Introduction to Digital Signal Processing

Digital Signal Processing

Notes

Contents

Digital Signal Processing
What is Digital Signal Processing?
Phase
Phasors and Complex Numbers

A Digial Signal Processing System

Notes

What is DSP?

Techniques include (*e.g.*)

- Filtering
- Frequency domain techniques (*i.e.* Fourier)
- Time domain techniques
- Random signals
- Prediction and Estimation (*e.g.* time series estimation)

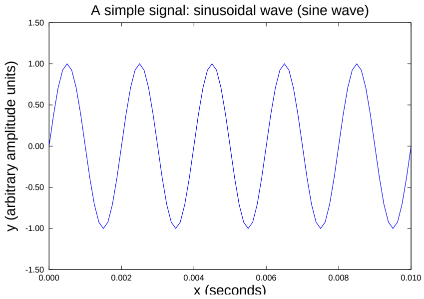
Example Applications

- Audio processing
- Communication systems
- Image processing
- Video processing
- Data compression
- Vehicle control
- Financial engineering

Notes

What is a Signal?

A simple example.

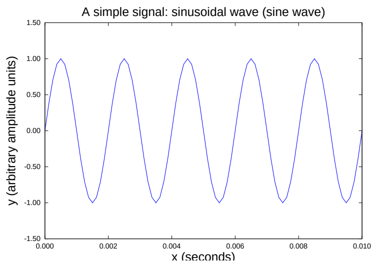


Notes

What is a Signal?

Can contain information for

- *Communication*
- *Storage*
- *Calculation*



Notes

Example of *What is a Signal?*

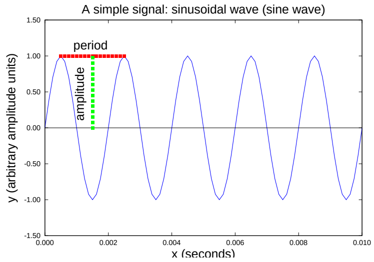
Information is carried in the

- *amplitude*, "a";
- *period*, "T";
- *frequency*, "$f = 1/T$";
- and *phase*, "ϕ".

Equation for a sine wave:

$$y(x) = a \sin(2\pi f x + \phi)$$

where "x" is time in seconds for this example. Amplitude "$a = 1$" controls the height of the wave.



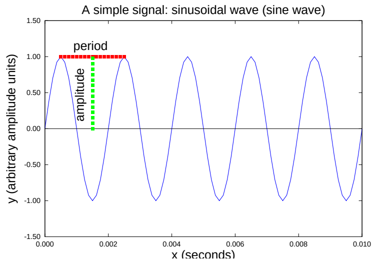
Notes

Frequency and Period

Equation for a sine wave:

$$y(x) = a\sin(2\pi f x + \phi)$$

- f is the frequency
- Measured in Hertz or Hz
- Here period, $T = 0.002$s
- $f = 1/T$ Hz, therefore $f = 1/0.002 = 500$Hz.



Notes

Phase

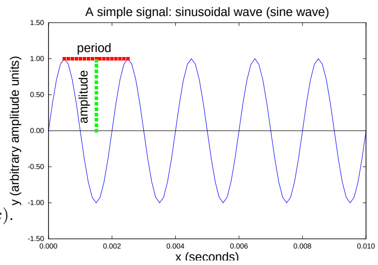
Equation for a sine wave:

$$y(x) = a\sin(2\pi f x + \phi)$$

- ϕ is the phase
- Here $\phi = 0$

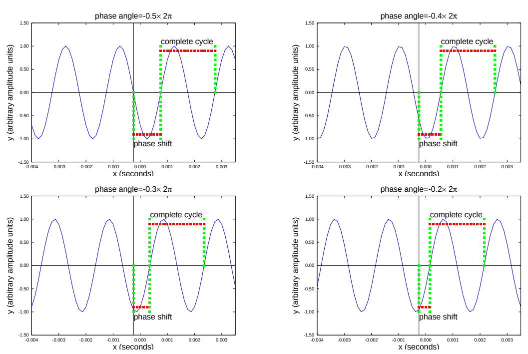
Therefore here,

$$y(x) = y(x, \phi = 0) = a\sin(2\pi f x).$$



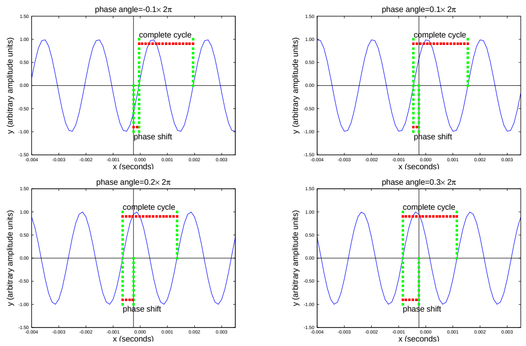
Notes

Phase *examples*



Notes

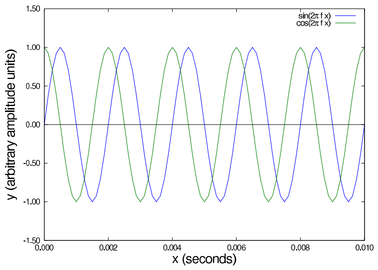
Phase *examples cont'd*



Notes

Cosine Vs Sine

Cosine and Sine functions are equivalent except for a phase shift ($1/4\times$period).



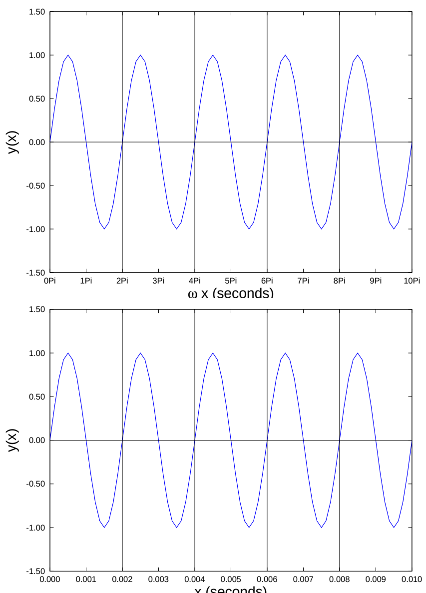
- $\cos(2\pi fx) = \sin(2\pi fx + \phi)$ where $\phi = \pi/2$.
- $\sin(2\pi fx) = \cos(2\pi fx + \phi)$ where $\phi = -\pi/2$.

Notes

Angular Frequency

- Frequency, $f = 1/T$
- Angular frequency, $\omega = 2\pi f$
- 1 period or cycle $= 2\pi$ radians

$$y(x) = \sin(2\pi fx + \phi)$$
$$= \sin(\omega x + \phi)$$



Notes

Phasor Representation

A cosine (or sine) wave:

$$y(x) = a\cos(\omega x + \phi)$$

can be represented as a phasor.
A phasor is a complex number:

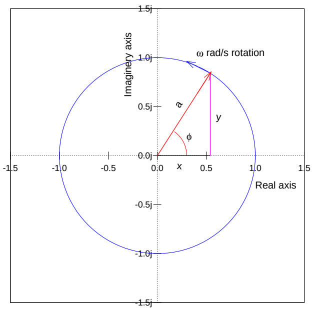
$$z = x + jy = a(\cos(\phi) + j\sin(\phi))$$

where x is known as the real part or $\mathrm{Re}(z) = x$ and y is known as the imaginery part or $\mathrm{Im}(z) = y$.

x and y can be calculated with $x = a\cos(\phi)$ and $y = a\sin(\phi)$.

Also remember $j = \sqrt{-1}$.

Argand *or* Phasor Diagram:



Notes

Complex Numbers

The square root of minus one is not defined so a symbol, j is used (sometimes i):

$$j = \sqrt{-1}.$$

Powers:

- $j^2 = -1$
- $j^3 = -j$
- $j^{-1} = 1/j = -j$

If $z = x + jy$ (rectangular form) then alternative representations are:

- Polar form: $z = a\angle\phi$
- Exponential form: $z = a\exp(j\phi)$

where $a = \sqrt{x^2 + y^2}$ and $\phi = \tan^{-1}(y/x)$.

Notes

Properties of Complex Numbers

If $z = x + jy$, $z_1 = x_1 + jy_1$ and $z_2 = x_2 + jy_2$ then

- Addition: $z_1 + z_2 = x_1 + x_2 + j(y_1 + y_2)$
- Subtraction: $z_1 - z_2 = x_1 - x_2 + j(y_1 - y_2)$
- Multiplication: $z_1 z_2 = a_1 a_2\angle(\phi_1 + \phi_2)$
- Division: $z_1/z_2 = a_1/a_2\angle(\phi_1 - \phi_2)$
- Reciprocal: $1/z = 1/a\angle(-\phi)$
- Square root: $\sqrt{z} = \sqrt{a}\angle(\phi/2)$
- Complex conjugate: $z^* = x - jy = a\angle - \phi$

The polar form simplifies some operations such as multiplication and division of complex numbers.

Notes

Phasor Representation

Euler's identity:

$$\exp(j\phi) = \cos(\phi) + j\sin(\phi)$$

Therefore

- $\cos(\phi) = \mathrm{Re}(\exp(j\phi)) \longrightarrow$ or the real part, x
- $\sin(\phi) = \mathrm{Im}(\exp(j\phi)) \longrightarrow$ or the imaginary part, y

Recall the cosine wave:

$$y(x) = \cos(\omega x + \phi)$$

which can be written as:

$$\begin{aligned} y(x) = \mathrm{Re}(a\exp(j(\omega x + \phi))) &= \mathrm{Re}(a\exp(j\omega x)\exp(j\phi)) \\ &= \mathrm{Re}(A\exp(j\omega x)) \end{aligned}$$

where A is the phasor representation of $y(x)$ given by

$$A = a\exp(j\phi) = a\angle(\phi).$$

Notes

Complex Exponentials, Sines and Cosines

Given

- $y_1(x) = b\exp(j\omega x) = b\cos(\omega x) + jb\sin(\omega x)$
- $y_2(x) = b\exp(-j\omega x) = b\cos(\omega x) + jb\sin(-\omega x)$

  as

  - $\cos(-\omega x) = \cos(\omega x)$ (even function)
  - $\sin(-\omega x) = -\sin(\omega x)$ (odd function)

Then

$$y_1(x) + y_2(x) = 2b\cos(\omega x).$$

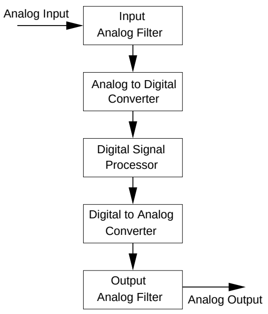
So that

$$b\cos(\omega x) = \frac{a}{2}\exp(j\omega x) + \frac{a}{2}\exp(-j\omega x).$$

A similar approach can be used to derive a sine function.

Notes

A Typical Digital Signal Processing System



- Input Analog Filter (antialiasing):
  Limits frequency range
- Analog to Digital Converter
  Converts signal to digital samples
- Digital Signal Processor
  Storage, Communication and or Calculations
- Digital to Analog Converter
  Convert to continuous signal
- Output Analog Filter
  Removes sharp transitions

Notes

Analog to Digital Converter

- Real world is typically *analog* (continuous)
- Digital signal approximates analog signal with discrete quantised samples
- ADC converts an analog signal to a digital signal
- Signal is digitised in two ways:
  - Signal is sampled at a sampling rate or frequency: Information is collected about the signal at regular intervals.
  - The continuous or analog signal is then quantised: *i.e.* put into digital form, where only a finite set of numbers are represented.

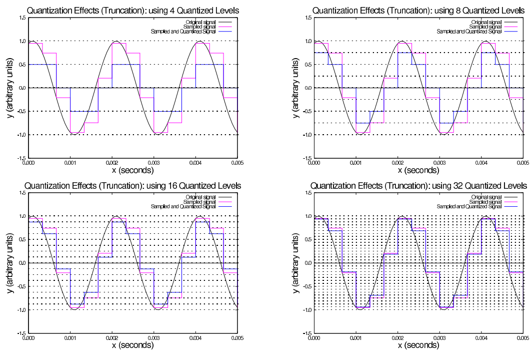
Notes

Quantisation using Truncation

- Signal can be quantised using *e.g.* truncation where numbers following specified position are removed.
- Examples:
  - 5.7 truncated to integer is 5
  - 5.11 truncated to 1 decimal place is 5.1
- Negative numbers are truncated in the same way (note <u>different</u> to the common *floor* function in matlab), *e.g.*
  - -5.78 truncated to integer is -5
  - -5.135 truncated to 2 decimal places is -5.13

Notes

Truncation Quantisation *examples*



- Errors can be seen between the sampled and the sampled and quantized signals.

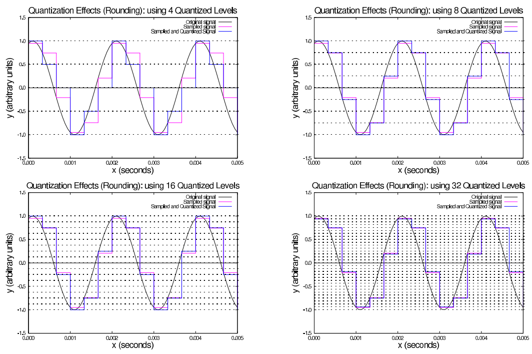
Notes

Quantisation using Rounding

- Rounding can be a quantization method associated with smaller errors, *e.g.*
  - 5.7 rounded to nearest integer is 6
  - 5.11 rounded to 1 decimal place is 5.1
  - -5.78 rounded to nearest integer is -6
  - -5.135 rounded to 2 decimal places is -5.14

Notes

Rounding Quantisation *examples*

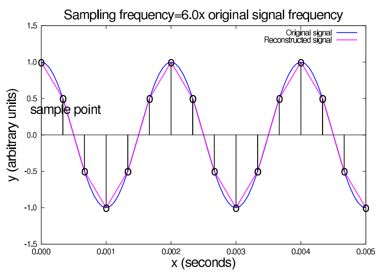


- Errors can be seen between the sampled and the sampled and quantized signals.

Notes

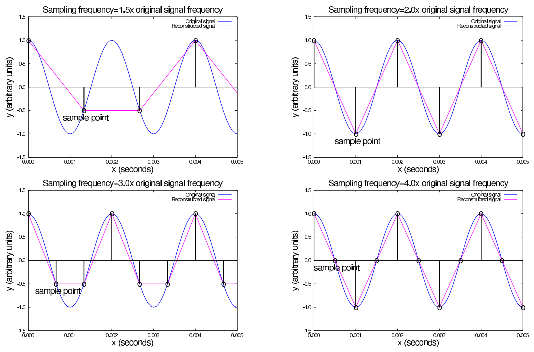
Sampling

- Sampling also affects the quality of the digitised signal.
- Higher sampling rate reduces error and enables better representation of the original analog signal in digital form.



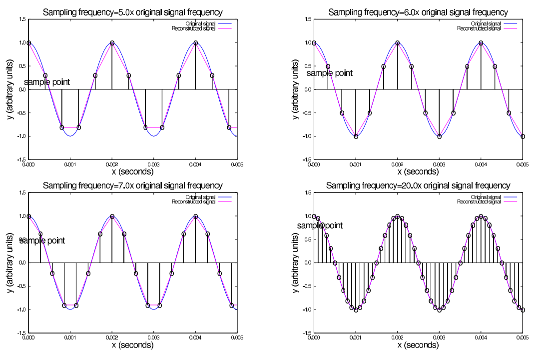
Notes

Sampling *examples*



Notes

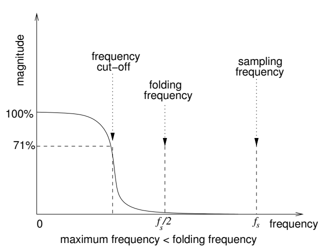
Sampling *examples cont'd*



Notes

Input Analog Filter: Antialiasing Filter

- Analog to Digital Converter (ADC) requires signal below a particular frequency (Nyquist Frequency)
- $\therefore$ Limit frequency range to below Nyquist frequency ($f_s/2$) before Analog to Digital Conversion.



- Otherwise next stage produces frequency errors (*i.e.* aliasing)
- Sampling produces copies of signal at multiples of sampling frequency
- Aliasing occurs when copies of signal overlap each other

Notes

Digital Signal Processor

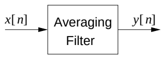
- After digitisation (with the ADC) digital signal processing may then be performed on the digitised signal.
- Simple example
  - Averaging filter:

$$y[n] = \frac{x[n] + x[n-1] + ... + x[n-k+1]}{k}$$

for window width $k = 3$

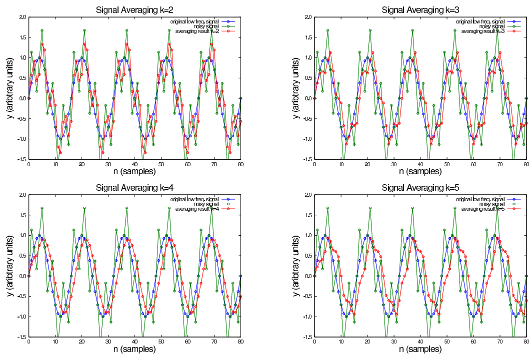
$$y[n] = \frac{x[n] + x[n-1] + x[n-2]}{3}$$

where $x[n]$ is an input value at sample time n and $y[n]$ is an output at sample time n



Notes

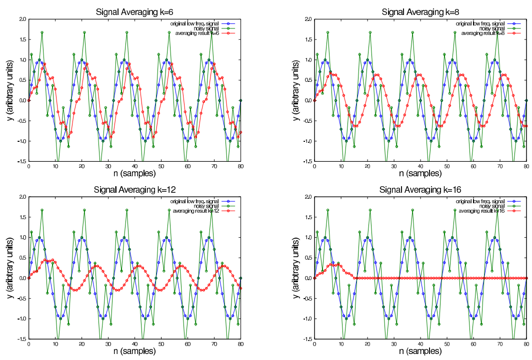
Averaging Filter *Examples*



Window width k controls the response of the filter. If k is too low, there is little benefit on output signal.

Notes

Averaging Filter *Examples cont'd*



Window width k controls the response of the filter. If k is too high, the filter removes all of the output signal.

Notes

Summary

- Definition of digital signal processing
- Description of phase
- Cosine and Sine functions
- Complex numbers and alternative representations
- A typical digital signal processing system

Notes

Notes

Notes