



Digital Filtering

preparing for the exam - part 2, IIR filters

Antonin Novak



Theoretical Examples

1. A digital filter is described by difference equation

$$y[n] = 0.1x[n] + 0.7x[n-1] - 0.6x[n-2] + 0.3y[n-1] - 0.19y[n-2].$$

- a Find the transfer function $H(z)$ of the filter.
b Write the filter coefficients in a Sigma Studio format [b0, b1, b2, a1, a2].
-

1) Find the Transfer Function

$$y[n] = 0.1x[n] + 0.7x[n-1] - 0.6x[n-2] + 0.3y[n-1] - 0.19y[n-2]$$

$$Y(z) = 0.1X(z) + 0.7X(z) \cdot z^{-1} - 0.6X(z) \cdot z^{-2} + 0.3Y(z) \cdot z^{-1} - 0.19Y(z) \cdot z^{-2}$$

$$Y(z) - 0.3Y(z) \cdot z^{-1} + 0.19Y(z) \cdot z^{-2} = 0.1X(z) + 0.7X(z) \cdot z^{-1} - 0.6X(z) \cdot z^{-2}$$

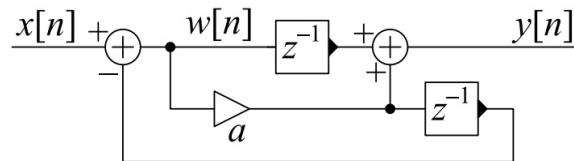
$$Y(z) [1 - 0.3 \cdot z^{-1} + 0.19 \cdot z^{-2}] = X(z) [0.1 + 0.7 \cdot z^{-1} - 0.6 \cdot z^{-2}]$$

$$H(z) = \frac{Y(z)}{X(z)} = \frac{0.1 + 0.7 \cdot z^{-1} - 0.6 \cdot z^{-2}}{1 - 0.3 \cdot z^{-1} + 0.19 \cdot z^{-2}}$$

2) Write the filter coefficients in Sigma Studio format.

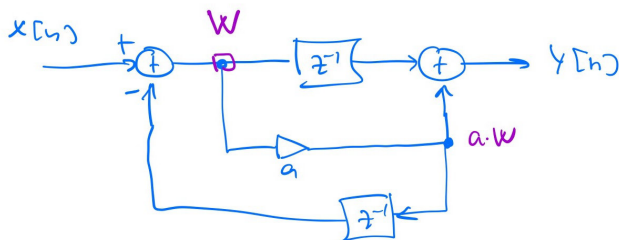
$$[0.1, 0.7, -0.6, -0.3, 0.19]$$

2. In the following block diagram, all elements are drawn with their outputs on the right. The input and output signals are $x[n]$ and $y[n]$ respective



- a Determine the transfer function of the system $H(z) = \frac{Y(z)}{X(z)}$.
- b Draw the transposed form of the block diagram.
-

1) Find the Transfer Function



$$W(z) = X(z) - z^{-1} \cdot a \cdot W(z)$$

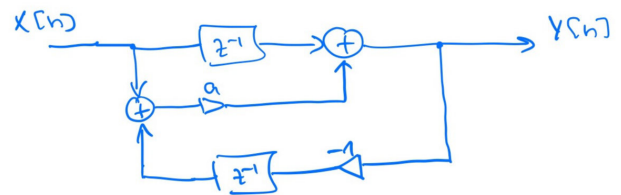
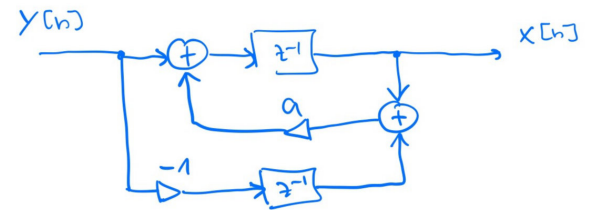
$$W(z) (1 + a \cdot z^{-1}) = X(z)$$

$$W(z) = \frac{X(z)}{1 + a \cdot z^{-1}}$$

$$Y(z) = z^{-1} \cdot W(z) + a \cdot W(z) = W(z) \cdot (a + z^{-1}) =$$

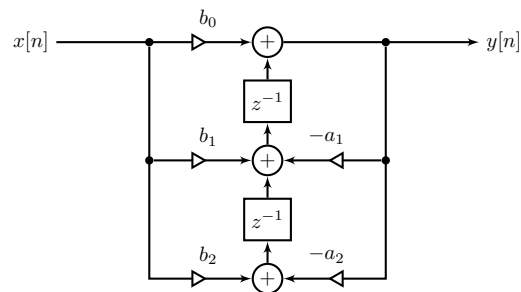
$$= X(z) \cdot \boxed{\frac{a + z^{-1}}{1 + a \cdot z^{-1}}}$$

2) Transposed form



Practical Examples

3. Complete a Python-like code that implements the following digital filter structure (sample by sample) without using `lfilter` function.



```
import numpy as np
import imdea
```

```
b, a = ... # filter coefficients b, a
x = ...   # input signal
```

```
y = np.zeros_like(x) # an empty array for the output signal
```

```
# complete the code here to get the output signal y
```

.....

```
d = [0, 0] # memory buffer
```

```
for n in range(N):
```

```
    y[n] = b[0] * x[n] + d[0]
```

```
    # memory update
```

```
    d[0] = b[1] * x[n] - a[1] * y[n] + d[1]
```

```
    d[1] = b[2] * x[n] - a[2] * y[n]
```

4. A 2nd order IIR filter has following parameters

$$b = [1, -2, 1]$$

$$a = [1, -1.9, 0.9]$$

Implement this filter in Sigma Studio without using Filter block. You can use Multiply, Add, Gain, Delay, Feedback, and T-connection.

.....

