

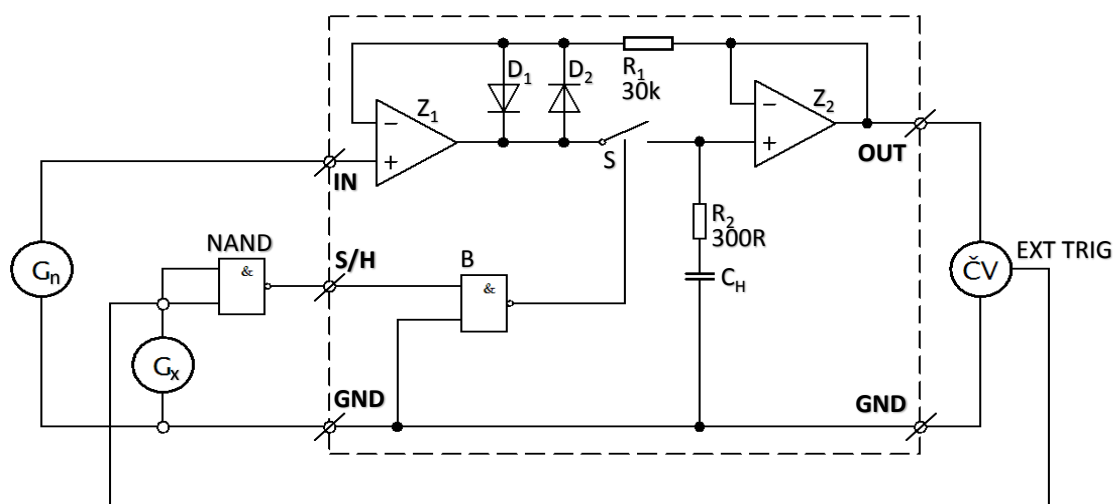
Datum měření:	<b>SPŠ Chomutov</b>	Přijmení:
19.12.2016		THEUMER
Číslo úlohy:	Vzorkovač - rekonstrukce průběhu a FFT analýza	Třída - skupina
12		A4-2

### Zadání:

Sestavte vzorkovač a naprogramujte program ve VEE, který pomocí odebraných vzorků zrekonstruuje průběh generovaný funkčním generátorem a dokáže vypočítat frekvenci i periodu zrekonstruovaného programu. Výsledky programu ověřte svými výpočty.

### Schémat zapojení:

Schéma 1: schéma zapojení vzorkovacího obvodu (LF398)



### Pozn.:

$G_x$  generátor, jehož signál vzorkujeme a rekonstruujeme v počítači

$G_n$  generátor, který nám taktuje odběr vzorků (TTL signál o vzorkovací frekvenci 10 Hz)

Ohraničená část obvodu představuje již sestavený přípravek. Jedná se o vzorkovací zesilovač typu LF398, který je určen k definovanému odběru vzorků vstupního napětí v okamžiku změny jeho řídicího signálu S/H. Obvod je tvořen vstupním zesilovačem  $Z_1$ , spínačem S, řízeným budičem B, externím paměťovým kondenzátorem  $C_H$  a výstupním zesilovačem  $Z_2$ , zapojeným ve funkci sledovače napětí. Záporná napěťová sériová zpětná vazba z výstupu obvodu na jeho vstup zajišťuje ve stavu sledování (sepnutém spínači S) jmenovitý přenos obvodu +1. Antisaturační diody  $D_1$ ,  $D_2$  zamezují přebuzení výstupu operačního zesilovače  $Z_1$  při přechodu obvodu ze stavu sledování do stavu pamatování (rozepnutí spínače S).

## Tabulka použitých přístrojů:

Tabulka 1: tabulka použitých přístrojů

Název	Označení	Údaje	Evidenční číslo
Stabilizovaný zdroj	$U_1$	2x 15 V/1 A, 5 V/1 A, typ: AUL 310	LE4 1045
Funkční generátory	$G_x$	HP33120A, 15 MHz	LE3 100
	$G_n$	METRIX GX240, 2MHz	LE2 57
Multimetr	ČV	HP34401A	LE3 94
Integrovaný obvod	TTL	IO MH7400	—
Vzorkovač	—	vzorkovací zesilovač LF398	LE2 2142

## Teorie:

### Vzorkování signálu:

Z měřeného signálu se odebírají v pravidelných intervalech délky  $T_s$  (index je z ang. sampling, vzorkování) vzorky, čili hodnoty signálu v okamžicích  $n \cdot T_s$ , kde  $n$  je celé kladné číslo. Tyto hodnoty se následně převedou na čísla, uloží do paměti a jsou určeny k číslicovému zpracování nebo k zobrazení. Původní analogový signál z nich můžeme rekonstruovat, byla-li dodržena tzv. vzorkovací věta: **Je-li vzorkován frekvenčně omezený signál s horní mezní frekvencí  $f_m$  v ekvidistantních okamžicích s periodou  $T_s = 1/f_s$ , pak můžeme původní signál získat zpět bez ztráty informace, jeli splněna podmínka  $f_s > 2 \cdot f_m$ .** Pokud dojde k porušení podmínky, nemůže rekonstrukce proběhnout správně a vznikne tzv. aliasing (zdeformovaný průběh).

### Kvantování signálu:

Analogově číslicový převodník (analog-to-digital converter) je obvod převádějící hodnotu napětí na svém vstupu na odpovídající číslo. Výstup z analogově-číslcového převodníku je číselný kód (zpravidla  $n$ -bitový binární nebo binárně-dekadický). Je ukládán do paměti a následně využit pro zobrazení v grafické podobě (číslcový osciloskop), případně před zobrazením zpracován (spektrální analyzátor, analyzátor výkonu). Pokud je použit přímo k číslicovému zobrazení výsledku měření na zobrazovací přístroje (číslcový voltmetr), je číslo zobrazeno v dekadické číselné soustavě a je mu přiřazena desetinná tečka odpovídající použitému vstupnímu rozsahu voltmetru.

### Měření základních parametrů periodických signálů:

Digitalizací periodického napětí získáme posloupnosti  $u_n$  nebo  $i_n$  pro  $n \in (0, n-1)$ . Pro posloupnost vzorků měřeného signálu použijeme společné označení  $x_n$ . Vzorkovací frekvenci upravíme tak, že sejmeme právě  $N$  vzorků za periodu signálu  $T$ . Upravíme definice efektivní hodnoty, stejnosměrné složky a střední aritmetické hodnoty tak, že integrály nahradíme součty dílčích integrálů přes sousední vzorkovací intervaly  $T_s$ . Pro efektivní hodnotu tak dostaneme výraz:

$$X_{ef} = \sqrt{\frac{1}{N} \sum_{n=0}^{N-1} \frac{1}{T_s} \int_{t_n}^{t_{n+1}} x^2(t) dt}$$

Pokud nahradíme měřenou veličinu  $x(t)$  tzv. schodkovou aproximací, čili předpokládáme konstantní hodnotu signálu rovnou amplitudě vzorku až do okamžiku odebrání následujícího vzorku, platí pro efektivní hodnotu výraz:

$$X_{ef} = \sqrt{\frac{1}{N} \sum_{n=0}^{N-1} x_n^2}$$

-pro stejnosměrnou složku platí:

$$X_{ef} = \frac{1}{N} \sum_{n=0}^{N-1} x_n$$

-pro střední aritm. hodnotu platí:

$$X_{stř} = \frac{1}{N} \sum_{n=0}^{N-1} |x_n|$$

Přesnější určení uvedených parametrů umožňuje použití lineární interpolace, při které spojíme jednotlivé sousední vzorky úsečkami.

#### Měření frekvenčního spektra:

Ovzorkujeme-li periodický měřený signál  $N$  vzorky/ $T$ , můžeme vypočítat koeficienty Fourierovy řady-spektrum signálu pomocí  $N$ -bodové diskrétní Fourierovy transformace (DFT).

$$x(k) = \frac{1}{N} \sum_{n=0}^{N-1} x(n) \cdot e^{-j\frac{2\pi}{N} \cdot n \cdot k}$$

kde  $k$  je řád harmonické (tj.  $f_k = f_1 \cdot k$ ),  $n$  je celé číslo (pořadí vzorku),  $N$  je počet vzorků za periodu signálu  $x(t)$ .  $N$  musí být vyjádřitelné jako celočíselná mocnina dvěma. Pro zvýšení rozlišitelnosti zobrazení FFT lze doplnit naměřené vzorky řadou nulových hodnot (zero padding). Výpočtem se získá  $N$  spektrálních čar představujících vzorky Fourierovy transformace vzorkovaného signálu ležícího v pásmu od 0 až po vzorkovací frekvenci.

Výpočet DFT se v praxi provádí pomocí algoritmů nazývaných souhrnně rychlá Fourierova transformace (FFT), které umožňují podstatně zkrátit dobu výpočtu. Tento způsob určení frekvenčního spektra se používá v přístrojích nazývaných FFT spektrální analyzátory.

#### **Postup:**

1. generátor  $G_x$  nám taktuje vzorkování, nastavíme TTL signál o vzorkovací frekvenci 10 Hz
2. generátor  $G_n$  budeme ovládat pomocí programu z počítače
3. voltmetr bude mít nastavenou funkci DC, integrační dobu (aperture) na 0,02 PLC a bude nastaven externí trigger (menu Trigger options > Trigger source = external, Trigger Count = 100 (počet spuštění) a Sample counts = 1) > to vše v ovladači voltmetru

#### **Konkrétní algoritmický popis práce programu je popsán níže ...**

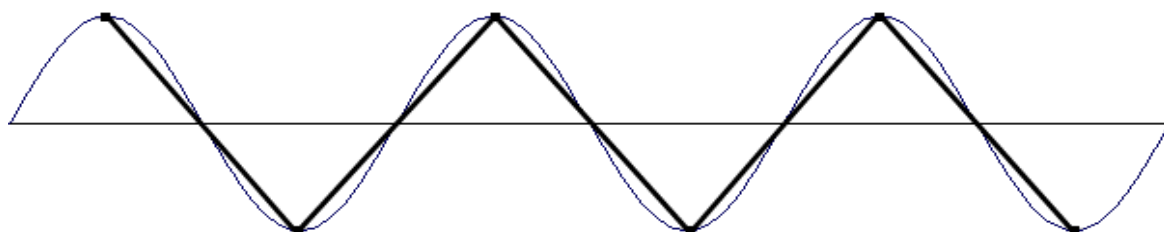
Je známo, že při nedodržení vzorkovací věty vznikne falešný obraz rekonstruovaného průběhu, který nazýváme aliasing.

## Aliasing:

Nedodržíme-li pravidlo, které udává vzorkovací věta, dojde k aliasingu, tedy jevu, kdy je zobrazen falešný obraz reálného průběhu. Aliasingový obraz průběhu může mít buď stejný tvar s odlišnou frekvencí (obvykle mnohem menší) nebo je zcela jiného, obvykle na první pohled zvláštního nepravidelného tvaru, který může připomínat šum. Vzorkovací pravidlo udává (Nyquistův teorém), že vzorkovací frekvence musí být vyšší, než je dvojnásobek nejvyšší frekvence harmonického průběhu, kterou měřené spektrum obsahuje.

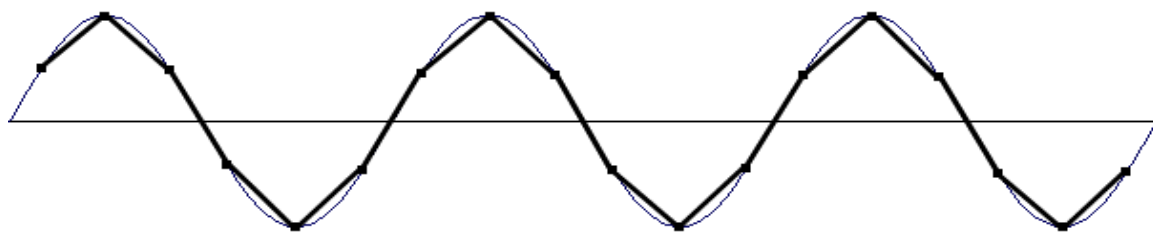
Na obrázku 1 je vidět co nastane, když bude vzorkovací frekvence 10 Hz a frekvence signálu přesně 5 Hz. Vzorkovací frekvence je tedy přesně 2x větší než rekonstruovaný průběh a jak je vidět, z funkce sinus vznikl trojúhelník.

*Obrázek 1: vzorkování sin. signálu se vzorkovací frekvencí 2x větší než frekvence průběhu*



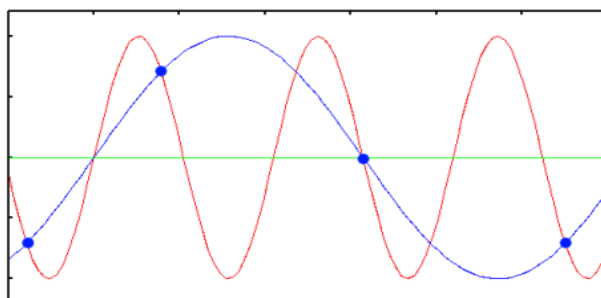
Na obrázku 2 je situace, kdy je sice vzorkovací frekvence větší, než dvojnásobek frekvence průběhu, nicméně jen o trochu. Rýsuje se tedy již požadovaná funkce sinus, ale obraz je stále zašpičatělý. I toto není ještě v toleranci.

*Obrázek 2: vzorkovací frekvence o trochu vyšší než dvojnásobek frekvence průběhu*



Poslední obrázek ukazuje stav, kdy je vzorkovací frekvence mnohem menší, než dvojnásobek frekvence signálu. Buď vznikne stejný průběh o jiné frekvenci, nebo úplně jiný průběh.

*Obrázek 3: vzorkovací frekvence mnohem nižší než dvojnásobek frekvence signálu*



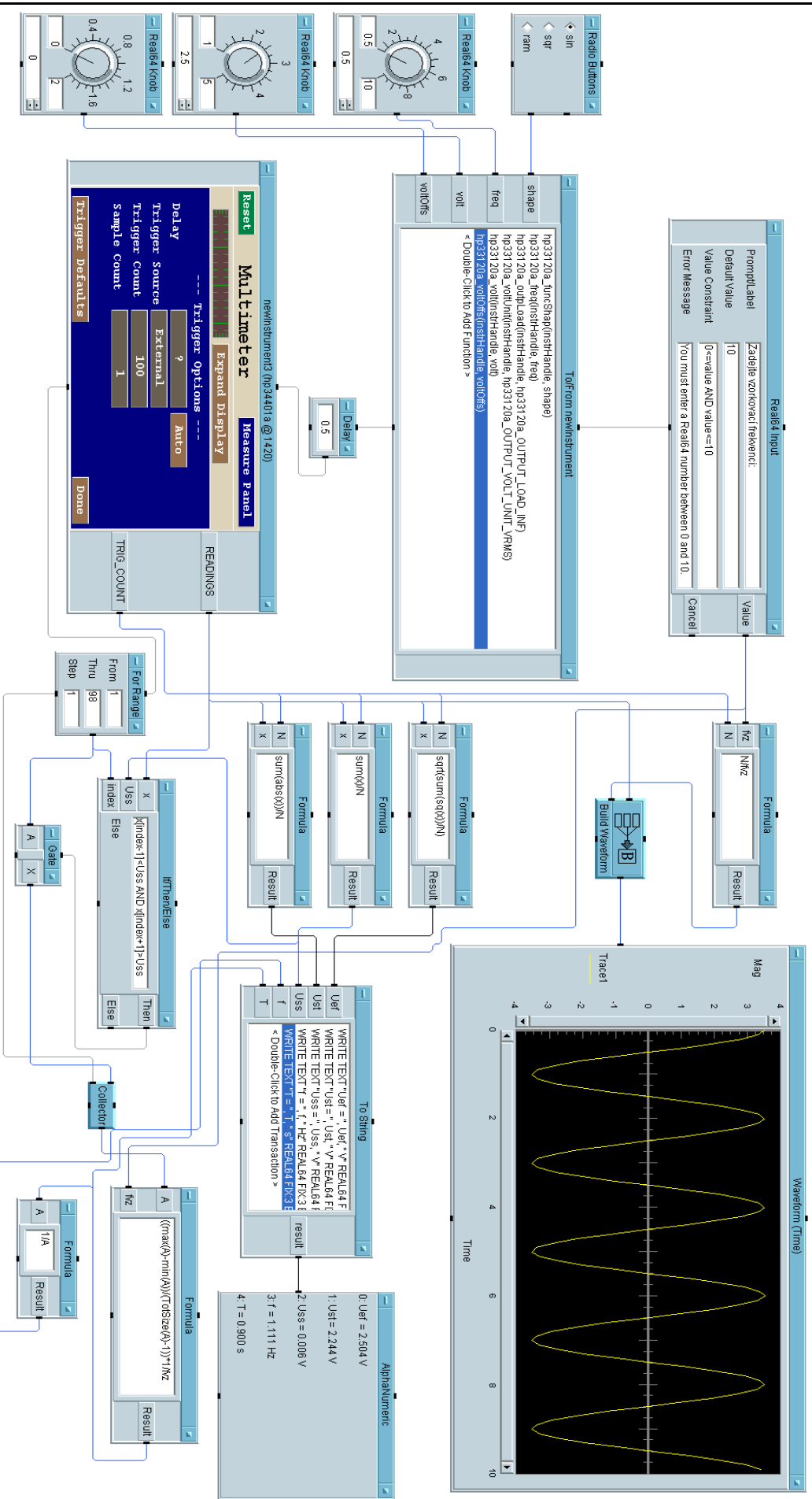
## Program:

1. V komponentě To/From Generator nastavíme parametry funkčního generátoru. Určíme mu, že hodnota napětí mu bude dávana v efektivní hodnotě a že bude pracovat do vysoké impedance. Ostatní parametry > napětí, frekvence, tvar a offset nastavíme tak, abychom je mohli libovolně zadávat, tedy nejsou pevně nastavené.
2. Hodnotu napětí, offsetu a frekvence budeme zadávat komponentou Real64Knob a tvar budeme zadávat komponentou RadioButtons. Výstupy těchto komponent přivádíme na vstupy To/From Generatoru.
3. Komponenta Real64Input nám udává vstupní informaci v podobě vzorkovací frekvence. Je možno jí měnit, ale pro naše pokusy budeme potvrzovat stále stejnou, a to 10 Hz.
4. Po tom, co se odešle příkaz pro nastavení generátoru, necháme zpoždění 0,5 s kvůli dynamice přístroje. Po tomto zpoždění nastavený voltmetr (viz. Postup) odečte hodnotu napětí a ke vzorku uloží jeho pořadí. Obě hodnoty jsou k dispozici na výstupech READINGS (vzorek) a TRIG\_COUNT (pořadí vzorku).
5. Hodnota vzorkovací frekvence 10 Hz z komponenty Real64Input je přiváděna do formule, kde je výpočtem počet vzorků N děleno vzorkovací frekvencí získán časový údaj pro osu x. Následně je tento časový údaj společně s daným vzorkem (příslušícím danému pořadí) přiveden do komponenty Build WaveForm, kde jsou data shromažďována. Okamžitě po doběhnutí vzorkování je z těchto kompletních nasbíraných dat v komponentě Waveform zobrazen průběh jako graf s popsány osami.
6. V dalším kroku je pomocí třech formulí proveden výpočet efektivní a střední hodnoty a stejnosměrné složky signálu. Vstupem těchto formulí je vzorek a jeho pořadí, tedy výstupní data z voltmetru. Tři vypočítané hodnoty jsou přiváděny do komponenty ToString, kde jsou formátovány s textem pro výpis do tabulky.
7. Další na řadě je komponenta ForRange, která generuje čísla od 1 do 98 po každém odebrání vzorku. V každém kroku je hodnota inkrementována a je přiváděna na komponentu If/Then/Else, kam je zároveň přiveden z voltmetru odebraný vzorek i jeho pořadí. Tato komponenta provede test, zda je poslední vzorek menší než stejnosměrná složka signálu a následný vzorek větší než stejnosměrná složka signálu. Pokud ne, nic nedělá a pokud ano, otevře komponentu Gate, která propustí aktuální pořadovou hodnotu vzorku a uloží jí do komponenty Colector, kde jsou tyto propuštěné hodnoty ukládány.  
*pozn.: tyto pořadové hodnoty, které jsou Gatem propuštěny indikují stav, kdy signál mění svou polaritu (s ohledem na offset), což bude stěžejní pro výpočet frekvence a periody*
8. Data nashromážděná v Colectoru (viz bod 7) jsou nyní ve formuli zpracovávána a výpočtem je určena jedna perioda signálu (nutno zjistit rozdíl mezi dvěma uloženými pořadovými hodnotami). Z této vypočítané periody je v dalším Colectoru vypočítána frekvence. Obě tyto vypočítané hodnoty jsou přivedeny do komponenty ToString a spolu s efektivní, střední hodnotou a hodnotou stejnosměrné složky jsou vypsané do tabulky.

Tabulka 2: tabulka hodnot vypočítaných počítačem v porovnání s nastavenými parametry

NASTAVENÍ GENERÁTORU				VÝPOČET PROGRAMU VEE				
tvar	$U_{ef}$ [V]	$f$ [Hz]	$U_{of}$ [V]	$U_{ef}$ [V]	$U_{st}$ [V]	$U_0$ [V]	$f$ [Hz]	$T$ [s]
sin	1	0,5	0	1,001	0,901	0,000	0,500	2,000
sqr	1	0,5	0	1,000	1,000	-0,002	0,500	2,000
ram	1	0,5	0	1,007	0,876	-0,001	0,506	1,975

Obrázek 4: program zkonstruovaný v softwaru VEE Pro



**Závěr:**

Náplní práce byla v podstatě simulace digitálního osciloskopu. Během měření jsme pochopili základní problematiku vzorkování signálů a uvědomili si základní souvislosti týkající se vzorkovací frekvence. Naučili jsme se předcházet aliasingu díky znalosti Nyquistova teorému. Z naměřených hodnot je patrné, že program dokázal vypočítat hodnoty přesně na tisíciny, a zároveň je možno říci, že TFT analýza signálu byla úspěšná, protože jsme se programovým výpočtem dostali ke stejným cifrám, které jsme reálně nastavili. Lze si povšimnout, že nejhorší přesnost byla dosažena při výpočtech u trojúhelníkového signálu. I tak se zdaleka tyto výsledky vejdou do tolerance.