

Číslicové filtry

Honza Černocký, ÚPGM

Alias

- Digitální filtry
- Diskrétní systémy
- Systémy s diskrétním časem
- atd.

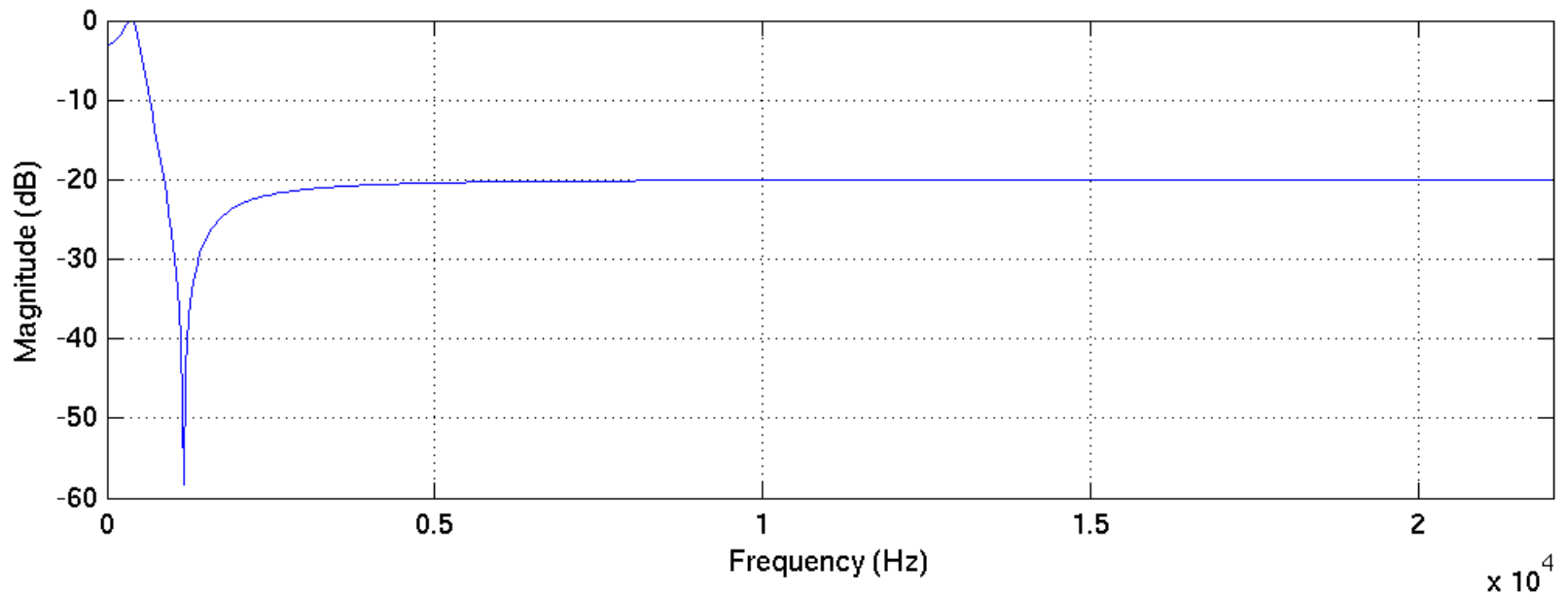
Na co ?

Úprava signálů

- Zdůraznění
- Potlačení
- Detekce

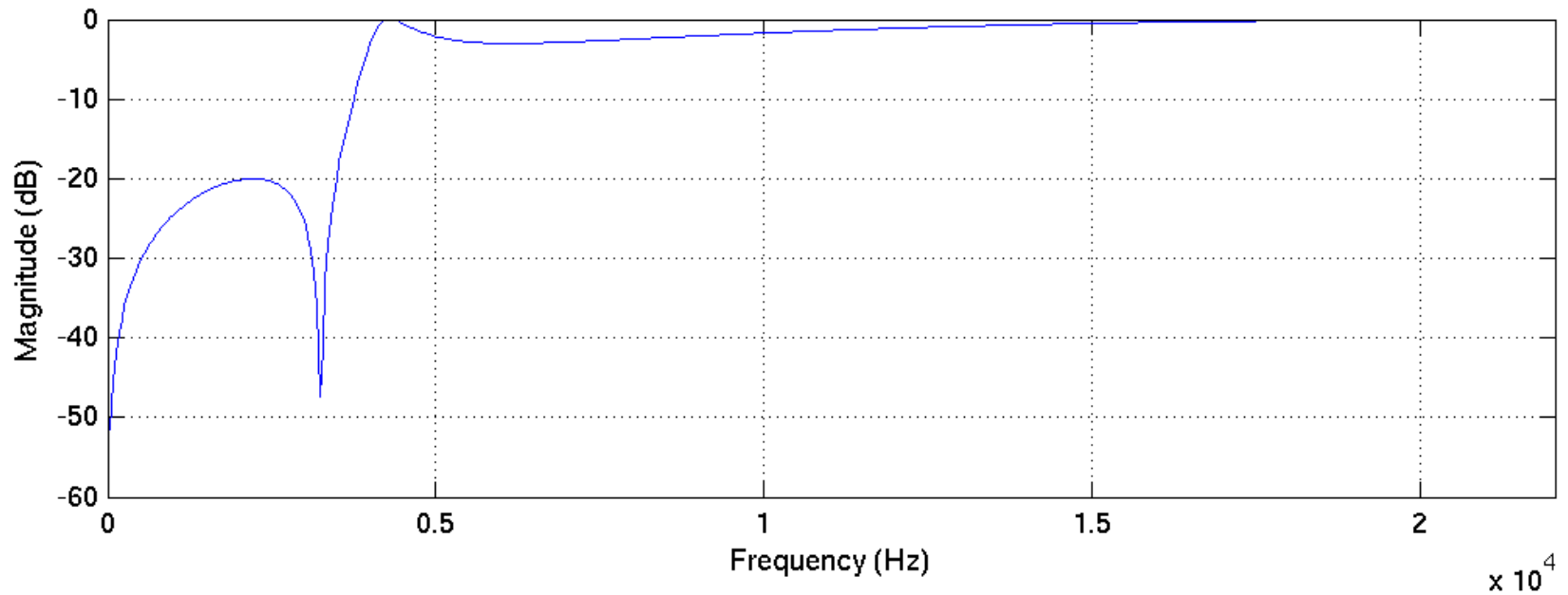
Zdůraznění

- basy



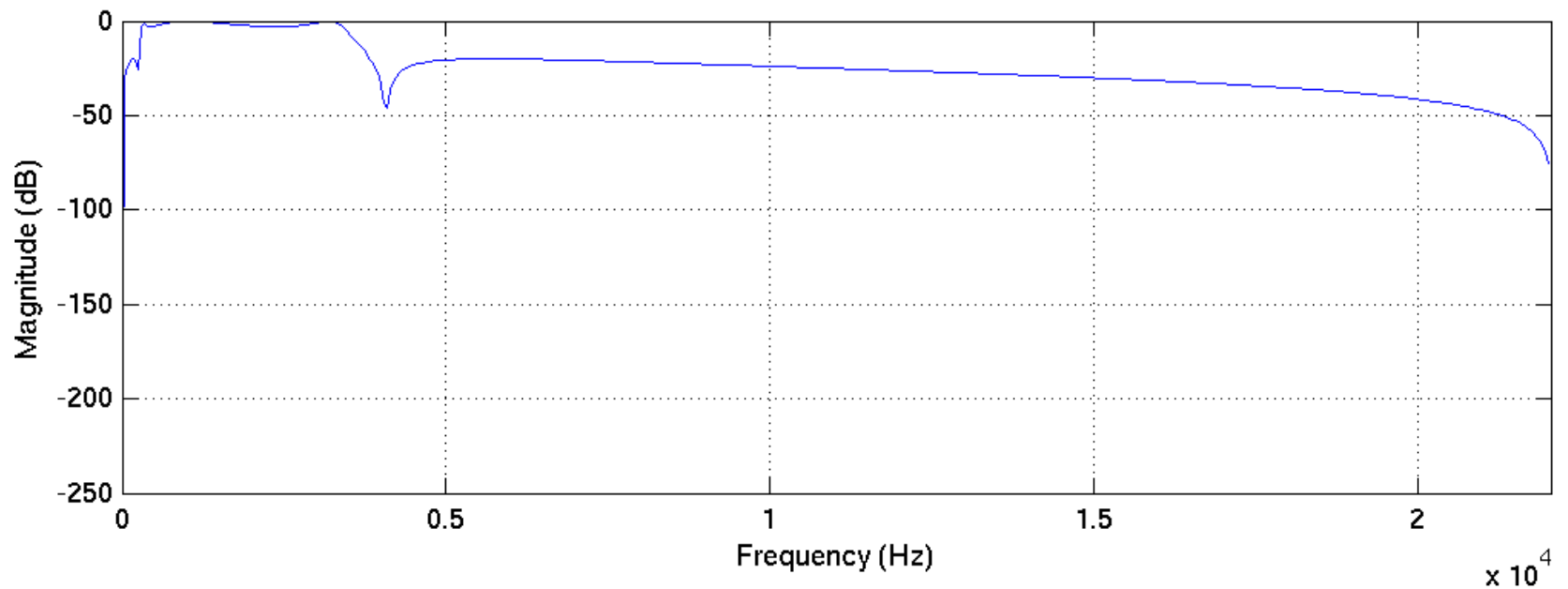
Zdůraznění

- výšky



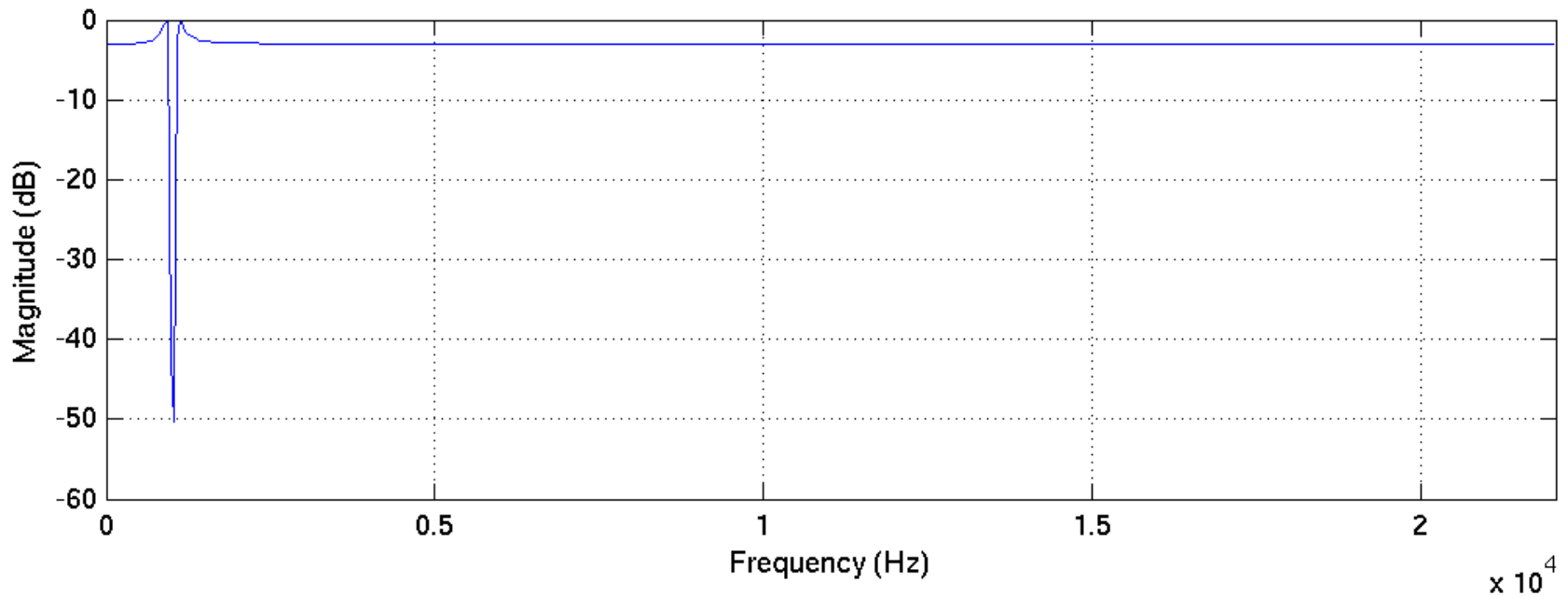
Zdůraznění

- Telefonní pásmo 300-3400 Hz



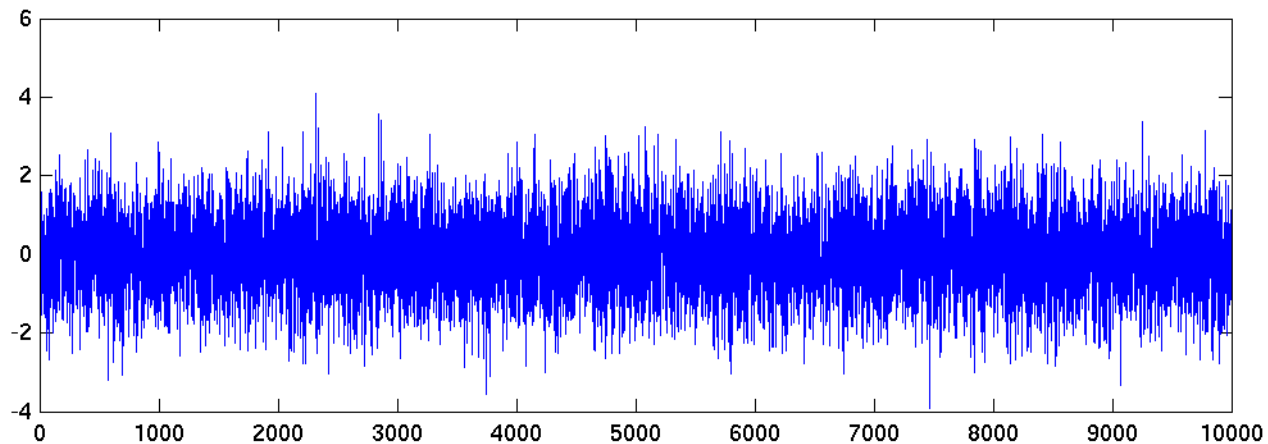
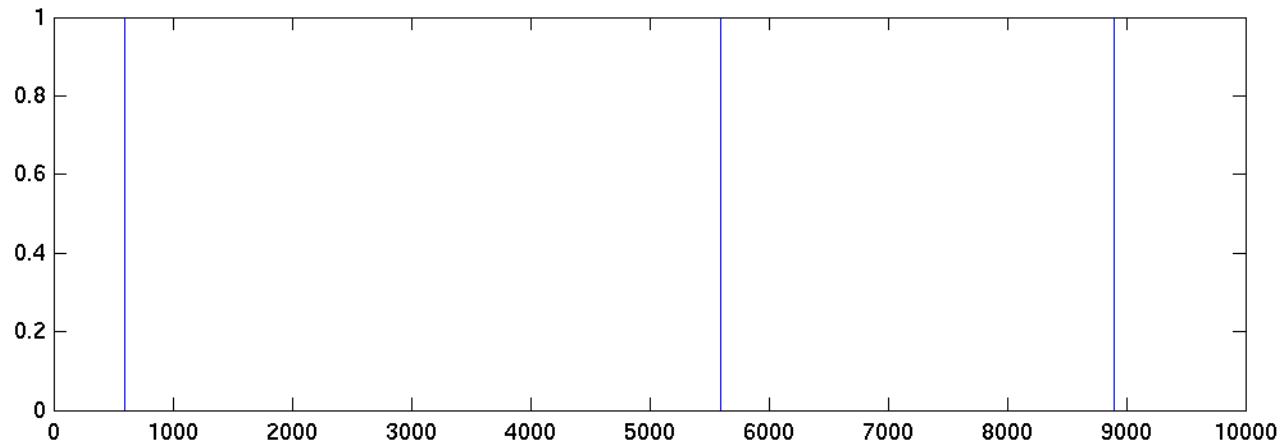
Potlačení

- Signál kontaminovaný 1kHz a jeho čištění ostrou pásmovou zadrží...



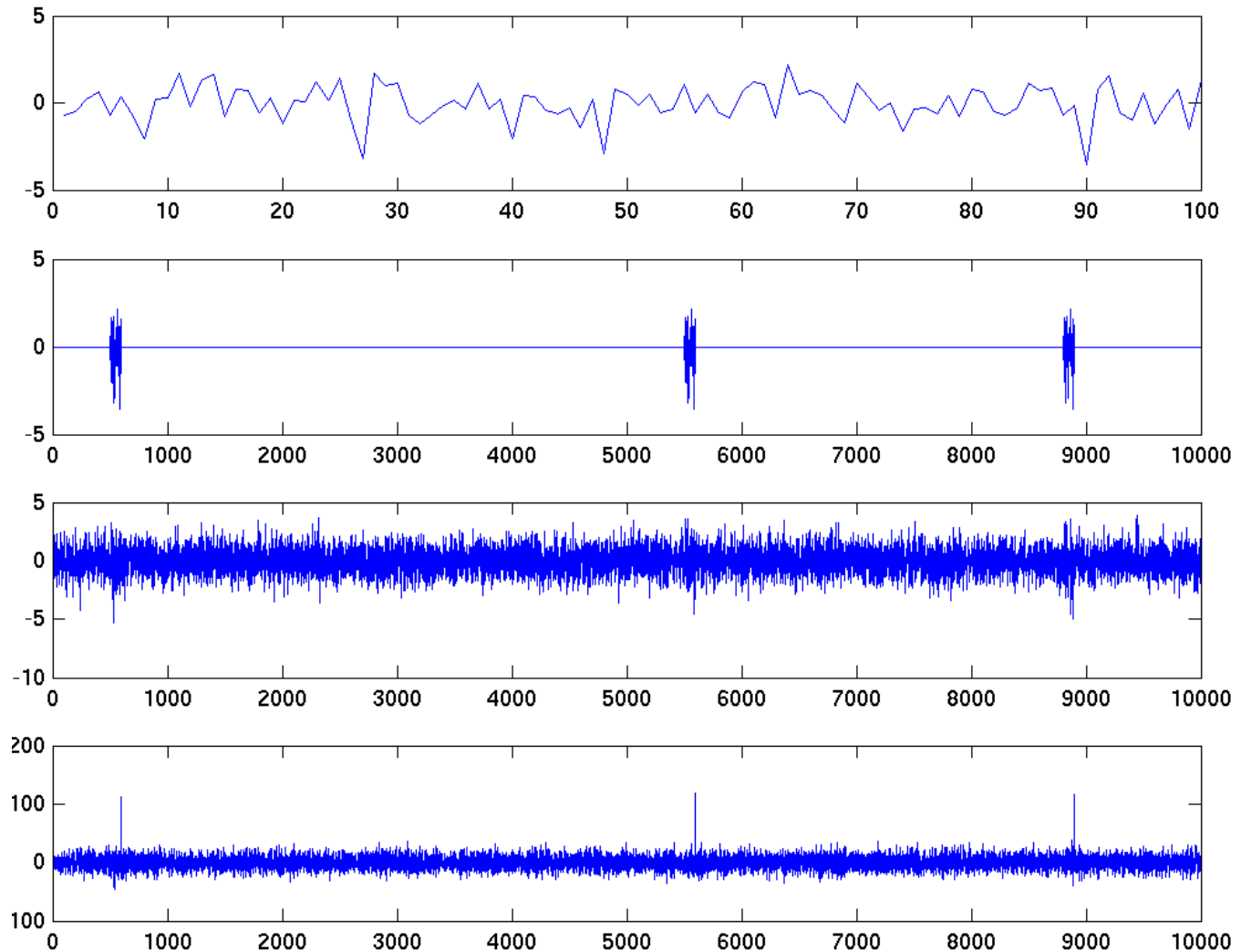
Detekce

- Pár špiček signálu v šumu



Detekce

- Přizpůsobený filtr



- Pěkná aplikace: RICHTER Jiří. Echo-Based Distance Measurement on Mobile Phone, BP FIT, 2014/2015



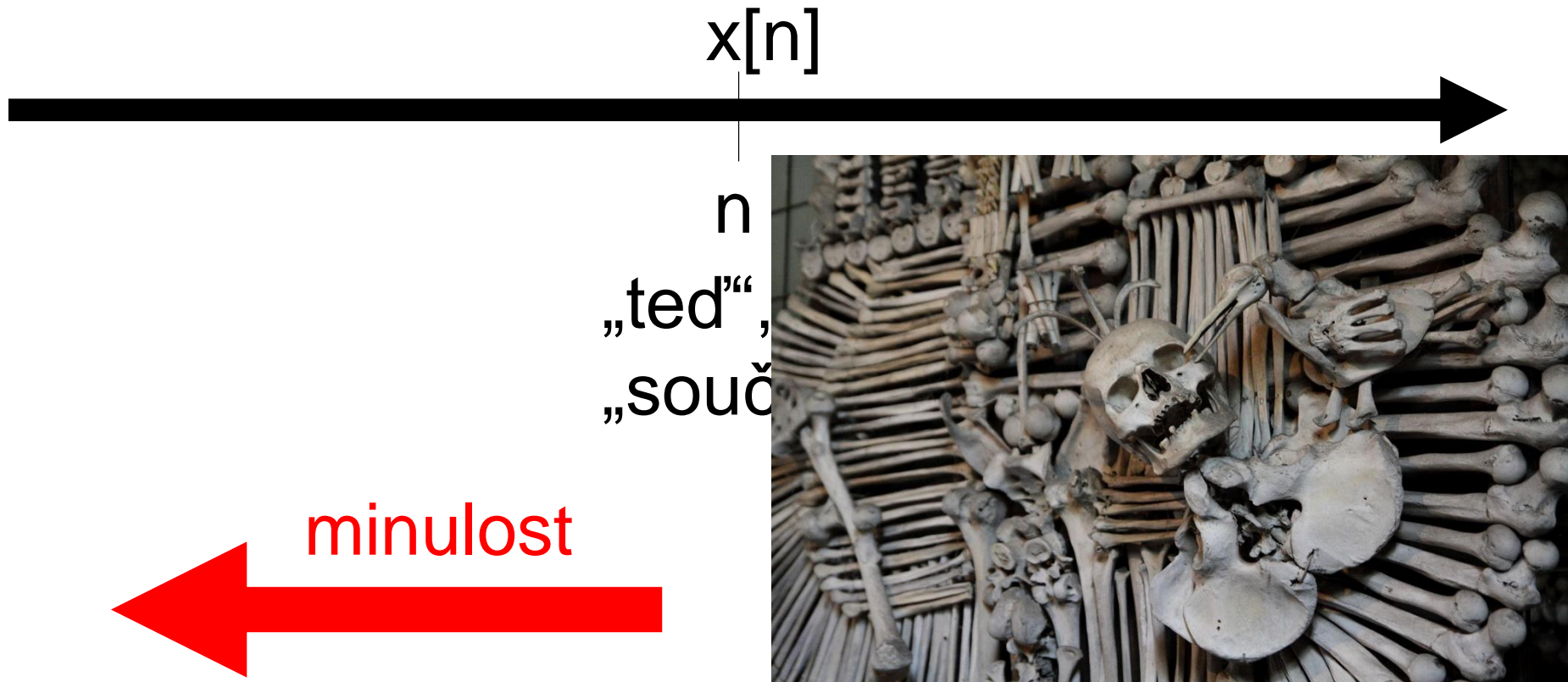
Co bude potřeba ?

- Vzorkovaný signál
 - na nějaké vzorkovací frekvenci (nejčastěji 8kHz, 16kHz, 44.1kHz (CD), 48kHz, 96kHz)
 - Pro zobrazování se na něj budeme dívat jako na spojitý ([stem vs. plot](#))
 - Pro ISS žádné speciální formáty (MP3, OGG, WAV), ale RAW
 - Bez hlavičky, sekvence 16-bit shorts
- ... [wire.c](#)

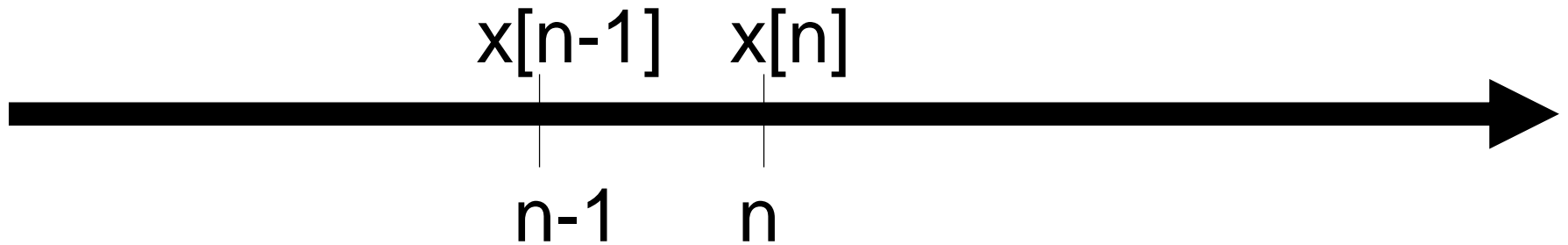
Co bude dál potřeba ?

- Pouze **3** základní operace
 - Násobení konstantou *
 - Sečítání +
 - Posun

Posun ? Pouze do minulosti



Posun off-line



... jen jiná hodnota ukazatele.

Posun on-line

- K dispozici je jen $x[n]$
- Funkce s **pamětí**

```
float filter (float xn) {  
    static float xn1, xn2;  
    ... nejake zpracovani ...  
    xn2 = xn1;  
    xn1 = xn;  
    return něco;  
}
```

Jednoduchý filtr

$$y[n] = b_0x[n] + b_1x[n-1] + b_2x[n-2] + b_3x[n-3]$$

... **diferenční rovnice**

Jak to funguje ?

--	--	--	--	--	--	--	--	--	--	--	--

Implementace off-line

- Alokace místa pro výstupní signál
- Pak naprosto doslovná implementace diferenční rovnice

[fir_offline.c](#)

Implementace on-line

- Funkce pro filtrování volána pro každý vzorek
- Musí si pamatovat minulé vzorky

[fir_online.c](#)

Co to asi bude dělat pro ...

b0	b1	b2	b3
0.25	0.25	0.25	0.25
0.25	-0.25	0.25	-0.25
1	0	0	0
2	0	0	0
0	0	0	1

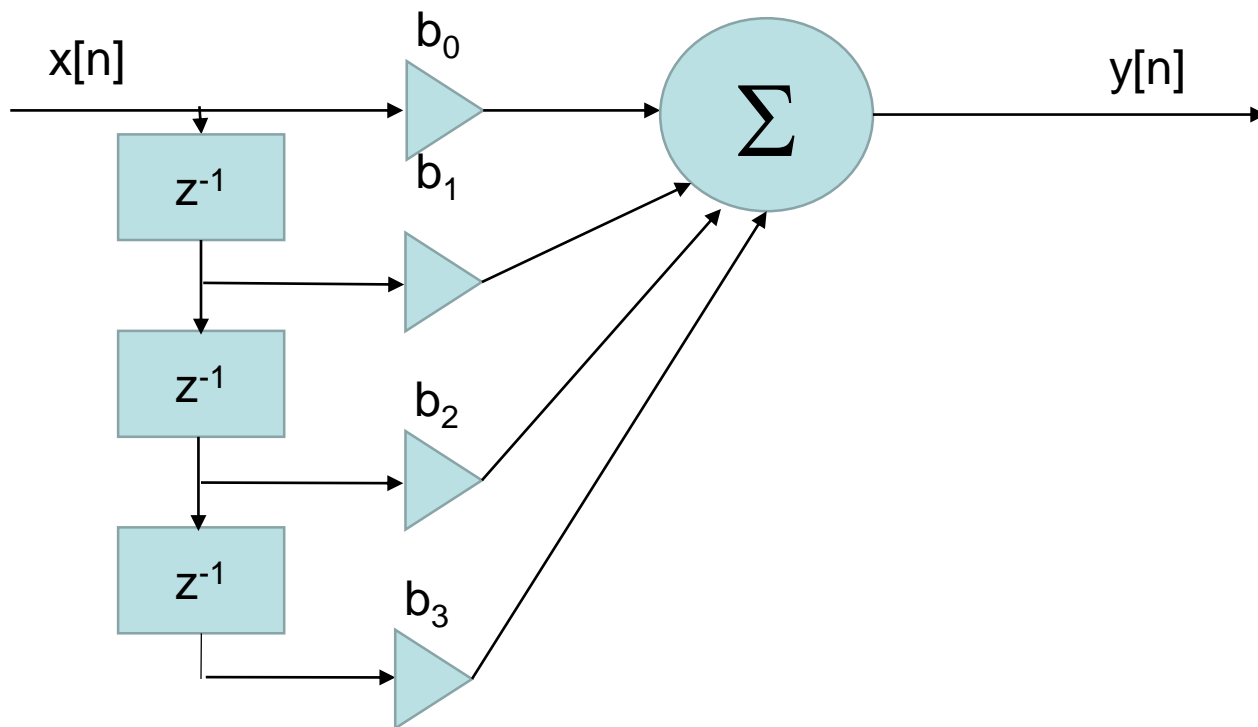
Ještě jedna možná implementace...



Jak zapsat a zakreslit filtr ?

$$y[n] = b_0x[n] + b_1x[n-1] + b_2x[n-2] + b_3x[n-3]$$

... **diferenční rovnice**



... **schema**

Impulsní odezva

- Reakce na jednotkový impuls

0	0	0	0	0	1	0	0	0	0	0	0
---	---	---	---	---	---	---	---	---	---	---	---

Konečná / nekonečná impulsní
odezva ?

Konvoluce

$$\begin{aligned} y[n] &= x[n] \star h[n] \\ &= \sum_{k=-\infty}^{+\infty} x[k]h[n-k] \end{aligned}$$

... demo na našem filtru.

Jde to i naopak (konvoluce je komutativní...)

$$y[n] = x[n] \star h[n] = h[n] \star x[n]$$

$$= \sum_{k=-\infty}^{+\infty} h[k]x[n - k]$$

... demo na našem filtru.

Konvoluce - souhrn

- Otočit impulsní odezvu
 - Posunout ji na dané „n“
 - Násobit
 - Sčítat
 - Napsat
- ... papírková metoda – [demo](#).

Využití **výstupu** filtru

- Zpětná vazba – **rekurzivní filtry**



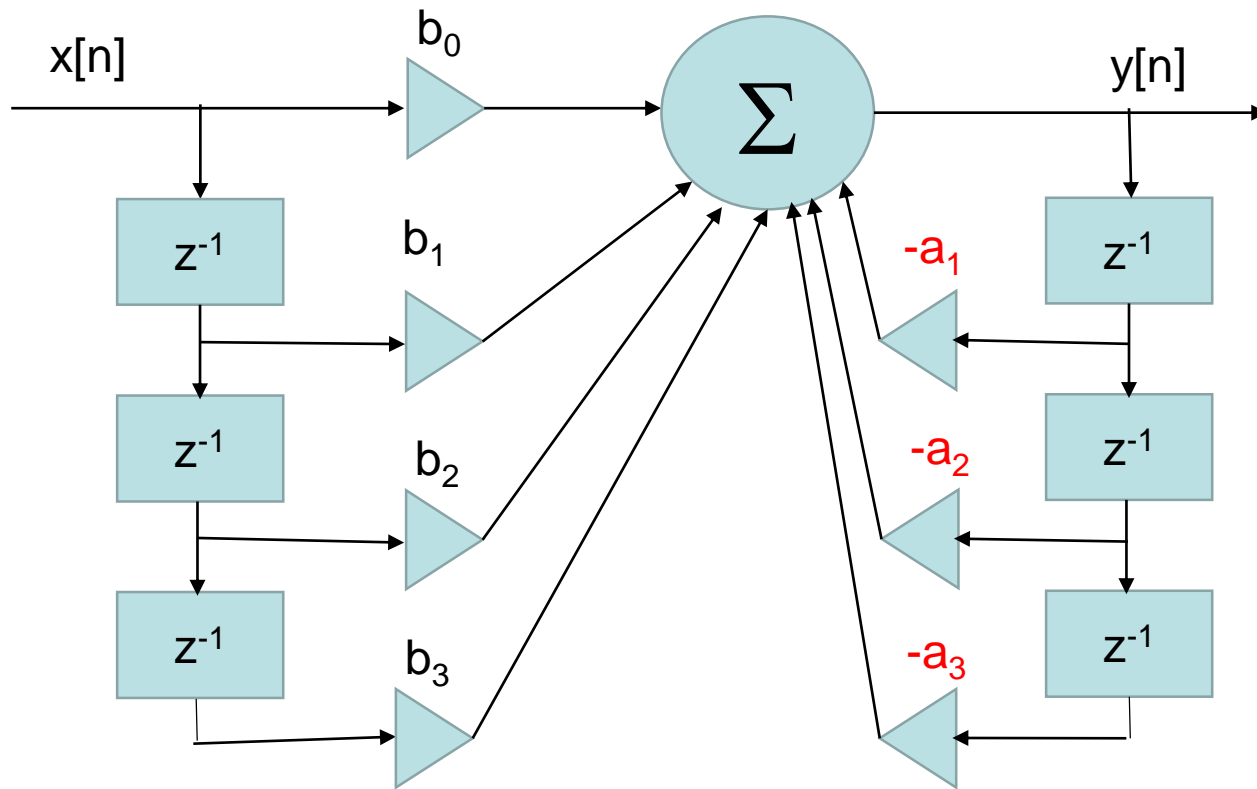
Diferenční rovnice

$$y[n] = b_0x[n] + b_1x[n-1] + b_2x[n-2] + b_3x[n-3] \\ - a_1y[n-1] - a_2y[n-2] - a_3y[n-3]$$

... proč tam není a_0 ?

... a proč jsou koeficienty zpětné vazby se záporným znaménkem ?

Schema



Implementace off-line

- Hrabeme i ve starých výstupech ...
... **iir_offline.c**

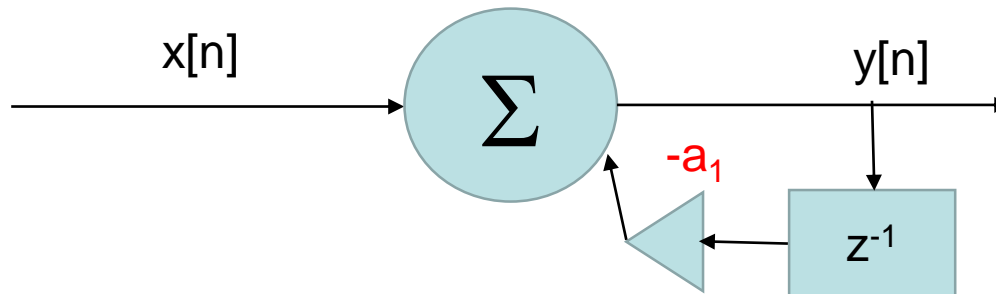
Implementace on-line

- Funkce si musí pamatovat i minulé výstupy.

... **iir_online.c**

Impulsní odezva

- Například jednoduchého čistě rekurzivního filtru (žádná vstupní část)



- **Nekonečná impulsní odezva - IIR**

Stabilita

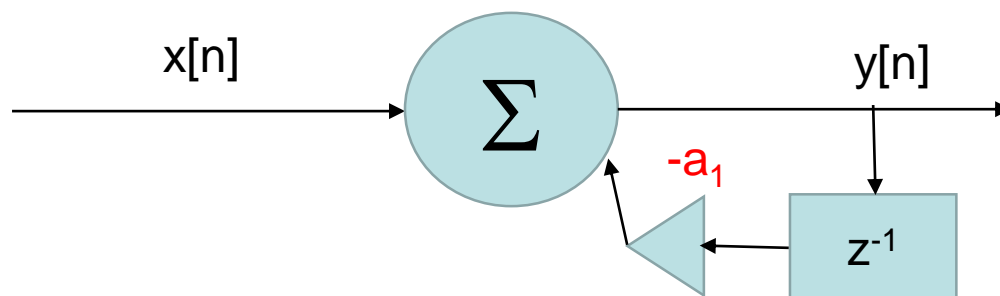
- Matematicky „Bounded Input Bounded Output“
- Lidově “je-li něco rozumného na vstupu, očekávám něco rozumného i na výstupu”

Stabilita FIR

Stabilita IIR

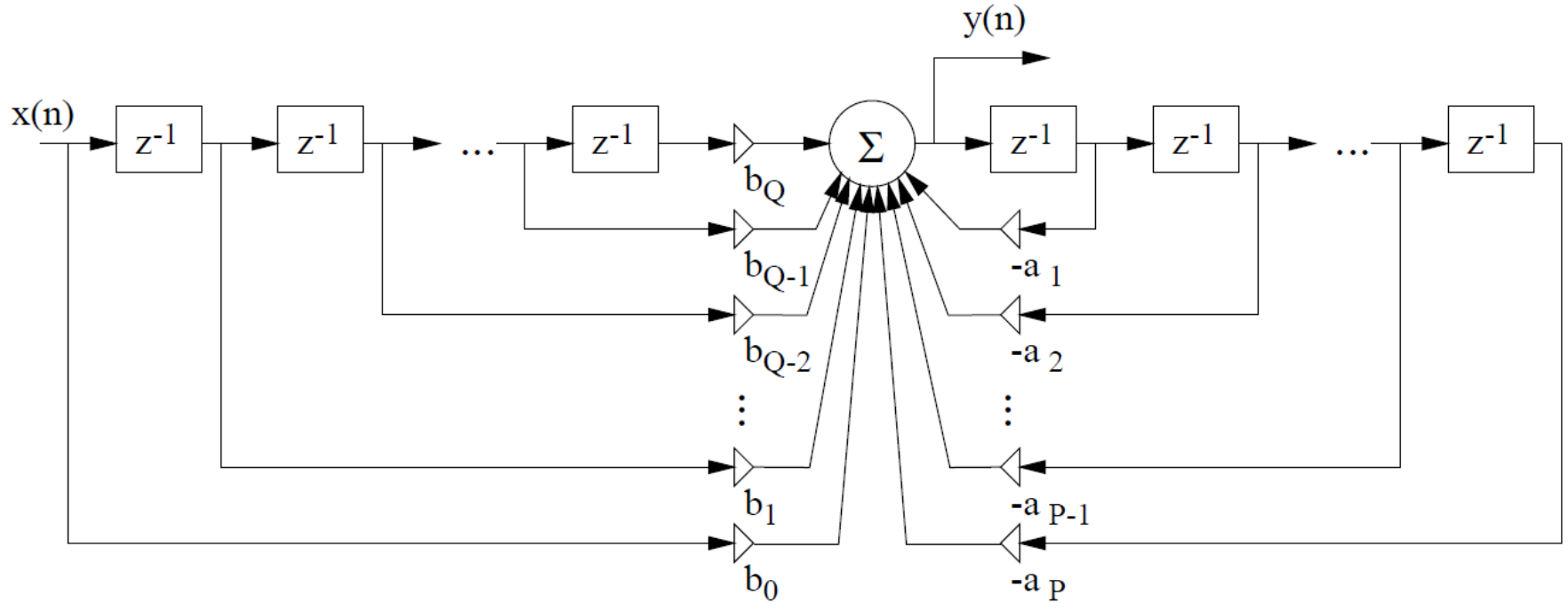


IIR prvního řádu



- Co musí platit ?
- A co musí platit pro složitější IIR filtry ?

Obečný filtr



Obecný filtr

- Řád vstupní části Q
- **Řád výstupní části P**

Diferenční rovnice

$$y[n] = b_0x[n] + b_1x[n-1] + \dots + b_Qx[n-Q] \\ - a_1y[n-1] + \dots + a_Py[n-P]$$

$$y[n] = \sum_{k=0}^Q b_k x[n-k] - \sum_{k=1}^P a_k y[n-k]$$

Implementace off-line

- Pořád to samé, ale s cykly.

iirbig_offline.c

Implementace on-line

- Cykly musí běžet odzadu (jinak by staré hodnoty přepisovaly nové).
- Cyklus pro rekurzivní část se musí zastavit na 1.
- „Odkládací“ paměťová pole, která nebudou nikdy využita, kvůli zjednodušení a úspoře „if“ů.

iirbig_online.c

Frekvenční charakteristiky

- Pro FIR filtry:
- Koeficienty filtru (resp. jeho impulsní odezvy) jsou jako **obráběcí nástroj** – výsledek mu bude podobný.

Testy

- 2s komorního „a“ (440 Hz) plus nějaký šum

Viz Matlab soubor [matlab_filtry.m](#) (sekce “filcy atd”)

Zednický „filc“

- 20 vzorků s hodnotami 1/20
- Vyhlazení, méně šumu.

=> Dolní propust'



Zednický „superfilc“

- 181 vzorků s hodnotami 1/181
- Vyhlazení, méně šumu, ale také skoro žádný signál ☹

=> Ještě více dolní propust'



Řezačka s vroubkovaným okrajem



=> Horní propust'

Pásmová propust' ?

- chceme vybrat 440 Hz.
- Výstup se bude podobat impulsní odezvě
- ...

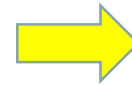
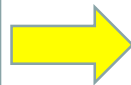
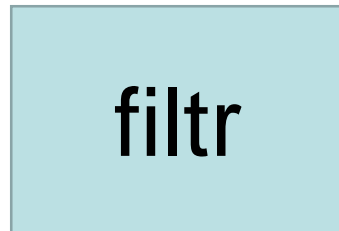
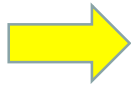
... test také na praseti.

Frekvenční charakteristika přesněji

- Bude nás zajímat (ted') pouze amplituda
- Jak budou jednotlivé frekvence zesíleny/zeslabeny
- ... ne jak budou posunuty.

Viz závěr [matlab_filtry.m](#)

Metoda 1 – proměřit !



Metoda 2 – pomocí impulsní odezvy

- Vygenerovat impulsní odezvu a udělat její frekvenční transformaci.
- impulsní odezva ovšem může být **dloooouuuuhhháááá**

The ultimate solution

z-transformance

Základy

$$x[n] \Rightarrow X(z)$$

$$a x[n] \Rightarrow a X(z)$$

$$x[n-k] \Rightarrow X(z) z^{-k}$$

Diferenční rovnice => přenosová funkce

$$y[n] = b_0x[n] + b_1x[n-1] + \dots + b_Qx[n-Q] \\ - a_1y[n-1] + \dots + a_Py[n-P]$$

Přenosová funkce => frekvenční charakteristika

$$Z \Rightarrow e^{j2\pi f}$$

- f je normovaná vzorkovací frekvencí
- Vyjde komplexní číslo, z toho se musí vzít jen absolutní hodnota
- Matlab: Ručně nebo pomocí `freqz`

Jak je to s časem a se spektry ?

$$y[n] = x[n] * h[n] \quad (\text{konvoluce})$$

$$Y(f) = X(f) H(f) \quad (\text{násobení})$$

- Pozor, ve spektrech při násobení musíme ponechat komplexní čísla.

Summary

Popis filtru

- Schéma
- Diferenční rovnice
- Impulsní odezva
- Přenosová funkce
- Frekvenční charakteristika

Summary II.

- Typy filtrů
 - FIR
 - IIR
- Implementace
 - Násobení
 - Sčítání
 - Posun
 - Off-line
 - On-line

Summary III.

- Výpočet frekvenční charakteristiky
 - Hnusně:
 - „měření“
 - Frekvenční analýza $h[n]$
 - Čistě
 - Přenosová funkce – záměna z za $e^{j2\pi f}$
 - f je normovaná frekvence

Zbývá

- Fáze – posuny signálů
- Stabilita u složitějších IIR filtrů
- Návrhy filtrů
- Učení filtrů na datech



The END