

Low-Pass Single-Pole IIR Filter

Summary: This article shows how to implement a low-pass single-pole IIR filter. The article is complemented by a [Filter Design tool](#) that allows you to create your own custom versions of the example filter that is shown below.

The low-pass single-pole IIR filter is a very useful tool to have in your DSP toolbox. Its performance in the frequency domain may not be stellar, but it is very computationally efficient.

Definition

A low-pass single-pole IIR filter has a single design parameter, which is the decay value d . It is customary to define parameters $a = d$ and $b = 1 - d$ (the logic behind this follows from the general case below). For a typical value of $d = 0.99$, we have that $a = 0.99$ and $b = 0.01$. The recurrence relation is then given by

$$y[n] = bx[n] + ay[n - 1],$$

where the sequence $x[n]$ is the input and $y[n]$ is the output of the filter.

The recurrence relation directly shows the effect of the filter. The previous output value of the the filter, $y[n - 1]$, is decreased with the decay factor $a = d$. The current input value, $x[n]$, is taken into account by adding a small fraction $b = 1 - d$ of it to the output value.

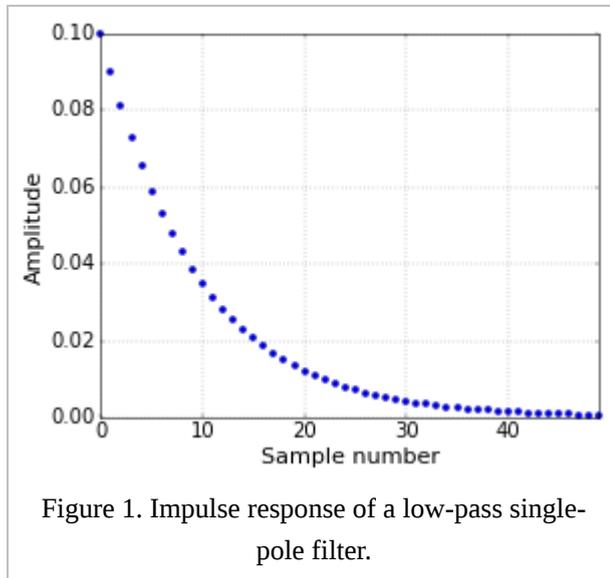
Substituting $b = 1 - a$ in the given recurrence relation and rewriting leads to the expression

$$y[n] = y[n - 1] + b(x[n] - y[n - 1]).$$

This then leads to compact update expressions such as `y += b * (x - y)`, in programming languages that support the `+=`-operator (see the Python code below for an example).

Impulse Response

For windowed-sinc filters (see, e.g., [How to Create a Simple Low-Pass Filter](#)), the impulse response is the filter. To apply the filter, you *convolve* the impulse response of the filter with the data. This is different for the single-pole IIR filter. Its action is essentially defined on a sample-by-sample basis, as described by the recurrence relation given above. The impulse response of a filter with $d = 0.9$ ($b = 0.1$) is shown in Figure 1.



Of course, this impulse response is actually *infinite*. I've plotted the first 50 samples here, and at that point it is quite close to zero, but it never actually *reaches* zero.

Properties

The response of this filter is completely analogous to the response of an electronic low-pass filter consisting of a single resistor and a single capacitor.

The decay value d is related to the *time constant* τ of the filter with the relation

$$d = e^{-1/\tau}.$$

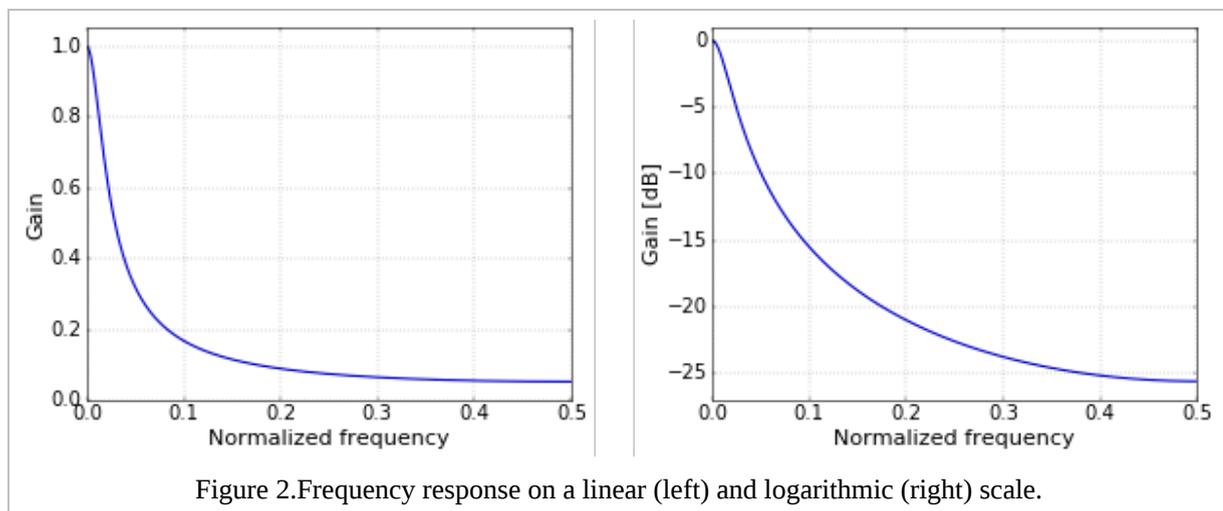
Hence, if d is given, the value of τ can be computed as $\tau = -1/\ln(d)$. As for an electronic RC-filter, the time constant τ gives the time (in samples for the discrete case) for the output to decrease to 36.8% ($1/e$) of the original value.

Another useful relation is that between d and the (-3 dB) *cutoff frequency* f_c , which is

$$d = e^{-2\pi f_c}.$$

Hence, if d is given, the value of f_c can be computed as $f_c = -\ln(d)/2\pi$.

The frequency response of the filter with the impulse response of Figure 1 is given in Figure 2.



Python code

In Python and in most other programming languages, the recurrence relation can be implemented through the already mentioned expression $y += b * (x - y)$. Below is a small Python class that implements this expression in its `filter()` member.

```
decay = 0.9 # Decay between samples (in (0, 1)).

class LowPassSinglePole:
    def __init__(self, decay):
        self.b = 1 - decay
        self.reset()
    def reset(self):
        self.y = 0
    def filter(self, x):
        self.y += self.b * (x - self.y)
        return self.y

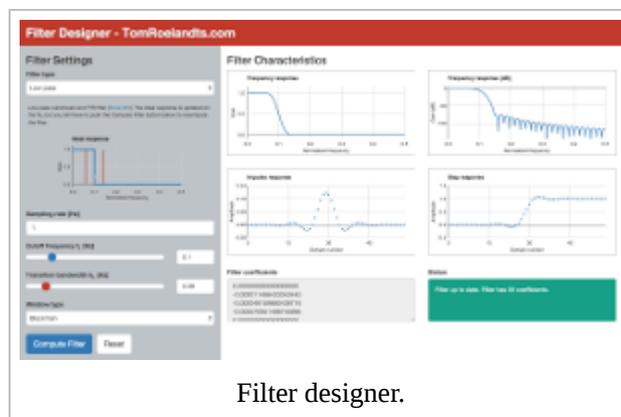
low_pass_single_pole = LowPassSinglePole(decay)
```

This filter can then be applied by calling the `filter()` member for each new input sample x , resulting in a new output sample y :

```
y = low_pass_single_pole.filter(x)
```

Filter Design Tool

This article is complemented with a [Filter Design tool](#). Experiment with different values for d , visualize the resulting filters, and download the filter code. Try it now!



Filter designer.

General Case

This last section is mainly here to have everything in one place for now. I plan to add a separate article on the Z-transform later.

A single-pole low-pass infinite impulse response (IIR) filter is given by the Z-transform

$$H[z] = \frac{bz}{z - a} = \frac{b}{1 - az^{-1}},$$

where $a + b = 1$ results in a filter with unity gain at DC.

The general form of this equation is

$$H[z] = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2} + \dots}{1 - a_1 z^{-1} - a_2 z^{-2} + \dots} = \frac{\sum_{n=0}^{\infty} b_n z^{-n}}{1 - \sum_{n=1}^{\infty} a_n z^{-n}}.$$

Tags: [Signal Processing](#) [Filter Design](#)

no body
(not
verified)

Wed,
02/07/2018
- 20:06

[Permalink](#)

"where a+b=1 results in a filter with unity gain." Should be "... unity gain at DC."

[Reply](#)

Tom

Thu,
02/08/2018
- 13:13

[Permalink](#)

Yes, at DC. I've added it in the article for completeness.

[Reply](#)

Mark
(not
verified)

Tue,
12/25/2018
- 16:25

[Permalink](#)

Hi!

I've encountered two conflicting formulas for the decay coefficient and decided to ask about that on Stack Exchange.

The formula you provide is an approximation, see <https://dsp.stackexchange.com/questions/54086/single-pole-iir-low-pass-....>

It seems to work fine with lower frequencies, but its value differs significantly when the center frequency is too high.

Thanks for your attention! :)

[Reply](#)

Tom

Fri,
12/28/2018
- 15:47

[Permalink](#)

Indeed, it's an approximation. Maybe I should have mentioned that. In practice, however, I think that this filter is often used with relatively small cutoff frequencies. For example, for d=0.99, the (estimated) cutoff frequency is about 0.0016 (relative to 0.5; the plot in the StackExchange answer that you refer to is relative to 1), which is clearly in the usable range of the approximation.

[Reply](#)

[Mark P \(mp035\) \(not verified\)](#)

Tue, 02/23/2021 - 11:05

[Permalink](#)

Thank you for this article. As an absolute amateur in DSP I found it very helpful.

[Reply](#)

ian (not
verified)

Sun,

Can you comment on the group delay of the one pole filters?

01/29/2023
- 02:00
[Permalink](#)

[Reply](#)

Tom
Sun,
01/29/2023
- 12:33
[Permalink](#)

I'll consider adding an article on group delay, because that is indeed missing for now...

[Reply](#)

Add new comment

Your name

Email

The content of this field is kept private and will not be shown publicly.

Homepage

3 x 4 = ?

Spam avoidance measure, sorry for this.

Comment

[About text formats](#)

- Allowed HTML tags: `<a href hreflang>` `` `` `<cite>` `<blockquote cite>` `<code>` `<ul type>` `<ol start type>` `` `<dl>` `<dt>` `<dd>` `<h2 id>` `<h3 id>` `<h4 id>` `<h5 id>` `<h6 id>`
- Lines and paragraphs break automatically.
- Web page addresses and email addresses turn into links automatically.

Save

Preview

Submitted on 14 February 2016