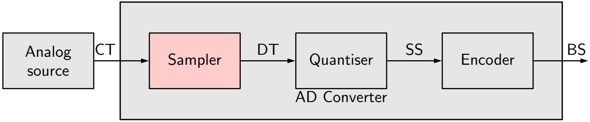
$$\begin{array}{rcl} -0.75 & \rightarrow & 00 \\ -0.25 & \rightarrow & 01 \\ 0.25 & \rightarrow & 10 \\ 0.75 & \rightarrow & 11 \end{array}$$

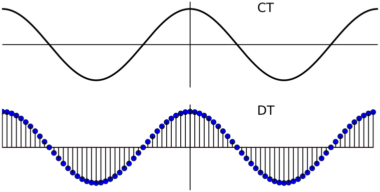
We produce the following conversion from SS to BS:

$$[-0.75, -0.75, 0.25, -0.25, 0.75, 0.25] \rightarrow 000010011110$$

Sampling: Principles

Sampling is the first state of the digitisation pipeline and creates a DT signal from from a CT signal.



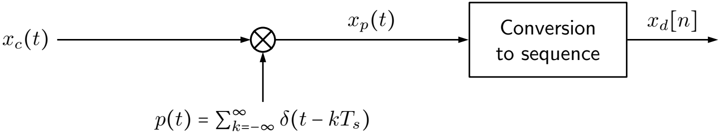


Sampling: The time domain

Mathematically, sampling can be described as a two step process:

1. Sample extraction: multiplication by a CT impulse train.
2. DT sequence generation: conversion to a DT impulse train.

Schematically:

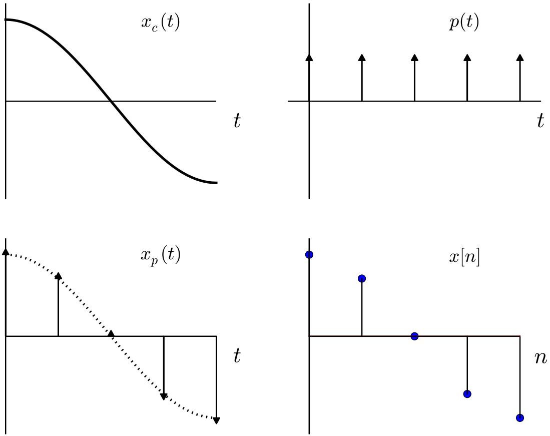


where T_s is known as the **sampling period**, $\omega_s = 2\pi f_s = 2\pi/T_s$ is the **sampling frequency** and

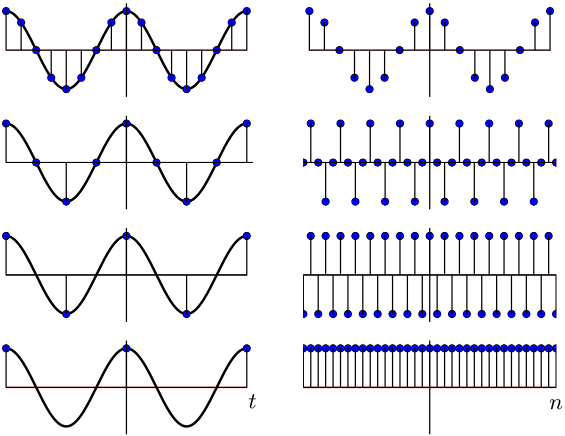
$$x_p(t) = x_c(t)p(t) = x_c(t) \sum_{k=-\infty}^{\infty} \delta(t - kT_S) = \sum_{k=-\infty}^{\infty} x_c(kT_s)\delta(t - kT_s)$$

$$x_d[n] = \sum_{k=-\infty}^{\infty} x_c(kT_s)\delta[n-k] \quad \longrightarrow \quad x_d[n] = x_c(nT_s)$$

Sampling: The time domain



Sampling: Sinusoids



Sampling: The frequency domain

We will now relate the frequency domain descriptions of $x_d[n]$, $x_\delta(t)$ and $x_c(t)$. We know that

$$x_p(t) = x_c(t)p(t)$$

and hence

$$X_p(\omega) = \frac{1}{2\pi} X_c(\omega) \star P(\omega)$$

where $X_c(\omega)$ and $P(\omega)$ are the Fourier transforms of $x_c(t)$ and $p(t)$. We also know that

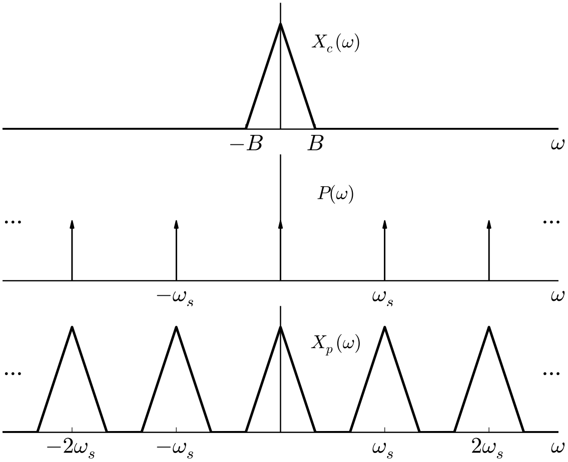
$$P(\omega) = \frac{2\pi}{T_s} \sum \delta\left(\omega - k\omega_s\right)$$

Hence

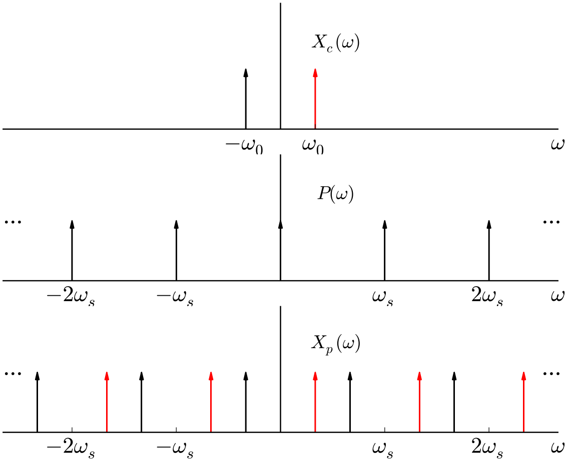
$$X_p(\omega) = \frac{1}{2\pi} X_c(\omega) \star \frac{2\pi}{T_s} \sum \delta\left(\omega - k\omega_s\right) = \frac{1}{T_s} \sum X_c\left(\omega - k\omega_s\right)$$

Hence, $X_p(\omega)$ is **periodic** and consists of **frequency-shifted** replicas of $X(\omega)$.

Sampling: The frequency domain



Sampling: The frequency domain



Sampling: The frequency domain

Signal $x_p(t)$ can be expressed in terms of $x[n]$ as follows:

$$x_p(t) = \sum_{n=-\infty}^{\infty} x[n]\delta(t - nT_s)$$

Hence, its Fourier transform $X_p(\omega)$ is

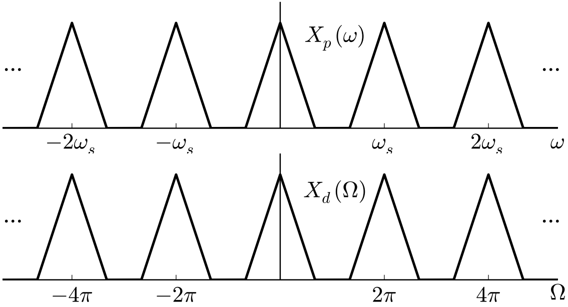
$$\begin{aligned} X_p(\omega) &= \int_{-\infty}^{\infty} x_p(t)e^{-j\omega t}dt \\ &= \int_{-\infty}^{\infty} \sum_{n=-\infty}^{\infty} x[n]\delta(t - nT_s)e^{-j\omega t}dt \\ &= \sum_{n=-\infty}^{\infty} x[n] \int_{-\infty}^{\infty} \delta(t - nT_s)e^{-j\omega t}dt \\ &= \sum_{n=-\infty}^{\infty} x[n]e^{-j\omega nT_s} = X_d(\omega T_s) \end{aligned}$$

where $X_d(\Omega)$ is the Fourier transform of $x[n]$.

Sampling: The frequency domain

In summary, $X_p(\omega) = X_d(\omega T_s)$ and $X_d(\Omega) = X_p(\frac{\Omega}{T_s})$. Specifically,

$$X_p(\omega_s) = X_p\left(\frac{2\pi}{T_s}\right) = X_d\left(\frac{2\pi}{T_s}T_s\right) = X_d(2\pi)$$



Sampling: The frequency domain

In summary, if $x_d[n]$ is the DT signal obtained by sampling a CT signal $x_c(t)$ with a sampling rate $f_s = 1/T_s$, its Fourier transform $X_d(\Omega)$ can be expressed as

$$X_d(\Omega) = \frac{1}{T_s} \sum_{k=-\infty}^{\infty} X_c\left(\frac{\Omega}{T_s} - k\frac{2\pi}{T_s}\right)$$

where $X_c(\omega)$ is the Fourier transform of $x_c(t)$.

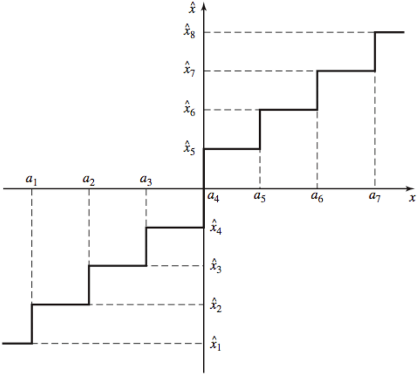
Quantisation

By sampling an analog signal we obtain a **sequence of continuous values**. For representing one of such values we might need an **infinite sequence of digits**, for instance, $3.14159265359\ldots$

If we want to obtain a digital representation of an analog waveform, after sampling the original waveform we need to **quantise the resulting amplitudes**. Quantisation consist of mapping continuous values into discrete ones that can be represented digitally, i.e. that can be represented as symbols. This leads to errors known as **quantisation errors**.

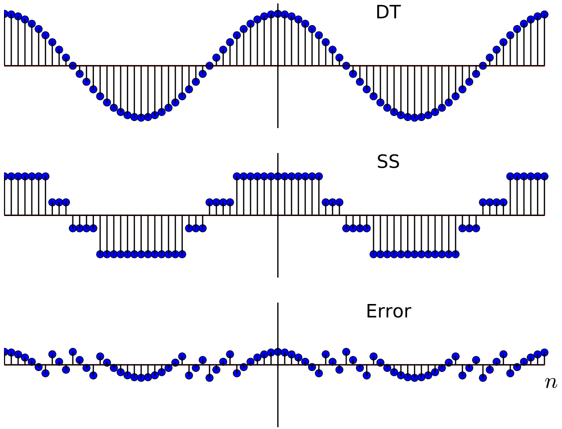
Uniform quantisation

In uniform quantisation the amplitude range is divided into equally-sized intervals known as **quantisation levels**. If the size of a quantisation level is q, the error for each sample is no larger than $\pm q/2$.



Quantisation error

The quantisation error is the difference of the DT signal and the SS signal. The smaller the quantisation interval q, the smaller the quantisation error.



AD conversion: Final considerations

A signal $x(t)$ with a duration of 2 s is sampled at 1 KHz. Each sample is quantised by a uniform quantiser with 256 levels. How can we determine the size of the memory that we need to store the digital version of $x(t)$?

By increasing the sampling rate and the number of quantisation levels the quality of the digitised versions of analog signals increases. However, it increases at the expense of increasing the storage needs. How can we determine the minimum requirements for obtaining digital versions of analog signals with sufficient quality?

Agenda

Quick review

Introduction to sampling

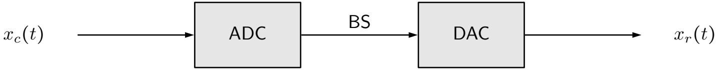
AD conversion

DA conversion

Principles of DA conversion

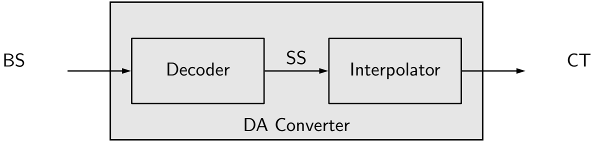
The process of DA conversion converts a binary sequence (discrete in time and in amplitude) into a signal which is continuous in time and in amplitude.

If we assume that the binary sequence is the result of digitising a signal $x_c(t)$ then our aim is for the result of the process of DA conversion, $x_r(t)$, to be as close to $x_c(t)$ as possible

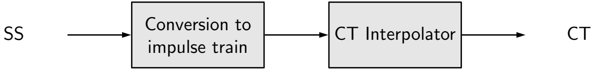


Principles of DA conversion

The process of DA conversion converts a binary sequence (discrete in time and in amplitude) into a signal which is continuous in time and in amplitude.

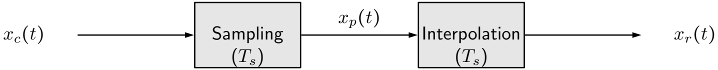


The interpolator first converts the symbol sequence into a CT impulse train and then proceeds to interpolate between the impulses.



Principles of DA conversion

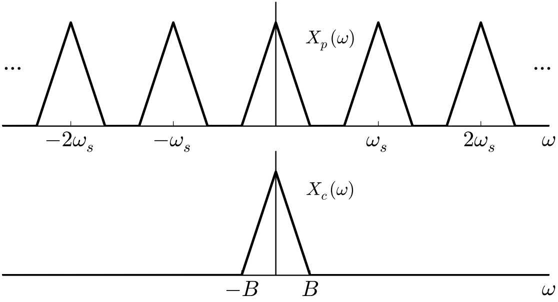
In our analysis of DA conversion, we will not consider the effects of quantisation, only sampling. The **ADC will be modelled as a sampler** and the **DAC as an interpolator**.



We will ask ourselves the following question: how can we design the sampling and interpolation stages so that $x_r(t) = x_c(t)$?

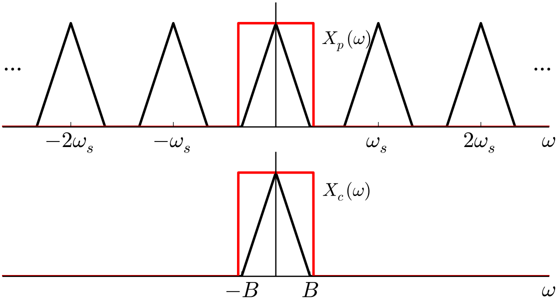
Interpolation: The frequency domain

Given $x_p(t)$, how can we obtain $x_c(t)$? (*Hint: Look at the frequency domain, i.e. at $X_p(\omega)$ and $X_c(\omega)$*)



Interpolation: The frequency domain

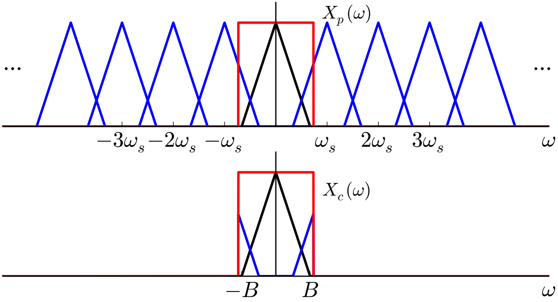
By looking at the frequency domain, we can conclude that by low-pass filtering the sampled signal $x_p(t)$ we could obtain $x_c(t)$.



Since $X_p(\omega) = (1/T_s) \sum X_c(\omega - k\omega_s)$, the interpolator will be a lowpass filter with frequency response $H_I(\omega) = T_s$, for $|\omega| \leq \omega_s/2$, and $H_I(\omega) = 0$ elsewhere.

Interpolation: Undersampling and aliasing

If we undersample a signal (ω_s is too low), there is **overlap** between the replicas of $X(\omega)$ in $X_p(\omega)$. This is called **aliasing**.



If there is aliasing, $X_c(\omega)$ will contain both $X(\omega)$ and part of the replicas $X(\omega + \omega_s)$ and $X(\omega - \omega_s)$. In order for us to avoid it, we need to make sure that the sampling frequency is high enough, specifically $\omega_s > 2B$.

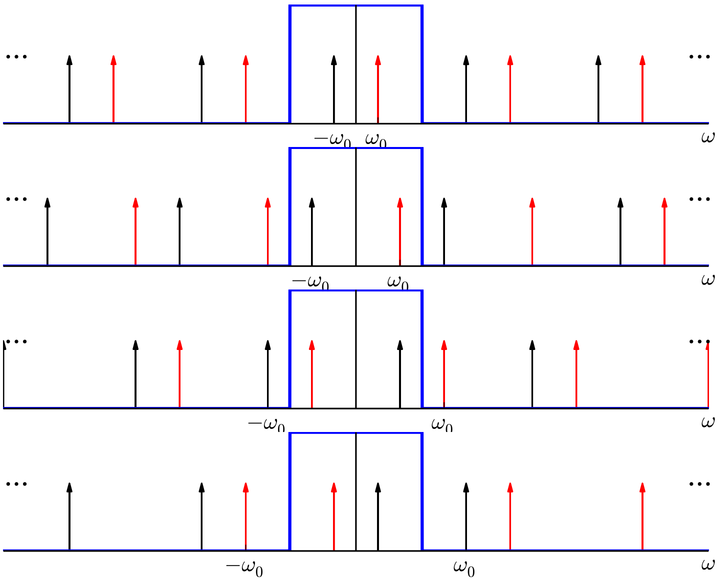
The Sampling Theorem

Let $x(t)$ be a band-limited signal with $X(\omega) = 0$ for $|\omega > B$. Then $x(t)$ is uniquely determined by its samples $x(nT_s), \quad n = 0, \pm 1, \pm 2, \pm 3, \ldots$ if $\omega_s > 2B$, where $\omega_s = 2\pi/T_s$.

Given these samples, we can reconstruct $x(t)$ by generating an impulse train in which successive impulses are separated by T_s units of time and have the amplitudes of successive samples. If this impulse train is processing it through a lowpass filter with gain T_s and cutoff frequency ω_c such that $B < \omega_c < \omega_s - B$, the resulting signal is **exactly** $x(t)$.

This theorem is also called Nyqvist Theorem or Nyquist-Shannon Theorem and $2B$ is known as Nyqvist frequency.

Interpolation: Undersampling and aliasing



Interpolation: The time domain

In the time domain, an ideal lowpass filter is characterised by a sinc impulse response. Let $H(\omega) = T_s$ for $|\omega| < B$ and $H(\omega) = 0$ for $|\omega| > B$. Then,

$$h(t) = \frac{T_s \sin(\pi t/T_s)}{\pi t}$$

and $x_c(t)$ can be obtained in the time domain from $x_p(t)$ as follows:

$$\begin{aligned}
x_c(t) &= x_p(t) \star h(t) \\
&= \left(\sum_{n=-\infty}^{\infty} x(nT_s)\delta(t - nT_s) \right) \star h(t) \\
&= \sum_{n=-\infty}^{\infty} x(nT_s)h(t - nT_s) \\
&= \sum_{n=-\infty}^{\infty} x_d[nT_s] \frac{T_s \sin\left(\pi(t/T_s - n)\right)}{\pi(t - nT_s)}
\end{aligned}$$

Interpolation: The time domain