

Spektrální analýza zvuku

aneb realtime FFT na AVR

bkralik
Hradec Králové, 2012

Úvod:

Projekt zprovoznění Fourierovy transformace jsem měl v plánu již delší čas. Zajímám se o matematiku, fyziku a elektroniku, takže se mi líbila možnost zkombinovat všechny uvedené obory do jednoho a naučit se při tom něco nového.

Teorie:

Fourierova transformace je integrální transformace definovaná vztahem

$$S(\omega) = \int_{-\infty}^{\infty} s(t) e^{-i\omega t} dt$$

a slouží pro převod periodického signálu (funkce) z časové do frekvenční oblasti.

Fourierova transformace může být jak ve spojitém tak v diskrétním čase. Protože data z A/D převodníku jsou navzorkována právě v diskrétních časových intervalech, použijeme tuto variantu, která je definována vztahem pro k-tý člen

$$X_k = \sum_{n=0}^{N-1} x_n \cdot e^{-i2\pi \frac{k}{N} n} \quad k \in 0, \dots, N-1$$

X_k je k-tý člen výsledného frekvenčního spektra, N je počet vzorků a x_n je vzorek číslo n . X_k a x_n jsou komplexní čísla, pro naši potřebu je imaginární složka x_n rovna nule a na display vypisujeme absolutní hodnotu X_k .

Protože však má diskrétní Fourierova transformace časovou složitost $O(n^2)$, je velmi těžké ji přímo implementovat na mikroprocesoru. Naštěstí pro nás, v roce 1956 byl publikován algoritmus rychlé Fourierovy transformace Cooley-Turkey, který snižuje časovou složitost na $n \cdot \log_2(n)$, což už umožňuje použít pro zpracování prakticky v reálném čase i levný mikrokontrolér jako je právě řada procesorů Atmel AVR Atmega.

Algoritmus Cooley-Turkey pracuje na principu „rozděl a panuj“, při kterém dělí existující DFT na 2 menší a tím i časově jednodušší DFT. Samozřejmě, tento postup můžeme opakovat a dosáhnout tak rozložení na mnoho malých součinů s tzv. twiddle faktory. Díky tomu ušetříme velmi mnoho času a můžeme tak provozovat FT pro velký počet výstupních frekvencí.

Samozřejmě, i FFT má svoje nevýhody. První nevýhoda plyne již z její definice – stupeň transformace musí být vždy mocninou čísla 2, což je pro použití v audiotechnice velmi nevýhodné. Člověk slyší „logaritmicky“ a proto by bylo lepší mít X i Y osu s logaritmickými měřítky, to však FFT neumožňuje.

Další problém nastane ve chvíli, kdy si uvědomíme, že Fourierova transformace se dá použít pouze pro periodické signály, což povětšinou hudba není. Naštěstí, lze použít tzv. okna, což jsou funkce, kterými se sada vzorků pronásobí a zajistí se tak periodičnost vzorku. Bohužel jsou tu opět nevýhody – každé okno má i svoji vlastní parazitní frekvenci, která se poté negativně projeví ve výstupním spektru.

Implementace algoritmu:

Protože už samotné pochopení rychlé fourierovy transformace není zrovna jednoduché, rozhodl jsem se, že použiji pro výpočet již hotový algoritmus, kterých se po internetu nalézají již dost (zdá se mi zbytečné znovu vynalézat kolo). Rozhodl jsem se pro knihovnu napsanou v Assembleru ze stránek elm-chan.org, kde ji publikoval koncem roku 2005 ChaN.

Knihovna vyniká výbornou rychlostí, zabudovaným Hammingovým oknem a jednoduchostí použití. Skládá se ze tří funkcí, `fft_input()`, `fft_execute()`, `fft_output()`. Funkce `fft_input` připraví hodnoty do vstupního bufferu, vynásobí je Hammingovým oknem a převede je do komplexních čísel. Převod je pouze formální, imaginární část je rovna nule. Funkce `fft_execute` provede vlastní „butterfly operace“, při kterých se hodnoty prokříží podle schématu a pronásobí se s twiddle faktory. Funkce `fft_output` seřadí hodnoty správně po výpočtu a převede komplexní čísla do reálných.

Pomocné rutiny:

V programu jsou použity další pomocné rutiny, zejména knihovna pro vykreslování na displayi a funkce pro navzorkování nových dat. Vše krom FFT knihovny je napsáno v jazyce C, použitému kvůli rychlosti vývoje a srozumitelnosti.

Knihovna pro práci s displayem má základ „vypůjčený“ ze stránek řeckého programátora Vassilise Serasidise, ovšem byla téměř ve všech funkcích upravena, zachovalo se pouze základní časování komunikace s displayem. Knihovna taktéž obsahuje funkci tzv. Slow Decay, sloužící pro lepší znázornění dat na displayi. Zvuk a hudba vůbec se totiž i ve frekvenčním spektru mění velmi rychle a bez zpomalení padání sloupce zpět do nuly by hodnotu mnohdy ani nebylo možné postřehnout.

Funkce pro navzorkování nových dat je celkově velmi jednoduchá, v úvodu se nastaví vstup A/D multiplexeru, referenční napětí, freerunning mod A/D převodníku a pustí se vzorkování. Po navzorkování jsou hodnoty uloženy do bufferu, na který je poté puštěna již zmíněná funkce `fft_input`.

Popis zapojení:

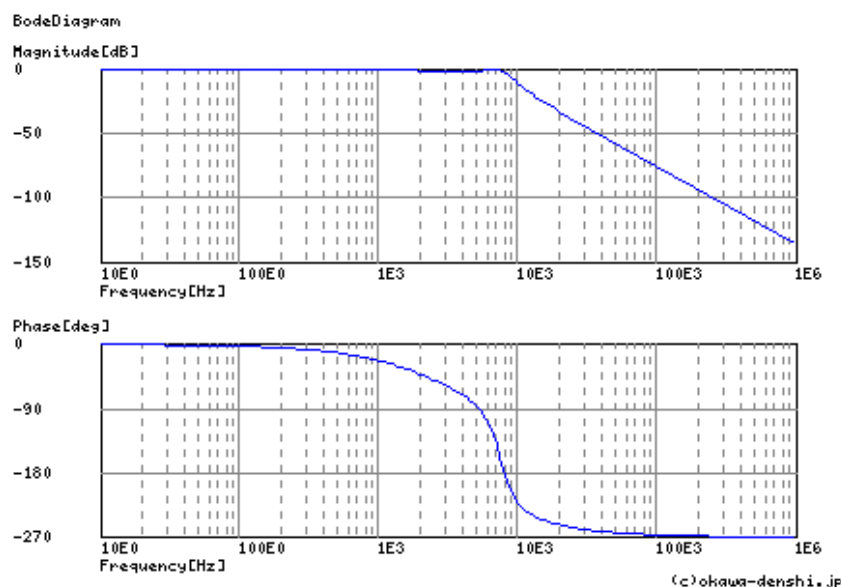
Jádro obvodu tvoří integrovaný procesor Atmega 32, obsahující 32 kB paměti flash a 2 kB paměti RAM. Při taktu 16 Mhz dosahuje výpočetního výkonu 16 mips, což by mělo být pro tuto aplikaci dostatečné.

K vzorkování je použit vnitřní A/D převodník Atmelu. Ten nevyniká sice oslňující kvalitou a je rušen zbytkem procesoru, ale protože se jedná pouze o informační zobrazení spektra, nemá cenu se zabývat externím A/D převodníkem.

Podle Shannonova-Nyquistova-Kotělnikovova teorému je nutné zvolit minimálně dvakrát vyšší vzorkovací frekvenci než bude nejvyšší zobrazovaná frekvence. Protože byla zvolena „kolem“ 8 kHz, je A/D převodník nastaven na 19230 Hz. Nejvyšší zobrazitelná frekvence je poté 9.6 kHz. To sice není moc, ale v daném konci spektra se logaritmická křivka již velmi natahuje a nemá cenu zpracovávat vyšší frekvenci.

Vstupní filtr:

Protože podmínkou úspěšného vzorkování je oříznutí vstupního spektra na max. frekvenci, bylo nutné do obvodu umístit vstupní filtr. Po zkouškách více typů dolních propustí byla nakonec zvolena dolní propust třetího stupně topologie Sallen-Key. Pomocí nástrojů na stránce <http://sim.okawa-denshi.jp/en/Sallen3tool.php> byl vyprojektován filtr s dělicí frekvencí 8 kHz. Vypočtená charakteristika filtru je vidět na následujícím obrázku:



Bylo taktéž nutné zařídit oddělení stejnosměrné složky vstupního napětí, k čemuž slouží vstupní vazební kondenzátor. Aby byl filtr napájen „tvrdým napětím“, byl ještě do obvodu vložen druhý operační zesilovač, poskytující navíc možnost zesílení vstupního signálu 1x-10x.

Ostatní součásti zapojení:

Obvod je doplněn standartním stabilizátorem 7805, LCD displayem ATM12864D o rozměrech 128*64 bodů, krystalem o taktu 16 Mhz a množstvím diagnostických konektorů. Pozornost byla také věnována vyhlazení napětí A/D převodníku, který se napájí přes LC článek složený z tlumivky 1 mH a kondenzátoru 100 nF.

Postup osazení a oživení:

Při osazování postupujeme jako obvykle, začínáme od nejnižších součástek po nejvyšší. Integrované obvody je vhodné vsadit do patice a vložit je do desky až po osazení. Externí součásti připojujeme nejlépe pomocí konektorů, abychom byli schopni vyndat plošný spoj z krabičky.

Obvod nijak oživovat nemusíme, pouze poprvé desku zapneme bez integrovaných obvodů a změříme napájecí napětí na příslušných bodech patic. Pokud se všude rovná pěti voltům, můžeme připojit display, vložit procesor a operační zesilovače a desku zapnout.

Pokud se nám při startu na obrazovce nic nezobrazí, je nejspíš nutné doladit kontrast. To se provádí pomocí trimru R8, kde šroubovákem natočíme takovou hodnotu, při které bude poměr kontrastu zobrazený/nezobrazený pixel co největší.

Měření:

Protože jsem chtěl ověřit charakteristiku vstupního filtru, změřil jsem jeho průběh. Použil jsem k tomu obyčejnou zvukovou kartu počítače s programem winscope (s běžícím FFT) a vygenerovaný vzrůstající tón 0-22kHz.

Na naměřených datech můžete vidět, že filtr nefunguje špatně, cutoff frekvence je opravdu na 8 kHz jako v návrhu, bohužel se do propustné části filtru připlétla nějaká nepřesnost a na 3.5 kHz se vyskytl útlum asi 1 decibel (hrubý propočet, měřítko winscope je lineární).

Celou konstrukci jsem otestoval i vizuálně, pustil jsem do vstupu sinus/obdelník a sledoval jsem zda hodnoty na displayi odpovídají frekvenci. Změřené hodnoty můžete sami posoudit v příloze.

Závěr:

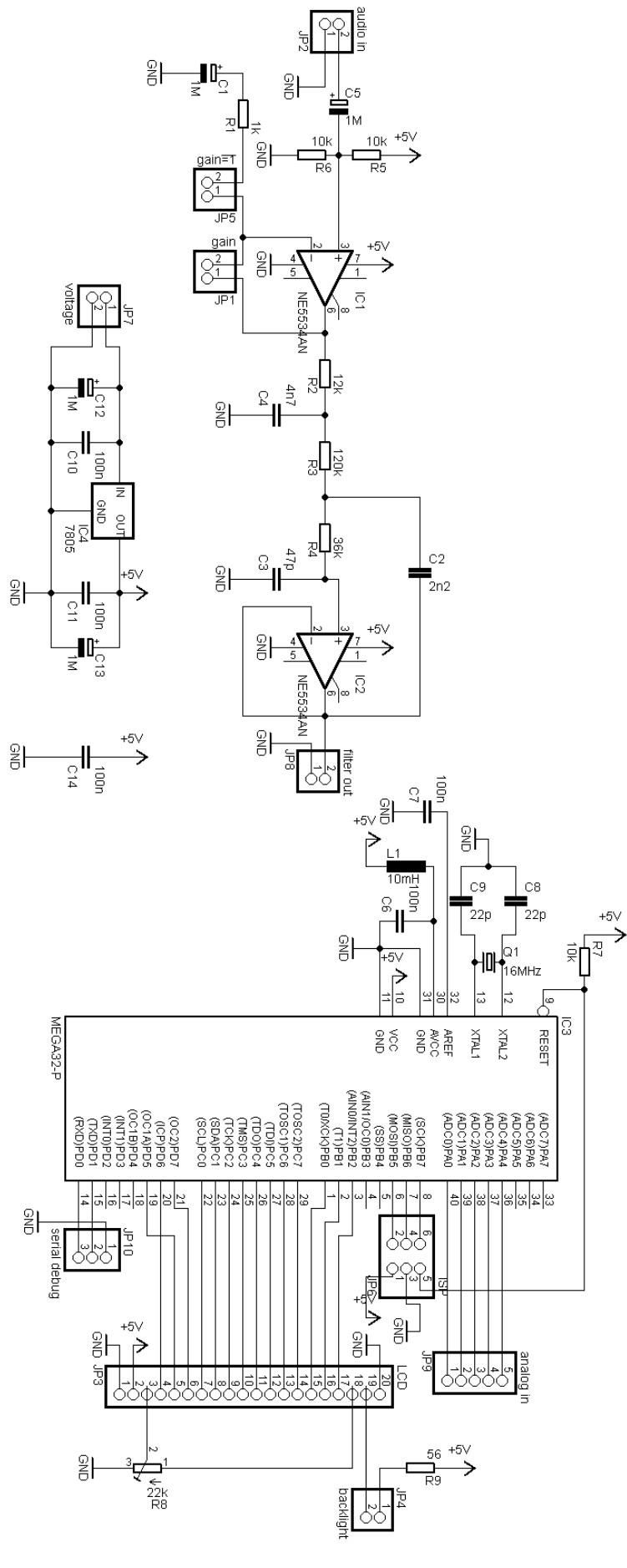
Po realizaci zapojení se ukázalo několik nedostatků v návrhu. Zprv je použitý LCD display velmi pomalý. Zobrazení pixelu mu trvá až 100 msec, takže pro vyšší obnovovací frekvence je nepoužitelný. Bohužel, tímto neduhem trpí většina „amaterských“ LCD displayů a je možné ho odstranit pouze použitím OLED displaye, který stojí ovšem při stejných rozměrech víc peněz. Druhým problémem je pomalá komunikace s displayem a pomalé výpočty slow-decay + logaritmu. To by mělo jít odstranit přepsáním Cčkových rutin do assembleru, na což bohužel není v letošním maturitním ročníku čas.

Na závěr bych jen dodal, že projekt se vydařil nad má očekávání, jako domácí efekt k audiosoustavě se hodí výborně. Pro studiové použití by bylo potřeba několik věcí vylepšit, což je však spíše otázka financí než práce (drahé DSP, OLED display, výkonnější procesor, lepší A/D převodník...).

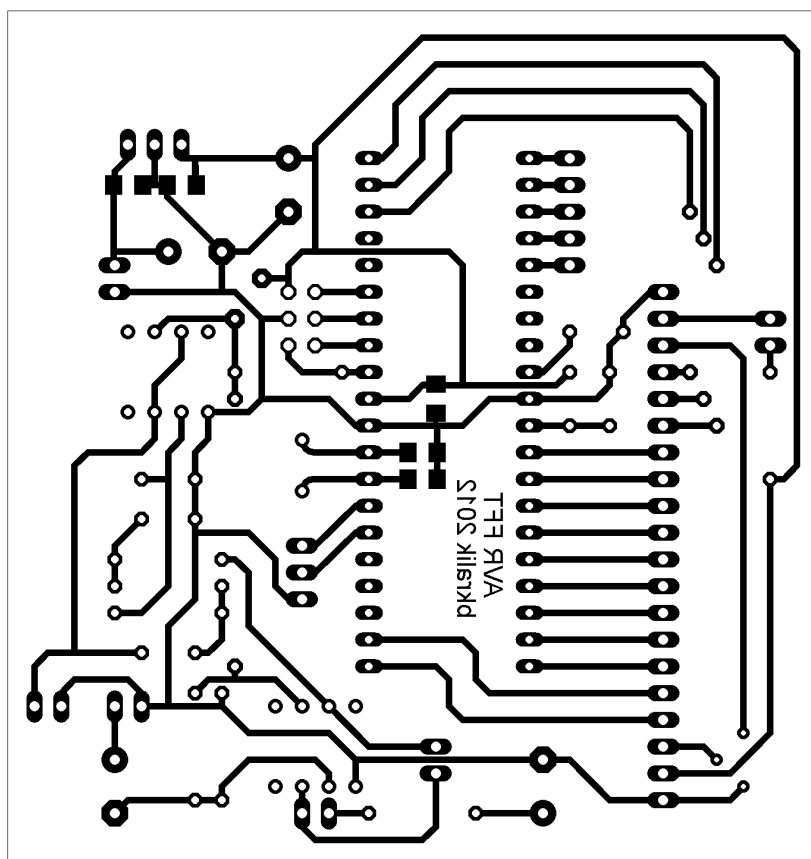
Přílohy:

- Příloha 1. - Schéma zapojení
- Příloha 2. - Plošný spoj ze strany součástek
- Příloha 3. - Osazovací plán
- Příloha 4. - Zapojení konektorů
- Příloha 5. - Seznam součástek
- Příloha 6. - Měřicí protokol
- Příloha 7. - Ukázka naměřených hodnot

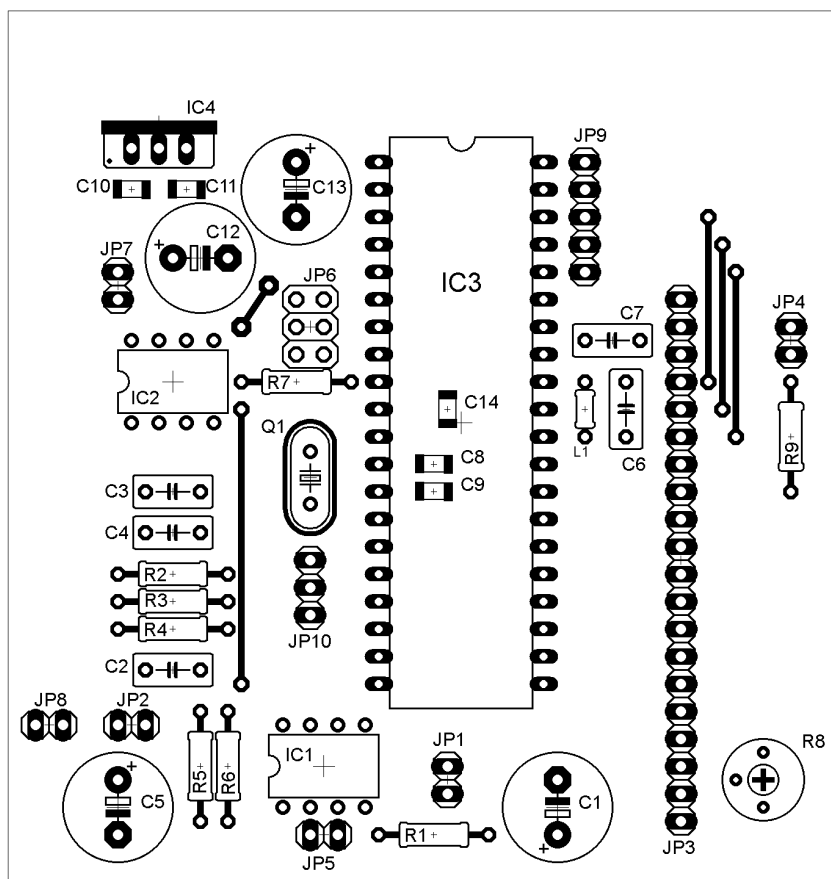
Příloha 1. - Schéma zapojení



Příloha 2. - Plošný spoj

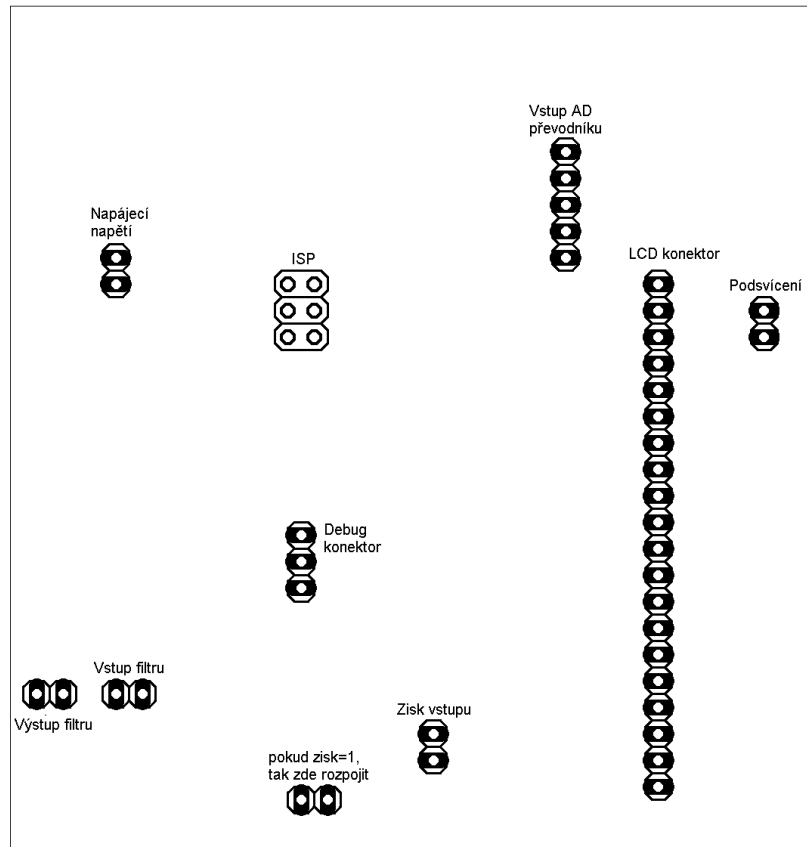


Příloha 3. - Osazovací plán



Pozn. Oba obrázky jsou ze strany součástek.

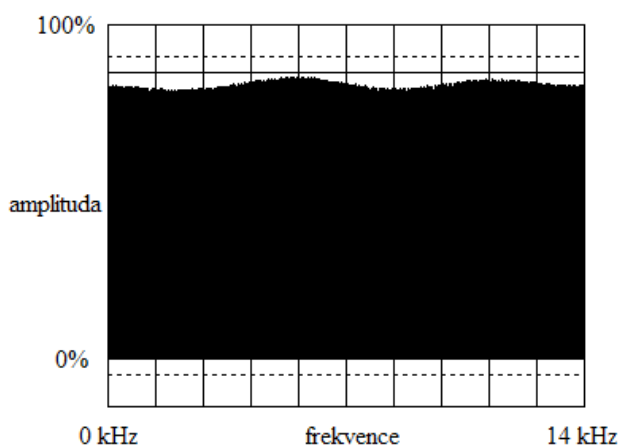
Příloha 4. - Zapojení konektorů



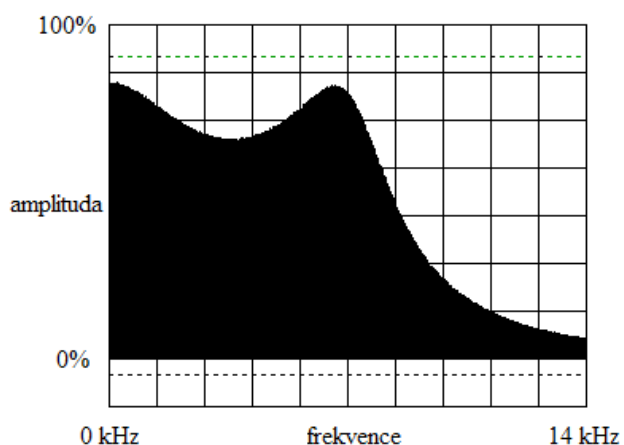
Příloha 5. - Seznam součástek

Označení	Hodnota	Podrobnosti
<u>Rezistory:</u>		
R1	1k	
R2	12k	
R3	120k	
R4	36k	
R5	10k	
R6	10k	
R7	10k	
R8	22k	
R9	56R	
<u>Kondenzátory</u>		
C1	1M/10V	elektrolyt
C2	2n2	keramika
C3	47p	keramika
C4	4n7	keramika
C5	1M/10V	elektrolyt
C6	100n	keramika
C7	100n	keramika
C8	22p	SMD keramika
C9	22p	SMD keramika
C10	100n	SMD keramika
C11	100n	SMD keramika
C12	1M/10V	elektrolyt
C13	1M/10V	elektrolyt
C14	100n	SMD keramika
<u>Integrované obvody:</u>		
IC1	NE5534AN	
IC2	NE5534AN	
IC3	MEGA32-P	
IC4	7805	
<u>Ostatní:</u>		
JP1-JP10	pinová lišta	40 pinů
L1	1mH	tlumivka
Q1	16MHz	krystal v pouzdře HC49

Příloha 6. - Měřicí protokol

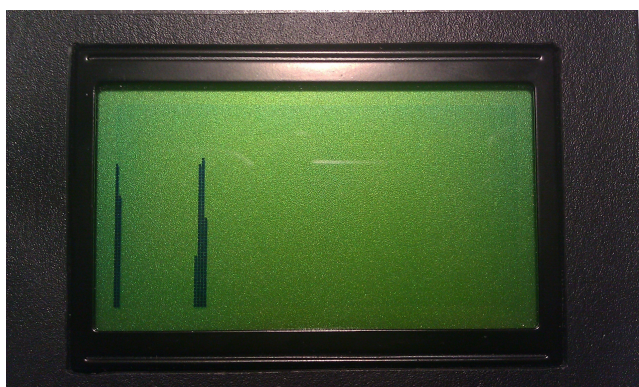


Ilustrace 1: Testovací signál před vstupem



Ilustrace 2: Charakteristika vstupního filtru

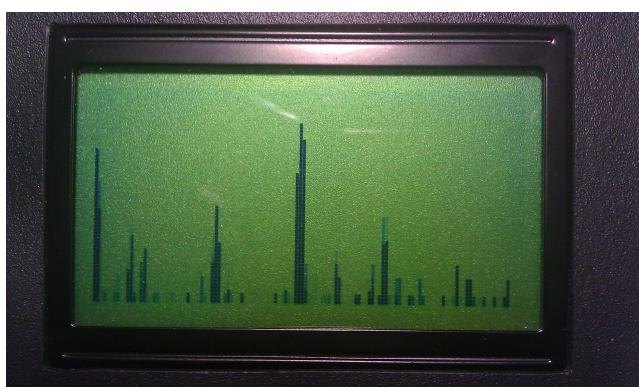
Příloha 7. - Ukázka naměřených hodnot



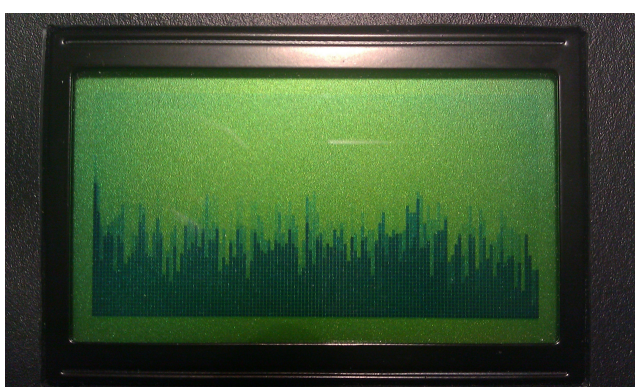
Ilustrace 3: Sinus 2 kHz



Ilustrace 4: Sinus 9.6 kHz



Ilustrace 5: Obdelník 4.6 kHz



Ilustrace 6: Bílý šum (celé spektrum)

Poznámka: Spektrální čára na začátku spektra je stejnosměrná složka, trvale v půlce protože AD převodník nemá symetrický vstup, t.zn. nula je 2.5V.