

Datum: 15.2.2024	Střední průmyslová škola, Chomutov, Školní 50, příspěvková organizace	Třída: A4
Číslo úlohy: 17.	Programování AMC – digitální filtr (Keysight VEE)	Jméno: T. Kubanek

Zadání:
Realizujte digitální filtr typu IIR (infinite impulse response) dolní propust v programu Keysight VEE, který ze sinusového signálu o frekvenci 800 Hz odstraní nežádoucí šum.

Schéma zapojení:

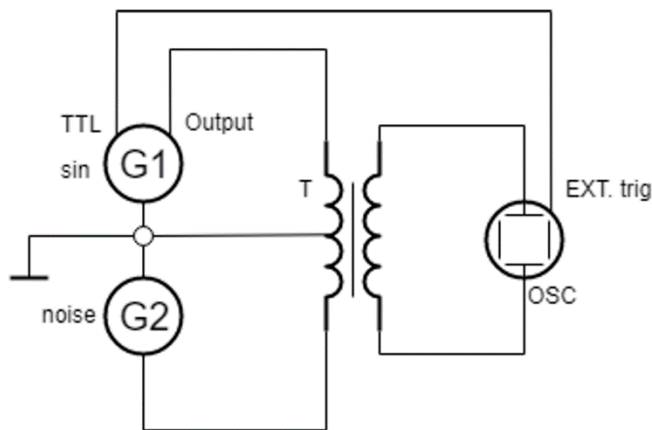
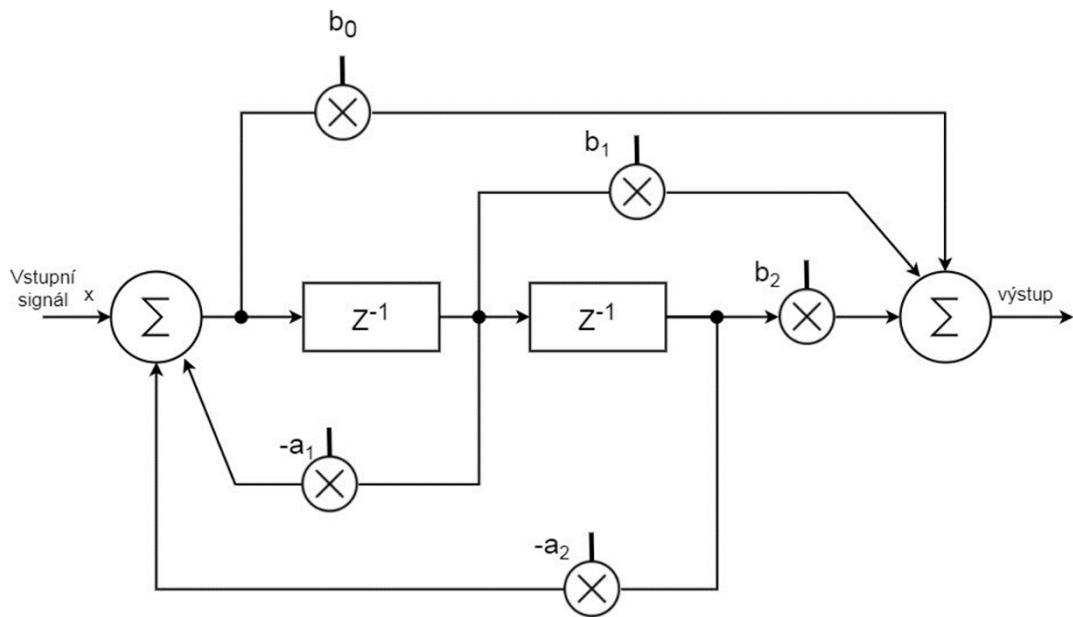


Schéma filtru s nekonečnou impulzní odezvou:



Použité přístroje:

Název přístroje:	Označení:	Údaje:	Inv. číslo:
Funkční generátor	G1	MTX 3240	LE3 5024
Funkční generátor	G2	33120A	LE3 104
Transformátor	T	-	-
Osciloskop	OSC	Keysight EDUX10526 50MHz 1GSa/s	LE 5125

Postup:

1. Obvod jsme zapojili dle schématu.
2. Na MTX 3240 jsme nastavili frekvenci 800 Hz.
3. Zapnuli jsme generátor 33120 A.
4. Otevřeli jsme program VEE, kde jsme postupovali dle zadání (postup viz výpis programu).
5. Vypočetli jsme si frekvence dle zadání:
 - $f_p=1200\text{Hz}$
 - $f_s=1400\text{Hz}$
6. Program jsme sestrojili pro filtr 6. řádu – vycházeli jsme ze schématu filtru s nekonečnou impulzní odezvou (viz schéma).
7. Program jsme odladili a odstranili šum.
8. Výsledky jsme zpracovali.

Výpis programu:

Viz. poslední strana.

Komentář k programu:

1.: Nastavení Driver osciloskopu:

- CH1=1 V
- TIMEBASE = 1 ms
- Nastavení ext. Triggeru
- Waveform panel – WAV points 2000 (nastavit počet vzorků na 2000)
- OUTPUT WF_ch1 (nastavení výstupu)

2.: Formule: Vybrání vzorků od 1 do 2000.

3.: Formule: Ze vzorků a času jsme si vypočetli vzorkovací frekvenci.

4.: Build Waveform: rozloží nám data z osciloskopu a vloží nám vzorky do pole.

5.: XY Trace: znázornění průběhu v závislosti na čase (signál + šum).

6.: Real64: 0 pro zapsání do pole frekvencí.

7.: Real64: Řád filtru.

8.: Formule: vydělili jsme f_p polovinou vzorkovací frekvence (znormováno).

9.: Formule: vydělili jsme f_s polovinou vzorkovací frekvence (znormováno).

10.: Real64: 1 pro zapsání do pole frekvencí.

11.: Real64Array: pole vektorů amplitud.

12.: Concatenator: shromáždili jsme všechny frekvence a vložili je do pole.

13.: MATLAB Script YULEWALK: matlabová funkce pro výpočet koeficientů a, b filtru.

14.: For Range: smyčka pro indexování vzorků.

15.: Formule: součet vzorku (s aktuálním indexem z For Range) s výsledkem formule, kde dochází k rozdílu součinů koeficientů b s hodnotami v posuvném registru.

16.: Real64: 0, slouží jen pro aktivaci programu, bez ní by se program nerozjel, jelikož by nemohl nijak začít.

17.: Uzel: pro hodnotu z formule nebo 0 (v případě počátku programu, kdy by formule nedávala žádný výstup).

18.: Shift register: posuvný registr, do kterého zapisujeme součet vzorků a koeficientů a.

19.: Formule: pro výpočet rozdílu součinů hodnot z posuvného registru a koeficientů a (index 1-6 viz schéma), výpočet slouží jako „zpětná vazba“. Výsledek se sčítá se vzorky, které následně plní posuvný registr.

20.: Formule: pro výpočet součtu součinů hodnot z posuvného registru a koeficientů b (index 0-6). Jeho výstup udává výsledný vyfiltrovaný sinusový signál. Jeho vstupem je registr X0 (výsledek formule 14) a nadále posuvný registr a koeficienty b.

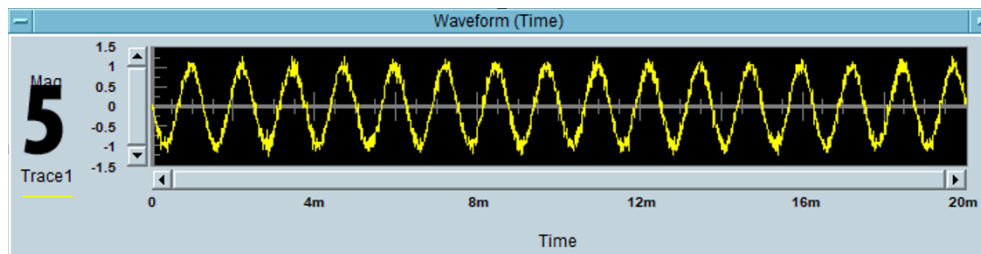
21.: Collector: shromáždění dat výsledků z formule 19 do pole.

22.: Build Waveform: slouží nám vypočtené vzorky v čase.

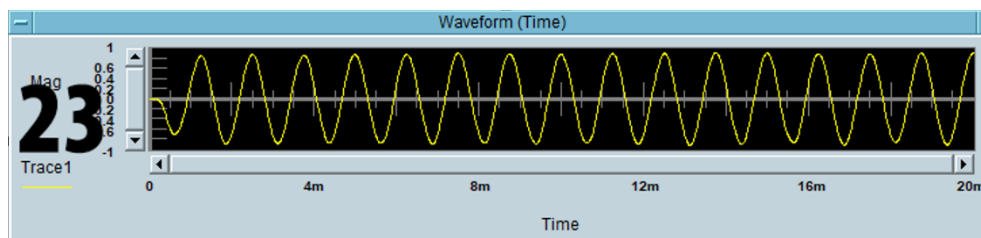
23.: XY Trace: zobrazí již vyfiltrovaný signál.

Grafy:

- Znehodnocený signál:



- Vyhlazený signál:



Závěr:

Návrh digitálního filtru je velmi složité a vyžaduje vysokou přesnost a bezchybnost jednotlivých kroků. Důkazem je toho, že pokud bychom změnili pouze jednu hodnotu koeficientů b , tak bychom na výstupu našli odlišný signál než ten, který jsme přivedli. Výsledkem našeho měření je vyfiltrovaný signál, který jsme vyfiltrovali pomocí našeho navrženého filtru 6. řádu, na začátku je viditelná menší chyba našeho filtru, která by v praxi byla zanedbatelná. Tuto chybu nejsme bohužel schopni za našich podmínek kompletně odstranit.

