

Dokumentace k projektu

ISS/VSG Projekt 2021/22

Obsah

1	Úvod	3
2	Otázky	3
2.1	Otázka 1	3
2.2	Otázka 2	3
2.3	Otázka 3	4
2.4	Otázka 4	5
2.5	Otázka 5	6
2.6	Otázka 6	6
2.7	Otázka 7.3	7
2.8	Otázka 8	8
2.9	Otázka 9	8
2.10	Otázka 10	10
3	Nestandardní zdroje	10

1 Úvod

Cílem projektu bylo analyzovat zadaný signál a nalézt čtyři harmonicky vztažené cosinusovky, které do něj byli přidány. Poté vytvořit filtr/y a ten/ty následně použít pro filtraci daného signálu. Tento projekt je řešen v jazyce Python. K výpočtům dopomohly funkce z knihoven *numpy* a *scipy*, pro tvorbu grafů byla využita knihovna *matplotlib*.

2 Otázky

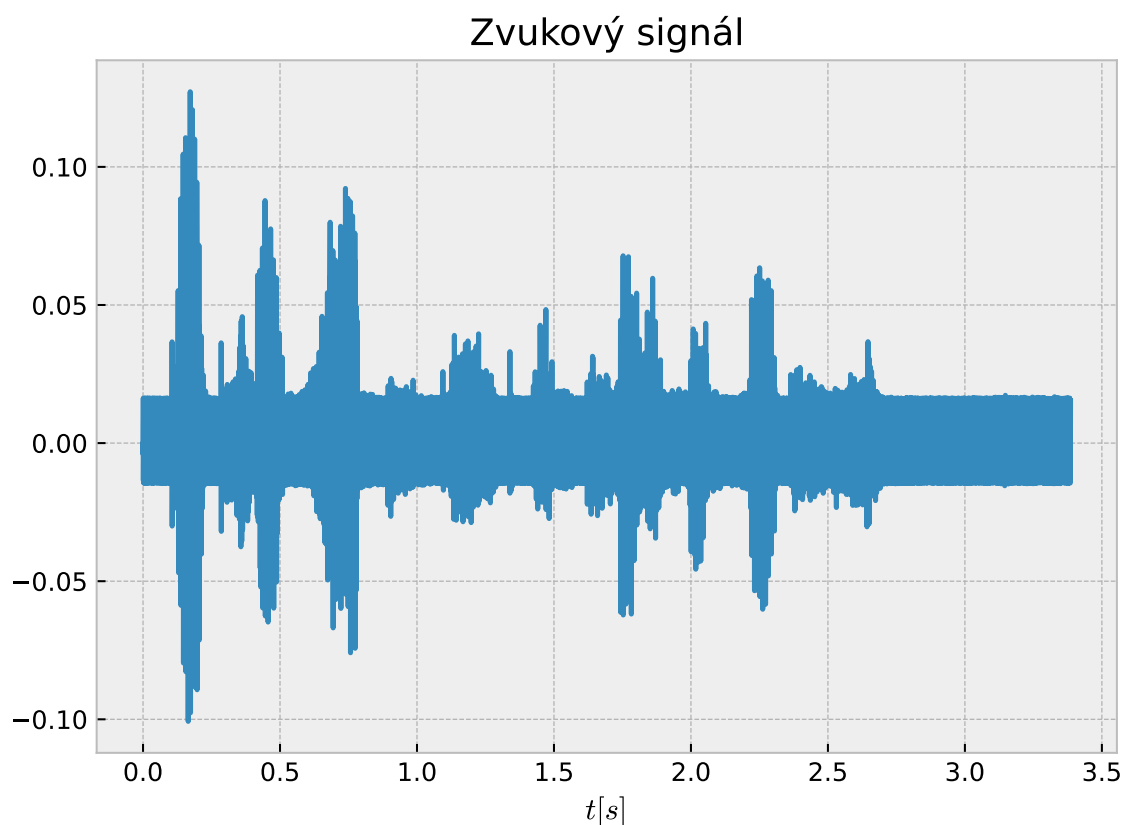
2.1 Otázka 1

Délka signálu:

- ve vzorcích: 54170
- v sekundách: 3.385625 sekund

Minimální hodnota: -0.100738525390625

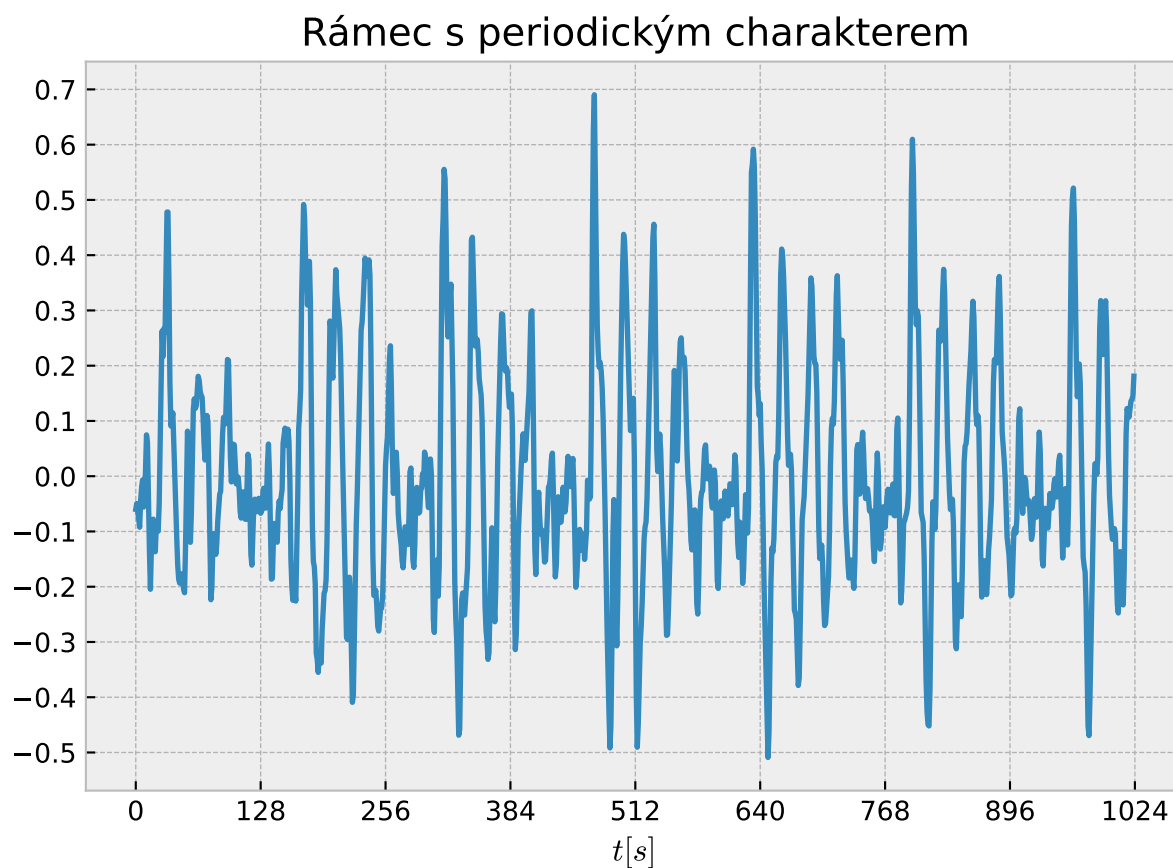
Maximální hodnota: 0.127197265625



Pro získání údajů byl využit modul *wavfile* s postupem dle Jupyter notebooku.

2.2 Otázka 2

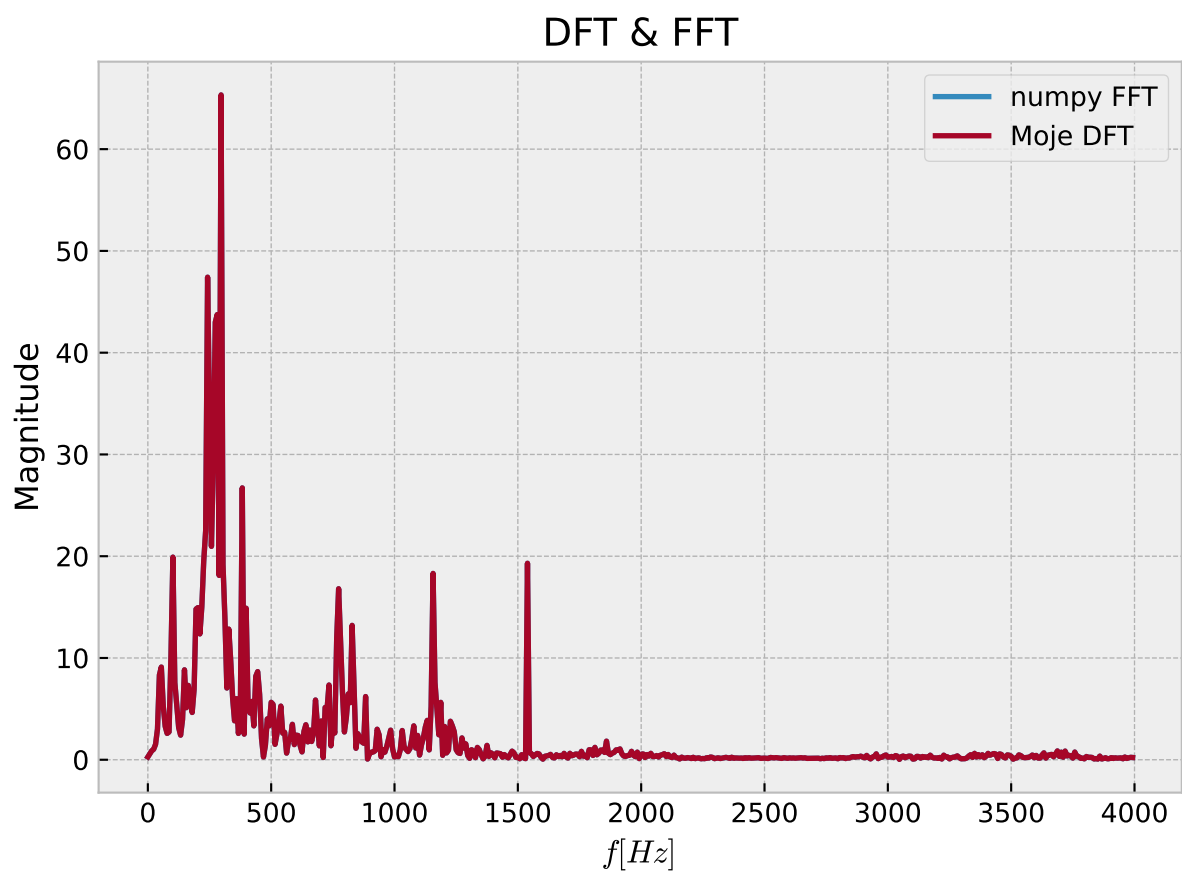
Dle zadání byl signál ustředněn a normalizován. Do řádků matice byly postupně uloženy jednotlivé úseky pomocí jednoduchého while cyklu. Celá matice byla nakonec transponována. Pro zobrazení



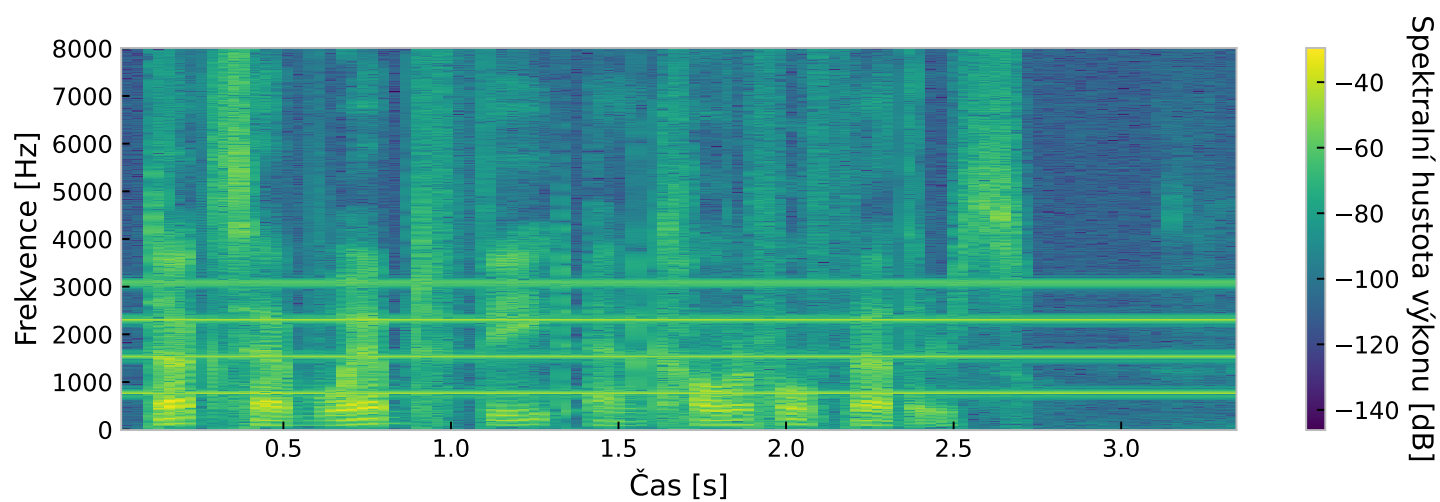
byl vybrán 14. rámec.

2.3 Otázka 3

DFT funguje správně, o čemž se lze přesvědčit srovnáním s *numpy* FFT. Bohužel z důvodu implementace (implementováno vnořenými while cykly), trvá DFT dlouho. Při snaze o před počítání části výrazu před while cykly docházelo k transformaci z komplexního čísla s čistě imaginární složkou na komplexní číslo s reálnou a imaginární složkou. Po neschopnosti opravy tohoto problému byl proto zvolen tento jednodušší, byť značně pomalejší způsob.



2.4 Otázka 4



Pro postup při tvorbě spektrogramu byl využit postupem z Jupyter notebooku.

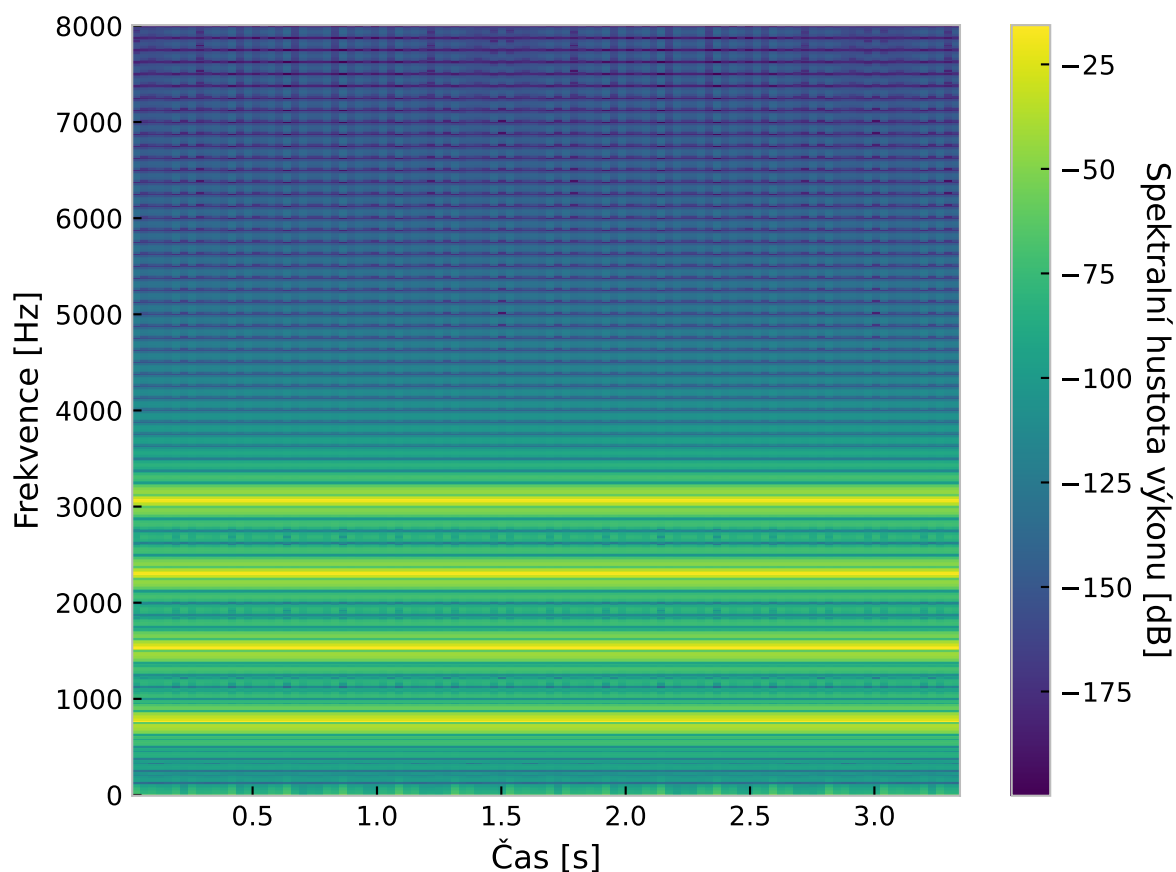
2.5 Otázka 5

Rušivé frekvence:

- frekvence 1: 650 Hz
- frekvence 2: 1300 Hz
- frekvence 3: 1950 Hz
- frekvence 4: 2600 Hz

Frekvence byly nalezeny ručně vytvořením spektrogramu s menšími kroky na frekvenční ose a přidáním dobře viditelné mřížky na oné ose.

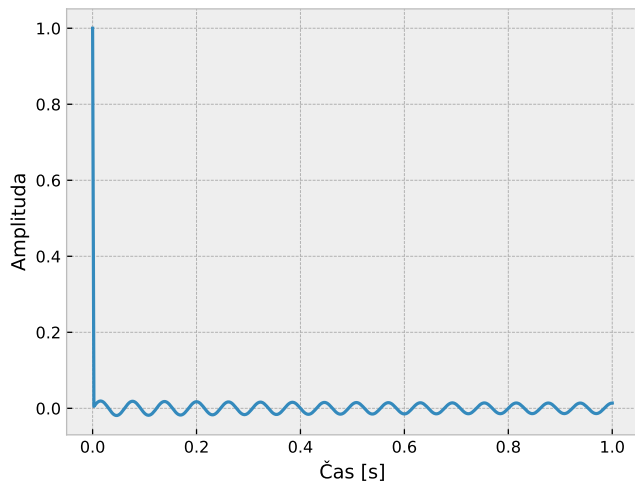
2.6 Otázka 6



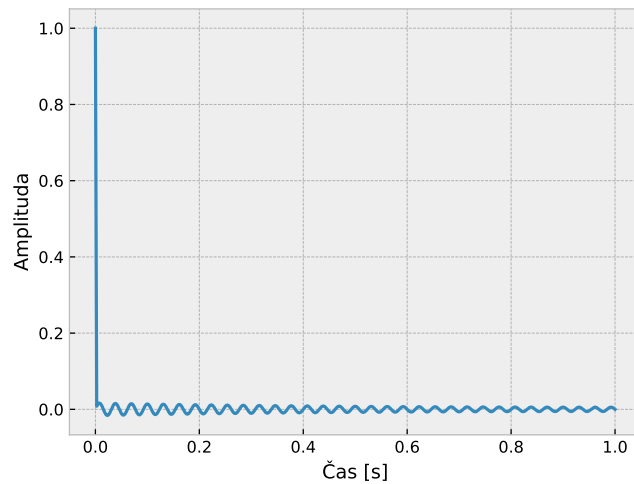
Po vygenerování cosinusovek s frekvencemi uvedenými v otázce 5, jejich poslechu a následné panice, bylo zjištěno, že při vygenerování cosinusovek o čtyřnásobné frekvenci vzniknou požadované cosinusovky. Tento signál zní a vypadá stejně, pouze je hlasitější. Pro vygenerování cosinusovek byla využita funkce *cos* z knihovny *numpy*. Pro jejich zápis byl opět využit modul *wavfile*.

2.7 Otázka 7.3

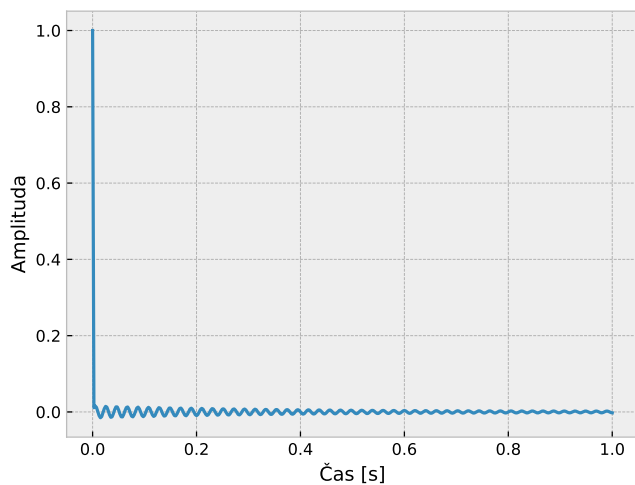
Impulzní odezvy filtrů:



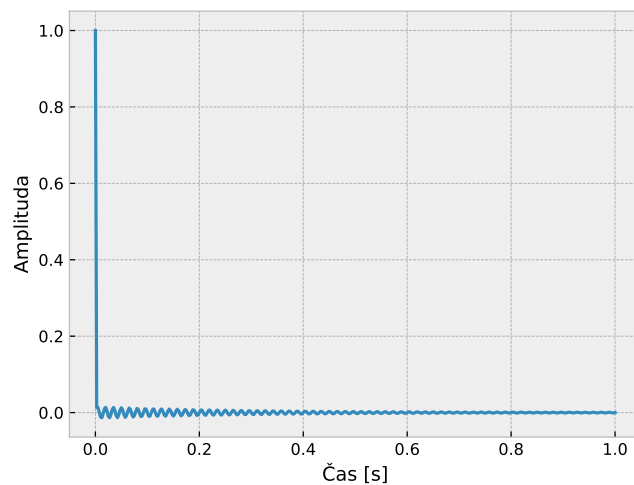
Obrázek 1: Filtr 1



Obrázek 2: Filtr 2



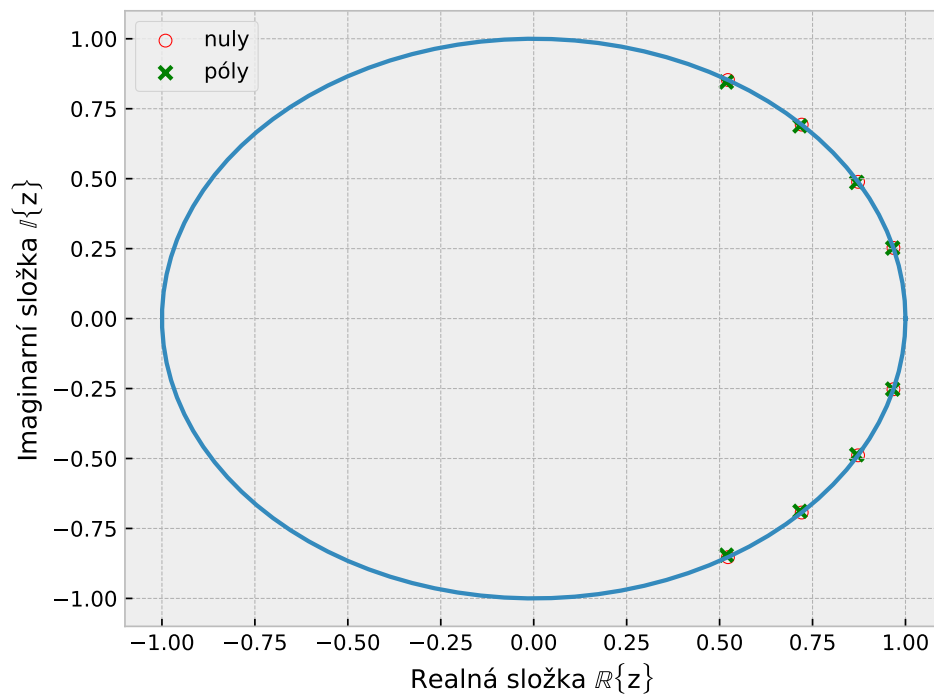
Obrázek 3: Filtr 3



Obrázek 4: Filtr 4

Po neúspěšných pokusech o vytvoření filtrů pomocí funkce *butter* z knihovny *signal* byla pro tvorbu filtrů využita funkce *iirnotch* z téže knihovny.

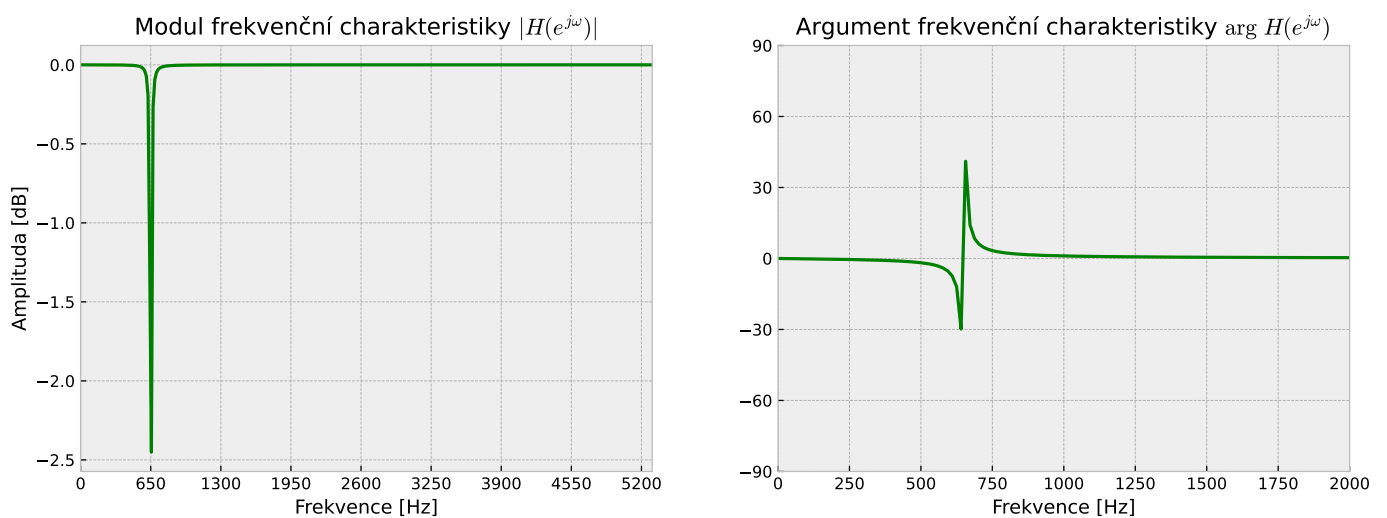
2.8 Otázka 8



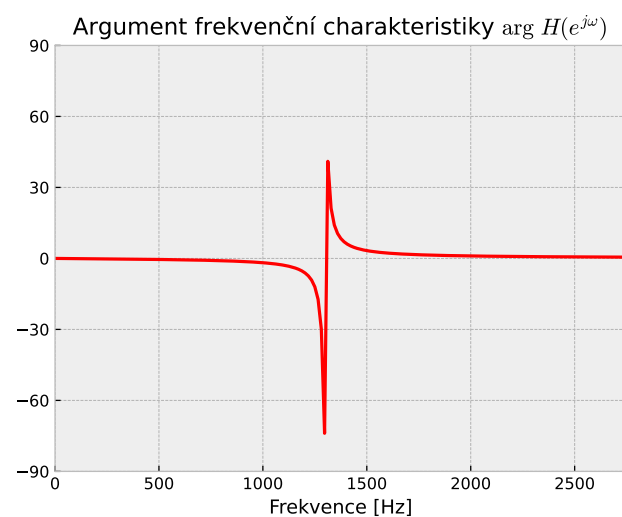
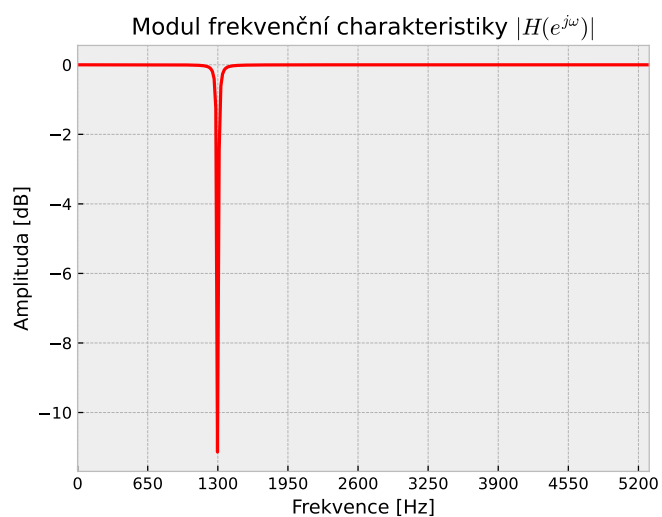
U této otázky byl opět využit postup z Jupyter notebooku.

2.9 Otázka 9

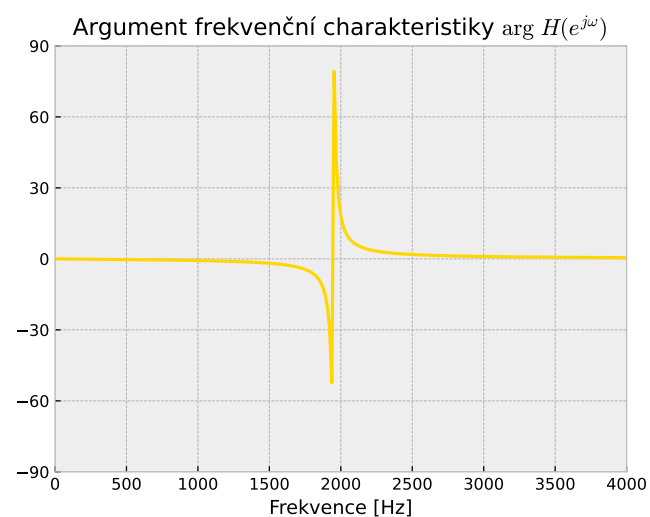
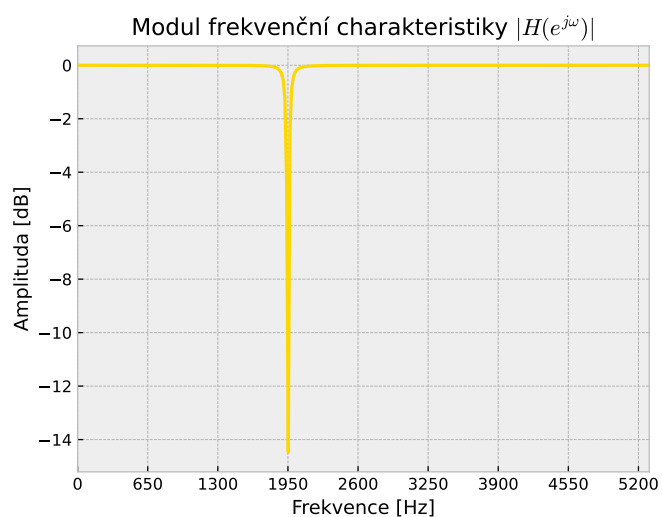
Frekvenční charakteristiky byly vytvořeny s pomocí oficiální *scipy* stránky pro funkci *irnotch*.



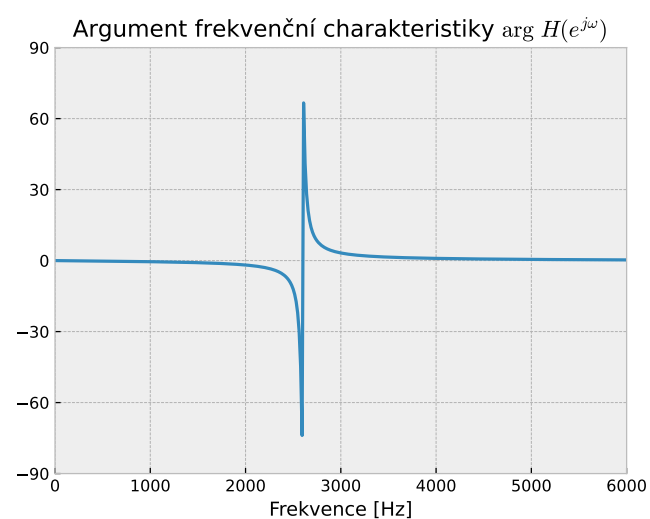
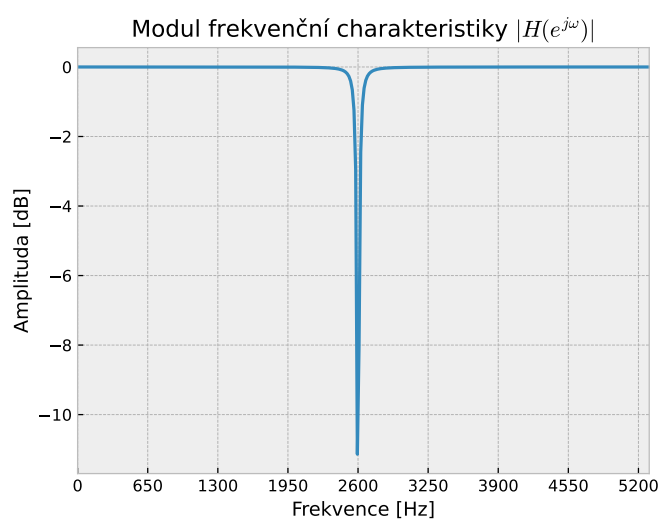
Obrázek 5: Filtř 1



Obrázek 6: Filtr 2



Obrázek 7: Filtr 3



Obrázek 8: Filtr 4

2.10 Otázka 10

Filtrace signálu byla provedena pomocí funkce *filtfilt* a filtru připraveného v předchozí části. Uloženy jsou znovu pomocí modulu *wavfile*. Filtr však nesplňuje očekávané chování.

3 Nestandardní zdroje

V projektu byly mimo standardních stránek využity také stránky pro knihovnu *numpy*, dále byly pro projekt využity oficiální *scipy* stránky, zejména pak stránky pro funkce *iirnotch*, *freqz*, *filtfilt*.