



Budapesti Műszaki és Gazdaságtudományi Egyetem
Villamosmérnöki és Informatikai Kar
Méréstechnika és Információs Rendszerek Tanszék

Mályusz Anna

Univerzális laborműszer megvalósítása jelfeldolgozó processzorral

Konzulens
Orosz György
Budapest, 2011

Tartalomjegyzék

Tartalomjegyzék	2
Hallgatói nyilatkozat	4
Kivonat.....	5
Abstract.....	6
1. Bevezetés	7
2. Rendszerterv és specifikáció.....	11
3. Felhasznált eszközök	13
4. Hagyományos mérési módok.....	16
4.1 Vizsgálójelek.....	16
4.1.1 Szinuszjel.....	16
4.1.2 Háromszög- és négyszögjel	17
4.1.3 A háromszög- és négyszögjel előállításának hibája	18
4.1.4 A jel additív szintézissel történő előállítása.....	21
4.2 Egyszerűbb mérési funkciók	23
4.2.1 Csúcsértékmérés	23
4.2.2 Periódusidő és frekvenciamérés.....	24
4.2.3 Effektívérték-mérés	25
4.2.4 A mérések hibája	26
4.2.5 A mérés pontosítása interpolációval	27
5. Rezonátoros struktúra és alkalmazásai	31
5.1 A rezonátoros algoritmus implementációja	33
5.2 Rezonátor alapú átviteli függvény mérés	36
5.2.1 Átvitel mérés a gyakorlatban	39
5.3 Impedanciamérés.....	42
5.3.1 DSP-s alkalmazás	49
5.3.2 A mérés korlátai.....	51

6.	Hangolható szűrő megvalósítása	52
6.1.1	Diszkrét idejű szűrők osztályozása	54
6.1.2	Szűrőtervezés ablakozással.....	58
6.1.3	FIR szűrő megvalósítása	63
7.	Kapcsolat a PC és a DSP között	67
7.1	Beolvasás.....	67
7.2	Adatok megjelenítése	68
8.	Eredmények, ajánlott elrendezések.....	70
8.1	PC-s interfész bemutatása	70
8.2	Egyszerűbb mérések.....	73
8.3	Rezonátor alapú átviteli függvény mérés	76
8.4	Impedanciamérés.....	83
8.4.1	Ajánlott mérési elrendezés impedanciaméréshez	85
8.5	Hangolható szűrő	86
9.	Összefoglaló értékelés és kitekintés	90
	Irodalomjegyzék	91

Hallgatói nyilatkozat

Alulírott **Mályusz Anna**, szigorló hallgató kijelentem, hogy ezt a szakdolgozatot meg nem engedett segítség nélkül, saját magam készítettem, csak a megadott forrásokat (szakirodalom, eszközök stb.) használtam fel. Minden olyan részt, melyet szó szerint, vagy azonos értelemben, de átfogalmazva más forrásból átvettem, egyértelműen, a forrás megadásával megjelöltem.

Hozzájárulok, hogy a jelen munkám alapadatait (szerző(k), cím, angol és magyar nyelvű tartalmi kivonat, készítés éve, konzulens(ek) neve) a BME VIK nyilvánosan hozzáférhető elektronikus formában, a munka teljes szövegét pedig az egyetem belső hálózatán keresztül (vagy autentikált felhasználók számára) közzétegye. Kijelentem, hogy a benyújtott munka és annak elektronikus verziója megegyezik. Dékáni engedéllyel titkosított diplomatervek esetén a dolgozat szövege csak 3 év eltelte után válik hozzáférhetővé.

Kelt: Budapest, 2012. 03. 05.

.....
Mályusz Anna

Kivonat

Dolgozatom célja, hogy egy olyan virtuális műszert hozzak létre, amely jelfeldolgozó processzoron (DSP: Digital Signal Processor) valósít meg különféle hagyományos és speciális mérési funkciókat. Mivel a jelfeldolgozó processzorok nagy számítási teljesítménnyel rendelkeznek, így kiválóan alkalmasak a mérésekhez kötődő jelfeldolgozási feladatok valós időben történő elvégzésére.

A virtuális műszerezés napjainkban az iparban és az oktatásban is elterjedt mérési forma. A virtuális műszer olyan szoftver, amely általános funkcionalitással rendelkező hardveren képes tesztelő és mérő rendszert megvalósítani.

Dolgozatomban arra törekszem, hogy bemutassam a megvalósított funkciókat az alkalmazott elméleti ismeretek tükrében, majd megvizsgáljam, és lehetőségek szerint korrigáljam az egyes funkciók megvalósításakor fellépő hibákat. Próbáltam a feladathoz mérten komplexitás és teljesítőképesség szempontjából minél kedvezőbb módszereket választani a feladatok megoldása során.

A létrejött műszer funkcióit az egyes fejezetekben tárgyalom, először az elméleti hátteret, majd a gyakorlati alkalmazást bemutatva. Az olvasó betekintést nyerhet az analóg jelek és rendszerek vizsgálatába a hagyományos jelgenerálási és mérési feladatok, valamint a komplexebb átviteli függvény, impedanciamérő és szűrő funkciók ismertetésén keresztül.

Abstract

The purpose of this thesis is the design and development of a digital signal processor based virtual instrument which realizes several traditional and special measurement functionalities. Due to the large computational capacity of signal processors, they can be deployed with success in such applications where complex, real-time measurement tasks should be performed.

Nowadays, virtual instruments have become wide-spread both in industry and education. A virtual instrument is such a system which can realize different measurement and test functionalities on the same hardware.

In the thesis, I introduce both theoretical and practical aspects of the measurement functionalities which have been implemented in the instrument. The measurement errors are also investigated, and solutions are presented which alleviate the errors. Measurement algorithms are chosen by considering their complexity and performance, furthermore taking into account the resources of the processor.

The instrument also realizes traditional functionalities like signal generation and measurement, but more complex algorithms have been implemented like automatic transfer function measurement, impedance measurement, and digital filter design.

1. Bevezetés

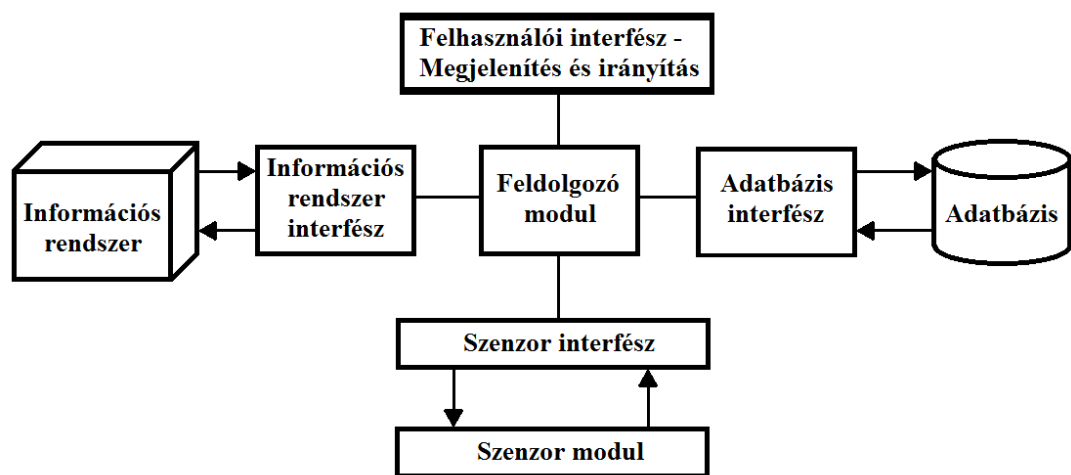
Az elektronikus műszerek funkcionális felépítése, a gyors technológiai fejlődés ellenére, az elmúlt évtizedben nem változott lényegében. A műszerek blokkvázlata nagyban hasonlít egymásra, mivel mindegyikben megtalálható a bemeneti, a feldolgozó és a megjelenítő egység. Azonban az alkalmazott technológiától, a felhasználási területtől és a műszerek fajtájából adódóan az egyes egységek bonyolultsága eltérő lehet. Egyszerűbb műszerekben a feldolgozó egység egy lehetséges megvalósítása az analóg átlagoló, míg a bonyolultabb esetekben inkább FFT-t számoló célhardverrel készítik el a műszert, vagy éppen a megjelenítő egységet egy mutatós műszerrel vagy grafikus kijelzővel valósítják meg.

Az 1990-es évektől kezdődően a PC-alapú készülékfejlesztés nagyjából két irányvonalat követett. A hagyományos készülékgyártókra jellemző, hogy megőrizték a szokásos készülékformát, kezelőfelületet, és a doboz belsejébe helyezték a speciális, külön erre a célra kifejlesztett PC-t. Ezzel szemben a másik irányvonal esetében egy hagyományos asztali PC-ről beszélhetünk, amit kiegészítenek analóg és digitális I/O perifériákkal és alkalmas szoftverrel. Ez utóbbi vezetett a „virtuális műszer” fogalmához [1].

A virtuális műszerezés rendszer tehát egy olyan szoftver, amit a felhasználó használ egy olyan számítógép által vezérelt tesztelő és mérő rendszer kifejlesztésére, amely képes egy külső, általános funkcionalitással rendelkező hardver eszközt vezérelni a számítógépről, és a tesztelési és mérési eredményeket is meg tudja jeleníteni a PC-n [2]. Az eljárásra számos definíciót találhatunk a világirodalomban. A Santori [3] féle definíció szerint a virtuális műszerezés egy olyan műszer, amelynek az általános funkciója és kapacitása egy szoftverben van meghatározva. Goldberg [4] leírja, hogy a virtuális műszerezés specializált alegységből, általános célú számítógépből, szoftverből és „know-how”-ból áll. Bár ezek informális definíciók, mégis jól tükrözik az általános alapelveket, azaz, hogy bármely számítógép képes szimulálni egy másikat, ha megvalósítjuk szoftverszimulálásként az adott számítógépen.

A virtuális műszerezés a következő komponensekből tevődik össze:

- szenzor modul
- szenzor interfész
- információs rendszer interfész
- feldolgozó modul
- adatbázis modul
- felhasználói interfész



- 1.1. ábra Egy virtuális műszer általános felépítése [2]

A 1.1. ábrán látható a virtuális műszerezés általános felépítése. A szenzor modul detektálja a fizikai jelet a környezetből és az itt megtalálható jelkondicionáló rész először kondicionálja a jelet, majd ezt követően az A/D átalakító digitális formába transzformálja a későbbi manipulálhatóság végett. A szenzor-interfészen keresztül kommunikál a szenzor-modul a számítógéppel, mely kommunikáció lehet vezetékes vagy vezeték nélküli. Amint az adatok digitális formában a számítógépen megtalálhatóak, tetszés szerint manipulálhatók (feldolgozó modul), vagy eltárolhatóak (adatbázis-modul). Ezt követően lehetséges az adatok megjelenítése a felhasználói interfészen keresztül, vagy vissza lehet alakítani analóg formába további folyamatvezérlés végett. A virtuális műszerezést gyakran integrálják más információs

rendszerekkel. Ilyenkor a konfigurációs beállítások és a mérési adatok az elérhető rekordokkal vannak tárolva és párosítva.

Az univerzális „műszermagból” és kiegészítő perifériákból sokféle konkrét műszert lehet kialakítani annak függvényében, hogy az adott esetben milyen alkalmazói szoftvert futtatunk a műszermagon, vagyis a számítógépen. Az alkalmazói szoftver cseréjével, ugyanazon a hardver- és alapszoftver készleten több, funkcionalitásukban különböző műszer implementálható, úgymint spektrumanalizátor, oszcilloszkóp, függvénygenerátor, stb [1]. Az előállított műszert nevezhetjük valódinak abban az értelemben, hogy funkcionálisan megegyezik egy célműszerrel, de ugyanakkor virtuális, mivel a fizikai megjelenése lényegesen eltér azokétól.

A virtuális műszer koncepció egyik vezető képviselője az amerikai National Instruments [5] cég. Kezdetben analóg és digitális I/O perifériákat kezdett gyártani számítógépekhez, majd PC-khez beépíthető kártya és külön bedobozolt egységek formájában. A cég virtuális műszer koncepciója alkalmas szoftver nélkül aligha ért volna el átütő sikert. A moduláris és univerzális hardverelemekhez szintén moduláris és univerzális programozási nyelvre és környezetre van szükség. A NI többféle szoftverrendszer is kifejlesztett a hardvereihez, a legnépszerűbb a LabVIEW adatfolyam alapú, magas szintű grafikus programozási nyelv és környezet. Természetesen a LabVIEW-n kívül más grafikus programozási nyelvek is léteznek, pl. Simulink (MathWorks), VEE (Agilent).

Dolgozatom célja, hogy egy olyan virtuális műszert hozzak létre, mely egy úgynevezett jelfeldolgozó processzoron (DSP: Digital Signal Processor) valósít meg különféle hagyományos és speciális mérési funkciókat. Mivel a jelfeldolgozó processzorok nagy számításai teljesítménnyel rendelkeznek, így kiválóan alkalmasak a mérésekhez kötődő jelfeldolgozási feladatok valós időben történő elvégzésére.

A feladathoz rendelkezésre álló processzor egy ADSP21364 típusú 32 bites lebegőpontos processzor, mely egy hangfrekvenciás tartományban működő, 24 bites analóg-digitális (AD) és digitális-analóg (DA) átalakítóhoz (kodek) kapcsolódik.

A számábrázolási pontosság és a relatív nagy bitszámú analóg interfész várhatóan megfelelő precizitást biztosít a feladat elvégzésére.

Az elkészített műszer vezérlése, és a mérési eredmények megjelenítése PC segítségével történik. A PC-s vezérlés és megjelenítés nem kíván különleges szoftvereket, mely az eszköz használhatósága miatt fontos szempont.

Dolgozatomban célul tűztem ki, hogy megvizsgáljam és lehetőségek szerint korrigáljam az egyes funkciók megvalósításakor fellépő hibákat. Próbáltam a feladathoz mérten komplexitás és teljesítőképesség szempontjából minél kedvezőbb módszereket választani a feladatok megoldása során.

A dolgozatom a következőképpen épül fel. A második fejezetben a rendszer felépítéséről adok átfogó képet. A harmadik fejezetben a felhasznált eszközöket ismertetem fontosabb funkciójuk szerint. A negyedik fejezetben a vizsgálójelek megvalósításának módjaival és hibáival foglalkozom, majd a periodikus jelek paramétereinek mérési lehetőségeivel foglalkozom, és a méréseknél felmerülő hibák megoldására teszek kísérletet. Az ötödik fejezetben egy komplexebb struktúrának a bemutatására vállalkozom. Az ötödik fejezet tartalmazza a rezonátoros struktúra ismertetését, a struktúra felhasználásával megvalósuló átviteli függvény mérés és impedanciamérés ismertetését. A hatodik fejezetben a DSP-n megvalósított hangolható szűrő specifikációjának kérdésével és implementálásával foglalkozom. A hetedik fejezetben bemutatom a DSP és PC közötti felhasználói interfészt. A nyolcadik fejezet tartalmazza a fontosabb mérések eredményeit, és a felhasználói interfész rövid használati útmutatóját. Az utolsó fejezetben összegzem a szakdolgozat során megvalósított rendszer eredményességét.

2. Rendszerterv és specifikáció

A feladat során célom egy jelfeldolgozásra kiválóan alkalmas jelfeldolgozó processzort (DSP-t) úgy felhasználni, hogy az képes legyen a laborokban elterjedten használt alapvető berendezések funkcióit megvalósítani. A megvalósítás teljesen virtuális módon, szoftveresen, kiegészítő hardver-elemek nélkül történik.

A feladat végeztével helytakarékos, egy lapra integrált műszert kapunk kézhez, mellyel gyorsan tudjuk egy rendszer alapvető funkcióit tesztelni, egyszerre tudunk a bemenetére vizsgálójelet előállítani, és a kimenetén leolvasni a válaszjelet. Az elkészült műszernek nem feltétlenül precíziós mérésekre kell alkalmasnak lenni, hanem tájékoztató jellegű mérések elvégzésére.

A DSP-vel való kommunikáció PC-n keresztül oldható meg.

A DSP-n megvalósítandó funkciók:

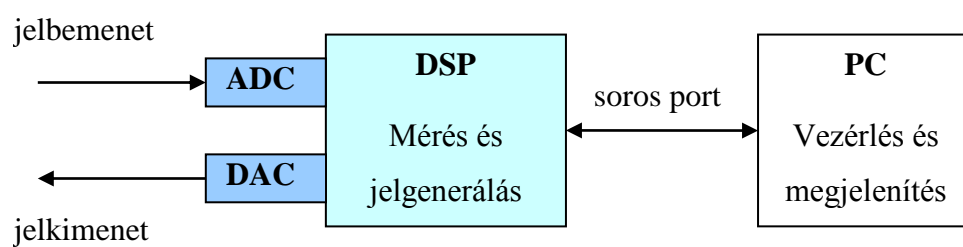
- vizsgálójelek előállítása (szinusz-, háromszög-, négyszögjel)
- periodikus jelek paramétereinek mérése (amplitúdó, periódusidő- és frekvencia-mérés, effektív érték mérése)
- átviteli függvény és impedanciamérés
- hangolható szűrő megvalósítása

A DSP és a felhasználó közötti kapcsolatot egyszerű PC-s interfészen valósítjuk meg, mely soros porton fogadja a szöveges formájú üzeneteket, ezért egyszerű soros port kezelő szoftvert felhasználva kezelhető.

A DSP-re megvalósítandó rendszer feladata lesz a bemenetére érkezett jelek feldolgozása, az eredmények elküldése a felhasználó felé, és funkciótól függően kimeneti jel előállítása.

A DSP-vel megvalósított műszer virtuális műszerként fogható fel, hiszen rendelkezésre áll egy általános funkcionalitásokkal rendelkező hardver, a DSP kártya, mely biztosítja

a megfelelő analóg hardveres interfészeket. A funkcionalitást mi implementáljuk a megfelelő szoftver megírásával.



2.1. ábra. A rendszer sematikus blokkdiagramja

3. Felhasznált eszközök

A tanszéken rendelkezésre állt egy, az Analog Devices által gyártott, ADSP-21364 EZ-KIT Lite fejlesztői kártya.

A kártya egy 32 bites lebegőpontos számábrázolású, 330 MHz maximális frekvencián működő ADSP-21364-es típusú DSP processzorral van felszerelve. 3 Mbit RAM áll rendelkezésre, két bemeneti AD és nyolc kimeneti DA csatornával.

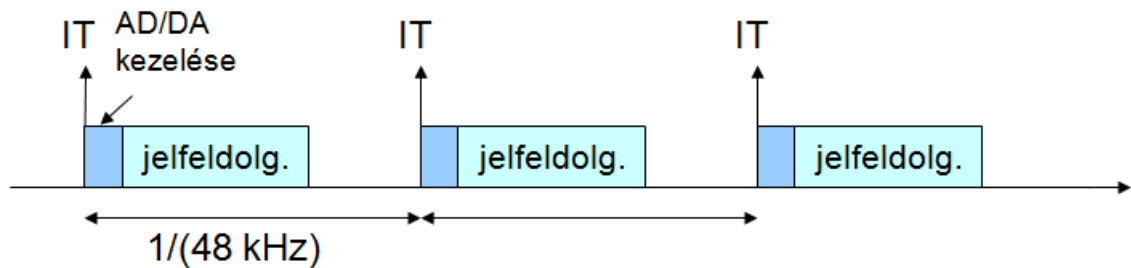
A fejlesztői kártya DSP-n kívül felhasznált főbb perifériái a következők:

- AD1847-es sztereo kodek
- JTAG csatlakozó
- UART illesztő IC
- Flash memória

Az AD1847 sztereo kodek biztosítja az analóg-digitális és digitális-analóg átalakítást. Ezen keresztül kapcsolódik a DSP az analóg világhoz. A JTAG csatlakozó JTAG-en keresztül történő programfejlesztést tesz lehetővé.

A PC és a DSP között az UART illesztő IC-t felhasználva, egy kommunikációs kártya teszi lehetővé az információközlést soros porton.

A JTAG segítségével történő fejlesztés során szabványos interfészen keresztül, egy programozó áramkör felhasználásával férünk hozzá a processzor különböző egységeihez. Segítségével a program letöltése, elindítása, megállítása, különböző regiszterek és perifériák állapotának lekérdezése. Mivel a DSP nem tartalmaz flash memóriát, ezért a program a DSP SRAM memóriájába töltődik be. Ebből következően szükséges a programok újbóli letöltése a DSP kártyára minden újraindítás után, mert az SRAM cellák a tápfeszültség megszűnése következtében „elfelejtik” a tartalmukat. A fejlesztői kártyán található flash memória segítségével a programok akár permanensen is tárolhatók, innen képes a DSP bootolást végrehajtani.



3.1. ábra

A DSP-n futó programra a 3.1. ábrán ábrázolt megszakítás alapú programszervezés jellemző. A megszakításokat az analóg jeleket kezelő analóg-digitális (AD) és digitális-analóg (DA) átalakításra használható kodek szolgáltatja. Minden megszakítást a DSP AD/DA átalakítójára jellemző periódusidő alatt, $f_s = 48 \text{ kHz}$ -es mintavételi frekvencián, tehát $1/48 \text{ ms}$ alatt dolgozunk fel. Egy megszakítási rutin első részében található a fizikai szint lekezelésére, mely az ábrán a sötétkék szakasz, ez tartalmazza az AD és DA átalakító buffereinek olvasását és írását. A következő fontos szakasz a jelfeldolgozási ciklus, mely során a specifikált feladatokat hajtjuk végre. Általában itt történik az AD átalakító jeleinek feldolgozása, és a DA átalakítóra kiadandó jel generálása. A jelek generálása természetesen történhet a bemenet felhasználásával és anélkül is.

A jelfeldolgozási ciklus során kell elvégezni a különböző adminisztratív teendőket, mely a program végrehajtását vezérli, például üzemmód beállítása, az üzemmódnak megfelelő függvények meghívása, értékek kijelzése, konverziók elvégzése.

Az elvégzendő feladatok futási idő szempontjából két csoportra oszthatók. Egyrészt vannak hard-realtime feladatok, melyeknek minden mintavételi periódus során le kell futniuk. Ezek például különböző mérési és jelgenerálási funkciók, melyek minden mintavétel során új adatot generálnak vagy dolgoznak fel.

Léteznek azonban olyan feladatok is, amiket csak ritkán kell számolni, de hosszú ideig tartanak (pl. Fourier-együttható számítás, szűrőtervezés, stb.). Ezeket külön kezeljük, mert nem hard-realtime feladatok, és általában üzemmód váltásakor kell végrehajtani

őket, amikor nem okoz problémát, ha valamilyen tranziens keletkezik a ki- vagy bemeneten abból fakadóan, hogy a számítások túlnyúlnak egy mintavételi perióduson. Ennek megfelelően ezeket a feladatokat csak akkor szabad engedélyezni, amikor a felhasználó valamilyen paramétert megváltoztat.

4. Hagyományos mérési módok

4.1 Vizsgálójelek

A laborműszerek közé tartoznak a függvénygenerátorok. A DSP-n megvalósított jelgeneráló funkciókkal olyan vizsgálójeleket állítunk elő, amik segítségével sokoldalú mérések végzése válik lehetővé. Tipikusan jelfeldolgozási algoritmusok, áramkörök tesztelésére, bemérésére, javítására gyakran kell olyan jeleket alkalmazni, melyek paraméterei ismertek, és a felhasználó által állíthatóak. Mivel a gerjesztés ismert, így a vizsgált rendszer válasza alapján annak helyes működése tesztelhető.

A DSP-n megvalósított vizsgálójelek a következők:

- szinusz
- változtatható kitöltésű háromszögjel
- változtatható kitöltésű négyszögjel

4.1.1 Szinuszjel

A szinuszjel alapját a DSP könyvtári függvényei között szerepelő szinusz függvény képezi. Egy fázisváltozót a frekvenciával arányos lépésközzel léptetve, az alábbi egyenletekben látható módon számítjuk:

$$x(n) = \sin(\varphi(n)) \quad (4.1)$$

$$\varphi(n+1) = \varphi(n) + 2\pi \frac{f}{f_s} \quad (4.2)$$

A szinusz függvényt előállító könyvtári függvény argumentumaként a szinuszjel fázisát kell megadni. A fázis a frekvencia integrálásával állítható elő, mely így elméletileg egy folytonosan növekvő szám. A fázis növelésének így történő direkt megvalósítása azonban számábrázolási problémák miatt nem megengedett. Ennek magyarázata, hogy a DSP-n véges hosszú lebegő pontos számot tudunk ábrázolni, amihez képest elég nagy érték után már elhanyagolhatóan kicsi egy lépésköz, tehát nem történik meg a fázisváltozó inkrementálása. Ennek megfelelően, mivel a szinusz periodikus függvény,

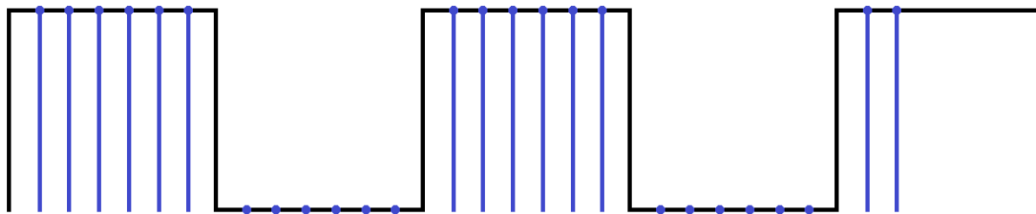
a szinusz függvény argumentumát 0 és 2π között tartjuk kihasználva, hogy a szinusz 2π szerint periodikus.

4.1.2 Háromszög- és négyszögjel

A háromszög- és négyszögjel előállításának két elterjedt formáját vizsgáltam.

- Wavetable (hullámtábla)
- Additív szintézis

Wavetable esetén úgy képzelhető el, mintha az ideális analóg jelet mintavételeznénk, ahogy a 4.1. ábra mutatja. A módszer előnye, hogy viszonylag egyszerű és gyors végrehajtási idejű az analitikusan előállítható függvények esetén, viszont a mintavételezésnél ismert átlapolódási jelenség léphet fel. A jel spektrumát háromszögjel esetében $1/n^2$ -tel csökkenő, négyszögjel esetében $1/n$ -nel csökkenő amplitúdójú frekvencia-összetevők képezik. A jelet mintavételezve, ami a frekvenciatartományban mintavételi frekvenciánkénti spektrumismétlődést jelent, átlapolódnak a mintavett spektrumok, ami a jel torzulását idézi elő.

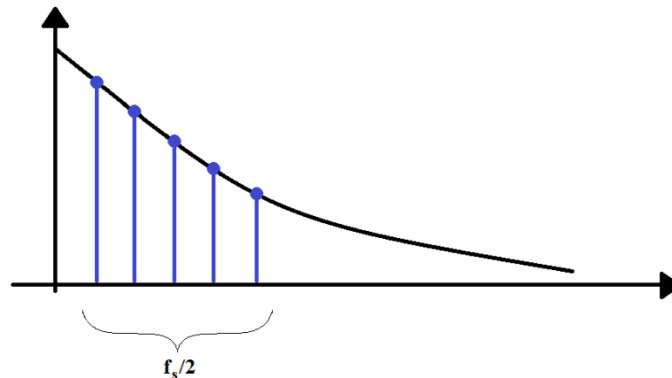


4.1. ábra Wavetable módszer

Additív szintézis a jel Fourier-komponensekből való előállítását jelenti. Mint ismert, periodikus jelek előállíthatóak szinuszos komponensek súlyozott összegeként. A 4.2. ábrán látható módon a periodikus jelek spektruma vonalas, egy-egy szinuszos komponens felel meg egy spektrumvonalnak.

A módszer előnye, hogy ebben az esetben nem lép fel átlapolódás, mivel csak a mintavételi frekvencia feléig állítjuk elő a komponenseket. Az additív szintézis hátránya, hogy számítási igénye nagyobb a wavetable módszerhez képest.

Két tipikus vizsgálójel Fourier-együtthatói láthatóak a (4.3) és (4.4) egyenletekben.



4.2. ábra Szinuszos komponensekből álló jel spektruma

A változtatható kitöltésű háromszögjel Fourier-sora [12]:

$$\sum_{n=1}^{\infty} \frac{2(-1)^n m^2}{n^2(m-1)} \sin\left(\frac{n\pi(m-1)}{m}\right) \sin(n2\pi f_0) \quad (4.3)$$

m : 100/kitöltés [%]

A változtatható kitöltésű négyszögjel Fourier-sora [12]:

$$\sum_{n=1}^{\infty} \frac{4A}{n\pi} \sin(n\pi m) \cos(n2\pi f_0) \quad (4.4)$$

m : kitöltés [%]/100

4.1.3 A háromszög- és négyszögjel előállításának hibája

Ebben az alfejezetben azt vizsgálom, hogy a két jelgenerálási módszer milyen hibával jellemezhető, és adott esetben melyik módszert érdemes alkalmazni. A két módszert jellemző hiba a $[-f_s/2 \dots f_s/2]$ tartományban ideálisan jelenlévő spektrumkomponensek effektív értékétől való eltéréssel jellemzem. Ezt az eltérést okozhatják belapolódó extra komponensek, illetve hiányzó komponensek.

Wavetable-nél a hibát a belapolódó komponensek okozzák, ezt illusztrálja a 4.3. ábra. Az ábrán késsel az $f_s/2$ tartományon belüli, míg pirossal a $f_s/2$ tartományon kívüli, átlapolódó komponenseket jelölöm. A hiba a $[-f_s/2, f_s/2]$ tartományon kívüli belapolódó komponensek effektív értékének és a $[-f_s/2, f_s/2]$ frekvenciasávba eső komponensek effektívértékének arányából számítható (4.5) képlettel.

$$\text{hiba} = \frac{\sqrt{\sum_{\forall b} |X_b|^2}}{X_{rms}} \quad (4.5)$$

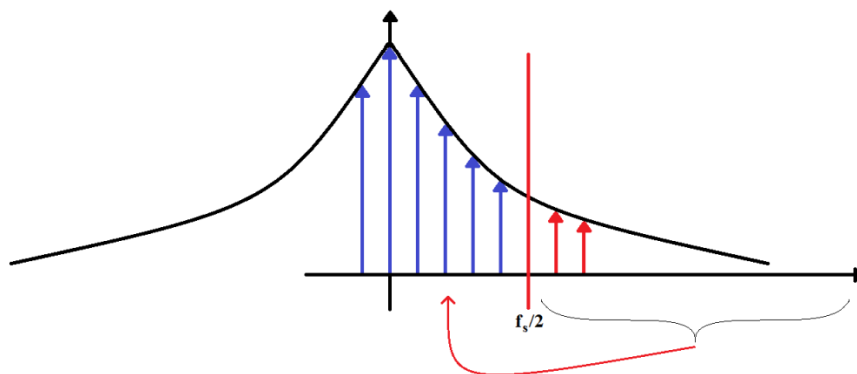
b : belapolódó komponensek

Additív szintézisnél a komponenseket valós időben generáljuk, ezért hibát okozhat, hogy nem tudunk akármennyi komponenst előállítani egy mintavételyi idő alatt, ez látható a 4.4. ábrán. A hibát a $[-f_s/2 \dots f_s/2]$ tartományból hiányzó és ideális esetben jelenlévő komponensek effektív értékének arányából számítható (4.6) alapján.

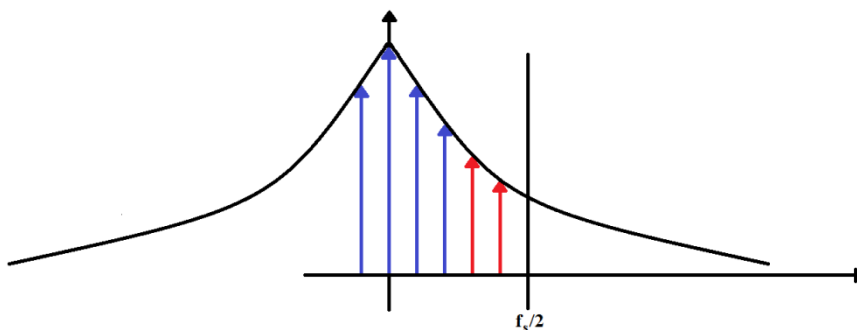
$$\text{hiba} = \frac{\sqrt{\sum_{\forall h} |X_h|^2}}{X_{rms}} \quad (4.6)$$

h : hiányzó komponensek

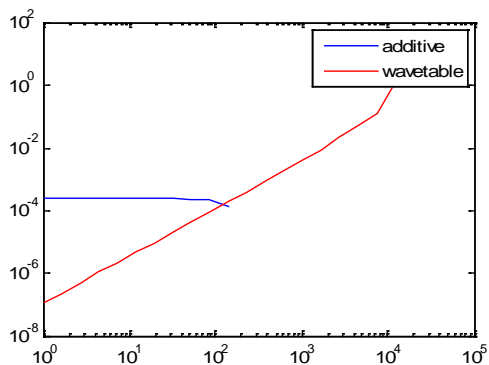
Ezen hibák alapján megállapítható, hogy milyen alapprofrekvenciájú jel, melyik módszerrel állítható elő kisebb hibával. A számításokat 50% kitöltésű négyszög illetve háromszögjelre végeztem. Habár a jelgenerálás során más kitöltési tényezőkre is fel kell készülni, az így kapott eredmények mégis irányadóak lehetnek ahhoz, hogy legalább hozzávetőleges becslést kapjunk arra vonatkozólag, hogy milyen nagyságrendű hibával számolhatunk bizonyos frekvenciákon. A számítások során azt feltételeztem, hogy 150 harmonikus megvalósítása lehetséges additív szintézis esetén (lásd következő szakasz), mert előzetes tesztek alapján ekkora számú periodikus komponens állítható elő egy mintavételyi idő alatt.



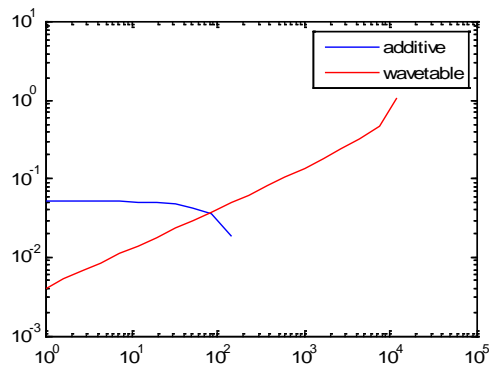
4.3. ábra Belapolódó komponensek wavetable módszer esetén (pirossal jelölve)



4.4. ábra Hiányzó komponense additív szintézis esetén (pirossal jelölve)



Háromszögjel



Négyszögjel

4.5. ábra Additív és wavetable hibájának frekvenciafüggése

A 4.5 ábrán a (4.5) és (4.6) egyenletek segítségével számított hibák alakulását ábrázoltam a jel frekvenciájának függvényében. Az ábra azt mutatja, hogy a wavetable kis frekvenciákon, kb. 70 Hz alatt, az additív módszer kb. 70 Hz-től biztosít

alacsonyabb hibát. Ez a jelenség a következő módon magyarázható. Minél kisebb a frekvencia, annál több felharmonikus fér el $f_s/2$ -ig, tehát additív szintézis során annál több harmonikust kell generálni, mely egy idő után számítási korlátba ütközik, tehát additív szintézis esetén a frekvencia csökkentésével egy adott frekvencia alatt nő az előállítás hibája. Alacsony frekvencián viszont a $[-f_s/2, f_s/2]$ tartományból kilógó komponensek száma csökken, tehát a wavetable módszer egyre kisebb hibával alkalmazható. Mivel a 70 Hz-es alsó határ igen alacsony a 24 kHz maximális frekvencia-összetevő mellett, ezért a feladat megoldását háromszög- és négyszögjel esetén az additív megvalósítás kódolásával kezdtem.

A szinusz előállítása gyakorlatilag a hullámtábla alapú szintézisnek felel meg, tehát a függvény pillanatértékét számítjuk pontról pontra. Mivel a szinuszjel sávkorlátozott jel, és csak egy frekvencián van komponense, tehát a belapolódás nem okoz problémát.

4.1.4 A jel additív szintézissel történő előállítása

Mivel a jel előállítása valós időben történik nem pedig előre eltárolt adatok segítségével, ezért oda kell figyelni, hogy milyen módszert alkalmazunk az egyes harmonikus komponensek előállításához.

Egyik lehetőség a szinuszos komponensek generálására a szinuszfüggvényt megvalósító könyvtári függvény alkalmazása. Annak meghatározása, hogy hány harmonikus előállítására van lehetőség, kísérleti úton történt. 50 %-os kitöltésű négyszögjelet állítunk elő a könyvtári függvény ismételt meghívásával, azaz minden frekvenciára meghívjuk, ezeket súlyozzuk, és összeadjuk az amplitúdó szerint, majd a kimenetre vezetjük az összeget. Ezt minden megszakítás-periódusban újra kell számolni. A fázisváltozót a (4.2)-ben leírt módszerrel számítjuk. Így 60 komponens kiszámítására van elegendő idő.

A könyvtári szinuszfüggvény sorozatos meghívása lassítja a jelgenerálást, mert feltehetőleg valamilyen interpolációs technikát használ a megfelelő pontosság eléréséhez, valamint a függvényhívás miatti overheadet is számításba kell venni. Emiatt

érdemesebb a szinuszfüggvényt csak ritkán alkalmazni. Ennek megfelelően a program csak egyszer hívja meg és a (4.7)-(4.9) egyenletek szerinti rekurzív módszerrel számolja a felharmonikusokat, így csökkentve a számításigényt. A rekurzív számításhoz azt az egyszerű algebrai azonosságot használjuk, miszerint $x^{(a+b)} = x^a \cdot x^b$.

$$\text{alapharmonikus: } e^{j\varphi} = \cos(\varphi) + j\sin(\varphi)$$

$$n. \text{ harmonikus: } e^{jn\varphi}$$

$$n + 1. \text{ harmonikus: } e^{j(n+1)\varphi}$$

$$e^{j(n+1)\varphi} = e^{jn\varphi} \cdot e^{j\varphi} = \text{alapharmonikus} \cdot n. \text{ harmonikus} \quad (4.7)$$

$$\cos((n + 1)\varphi) = \cos(\varphi) \cos(n\varphi) - \sin(\varphi) \sin(n\varphi) \quad (4.8)$$

$$\sin((n + 1)\varphi) = \cos(\varphi) \sin(n\varphi) + \sin(\varphi) \cos(n\varphi) \quad (4.9)$$

A számításhoz szükséges komplex számokat egy kételemű tömbben kezeljük, ezek nulladik, valós eleme a koszinuszos tag, első, képzetes eleme a szinuszos tag. Három komplex változópárt használunk fel az alapharmonikus, az n . és az $n + 1$. harmonikusok tárolására. Az alapharmonikusok értékei könyvtári függvény segítségével számított koszinusz és szinusz. Ezekkel egyenlők inicializáláskor az n . harmonikusokat tartalmazó változók. Az új értékeket ciklusba szervezve számítjuk a (4.7) alapján. Így első lefutás után az $n + 1$. harmonikusokat tároló komplex pár a második harmonikussal lesz egyenlő. Az $n + 1$. harmonikust mindig az alapharmonikus és az n . harmonikus szorzataként számoljuk. Ezek az egyszerű szorzások és összeadások a DSP-n hatékonyan végrehajthatóak.

Ezzel a módszerrel 150-re emelkedik a megvalósítható frekvencia-összetevők száma. A Fourier-együtthatók viszont, amikkel a frekvencia-összetevőket súlyozzuk, előre kiszámíthatóak és eltárolhatóak, így elég egyszer generálni egy együtthatótömböt, és csak akkor újra feltölteni, ha valamelyik paraméter változna. Ez jelentősen rövidíti a számítást.

4.2 Egyszerűbb mérési funkciók

Az egyszerűbb mérések közé tartoznak azok a mérések, ahol nem az általunk adott gerjesztés választ mérjük vissza, hanem önálló forrásból érkező jelet analizálunk. Olyan mérési funkciók megvalósítása itt a cél, amik elterjedten használtak általános mérő műszereken. Ilyenek:

- Maximum, minimum, csúcstól csúcsig mért érték
- Frekvencia- és periódusidő-mérés
- Effektív érték (RMS) mérés, RMS mérés szórásának számítása

A fenti értékek mérése során a DSP-vel folyamatosan mintavételezzük a bemeneti jelet, amíg a mérési funkció nincs leállítva.

A működést mérési periódusokra vagy más néven mérési ciklusokra bontjuk. Minden mérési periódus elején újratekintjük a mérendő paraméter számítása az adott időablakra, és egy mérési periódus végén a műszer megjeleníti a mért értékeket. A mérés periódusának időtartama minden mérés esetén változtatható. A mérés időtartama az az idő, amennyi ideig a bemeneti mintákat egymáshoz tartozóként figyeljük, ennek elteltével az addig számolt értékeket tekintjük a mérés végeredményének. A beállított mérési idő lejártá után új mérési periódus kezdődik, a program nullával inicializálja a változókat. A minél rövidebb mérési periódus nyilvánvalóan meggyorsítja a mérési folyamatot, de általánosságban elmondható, hogy minél hosszabb egy mérési ciklus, annál pontosabb a mérés.

A következőkben a mérés szó alatt a beállított mérési időtartam alatt elvégzett adatgyűjtés és kiértékelés értendő.

4.2.1 Csúcsértékmérés

Periodikus jelek szélső értékeinek szokásos elnevezése a csúcsérték. Definiálható a jel maximális, ún. pozitív csúcsértéke, X_{p+} , a jel minimális, ún. negatív csúcsértéke, X_{p-} , valamint a két szélsőérték különbsége, az ún. csúcstól csúcsig érték, X_{pp} .

A maximum és minimum érték periodikus jel esetén a pozitív és negatív csúcserőértéket jelenti. A jel szélsőértékeit a programmal iteratív módon számoljuk. A megadott mérési ideig keressük a jel maximumát és minimumát. Minden újabb mintára megvizsgáljuk, hogy az eddig maximumként illetve minimumként eltárolt értéknél nagyobb-e vagy kisebb, ha igen, a maximum ill. minimum változót felülírjuk a vizsgált értékkel. X_{PP} -t egyszer, a mérési időköz végén számoljuk ki, értéke a maximum és minimum különbsége.

4.2.2 Periódusidő és frekvenciamérés

A frekvencia és a periódusidő a periodikus jelek egyik paramétere.

A periodikus jelek közös jellemzője, hogy a t időpontban felvett $x(t)$ értéküket bármely $(t + kT)$ időpontban is felveszik, vagyis:

$$x(t) = x(t + kT); \text{ ahol } k \in \mathbb{Z} \quad (4.10)$$

A T ismétlődési időt periódusidőnek, az ehhez tartozó frekvenciát $f_0 = \frac{1}{T}$ alapfrekvenciának, az $\omega_0 = 2\pi f_0 = \frac{2\pi}{T}$ kifejezéssel definiált rad/s dimenziójú mennyiséget pedig körfrekvenciának nevezzük.

A (4.10) alapján egy egyenkomponens nélküli $x(t)$ jel minden periódusban kétszer metszi a nulla értéket, és csak egyszer úgy, hogy előtte negatív majd pozitív értékű.

A műszer alapkoncepciója, hogy állandó hosszúságú mérési ciklusok során dolgozzuk fel a jelet, melyhez legjobban a közismert módszerek közül állandó kapuidejű periódusidő-mérés illeszkedik, így ezt a módszert alkalmaztam. A módszer lényege, hogy a T_m mérési idő alatt megszámláljuk a beérkező periódusok számát (P) és a mérési időt leosztjuk ezzel az értékkel, tehát:

$$T = T_m/P \quad (4.11)$$

Az állandó kapuidejű periódusidő-mérés esetén hibát okoz, ha a T_m mérési idő nem egész számú többszöröse a T periódusidőnek. Ezt a problémát azzal az ismert

módszerrel küszöböltem ki, miszerint a mérési idő kiterjeszhető addig, amíg az eredeti mérési idő lejártát követően egy újabb periódus kezdődik.

A periódusidő-mérés esetén ismert, hogy egész számú periódus mérése esetén is az időmérés kvantált eredménye miatt úgynevezett kvantálási hiba keletkezik. Az időmérés felbontását a DSP mintavételi frekvencia adja, mely $f_s = 48$ kHz, tehát T_m mérési idő alatt $N = T_m f_s$ kvantálási lépcső található. Ebből ismert módon a kvantálási hiba maximális értéke [8]:

$$h = \frac{1}{N} = \frac{1}{T_m f_s} \quad (4.12)$$

A kvantálási hiba csökkentése érdekében érdemes a mérési időt minél hosszabbra választani.

A periódusidő mérését a jel negatívból pozitív irányba történő null-átmeneteinek regisztrálásával, egy periódust számláló változó null-átmenetenkénti inkrementálásával végezzük. A periódusidő két azonos irányú null-átmenet közt eltelt idő. A mérési idő és a periódusszámláló értékének hányadosa éppen a periódusidő. A mérés akkor hibátlan, ha a mérési idő az órajel és periódus szorzatának egészszámú többszöröse. A frekvenciát a periódusidőből származtatjuk, annak reciproka.

4.2.3 Effektívérték-mérés

Villamos jel esetén a jel effektív értéke fizikailag azzal az egyenfeszültséggel vagy egyenárammal egyenlő, amely egy adott ellenálláson ugyanakkora hőenergiát termel, mint a periodikusan változó feszültség vagy áram.

Az effektív érték definíciója diszkrét jelre:

$$x_{\text{eff}} = \sqrt{\frac{1}{N} \sum_{i=1}^N x_i^2} \quad (4.13)$$

Az effektív érték egy periódus mintáiból származik. A (4.13) egyenletet a programban úgy alkalmazzuk, hogy egy ehhez rendelt változóban minden mintára összegezzük a

változót az aktuális minta négyzetével. A mérési periódus végén a négyzetösszegeket a minták számával osztjuk, majd négyzetgyököt vonunk belőle.

A méréseknél érdekes lehet, hogy többszöri mérés esetén mennyire ingadozik a mérési eredmény egy jelnél. Ennek reprezentálására alkalmas a szórás. Számítása:

$$\bar{x} = \frac{1}{N} \sum_{i=1}^N x_i \quad (4.14)$$

$$\sigma = \sqrt{\frac{1}{N-1} \sum_{i=1}^N (x_i - \bar{x})^2} \quad (4.15)$$

A szórást egy függvénnyel számoljuk a programban, bemenetei az N változó és az N mintát tartalmazó tömb. A függvény először ciklikusan kiszámolja a tömb elemeinek összegét, az értéket leosztja N -nel, ez a (4.14) képlet szerinti átlagérték. Ezután újabb ciklusban a minták és az átlagértékének különbségét összegzi, ezt a részeredményt osztja $N - 1$ -gyel, majd a négyzetgyökét képezi a könyvtári sqrt függvénnyel. A függvény a szórás értékével tér vissza, és ezt kijelezhetjük a mért értékkel együtt.

A mérés szórásának számítása azért fontos, mert egy mérés bizonytalanságáról a legtöbb műszer esetén csupán a kijelzett érték ingadozása alapján lehet hozzávetőleges információt nyerni, pontos értékéről azonban nem.

4.2.4 A mérések hibája

Az eddigiekben említett mérésekben hibát okoz, hogy diszkrét idejű, mintavételezett jelekkel dolgozunk, míg a mért értékeket az analóg jelre kell megadni. A csúcsérték mérésénél a hiba az, hogy amit csúcsértékként eltárolunk, nem minden esetben a valódi maximum/minimum. Ez a hiba onnan ered, hogy két mintavétel között eltelt időben az analóg jel folyamatosan változik, és a mintavételi pontokban felvett értékektől pozitív/negatív irányba is eltérhet. Ugyanígy a null-átmenetnél is csak azt detektáljuk, hogy az előbbi mintavételi pontban a jel még negatív volt, a mostaniban már pozitív, de

azt, hogy a két mintavétel között pontosan mikor volt a jel amplitúdója nulla a megfigyelt bemeneten, nem tudhatjuk biztosan.

Ezen probléma kiküszöbölésének egyik lehetséges módja az interpoláció, amivel megbecsülhetjük, hogy két minta között, amik a maximum közvetlen közelében vannak, mi a valódi maximum várható értéke, hasonlóan a minimumot is közelíthetjük, és a null-átmenet helyét is pontosíthatjuk, ezzel csökkentve a hibát.

Az effektívérték mérésében hibát okozhat, ha nem egész számú perióduson számoljuk az effektív értéket. A mérési idő változtathatóságából ered, hogy olyan mérési időt is választhatunk, amely nem egészszámú többszöröse a mért jel periódusának, pedig minél rövidebb mérési időt választunk, annál nagyobb lehet a hiba, mert kevesebb periódus fér a mérési idő intervallumába, aminek nagyobb hányadát jelenti egy periódus töredéke. Ez kiküszöbölhető, ha úgy kompenzáljuk a mérési időt, hogy az a periódusok többszöröse legyen. Ezzel a kompenzációval rövid mérési időintervallumok esetén akár egy nagyságrenddel is csökken az effektív érték szórása. A mérési idő kompenzálását a műszer automatikusan elvégzi, amennyiben ezt beállítjuk.

Periódusidő-mérés esetén az időmérés kvantáltsága okozza a mérési hibát. Ezt a hibát lehetne tovább csökkenteni, ha a jel null-átmeneteinek időpontját becsülnénk például lineáris interpolációval.

4.2.5 A mérés pontosítása interpolációval

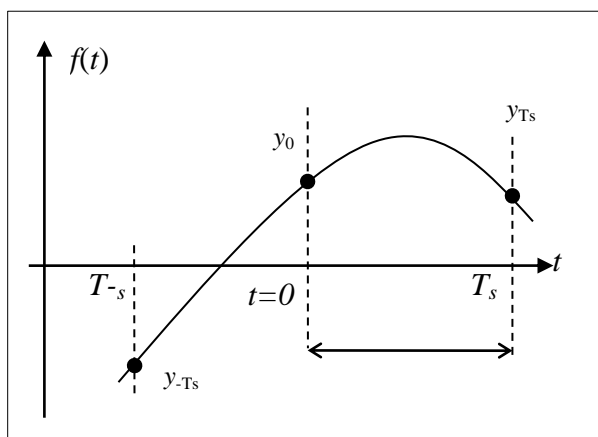
A mintavételezett jel minták közötti értékeit becsülhetjük interpolációval. Az interpoláció során egy interpolációs $i(x)$ függvénnyel közelítjük a mintavételezett $f(x)$ függvényt, úgy hogy a mintavett értékeket az interpolációs függvény is felveszi, tehát kielégíti az $i(x_i) = f(x_i)$ feltételt. Az interpolációs függvény segítségével becsülhetjük a folytonos idejű $f(x)$ -nek a mintavételezett értékei között felvett értékét.

A programban felhasznált parabolikus interpolációs függvény polinomiális, ami jó kompromisszumot jelent a számítási igény és az interpoláció minőségi jellemzői között. Polinomiális interpolációról beszélünk, amikor a közelítő függvények polinomok.

Ennek az interpolációnak a segítségével n pontra egyértelműen illeszthető egy $n - 1$ -ed fokú polinom, amely átmegy a közölt pontokon.

Másodfokú interpoláció

Másodfokú interpoláció alkalmazásakor két mintára egy másodfokú függvényt illesztünk a 4.6. ábrán látható módon.



4.6. ábra Másodfokú interpoláció

A becslő parabola középpontja a $t = 0$ pillanatban van, tőle a pozitív negatív irányban az időtengelyen egy mintavételi periódusra van egy-egy minta.

A másodfokú egyenletek általános alakja:

$$f(x) = a(x - x_1)(x - x_2) = ax^2 + bx + c$$

A keresett $f(t)$ -t a $[-T_s, T_s]$ intervallumban becslő függvény a , b , c paraméterei a (4.19), (4.20) és (4.16) egyenletekből számolhatók.

$$i(t = 0) = y_0 = c \tag{4.16}$$

$$i(t = T_s) = y_{T_s} = aT_s^2 + bT_s + y_0 \tag{4.17}$$

$$i(t = -T_s) = y_{-T_s} = aT_s^2 - bT_s + y_0 \tag{4.18}$$

(4.17) és (4.18) összegéből:

$$y_{T_s} + y_{-T_s} = 2aT_s^2 + 2y_0$$

$$a = \left[\frac{y_{T_s} + y_{-T_s}}{2} - y_0 \right] \frac{1}{T_s^2} \quad (4.19)$$

(4.17) és (4.18) különbségéből:

$$y_{T_s} - y_{-T_s} = 2bT_s$$

$$b = \frac{1}{T} \frac{y_{T_s} - y_{-T_s}}{2} \quad (4.20)$$

A programban a másodfokú interpolációt a csúcserőtelmérés pontosítására használjuk. Ez azt jelenti, hogy figyeljük az utolsó három mintát, ha teljesül rájuk, hogy

$$f(t = -T_s) \leq f(t = 0) \geq f(t = T_s)$$

tehát lokális maximumpontot találtunk, vagy épp ellenkezőleg,

$$f(t = -T_s) \geq f(t = 0) \leq f(t = T_s),$$

tehát lokális minimum pontot találtunk, akkor a jel az utolsó három mintavételezés alatt elérte a maximum vagy minimum amplitúdóját. Erre a három mintára illesztünk egy másodfokú görbét, kiszámolva az a, b, c paramétereit (4.19), (4.20) és (4.16) alapján, majd a függvény szélsőértékét keressük (4.21) szerint. Ha a kiszámolt érték nagyobb/kisebb, mint a $f(t = 0)$ értéke, akkor eltároljuk, mint a mért jel lokális maximuma/minimuma, ellenkező esetben a maximum/minimum az $f(t = 0)$ értéke marad. A mérési ciklus során ezen lokális minimumok illetve maximumok közül tároljuk a legkisebb illetve legnagyobb értékeket.

$$f'(x) = 0 = 2ax_m + b$$

$$x_m = -\frac{b}{2a}$$

$$f(x_m) = a \left(-\frac{b}{2a} \right)^2 - b \frac{b}{2a} + c = \frac{b^2}{4a} - \frac{b^2}{2a} + c$$

$$f(x_m) = c - \frac{b^2}{2a} \quad (4.21)$$

5. Rezonátoros struktúra és alkalmazásai

A rezonátoros algoritmust ismert frekvenciájú periodikus jelek Fourier-együtthatóinak rekurzív módon történő számítására alkalmazzuk. Ennek előnye DFT/FFT-vel szemben, hogy mintáról mintára rekurzív módon számolunk, tehát nem kell megvárni, míg a DFT felbontásának megfelelő számú bemenő értéket mintavételeztünk, ami nagyfelbontás esetén hosszabb ideig is eltarthat. Előnye továbbá, hogy nem az egész frekvenciatartományon számítja a jel spektrumát, hanem periodikus jelről lévén szó, csak a Fourier-együtthatókat számolja, de ezzel együtt hátránya, hogy csak ismert periódusú jel esetén alkalmazható hibátlanul. Ha pontosan ismerjük a mintavételezett jel frekvenciáját, akkor elkerülhetőek az DFT-nél fellépő tetőesés (picket fence) és szivárgás (leakage) jelenségek.

Az algoritmus ismertetéséhez először tekintsük a periodikus jelek Fourier-együtthatós felírását.

$$y_n = \sum_{i=-P}^P x_{i,n} e^{j\theta i n} = \sum_{i=-P}^P x_{i,n} c_{i,n}$$

és
$$c_{i,n} = e^{j\theta i n}, \quad (5.1)$$

ahol y_n periodikus jel, mely P számú harmonikust tartalmaz, a mintavételi frekvenciához viszonyított relatív frekvenciája $\theta = 2\pi f_s/f$, $x_{i,n}$ az i -edik felharmonikus Fourier-együtthatója, $c_{i,n}$ az i -edik harmonikus bázisfüggvénye az n . időpillanatban.

Az (5.1) egyenlet tömörebb formája:

$$\begin{aligned} \mathbf{c}_n &= [c_{-P,n} \dots c_{0,n} \dots c_{P,n}]^T \\ \mathbf{x}_n &= [x_{-P,n} \dots x_{0,n} \dots x_{P,n}]^T \\ y_n &= \mathbf{c}_n^T \mathbf{x}_n \end{aligned} \quad (5.2)$$

ahol felhasználjuk a vektor-szorzás definícióját, tehát két vektor skaláris szorzata a vektorok elmeinek páronkénti szorzatából álló összege.

Az y_n jelet úgy is tekinthetjük, mint egy lineáris rendszer kimenetét, amelynek állapotegyenlete:

$$\begin{aligned} \mathbf{x}_{n+1} &= \mathbf{x}_n \\ y_n &= \mathbf{c}_n^T \mathbf{x}_n \end{aligned} \quad (5.3)$$

Ahol az \mathbf{x}_n vektor az állapotváltozót jelöli, amely voltaképpen a jel Fourier-együtthatóit tartalmazó vektor.

A periodikus jelek állapotváltozós alakját felhasználva a Fourier-együtthatók egy úgynevezett állapot-megfigyelő segítségével mérhetőek [9]. Az állapot-megfigyelők olyan rendszerek, melyek a megfigyelendő rendszer struktúrájának és kimenetének ismeretében képesek meghatározni a megfigyelt rendszer állapotváltozóinak értékét. A rezonátoros megfigyelő a periodikus jelek koncepcionális jelmodellje alapján becsli az y_n jel x_n Fourier-együtthatóit az 5.1. ábrán látható algoritmus alapján. A megfigyelőt leíró egyenletek, az $\hat{\mathbf{x}}_n$ vektorral becsülve az \mathbf{x}_n együtthatókat [9]:

$$y'_n = \mathbf{c}_n^T \hat{\mathbf{x}}_n, \quad (5.4)$$

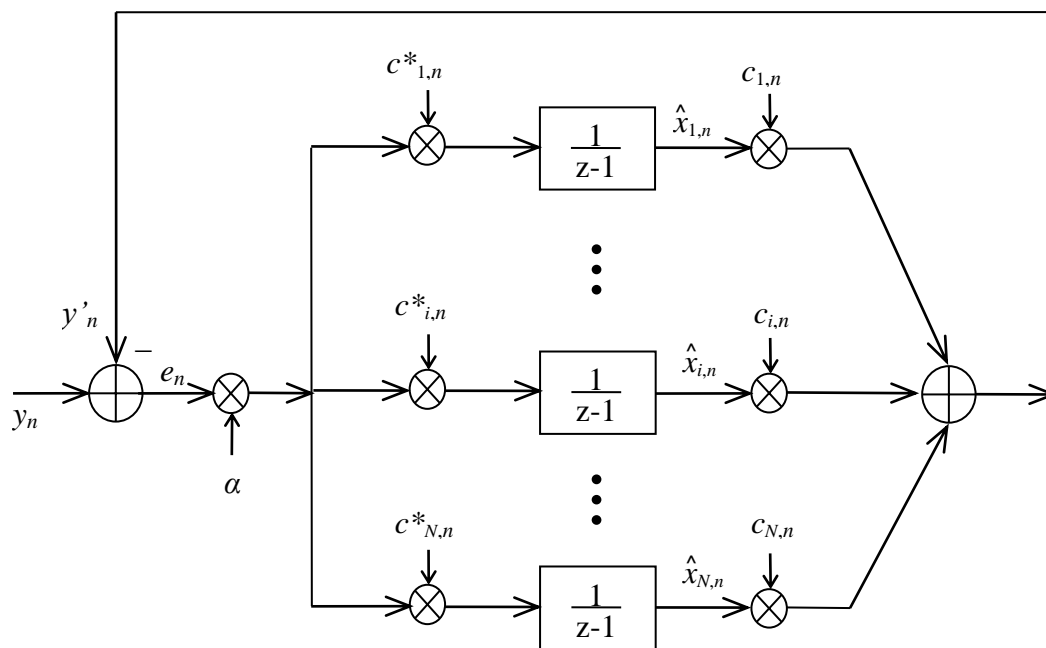
$$e_n = y_n - y'_n = y_n - \mathbf{c}_n^T \hat{\mathbf{x}}_n, \quad (5.5)$$

$$\hat{\mathbf{x}}_{n+1} = \hat{\mathbf{x}}_n + \alpha e_n \mathbf{c}_n^*, \quad (5.6)$$

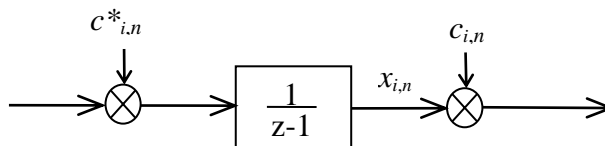
ahol a $*$ operátor jelöli a komplex konjugálást. A rezonátoros megfigyelő követi a rekurzív becslési algoritmusok szokásos alakját: a rendelkezésre álló $\hat{\mathbf{x}}_n$ becsülő paraméterek alapján becslést ad az y_n jelre. A becsült értéket y'_n -nel jelöljük. Az y_n becsült és valós értékéből kiszámolja a becslés hibáját (e_n), a hiba alapján pedig módosítja a becsülő paramétereit. Az α változó egy ún. bátorsági tényező, bizonyos határon belül minél nagyobb az értéke, annál gyorsabb a becslés, de minél kisebb annál kisebb a közelítés zavarérzékenysége.

A rezonátoros megfigyelő a mintavételezett jel Fourier-együtthatóit állítja elő a jel frekvenciájának ismeretében. A frekvencia ismeretének feltétele határt szab az

alkalmazhatóságának, de tipikusan alkalmas olyan vizsgálatoknál, amikor praktikus saját jellel gerjeszteni egy rendszert, és annak válaszát visszamérni. Ilyen alkalmazás a később megvalósított átviteli függvény mérés, impedanciamérés és torzításmérés.



egy rezonátor csatorna:



5.1. ábra A rezonátoros megfigyelő

5.1 A rezonátoros algoritmus implementációja

A következőkben a DSP-re készült, rezonátoros megfigyelőt megvalósító függvény implementációs részleteit szeretném bemutatni.

A rezonátoros megfigyelőt először torzításmérésnél használtam fel. Alkalmas függvénygenerátor periodikus jelének vizsgálatára, de a DSP által kiadott saját jel analizálására is.

A függvényben az (5.4) - (5.6) egyenleteket alkalmazom. Maximum húsz rezonátor csatornát számolunk, ennél több együttható esetén már alkalmanként túl sokáig tartott az állapotváltozók számítása, és kifutott az egy mintavételnyi időből a program lefutása. A csatornapárok maximális számát (P) a mintavételi frekvencia és a mért frekvencia határozza meg: $P = \lfloor \frac{f_s}{2f} \rfloor$. Az $\hat{\mathbf{x}}_n$, \mathbf{c}_n együtthatókat, valamint az y'_n , e_n változókat globálisan tárolom, mert újabb függvényhíváskor szükség van az előzőleg számolt értékekre a rekurzív számítási mód miatt. Az $\hat{\mathbf{x}}_n$, \mathbf{c}_n együtthatók komplex számok, a valós és képzetes részeket külön tömbökben tárolom. A \mathbf{c}_n együtthatókat a jel additív szintézissel történő előállításánál tárgyalta módon állítjuk elő. A $c_{1,n}$ komplex szám valós és képzetes részét számolom külön-külön beépített szinusz és koszinusz függvényekkel, majd ez alapján a többi $c_{i,n}$ -t az (5.7) és (5.8) azonosságok alapján.

$$\exp(j \cdot \theta \cdot (i + 1) \cdot n) = \exp(j \cdot \theta \cdot i \cdot n) \exp(j \cdot \theta \cdot n) \quad (5.7)$$

$$\exp(j \cdot x) = \cos(x) + j \sin(x) \quad (5.8)$$

Tehát a $c_{i,n}$ komplex számokat rekurzív módon tudjuk számítani egymásból egyetlen komplex szorzás segítségével. Ez számítási idő szempontjából kedvező, ugyanis a szinusz és koszinusz függvények futási ideje ennél hosszabb.

A mérés elején az $\hat{\mathbf{x}}_n$ becsült paraméterek nullával inicializálódnak, és a hibától függően addig nőnek, amíg az egyes frekvenciákon a hiba értéke közel nullává válik.

A jelet becsülő y'_n tag a becsülő paraméterek és a bázisfüggvények szorzatának összegeként számítható. Az (5.4) képlet alapján, ahogyan $\hat{\mathbf{x}}_n$ és \mathbf{c}_n , az y'_n jel is általánosságban lehet komplex szám, de fizikailag, a mért jelhez hasonlóan, valós szám kell, hogy legyen. Az y'_n jelet egy P hosszú ciklusban számoljuk úgy, hogy eleve a szorzatnak csak a valós részét számoljuk az alábbi módon:

$$y'_n = \text{Re}\{\mathbf{c}_n^T \hat{\mathbf{x}}_n\} = \hat{x}_{0,n} + \sum_{i=1}^P \text{Re}\{c_{i,n}\} \cdot \text{Re}\{\hat{x}_{i,n}\} - \sum_{i=1}^P \text{Im}\{c_{i,n}\} \cdot \text{Im}\{\hat{x}_{i,n}\} \quad (5.9)$$

Az $\hat{\mathbf{x}}_{0,n}$ a DC komponens együtthatója, ami ideális esetben elhanyagolható lenne, de fizikai mérésnél figyelembe kell venni, például az áramköri elemek offset-feszültsége miatt.

A rezonátoros megfigyelőn belül meghívjuk a torzítási tényezőt számoló függvényt is, ennek felépítését most tisztázom. A THD-t (Total Harmonic Distortion) számoló függvénynek a becsülő paraméterek valós és képzetes részeit tároló tömbökre mutató mutatókat, valamint a tömbök hosszát, azaz a rezonátor csatornák számát kell paraméterül megadni. A THD a definíció szerint:

$$THD = \sqrt{\frac{\sum_{i=2}^P V_i^2}{V_1^2}} \quad (5.10)$$

ahol $V_1 \dots V_P$ a jel harmonikus komponenseinek effektív értékét jelöli. A THD-t a programban a következő módon számítjuk:

$$THD = \sqrt{\frac{\sum_{i=2}^P \text{Re}\{\hat{x}_{i,n}\}^2 + \text{Im}\{\hat{x}_{i,n}\}^2}{\text{Re}\{\hat{x}_{1,n}\}^2 + \text{Im}\{\hat{x}_{1,n}\}^2}} \quad (5.11)$$

A rezonátoros megfigyelőt meghívva a bejövő jelre torzítást tudunk számolni, így például jelgenerátorok kimenetét tudjuk vizsgálni szinusz-, háromszög- vagy négyszögjelre, amiknek ismert a torzítási tényezője. Ezáltal jellemezhető a jelgenerátor által kiadott jel tisztasága. Alkalmazható a torzításmérés például erősítők, jelkondicionáló áramkörök linearitásának mérésére is.

5.2 Rezonátor alapú átviteli függvény mérés

Ebben az alfejezetben a rezonátoros megfigyelő segítségével megvalósított átviteli függvény mérést mutatom be.

A lineáris időinvariáns rendszereknél a harmonikus jelek kitüntetett szereppel bírnak. Ha a rendszer gerjesztése ω körfrekvenciájú, harmonikus időfüggvény az

$$x(t) = X \cos(\omega t + \alpha) \quad (5.12)$$

szerint, akkor a válasza is harmonikus lesz, ugyanazzal a frekvenciával:

$$y(t) = Y \cos(\omega t + \beta). \quad (5.13)$$

Így két valós számmal jellemezhető a rendszer, a $K = Y/X$ erősítéssel és a $\varphi = \alpha - \beta$ fázistolással. Ezt a két számot a H abszolút értékű φ fázisú komplex számba foglalva:

$$\bar{H} = K e^{j\varphi} \quad (5.14)$$

A rendszer teljes-körű jellemzéséhez úgy jutunk, ha minden lehetséges frekvencián megvizsgáljuk a K erősítését és φ fázistolását, így megadva a $K(\omega)$ amplitúdókarakterisztikát és $\varphi(\omega)$ fáziskarakterisztikát. Az amplitúdó- és fáziskarakterisztikából képzett komplex függvény a rendszer átviteli karakterisztikája:

$$H(j\omega) = K(\omega) e^{j\varphi(\omega)} \quad (5.15)$$

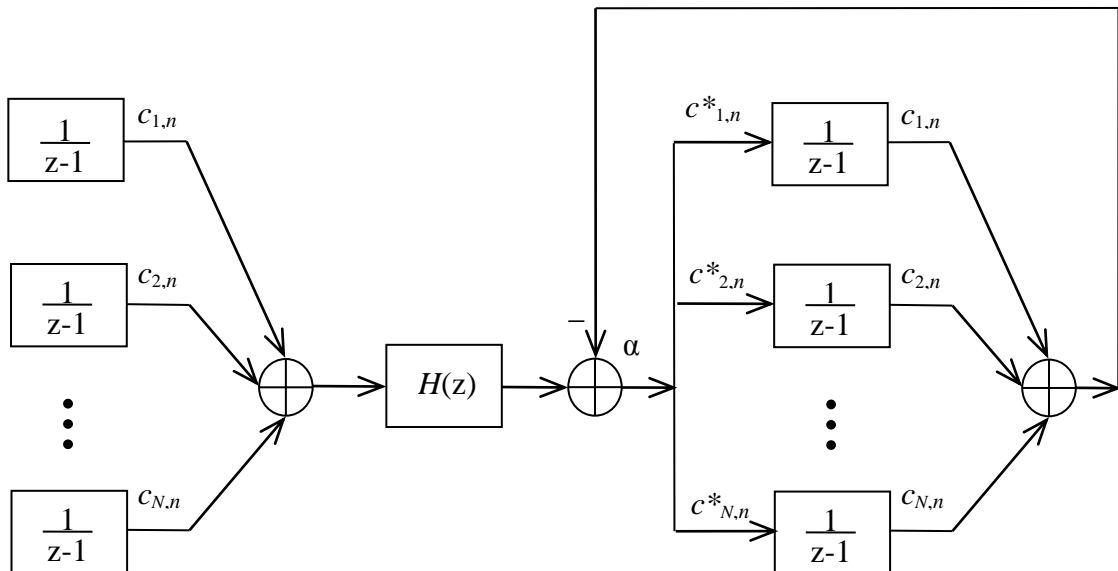
Mintavételezett rendszernél:

$$H(e^{j\vartheta}) = K(\vartheta) e^{j\varphi(\vartheta)} \quad (5.16)$$

Az átviteli karakterisztika mérése azért hasznos, mert az átviteli karakterisztika és a gerjesztés spektrumának ismeretében a válasz szorzással előállítható, ahol a ϑ jelöli a mintavételi frekvenciához képest a relatív frekvenciát.

A rezonátoros megfigyelővel megvalósított átviteli függvény mérésének módszeréről részletesen a [9] irodalomban olvashatunk. A mérési módszer a következő módon összegezhető. A rezonátoros megfigyelő segítségével a rendszert harmonikus (keskenysávú) módon gerjesztjük. Az 5.1. ábrán megismert rezonátoros megfigyelő az 5.2. ábrán láthatóra módosul. A korábbi ábrán a struktúráról tudtuk, hogy valamilyen

lineáris rendszer, aminek az állapotváltozói nem változnak az idő elteltével (időinvariáns). A jelen esetben a gerjesztésnek az egész struktúráját ismerjük, hiszen mi állítjuk elő a gerjesztő jelet, az ismeretlen tag csak a rendszer átviteli függvénye.



5.2. ábra $H(z)$ identifikációja rezonátoros megfigyelővel

A $H(z)$ ezzel a felépítéssel becsült állapotváltozók és az ismert állapotváltozók hányadosaként számítható, azaz:

$$H(z_i) = \frac{\hat{x}_i}{x_i} \quad (5.17)$$

Ekkor $x_n = 1$ választással, tehát egységnyi amplitúdójú, nullafázisú gerjesztés esetén:

$$H(z_i) = \hat{x}_i \quad (5.18)$$

\hat{x}_n a rendszer kimenetén mér szinuszos jel komplex amplitúdója (amplitúdó és fázis), x_n pedig a gerjesztés komplex amplitúdója. Definíció szerint a $H(z)$ átvitel a rendszer válaszának és gerjesztésének hányadosa. Elképzelhető lenne, hogy a rendszer bemenetére multiszinuszos gerjesztést adunk, és minden frekvencián mérjük a választ. A gyakorlatban azonban általában az alapharmonikus frekvencián történik csak

gerjesztés. Ennek oka, hogy a sok szinuszos komponensből összeállított jel csúcserőértékének nem szabad túllépnie a DA átalakító működési tartományát, így több komponens alkalmazása esetén kisebb a megengedett jelteljesítmény adott frekvenciákon, mintha csak egyetlen komponenssel gerjesztenénk, ami a jel/zaj viszonyt rontja.

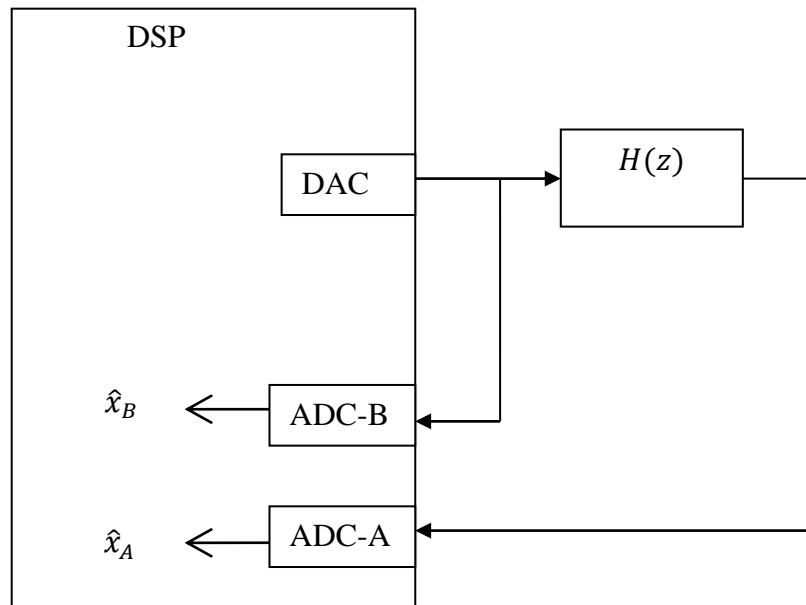
Gyakorlati mérések során hibát okozhat, hogy a jelfeldolgozó algoritmusok ki- és bemeneti értéke nem csatlakozik közvetlenül a mérendő rendszerhez, hanem a jelútban többféle komponens is található, úgymint DA átalakító, analóg jelkondicionáló áramkörök, AD átalakító. A kérdés azért is fontos, mert az adott jelfeldolgozó processzorhoz szigma-delta AD és DA átalakítók csatlakoznak, amelyek jelentős késleltetéssel rendelkeznek. A probléma megoldására az 5.3. ábrán látható elrendezést alkalmaztam. Ebben a megoldásban az x_i változókat egy másik rezonátorral közvetlenül visszamérjük, ekkor:

$$H(z) = \frac{\hat{x}_A}{\hat{x}_B}, \quad (5.19)$$

ahol x_A a rendszer kimenetén mért jel komplex amplitúdója, x_B pedig a rendszer bemenetén mért jel komplex amplitúdója. A két érték mérése egy-egy rezonátorbank segítségével történik, a két AD átalakító bemenetét feldolgozva.

A mérési eljárás úgy magyarázható, hogy láthatóan az ADC-B által visszamért \hat{x}_B jel valójában gerjesztőjelként fogható fel, tehát a $H(z)$ rendszer kimenetén mért \hat{x}_A értéket visszaosztva a gerjesztéssel, megkapjuk a $H(z)$ átviteli függvényt az adott frekvencián. A két visszacsatoló út közötti eltérést csupán a $H(z)$ átviteli függvény okozza.

A mérésnél nem foglalkoztunk a DSP ki- és bemeneti ellenállásainak hatásával. Azt feltételezzük, hogy a mérendő eszköz bemenete nagy impedanciás, a kimenete kis impedanciás. Ha nem így van, az hibát okozhat. Teljes körű megoldást az nyújtana, ha például rendelkezésre állna egy olyan analóg interfész, mely képes kisimpedanciás kimenetet, és nagy impedanciás bemenetet biztosítani.



5.3. ábra Átviteli függvény mérésének felépítése

5.2.1 Átvitelmérés a gyakorlatban

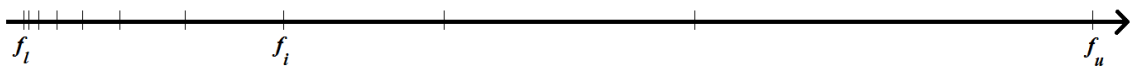
Az átvitelméréskor egy rendszer átvitelét mérjük a frekvenciatartományban. A rendszer bemenetét a DSP által előállított jel szolgáltatja, kimenetét pedig a DSP bemenetére csatlakoztatva vizsgáljuk. A frekvenciatartománybeli vizsgálatot diszkrét frekvencia pontokban végezzük, adott frekvenciájú szinuszzellel. Az átvitelmérés során felhasználjuk a rezonátoros algoritmust megvalósító függvényt, amivel lehetőség van egyszerre két rezonátorbankot üzemeltetni, azaz két csatornán mérve, mindkét jelre számolni a Fourier-együtthatókat. A dupla rezonátorbankkal kiküszöbölhetjük az AD és DA átalakítók hibáját. Amíg az egyik csatornán a vizsgált rendszer kimenetét mérjük, addig a másik csatornára a DSP saját kimenetét, a mérőjelet kapcsoljuk. Ezzel ellenőrzési lehetőséget is biztosítunk magunk számára annak vizsgálatára, hogy a megfelelő jel jutott a mért objektum bemenetére. A kimenetre állított szinusz jel a rezonátor által számolt alapharmonikus bázisfüggvénye, így a jelek szinkronban

vannak, és elkerüljük annak a lehetőségét, hogy a gerjesztő és mért jel frekvenciája elhangolódjon egymástól.

Mérés során a következő paraméterek hangolását biztosítja a rendszer:

- Gerjesztőjel amplitúdója
- Mérés időtartama
- Mérés által lefedett frekvenciasáv
- Lineáris vagy logaritmikus felbontás
 - Lineáris felbontásnál a mérési frekvenciák közötti lépésköz
 - Logaritmikus felbontásnál a mérési pontok száma

Az átvitelmérésnél a mérés fentebb látható legfontosabb paraméterei a felhasználó által állíthatók. Meghatározhatjuk, hogy a gerjesztés mekkora frekvenciatartományon történjen. Megadható az egy frekvencián történő mérés időtartama. Állítható, hogy lineárisan vagy logaritmikusan mérjük a frekvenciaskálán. Lineáris mérés során megadhatjuk a frekvencia lépésköz nagyságát, azaz hogy mennyivel növeljük a kimenetre adott jel frekvenciáját a beállított időköz elteltével. Logaritmikus frekvencia felosztásnál az osztópontok számát adhatjuk meg, amiből a program számolja ki a frekvencia növelés mértékét. Ha a frekvenciát logaritmikusan változtatjuk, az azt jelenti, hogy minden frekvenciapárra igaz, hogy azok hányadosa állandó. Ez a konstans a mérés paramétereiből származtatható. Ezt a szorzó számot a program a mérés megkezdésekor meghatározza a frekvenciahatárok és az osztópontok száma alapján, majd minden frekvencialéptetésnél az előzőleg kiszámított frekvencia értéket ezzel a szorzóval szorozza. A számítás menete:



5.4. ábra

$$f_i = f_l \cdot k^i$$

$$f_u = f_l \cdot k^G \quad \rightarrow \quad \frac{f_u}{f_l} = k^G \quad \rightarrow \quad \ln\left(\frac{f_u}{f_l}\right) = G \cdot \ln(k)$$

$$k = e^{\frac{\ln(\frac{f_u}{f_l})}{G}} \quad (5.20)$$

A levezetésben f_l és f_u jelölik rendre a mérés alsó és felső frekvenciakorlátait. G jelöli a mérési pontok számát, k a szomszédos frekvenciapárok hányadosát, valamint f_i az i -edik mérési frekvenciát.

A mérés a beállított f_l kezdő-frekvencián kezdődik, ekkor a program még nem a felhasználó által megadható időközig mér - alapesetben 0,2 másodpercig -, hanem hosszabban, egy másodpercig. Ennyi idő szükséges, hogy a visszamért jel becslése elég pontos legyen. A továbbiakban rövidebb mérés is elegendő a megnövelt frekvenciákon, mert a frekvenciák közötti lépés nem nagy, gyorsabban beállnak a rezonátorbankok.

A mérés tehát a következő módon történik. A megadott f frekvencián a rezonátorbankok beállított ideig mérik és becslik a beérkező jel Fourier-együtthatóit, és egyben előállítják a mérőjelet, ami praktikusán a becslők bázisfüggvényének valós része az alapfrekvencián, azaz $Re\{c_{1,n}\}$, ami épp egy f frekvenciájú, nulla-fázisú koszinusz. A következő frekvencián történő mérés előtt kiértékeljük az eredményeket, kiszámítjuk az átvitelt f -en. Az átvitel a mért jel és a mérő jel alapfrekvenciás komplex Fourier-együtthatóinak hányadosa. Az átvitel az f_i , i -edik mért frekvencián a rezonátornál megismert jelölésekkel, a második index most a rezonátorbankot jelöli:

$$H(z_i) = \frac{\hat{x}_{i,A}}{\hat{x}_{i,B}} = \frac{Re\{\hat{x}_{i,A}\} + jIm\{\hat{x}_{i,A}\}}{Re\{\hat{x}_{i,B}\} + jIm\{\hat{x}_{i,B}\}} \quad (5.21)$$

Az átvitel valós és képzetes részre bontva a komplex osztás szabályai szerint:

$$\begin{aligned} Re\{H(z_i)\} &= \frac{Re\{\hat{x}_{i,A}\}Re\{\hat{x}_{i,B}\} + Im\{\hat{x}_{i,A}\}Im\{\hat{x}_{i,B}\}}{Re\{\hat{x}_{i,B}\}Re\{\hat{x}_{i,B}\} + Im\{\hat{x}_{i,B}\}Im\{\hat{x}_{i,B}\}} \\ Im\{H(z_i)\} &= \frac{Im\{\hat{x}_{i,A}\}Re\{\hat{x}_{i,B}\} - Re\{\hat{x}_{i,A}\}Im\{\hat{x}_{i,B}\}}{Re\{\hat{x}_{i,B}\}Re\{\hat{x}_{i,B}\} + Im\{\hat{x}_{i,B}\}Im\{\hat{x}_{i,B}\}} \end{aligned} \quad (5.22)$$

A valós és képzetes részre történő felbontás azért szükséges, mert a DSP csak valós számokat képes számolni.

Egy adott frekvencián történő mérést követően a frekvenciát a beállítás szerint léptetjük, lineárisan a $f_{i+1} = f_i + \Delta f$ módon, vagy logaritmikusan $f_{i+1} = f_i \cdot k$ a (5.20) alapján.

5.3 Impedanciamérés

Az előző fejezetben bemutatott, általános lineáris rendszer átviteli függvényének mérésére szolgáló algoritmus egy speciális eseteként tekinthető az impedanciamérés. Ennek oka, hogy egy kétpólus árama (mint bemenet), és a rajta eső feszültség (mint kimenet) között annak impedanciája teremt kapcsolatot. Ennek eredményeként egy impedanciamérő funkció viszonylag egyszerűen megvalósítható az átviteli függvény méréséhez létrehozott műveletek segítségével.

Az áramköri modellezés és a gyakorlati kapcsolások építése során egyaránt a passzív kétpólusok a legegyszerűbb építőelemek (R, L, C). Ideális esetben az ellenállásnak csak ellenállása, a tekercsnek csak induktivitása, a kondenzátornak csak kapacitása lenne, de ezek a kétpólusok fizikailag csak közelítik az elméleti modelljüket, így például a tekercs huzalnak ellenállása, a vasznak vesztesége, a kondenzátornak pedig véges ellenállása és induktivitása is van. Az impedanciamérés szükségessége azért indokolt, mert a valóságban használt elemi kétpólusok tulajdonságainak ismerete elengedhetetlen az áramkörtervezéshez és -analizáláshoz.

Az impedanciamérésnél tehát az áramköri elemet nem egy paraméterrel modellezzük (ez lenne például kondenzátor esetén C), mert ez esetben a modell viselkedése jelentősen eltérne az objektum viselkedésétől. Emiatt a mérendő objektumot több paraméterrel modellezzük a pontosabb mérés céljából, ezzel csökkentve a modellezés hibáját.

A komplex impedanciát egyetlen frekvencián egy komplex mennyiséggel jellemezhetjük. A komplex impedancia valós és képzetes része egy-egy ideális kétpólusnak feleltethető meg. Ezért a pontosabb impedanciamérők, ahogy a

következőkben bemutatott megvalósítás is, mindig egy kételemű modell paramétereit mérik a megadott frekvencián.

A komplex impedancia váltóáramú elektromos hálózatban a komplex feszültség és komplex áram hányadosa, váltakozó áramú hálózatok ellenállása.

$$Z = \frac{\bar{U}}{\bar{I}} = R + jX = |Z|e^{j\varphi}$$

Az X reaktancia induktív jellegű, ha $X > 0$ és kapacitív jellegű, ha $X < 0$.

$$X_L = \omega L \quad (5.23)$$

$$X_C = -\frac{1}{\omega C} \quad (5.24)$$

Az impedanciákat két paraméterrel jellemezve soros R-L és soros R-C modellel közelíthetjük a valós impedanciát. A soros R-L modell esetén Z -re:

$$Z = R + j \omega L = R + jX_L$$

míg soros R-C modellnél:

$$Z = R + \frac{1}{j\omega C} = R + jX_C$$

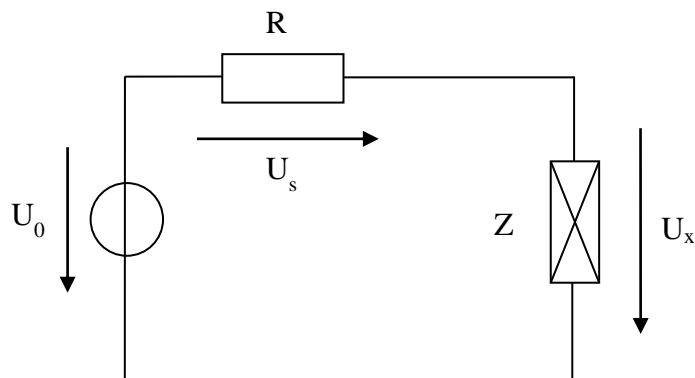
A R-C kapcsolás kis frekvencián kapacitív jelleget mutat (szakadás), nagy frekvencián ohmos jellegű, Az R-L kapcsolás kis frekvencián ohmos, nagy frekvencián induktív jellegű, szakadásként viselkedik.

Az impedancia-analizátorok általában komplex aránymérésen alapuló impedanciamérést valósítanak meg, és a megvalósított műszerben is ezt az eljárást alkalmaztam. A választás azért esett erre a módszerre, mert gyakorlatilag lényeges hardveres módosítás nélkül alkalmas a DSP kártya a feladat végrehajtására, a mérendő eszköz impedanciájára nézve elfogadható értéktartományban. A mérés általában egy ismert értékű R impedancia segítségével történik, amit legtöbbször sorba kötnek a Z_x mérendő objektummal, majd azonos nagyságú áramot bocsátanak rajtuk keresztül. A mérendő

objektumon és a precíziós ellenálláson eső feszültségek (U_x és U_s) komplex arányának mérésével az ismeretlen Z_x komplex impedancia kifejezhető:

$$Z_x = R \frac{\bar{U}_s}{\bar{U}_x} \quad (5.25)$$

A mérés alapelve a 5.5. ábrán látható

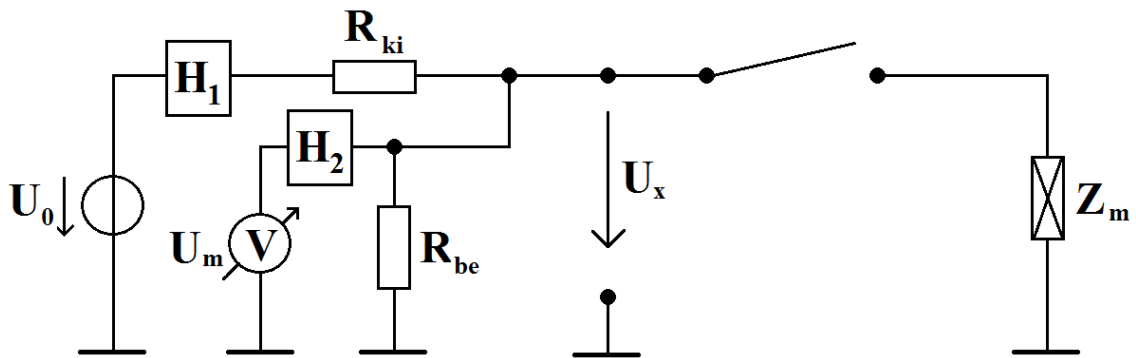


5.5. ábra

A konkrét rendszerben referenciaellenállásként a DSP kártya kimeneti ellenállását, feszültségforrásként pedig a DA átalakítót használjuk fel, így nem szükséges kiegészítő hardver a méréshez.

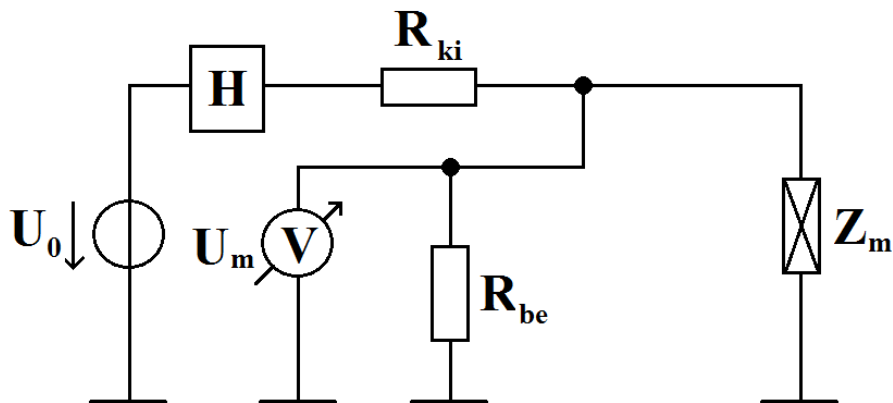
A saját alkalmazásban a (5.25) számítás némiképp bonyolódik a DSP perifériáinak felépítése miatt a következő módon.

Az 5.6. ábra modellezi a DSP mérésben részt vevő fontosabb részeinek és a mérendő impedanciának a kapcsolatát. Az R_{ki} és R_{be} a DSP kimeneti és bemeneti csatornáit terhelő ellenállások. A H_1 és H_2 a D/A és A/D átalakítók átvitelét jelzi. Az ábrán látszik, hogy a Z_m mért impedancián eső feszültség, U_x nem azonos U_m értékével, ami nálunk mérési eredményként megjelenik. U_m , gyakorlatilag a rezonátoros megfigyelő által szolgáltatott alapharmonikus Fourier-együttható. Ha az R_{be} ellenállás ∞ nagy lenne, a kimeneti ellenállást használhatnánk az (5.25) egyenletben szereplő R precíziós ellenállásként.



5.6. ábra

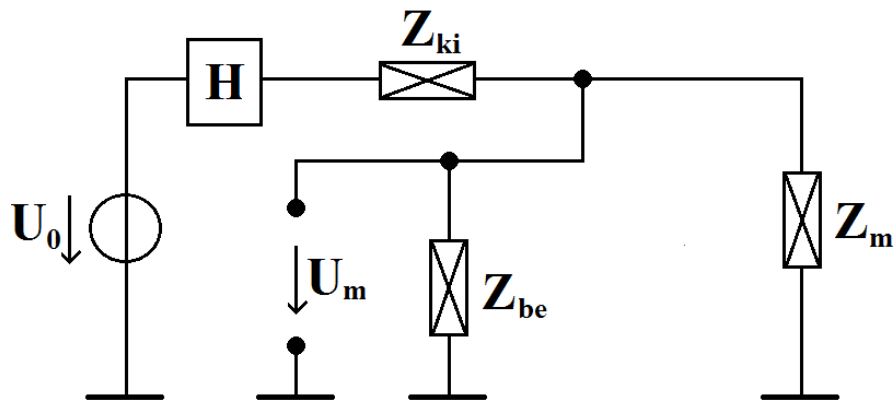
Az 5.6. ábrán látható áramkört három részre oszthatjuk. Az első rész a bemeneti U_0 feszültséget és a H_1 átvitelt, a második rész az R_{ki} , R_{be} , Z_m impedanciákat, a harmadik a H_2 átvitelt és a mért feszültséget tartalmazza. Belátható, hogy a H_2 blokk összevonható a H_1 blokkal, ezzel egy H átvitelt definiálva a rendszerben, ekkor az 5.7. ábrán látható egyszerűsített blokkséma alakul ki.



5.7. ábra

Az DSP kártya adatlapján kikereshető, hogy a kimeneti ellenállás 600Ω , a bemeneti ellenállás $11k\Omega$ névlegesen, ezek azonban hozzávetőleges értékek, amik használata rontaná az impedanciámérés pontosságát. A minél kisebb hibájú mérés érdekében tehát érdemes a kimeneti és bemeneti ellenállás értékét is ismeretlen komplex impedanciaként kezelni.

Az általános mérési elrendezés látszik a következő ábrán:



5.8. ábra általános eset

A 5.8. ábrán látható hálózat analízise alapján a mért U_m feszültség a következők szerint számolható („x” a „replusz” műveletet jelöli):

$$U_m = U_0 H \frac{Z_{be} \times Z_m}{Z_{ki} + Z_{be} \times Z_m}$$

$$U_m = U_0 H \frac{\frac{Z_{be} Z_m}{Z_{be} + Z_m}}{Z_{ki} + \frac{Z_{be} Z_m}{Z_{be} + Z_m}}$$

$$U_m = U_0 H \frac{Z_{be} Z_m}{Z_{ki} Z_{be} + Z_{ki} Z_m + Z_{be} Z_m}$$

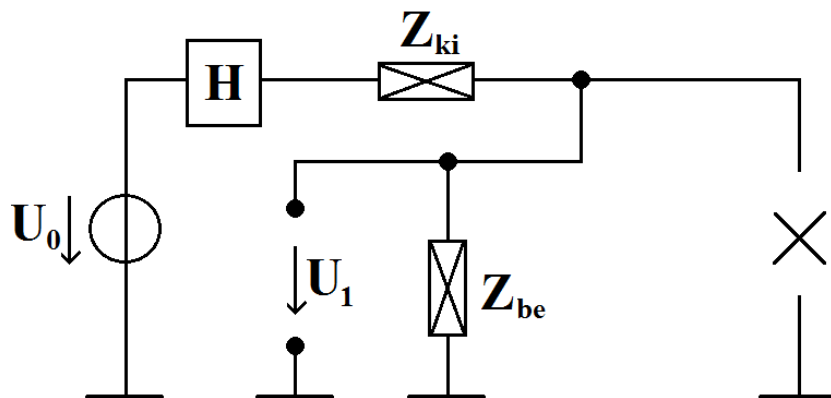
$$U_m = U_0 H \frac{Z_{be} Z_m}{Z_{ki} Z_{be} + (Z_{ki} + Z_{be}) Z_m}$$

$$U_m = U_0 H \frac{Z_{be}}{Z_{be} + Z_{ki}} \cdot \frac{Z_m}{Z_m + Z_p}$$

ahol

$$Z_p = Z_{be} \times Z_{ki} = \frac{Z_{be} Z_{ki}}{Z_{be} + Z_{ki}}$$

Nem ismerjük tehát sem a H átvitelt, sem a Z_p értékét, amik pedig szükségesek lennének a Z_m impedancia megismeréséhez.

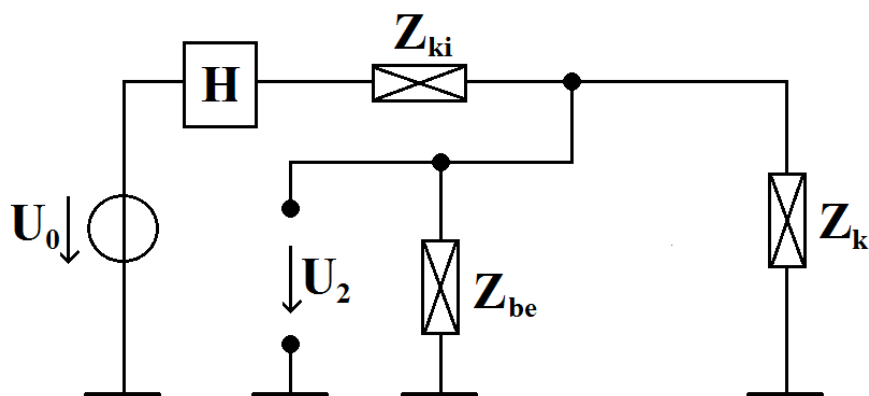


5.9. ábra Kalibráció üresjárásban

A H értékének kiszámításához a rendszert üresjárásban mérjük, ahogy az a 5.9. ábrán látható. Ekkor a DSP-vel mért U_1 feszültség:

$$U_1 = U_0 H \frac{Z_{be}}{Z_{be} + Z_{ki}} \quad (5.26)$$

Vegyük észre, hogy ez éppen az előző levezetésben az U_m kiszámításában szereplő első tag. Ebből külön H értékének ismerete már nem fontos, mert H -ra csak az U_m kiszámításához van szükségünk.



5.10. ábra

Az U_m számításához szükségünk van még Z_p értékére. A Z_p megismeréséhez az áramkört egy ismert impedanciájú, Z_k kalibrációs ellenállással zárjuk le, ezt illusztrálja 5.10. ábra. A kalibráló impedancia értékét nagy mérőpontosságú műszeren mérjük le. Az U_2 feszültség:

$$U_2 = U_0 H \frac{Z_{be}}{Z_{be} + Z_{ki}} \cdot \frac{Z_k}{Z_k + Z_p}$$

(5.26) ismeretében:

$$U_2 = U_1 \cdot \frac{Z_k}{Z_k + Z_p}$$

Z_p -re rendezve:

$$Z_p = Z_k \left(\frac{U_1}{U_2} - 1 \right) \quad (5.27)$$

Az első kalibráció során meghatározott U_1 és a második kalibráció során kiszámolt Z_p ismeretében az ismeretlen Z_m impedancia számítása a következő módon végezhető:

$$U_m = U_1 \cdot \frac{Z_m}{Z_m + Z_p}$$

$$Z_m + Z_p = \frac{U_1}{U_m} Z_m$$

$$Z_p = Z_m \left(\frac{U_1}{U_m} - 1 \right)$$

$$Z_m = Z_p \frac{1}{\frac{U_1}{U_m} - 1} \quad (5.28)$$

Habár a kalibrációhoz szükséges egy nagy pontosságú műszer vagy ismert impedancia, azonban egy kalibrálást követően a rendszer hosszú ideig használható a kalibrált értékekkel. Annak vizsgálata még nem készült el, hogy milyen időtartományban garantálható a kalibráció megfelelő pontossága.

5.3.1 DSP-s alkalmazás

Az impedancia a váltakozó áramú áramkörök ellenállása, annak értéke függ a váltakozó feszültség frekvenciájától. A mérést a programon belül az impedanciamérő függvény végzi, ami meghívja a rezonátoros megfigyelő függvényt, mivel a méréshez ismét a rezonátoros struktúrát használjuk fel a komplex feszültségek méréséhez. Az alapharmonikus összetevőt vezetjük a kimenetre, a rezonátor csatornákon visszamért értékek közül az alapharmonikus komplex Fourier-együtthatója lesz a számításhoz szükséges eredmény.

A DSP-vel történő impedanciamérés lépései a következők:

- Kalibráció üresjárásban, U_1 számítása és eltárolása, lásd 5.9. ábra és (5.26)
- Kalibráció kalibrációs ellenállással, U_1 , U_2 és Z_k ismeretében Z_p kiszámítása és eltárolása, lásd 5.10. ábra és (5.27)
- Ismeretlen impedancia mérése, U_1 , U_m és Z_p ismeretében Z_m kiszámítása (5.28) alapján

A Z_m kiszámításához felhasznált U_1 , U_2 , U_m , Z_p változók, ahogy a Z_m is, mind komplex számok. A program complextype típusú változóban tárolja őket, ami egy saját komplex számokat megvalósító struktúra, real és imag nevű float típusú változókból áll. A számítások során komplex számokat kell összegezni, szorozni és osztani egymással, így ezen műveletek megkönnyítése végett saját műveletvégző függvényeket hoztam létre. Az összeadást, kivonást, szorzást, osztást külön függvények valósítják meg, melyek bemeneti értéként két complextype változót várnak, és visszatérési értékük szintén komplex szám. Célszerű volt még a komplex számok hosszát számoló függvényt is írni, ami egy complextype változóra számolja ki az abszolútértékét, azaz a tagok négyzetösszegének gyökét, majd visszatér az értékkel egy float változóban.

Az egyes méréseket: első kalibráció, második kalibráció, ismeretlen impedancia mérése külön szeretném részletesen bemutatni, kitérve a gyakorlati használathoz szükséges megfontolásokra is.

1) Az első kalibrálás

Az első kalibrálás során a rendszert üresjárásban mérjük. Ez technikailag azt jelenti, hogy a kimeneti csatornát üresen hagyjuk. A mérés öt másodpercig tart, azaz ennyi ideig fut a rezonátoros megfigyelő, és ennyi idő alatt a rezonátoros megfigyelő már kellő pontossággal előállítja a bemeneti mintákból a becsülő paramétereket. A kimenetre az alapharmonikus került gerjesztőjelként, ennek komponenseit mérjük vissza. Az alapharmonikushoz tartozó valós és képzetes becsülő tekinthető U_1 értékének. Az impedanciamérő függvény ezt az értéket eltárolja egy complextype típusú U_1 változóban.

2) A második kalibrálás

A felhasználó feladata megadni, hogy a kimenetre milyen, praktikusán valós értékű impedancia került kalibrációs ellenállásként, ezt elmentjük Z_k -ként. Ezután indíthatjuk a második kalibrációt. Újabb öt másodpercig fut a rezonátoros megfigyelő, a végén az alapharmonikushoz tartozó becsült U_2 complextype típusú változóba mentjük. A mért feszültség és a kalibrációs ellenállás alapján (5.27) egyenletbe behelyettesítve kiszámoljuk és eltároljuk a Z_p változót.

Ezzel befejeződik a kalibráció, a következőkben már rátérhetünk a mérésre.

3) Mérés

Az impedanciamérés mérési ideje a felhasználó által állítható. A rezonátoros függvényel kiszámolt U_m és a kalibrációk során eltárolt U_1 és Z_p ismeretében a (5.28) egyenlettel Z_m meghatározható. Az eredményül kapott Z_m komplex számból a beállítás alapján soros R-L vagy soros R-C képet számolhatunk, ezt a konverziót a DSP-s program automatikusan elvégzi.

Z_m és az f frekvencia alapján a soros R-L modell paraméterei (5.23) ismeretében:

$$R = \operatorname{Re}\{Z_m\} [\Omega]$$

$$L = \frac{\operatorname{Im}\{Z_m\}}{2\pi f} [H]$$

Z_m és az f frekvencia alapján a soros R-C modell paraméterei (5.24) ismeretében:

$$R = \operatorname{Re}\{Z_m\} [\Omega]$$

$$C = -\frac{1}{2\pi f \cdot \operatorname{Im}\{Z_m\}} [F]$$

Induktivitásnál L -t érdemes -mH-ben, kondenzátornál a C -t pedig -nF-ban megadni:

$$L [\text{H}] = L \cdot 10^3 [\text{mH}]$$

$$C [F] = C \cdot 10^9 [\text{nF}]$$

5.3.2 A mérés korlátai

Próba mérések során kiderült, hogy Z_p értéke változik a frekvencia függvényében, de nem egy jól becsülhető karakterisztika mentén. Mivel nem lenne érdemes a DSP memóriáját a Z_p frekvenciafüggő értékeivel feltölteni és elfoglalni, ezért a mérési frekvencia változtatásakor újabb kétlépéses kalibráció szükséges.

6. Hangolható szűrő megvalósítása

A digitális szűrők témájában ismereteimet a [11] és [13] irodalmakból szereztem/bővíttem, valamint felhasználtam őket a szakdolgozatom elkészítéséhez irodalmi forrásként.

A digitális szűrők olyan diszkrét dinamikus rendszerek, amelyek tervezésénél az átviteli függvényt specifikáljuk. Klasszikus értelemben szűrőnek tekintjük azokat a rendszereket, amelyek amplitúdókarakterisztikája felosztható áteresztő, illetve zárótartományokra. Az áteresztőtartományokban konstans (jellegzetesen egységnyi), a zárótartományokban zérus átvitelt írunk elő. Az ilyen előírást pontosan teljesítő rendszer impulzusválasza azonban időben nem korlátos, ezért a gyakorlatban az áteresztő-, illetve zárótartományok között mindig van egy ún. átmeneti tartomány, ahol az amplitúdónak 1-nél kisebbnek kell lennie, továbbá megengedünk adott értékű ingadozást az áteresztő-, illetve zárótartományokban. Ez utóbbiban a (zérus körüli) ingadozást elnyomásnak nevezzük.

A digitális és analóg szűrők felhasználása három alapvető lépésre bontható:

1. specifikáció
2. tervezés
3. implementáció

A szakdolgozatomban alapvetően a 2. és 3. pont megvalósítása volt a cél. A specifikációt a felhasználó adja, és ez alapján kell a megfelelő lépéseket végrehajtani ahhoz, hogy a gyakorlatban is működő szűrőt tudjunk a jelfeldolgozó processzoron megvalósítani.

A fejezet következő szakaszában rövid leírást adok a digitális szűrőkhöz kötődő alapvető ismeretekről, amelyeket felhasználtam a feladat megoldásához, illetve szükségesek a tervezési lépések átlátásához, és a magabiztos gyakorlati alkalmazáshoz.

Az áttekintés során a következő jelöléseket alkalmazom.

Adva van egy általános $x(t)$ folytonos idejű analóg bemenő jel $X_A(\Omega)$ Fourier-transzformáltjával, ahol A index és az $\Omega = 2\pi f_A$ a klasszikus folytonos idejű Fourier-transzformáltra, ill. körfrekvenciára utal. Ahhoz, hogy ebből a folytonos idejű jelből egy diszkrét idejű $x(n)$ sorozatot kapjunk, az $x(t)$ jelet egyenletes T időközökben mintavételezni kell. A T időt mintavételi periódusidőnek, az ismétlődésnek megfelelő $\Omega_s = 2\pi/T$ frekvenciát pedig mintavételi frekvenciának nevezzük. A $x(t = nT) = x(nT)$ mintasorozatot mintavételezett jelnek nevezzük. A $T = 1$ választás a frekvenciát illetően azt jelenti, hogy az Ω_s mintavételi frekvenciának fog megfelelni a 2π relatív frekvencia, az $\Omega_s/2$ ún. Nyquist-frekvenciához pedig a π relatív frekvencia tartozik.

Egy lineáris diszkrét idejű rendszer átviteli függvénye z -ben racionális törtfüggvény:

$$H(z) = \frac{B(z)}{A(z)}, \quad (6.1)$$

ahol $B(z)$ és $A(z)$ polinomok. Az átviteli karakterisztikához $z = e^{j\vartheta}$ helyettesítéssel jutunk, ahol ϑ a diszkrét körfrekvencia. A folytonos rendszerekkel való kapcsolatban:

$$\vartheta = 2\pi \frac{f}{f_s}, \quad (6.2)$$

ahol f a frekvencia, f_s pedig a mintavételi frekvencia. Látható, hogy $H(\vartheta)$ 2π -ben periodikus függvény, vagy másként az átviteli karakterisztika f_s -sel periodikus.

Általánosságban a bemeneti $x[k-i]$ és a kimeneti $y[k-i]$ sorozatok között egy N -ed rendű lineáris differenciaegyenlet teremt kapcsolatot a

$$\sum_{i=0}^N a_i y[k-i] = \sum_{i=0}^M b_i x[k-i] \quad (6.3)$$

összefüggésnek megfelelően.

A digitális szűrők számítása általában (6.3) egyenlet $y[k]$ -ra rendezésével (6.4) alapján történhet, mely a gyakori $a_0 = 1$ választással:

$$y[k] = \sum_{i=0}^M b_i x[k-i] - \sum_{i=1}^N a_i y[k-i], \quad (6.4)$$

ahol $x[k]$ és $y[k]$ rendre a gerjesztés és a válasz mintái az k . időpillanatban, az a_i és b_i konstansok pedig rendre $A(z)$ és $B(z)$ együtthatói.

6.1.1 Diszkrét idejű szűrők osztályozása

Attól függően, hogy a jelsorozatok amplitúdója kvantált vagy nem kvantált, digitális illetve diszkrét idejű analóg rendszereket különböztetünk meg. A tervezési részproblémák eltérnek ugyan, de az alapstruktúra, a diszkrét idejű tervezési feladat ugyanaz lesz. A szűrőknek három osztályozási rendszere terjedt el. Az ARMA típusú rendszerek azon a tényen alapulnak, hogy a kauzális, lineáris és állandó együtthatós, időinvariáns rendszerek a (6.5) szerinti N -ed fokú állandó együtthatós differenciaegyenlettel írhatók le:

$$\begin{aligned} a_0 y[k] + a_1 y[k-1] + \dots + a_N y[k-N] &= \\ &= b_0 x[k] + b_1 x[k-1] + \dots + b_M x[k-M]. \end{aligned} \quad (6.5)$$

A válasz kifejezésében a $b_i x[k-i]$ alakú tagok összege a gerjesztés valamilyen mozgó átlagának (MA) tekinthető, az $a_i y[k-i]$ tagok pedig a visszahatást fejezik ki. A b_0, b_1, \dots, b_M együtthatókat előreecsatolási vagy Mozgó Átlagú (MA) konstansoknak hívjuk, míg az a_0, a_1, \dots, a_N együtthatókat visszacsatolási vagy Auto-Rekurzív (AR) konstansoknak. Így AR rendszerről beszélünk, amikor a b_0 kivételével mindegyik b_i együttható zérus, és MA rendszerről, amikor az a_0 kivételével mindegyik a_i együttható zérus. Az AR és MA rendszerek speciális esetei az ARMA („Auto Regressive, Moving Average”) típusú rendszereknek.

Az ARMA rendszerek, valamint a szűrők másik osztályozási rendszere az impulzusválasz hossza alapján történő megkülönböztetés.

Az egységimpulzusra adott válaszsorozat hosszúságától függően két alapvető, jól elhatárolható osztályt különböztetünk meg: a véges impulzusválaszú FIR (Finite

Impulse Response) és a végtelen impulzusválaszú IIR (Infinite Impulse Response) rendszereket, ill. szűrőket. Ebben az osztályozásban elfoglalt helyüktől függően a vizsgált rendszerek tervezési apparátusa és műszaki paraméterei jelentősen eltérnek egymástól.

A FIR rendszerek definíciójaként felírható, hogy impulzusválaszokra igaz:

$$h(n) \neq 0, \quad \text{ha} \quad N_1 \leq n \leq N_2$$

és

$$h(n) = 0, \quad \text{ha} \quad \begin{cases} -\infty < n < N_1 \\ N_2 < n < \infty \end{cases} \quad (6.6)$$

A harmadik osztályozási rendszer a szűrők realizálási technikája alapján történik, a rendszer differenciaegyenletének felírásától függően megkülönböztetünk rekurzív és nemrekurzív rendszereket. A korábban megismert MA rendszerek nemrekurzívak, mert a visszacsatolási együtthatók zérusok. A FIR és nemrekurzív, ill. az IIR és rekurzív fogalmakat gyakran keverik, mert a gyakorlati esetek túlnyomó többségében, a FIR szűrőket nemrekurzív módon, az IIR szűrőket rekurzív módon valósítjuk meg. Fontos azonban, hogy mind a FIR, mind az IIR szűrőket rekurzív vagy nemrekurzív technikával is meg tudjuk valósítani.

Stabilis egy rendszer, ha korlátos bemenő jelsorozatra korlátos kimeneti sorozatot kapunk válaszképpen. Egy lineáris időinvariáns rendszer akkor és csak akkor stabil, ha S alábbi értéke korlátos, vagyis

$$S = \sum_{k=-\infty}^{\infty} |h[k]| < \infty. \quad (6.7)$$

A nemrekurzív rendszerek mindig stabilak, míg a rekurzív rendszerek a kimeneti visszacsatolás miatt csak feltételesen stabilak.

A (6.3) egyenlet $N = 0$ helyettesítéssel a FIR szűrők nemrekurzív leíró egyenletét adja.

$$y[k] = \frac{1}{a_0} \sum_{i=0}^M b_i x[k-i] \quad (6.8)$$

Ugyanez az egyenlet $N \neq 0$ esetben éppen az IIR szűrők rekurzív leírása.

A digitális szűrők megjelenésekor az analóg tervezési módszerek már jól kidolgozottak voltak, ezért a legkézenfekvőbb módszer a digitális szűrők tervezésére az analóg szűrők transzformálása.

A digitális IIR szűrők tervezésének lépései:

- Specifikáció megadása, és a bilineáris transzformáció frekvenciatorzításának megfelelő előtorzítása,
- Analóg szűrő tervezése,
- bilineáris transzformáció.

Az analóg szűrő tervezésének lépései:

- a specifikáció transzformálása a referens aluláteresztő specifikációba,
- a kiválasztott approximáció alapján a karakterisztika tervezése,
- a referencia aluláteresztő szűrő visszatranszformálása.

Az egyes approximációk az ideális aluláteresztő karakterisztikát közelítik valamilyen matematikailag definiált módon.

Az IIR szűrőkkel szembeni előnyei miatt FIR szűrő felhasználása mellett döntöttem a DSP-re készült programban. A legfontosabb előnyei, melyek megalapozzák a döntésemet:

- mindig stabil
- elfogadható számításigényű implementáció
- lineáris fázismenet tervezhető

A (6.5) egyenlet szerint a kimenet a bemenet és a b_i együtthatókészlet diszkrét konvolúciójaként áll elő. Ez viszont azt is jelenti, hogy a b_i együtthatókészlet valójában az adott FIR szűrő impulzusválasza is.

A FIR szűrők egyik előnye az IIR szűrőkkel szemben, hogy könnyen biztosítható a lineáris fázismenet, azaz a szűrő tetszőleges frekvenciájú bemenőjelet ugyanolyan mértékben késleltet. Ennek a feltételeit a következőkben röviden ismertetem a [11] és [13] irodalom alapján.

A $H(z)$, ill $H(e^{j\vartheta})$ függvényekkel jellemzett FIR szűrők τ_f fáziskésleltetése és τ_c csoportfutási (csoportkésleltetési) ideje a $\varphi(\vartheta) = \arg[H(e^{j\vartheta})]$, fáziskarakterisztika ismeretében

$$\tau_f = -\frac{\varphi(\vartheta)}{\vartheta} \quad \text{és} \quad \tau_c = -\frac{d\varphi(\vartheta)}{d\vartheta} \quad (6.9)$$

összefüggésekkel definiálható.

Ha előírjuk, hogy τ_c csoportfutási idő a frekvenciától független α konstans legyen akkor

$$\varphi(\vartheta) = \beta - \alpha\vartheta, \quad (6.10)$$

ahol β is egy frekvenciafüggetlen konstans. Ebből a $h(n)$ együtthatókra a

$$\sum_{i=0}^{N-1} h(n) \sin[(\alpha - i)\vartheta - \beta] = 0 \quad (6.11)$$

megkötést kapjuk. A részletes levezetés megtalálható a [11] irodalomban.

Ha a FIR szűrők fázisfüggése ϑ -ban lineáris, akkor τ_f és τ_c is $-\alpha$ értékű lesz. Az (6.11) egyenlet $\beta = 0$ esetén a következő feltételekkel teljesül:

$$\alpha = (N - 1)/2 \quad (6.12)$$

és

$$h(n) = h(N - 1 - n) \quad 0 \leq n \leq N - 1, \quad (6.13)$$

ill. az egyenlet egyetlen $\beta = 0$ -tól eltérő megoldása esetén, amikor $\beta = \pm\pi/2$:

$$\alpha = (N - 1)/2$$

és

$$h(n) = -h(N - 1 - n). \quad (6.14)$$

A fentiekből következik, hogy adott N fókuszszámra csak egyetlen α érték mellett kapunk lineáris fázismenetet a (6.11) szimmetriafeltételek mellett. A $h(n)$ impulzusválasz sorozat szimmetrikus, és szimmetriatengelye páros értékű N -nél a két középső mintára esik, páratlanértékű N -nél egybeesik a középső $(N - 1)/2$ indexű mintával. Az (6.14) teljesülésekor is szimmetrikusak az impulzusválasz mintái, de a tükrözésen túlmenően a minták előjele is ellentétesre változik, ilyenkor antiszimmetrikus impulzusválaszról beszélünk. Mivel a FIR szűrők b_i együtthatói megegyeznek az impulzusválasz értékeivel: $b_i = h(i)$, így a fent bemutatott kényszerek igazak a b_i együtthatókra is.

A lineáris fázismenetű FIR szűrőknek négy fajtája van, N paritásától és $h(k)$ mintáinak szimmetriájától függően.

6.1.2 Szűrőtervezés ablakozással

A FIR szűrők tervezésére, tehát a specifikációk alapján a szűrőegyütthatók kiszámítására a leggyakoribb eljárások a következők:

- szűrőtervezés ablakozással
- szűrőtervezés Remez algoritmussal
- szűrőtervezés legkisebb négyzetes hibájú illesztéssel

Szakdolgozatom elkészítése során az ablakozással történő szűrőtervezést választottam.

Ennek okai a következők:

- jól kidolgozott, robusztus algoritmus
- DSP-n történő megvalósításhoz rendelkezésre állnak a szükséges DFT és inverz DFT műveletek
- egylépéses, nem iteratív algoritmus, így konvergencia probléma nem jelentkezik
- determinisztikus számítási idő (real-time rendszerben kell implementálni)

A következő szakaszban a [13] irodalom alapján ismertetem az ablakozással történő tervezés lépéseit.

Az ablakozással történő szűrőtervezés az ún. frekvencia-mintavételi eljárás továbbfejlesztése. A FIR szűrők együtthatói megegyeznek az impulzusválaszokkal, eszerint tehát a specifikáció (mint ideális átviteli karakterisztika) egyszerű inverz Fourier-transzformációjával előállítható a keresett együtthatókészlet.

A szűrőtervezés lépései:

1. A szűrőspecifikáció megadása egy vektorban. A vektor elemei valósak, és szimmetrikusak a DFT-nek megfelelő módon:

$$D(n) = D(N - n), n = 1 \dots N/2,$$

ahol $D(n)$ a specifikáció egy mintája, N a DFT pontszáma és páros. Ez utóbbi nem feltétlenül szükséges, de a tárgyalást megkönnyíti. Nem is megszorítás, mert N -et célszerű nagyra választani, megvalósítandó szűrő együtthatóinak számánál lényegesen nagyobbra, mely biztosítja a megfelelő felbontást a frekvenciatartományban. $D(0)$ a szűrő DC átvitele. Egy tipikus szűrőspecifikáció a 6.1.a ábrán látható.

2. IDFT alkalmazásával kapunk egy N hosszúságú impulzusválaszt, amely a prototípus szűrő:

$$h_0 = \text{IDFT} \{D\}$$

Ennek az impulzusválasznak valósnak kell lennie, és szimmetrikusnak a 0 időpillanatra. Minthogy azonban DFT-t alkalmaztunk, a negatív időnek a prototípus szűrő második felének mintái felelnek meg, ahogyan a 6.1.b ábrán látható.

3. Legyen az M páratlan szám a megvalósítandó szűrő együtthatóinak száma. (A tervezés természetesen megoldható páros M -re is, de így lényegesen egyszerűbbek az összefüggések. Mivel M rendszerint több 10, ez nem okoz gondot.) Az N mintát csonkolnunk kell úgy, hogy M mintát is kapjunk, és az előírt karakterisztikát is minél jobban közelítsük. Az N mintát úgy rendezzük sorba, hogy a prototípus szűrő második felének utolsó $(M - 1)/2$ mintája

után illesztjük a szűrő első felének első $(M - 1)/2 + 1$ mintáját, így először a prototípus szűrő M mintáját szimmetrikussá tesszük a zérus időpont körül, majd eltoljuk $(M - 1)/2$ mintával a sorozatot, így a negatív időpontokban zérus lesz az impulzusválasz, ezzel biztosítva a szűrő kauzalitását. Az N minta M -re csonkolása egy négyszög alakú ablakkal történő szorzást jelent. Az előállított impulzusválasz tehát:

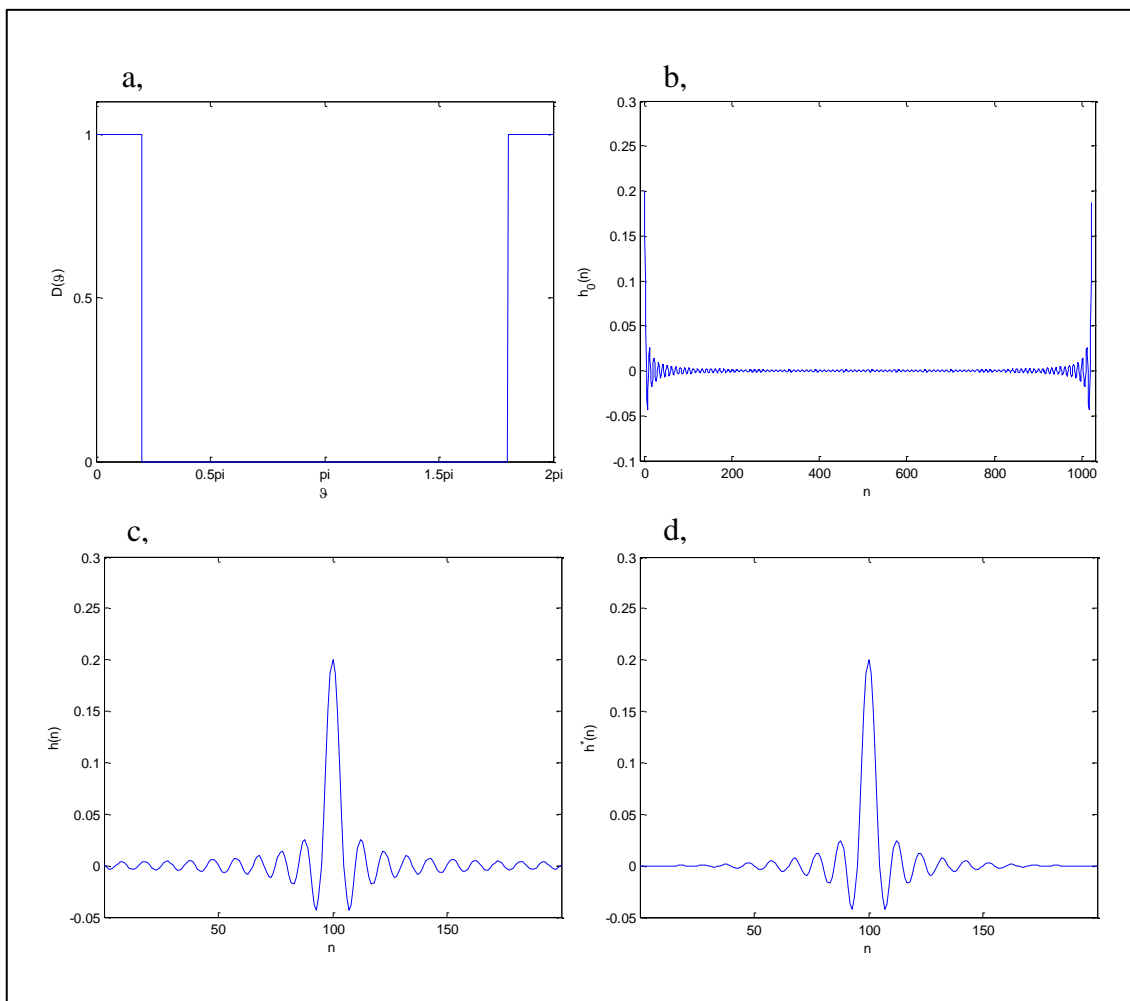
$$h(n) = \left[h_0\left(N - \frac{M-1}{2}\right), \dots, h_0(N - 1), h_0(0), h_0(1), \dots, h_0\left(\frac{M-1}{2}\right) \right] \quad (6.15)$$

Ez látható a 6.1.c ábrán. Az $(M - 1)/2$ eltolás miatt a szűrő karakterisztikája már nem lesz valós, de a (6.13) egyenlet alapján lineáris lesz a fázismenete. A szűrő karakterisztikája azonban a D specifikáció és a $h(n)$ impulzusválasz N pontos DFT-je alapján ellenőrizhető.

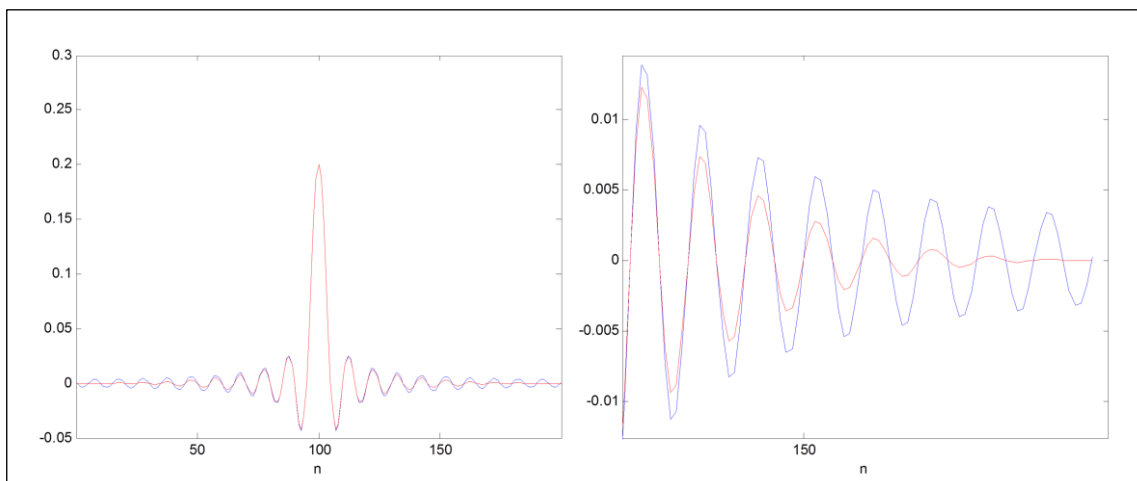
4. Az impulzusválasz mintáinak csonkítása a karakterisztikában túllendülést és lengéseket okoz, emiatt a karakterisztika a levágási pontok közelében jelentősen eltérhet a specifikációtól, még nagy M -ek esetében is. Ez az ún. Gibbs-oszcilláció. A jelenség csökkenthető, ha az előzőleg alkalmazott négyszög ablakfüggvény helyett másféle ablakfüggvényt választunk, és a $h(n)$ vektort megszorozzuk ezzel a szintén M hosszúságú függvénnyel (pl. Hanning-ablakkal):

$$b_n = h^*(n) = w(n)h(n), \quad n = 0 \dots M - 1,$$

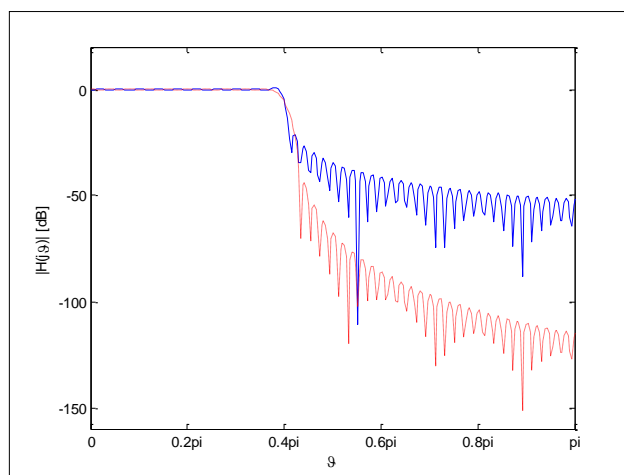
ahol $w(n)$ az ablakfüggvény. A 6.1.d ábra az ablakozott impulzusválaszt mutatja. A karakterisztika ismét a $h^*(n)$ impulzusválasz N pontos DFT-je alapján ellenőrizhető. Ha a szűrő valamilyen tervezési igényt nem elégít ki, az ablakfüggvény vagy M értéke módosítható, ilyenkor a 3. és 4. pont szerinti műveleteket újra el kell végezni.



6.1. ábra Szűrőtervezés ablakozással (a) aluláteresztő specifikáció, $N=1024$; (b) prototípus szűrő; (c) a szűrő impulzusválasza négyszög ablakkal, $M=201$; (d) a szűrő impulzusválasza Hanning-ablakkal



6.2. ábra Az ablakozás hatása az impulzusválaszra; Kék: ablakozás nélkül,
piros: ablakozott



6.3. ábra Ablakozás hatása az átviteli karakterisztikára; Kék: ablakozás nélkül,
piros: ablakozott

A 6.2. ábrán látható módon, az ablakozás hatására az impulzusválasz szélei jobban belesimulnak a környező nulla szintbe, ezáltal növeli az átmeneti tartomány meredekségét, tehát végeredményben a zárótartományban az elnyomást, melyet a 6.3. ábra szemléltet.

6.1.3 FIR szűrő megvalósítása

A FIR szűrő tervezése és a szűrés folyamata a programon belül elkülönül. A tervezést minden újabb szűrő specifikáció megadásakor lefuttatjuk a main függvényben, így váratlan megszakításkor sem vesznek el az eddig kiszámolt adatok, és nem akadályozzák egymást a működésükben. A szűrést azonban a process függvény hívja meg minden bejövő adatra, minden megszakításra. A tervezésnél nem szempont, hogy az egy periódus alatt lefusson, mivel csak a szűrés elején kell elvégezni. A process függvény a szűréssel együtt azonban értelemszerűen nem tarthat tovább, mint egy órajelperiódus, mert az hibát okozna a bemeneti/kimeneti adatokon végzett műveletekben.

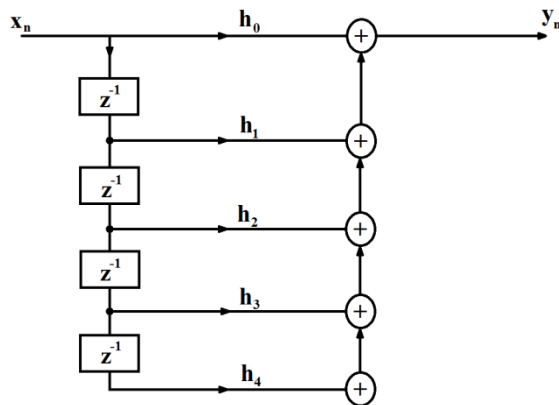
A FIR szűrő tervezésekor négyféle átviteli karakterisztikát adhatunk meg, alul-, felül-, sáváteresztő, sávzáró, az ezeket meghatározó frekvenciákat, valamint a szűrő együtthatóinak számát, szigorúan páratlan számként. A frekvenciatartományban a szűrőt úgy képzeljük el, hogy a kívánt frekvenciákon tökéletesen átereszt, 0 dB csillapítással, a többi frekvencián pedig a jel 0-szorosát engedi át, végtelen csillapítással. A karakterisztikát ezen megfontolásból egyesekkel és nullákkal alakítjuk ki, és ebből számoljuk a szűrő súlyfüggvényének együtthatóit frekvenciatartományból időtartományba transzformálással, inverz Fourier-transzformációval. A fordítóprogramban több inverz FFT-t megvalósító függvény is rendelkezésre áll, ezek közül a `ifft1024` függvényt használom, ami 1024 pontra számolja ki az időtartománybeli átvitelt. Ez a függvény az `argumentumlistájában` két, egyenként 1024 elemű `complex_float` típusú tömbre mutató mutatót vár. A `complex_float` típus a komplex számokat megvalósító struktúra, amely `real` és `imag` nevű float változóból áll. A rendelkezésre álló memóriaméret miatt esett a választás az 1024 pontos inverz FFT-re, mert az ez által felhasznált 1024 elemű komplex tömb a legnagyobb méret, ami még elfér a memóriában a program által használt egyéb adattömbök mellett. A szűrővel a mintavételi tétel miatt a mintavételi frekvencia felénél kisebb frekvenciájú jelek szűrhetők. Valós átviteli karakterisztikát definiálva a tömbnek a képzetes, `imag` együtthatóit 0-nak definiálom, a következő bemutatásban a változók a tömb valós, `real`

elemeit jelentik. Mivel az $\text{fft}(1024)$ függvény 1024 pontos karakterisztikát számol, az f_s -t 1024-re normáljuk. Aluláteresztő szűrő esetére tekintsük a tömb felépítését. Ha a beállított vágási frekvencia f_v , akkor a szimmetria feltételt szem előtt tartva a 0-dik elem 1 (DC együttható), a további 1023 elem első és utolsó f_v/f_s része 1, a maradék pedig 0 értékű lesz. Az transzformációt elvégezve helyes bemeneti szűrőkarakterisztika esetén az eredményként kapott szűrőegyütthatók képzetes értéke nem tér el 0-tól, így a továbbiakban is a struktúra valós értékeit használjuk fel. Ezzel elkészült a prototípus szűrő. A beállított M pontos felbontású szűrőt, ahol M legfeljebb 201, a prototípus szűrő valós értékeiből állítjuk össze, már nem `complex_float`ként, hanem `float`ként, a felbontásnál megállapított számú minták egymáshoz fűzésével, a (6.15) által ismertetett módon. Utolsó lépésként Hanning ablakozással simítjuk az átvitelt, végigfutunk a szűrőegyütthatókon, és egyenként megszorozzuk őket az M hosszúságúra tervezett Hanning ablak aktuális léptetett értékével.

Szűrésnél a szűrő kimenetét konvolúcióval állítjuk elő, a bemenő jel mintáiból és a szűrő súlyfüggvényéből. A szűrést végző függvény a következő paramétereket kapja:

- szűrőegyütthatókat tartalmazó buffer címe
- bejövő adatokat tartalmazó buffer címe
- aktuális adat pozíciója a bejövő adatokat tartalmazó bufferben
- szűrő hossza

A szűrő működését a 6.4. ábra mutatja be szemléletesen. A szűrő késleltető elemekből áll, melyek kimeneteit különböző súlyozással összegezzük. Az ábrán a késleltetők mindegyike egy-egy szám tárolására képes, amelyek minden lépésben egy késleltetővel tovább lépnek.



6.4. ábra

A késleltetést egy erre rendelt tömbbel valósítom meg, ez tartalmazza az utolsó M számú bemeneti mintát. A konvolúciót megvalósító függvény rendelkezésemre állt, amit assembly nyelven írtak. A szűrőegyütthatók az ábrán is látható módon súlyozzák a bemenő jelet, ez azt jelenti, hogy a bemeneti jel mintáit léptetjük, azaz ami előbb a tömb i . eleme volt, most az $i - 1$. eleme lesz. Ezt a programban megvalósíthatnánk egy olyan veremszerű tömbbel, aminek mindig eggyel eltoljuk az értékeit, azaz az utolsó „kiesik” az alján, a jelenlegi minta pedig a tetejére kerül a veremnek. Ebben az esetben minden új beérkező mintát úgy tudnánk a tömbbe menteni, hogy egy ciklusban az M hosszú tömböt önmagába másoljuk, az M . helyre az $M - 1$. elemet végig az első elemig, majd a nulladik indexszelésű helyre mentenénk a frissen mért értéket. Ez a bufferelési megoldás hátrányos lenne a programunkban, mert feleslegesen sok időbe kerül. Ehelyett a megoldás helyett egy M hosszúságú tömböt használok úgynevezett cirkuláris bufferként, ahol mindig a legrégebbi minta helyére kerül az új minta, és a legfrissebb minta indexét egy változóban tárolom és léptetetem. A cirkuláris buffer használatát az indokolja, hogy ennek kezelését a DSP hardveresen is támogatja, ezáltal jelentősen felgyorsítva a műveletek elvégzését. Végül a konvolúciós függvény megkapja a szűrőegyüttható tömböt, a bemeneti értékeket tároló tömböt és a tömbben az aktuális bemenet által elfoglalt hely címét.

Ha a szűrő bemenetére (egy impulzusnak megfelelő) egyetlen 0-tól különböző bináris szám érkezik, a kimeneten annyi (nyilvánvalóan véges) ideig kapjuk a megfelelően súlyozott bináris számokat, ameddig az összes késleltető ki nem ürül.

7. Kapcsolat a PC és a DSP között

A PC és DSP kártya közti kommunikáció teszi lehetővé, hogy a feldolgozott adatokat megjelenítsük, és a program futása közben állítsuk a mérési és a jelgeneráló beállításokat. Egy egyszerű kommunikációs felületet alakítottam ki, ami a Windows *HyperTerminal* nevű programjára épül, melyen keresztül soros port segítségével valósul meg az adatcsere. A HyperTerminal program a Windows operációs rendszer része, így a legtöbb PC-n megtalálható. Maga a program karakterek küldését és fogadását végzi, tehát bármilyen terminálprogram felhasználható a célra. A rendszer magában hordozza egy felsőbb szintű vezérlőprogram létrehozását is, mely például egy GUI (Graphical User Interface) segítségével előállítja a vezérlő parancsokat és értelmezi a DSP által küldött eredményeket.

A HyperTerminal-ban megjelenő adatok akár további feldolgozást is lehetővé tesznek, ha azokat elmentjük egy szöveges fájlba, majd a fájlt feldolgozzuk, például MATLAB segítségével. Mivel a megjelenített adatok formátuma kötött, így ez viszonylag egyszerűen elvégezhető. A dolgozatban közölt eredmények is ilyen módszerrel készültek.

7.1 Beolvasás

A felhasználó által küldött parancsszavak beolvasása a következő módon történik a DSP-n. A PC felől a kommunikációs kártyán keresztül sorosan érkező ASCII kódolású adatokat egy string tömbbe olvassuk, amelyből a karakter típusú változók könnyen kiolvashatók. Ebben a tömbben található tehát a felhasználó által küldött konfigurációs parancs. A parancsok tartalmazhatnak szöveges formátumban érkező számokat is, amiket egy függvény segítségével alakítunk float vagy integer típusú változókká, melyek aztán a program bármely fázisában kiolvashatóak.

A parancsok alapvetően a bemutatott funkciók szerint oszthatóak csoportokra:

- vizsgálójelek
- egyszerűbb mérések
- torzításmérés

- impedanciamérés
- hangolható szűrő

paramétereinek állítása.

A különböző funkciókhoz köthető parancsok más betűvel kezdődnek. Az egyszerűbb mérési beállítások módosításához használt parancsok M, a jelgeneráláshoz köthető parancsok G, a torzításmérőhöz tartozó parancsok R, az átviteli függvényt mérő parancsok A, az impedanciamérést beállító parancsok I, míg a szűrő paramétereit állító parancsok F betűvel kezdődnek. Az M, G, R, A, I és F betűket követő karakterek a paraméterbeállításokat tartalmazzák a következő formában (szóközök nélkül):

paraméter kód Paraméterérték

Egy parancsban több ilyen paraméter kód-paraméterérték pár szerepelhet. A paraméter kódok nagyrészt egybetűs kódok, a paraméterértékek pedig numerikus vagy karakter típusúak lehetnek. Jelgenerálás esetén állítható a jelalak (szinusz, háromszög, négyszög), frekvencia, amplitúdó, háromszög- illetve négyszögjel esetén a kitöltési tényező. Mérés esetén beállítható a mérés típusa (RMS, csúcsérték, frekvencia), mérés ideje, illetve a kompenzáció engedélyezés/tiltás. Torzításmérésnél a mért jel frekvenciája adható meg. Átviteli függvény mérésnél beállítható a mérőjel amplitúdója, a mérési frekvenciatartomány eleje és vége, hogy lineáris vagy logaritmus a frekvencia osztás, lineáris osztás esetén a frekvencia-lépésköz, logaritmikus osztás esetén az osztópontok száma, a mérési periódus hossza. Impedanciamérésnél a mérőjel amplitúdója és frekvenciája állítható, valamint karakteres parancsokkal jelezhető, hogy előkészültünk az első kalibrációhoz, a második kalibrációhoz, majd a méréshez. Szűrésnél a szűrőtípus, a törésponti frekvenciák értéke, és a szűrő felbontása adható meg.

7.2 Adatok megjelenítése

A feldolgozott adatok, jellemzően a mérési eredmények kiírását a felhasználó felé a PC-n valósítjuk meg. A kiíratáshoz az *sprintf* nevű könyvtári függvényt használjuk,

mely karaktereket és számokat konvertál egy string típusú változóba. Az így elkészült adatsomagot küldjük át a soros porton a PC felé, amit a HyperTerminal jelenít meg.

A kezelői felületen alapesetben állandóan jelen van az éppen aktuális mérés eredménye. Új parancs megadásakor a mérési adatok helyett a beírt parancs jelenik meg a képernyőn, mely segíti a felhasználót a parancs helyes begépelésében. Ebben az esetben a DSP-re érkező adatokat módosítás nélkül visszaküldjük a PC felé.

Egy parancsot Enter lenyomásával nyugtázhatunk. Nem értelmezhető parancs esetén a program futása a paraméterek módosítása nélkül folytatódik.

8. Eredmények, ajánlott elrendezések

8.1 PC-s interfész bemutatása

A PC-s interfész tetszőleges soros port-kezelő programmal futtatható (ajánlott: HyperTerminal). A soros port kezelő programmal ki kell választani azt a portot, amin keresztül a DSP kapcsolódik a PC-hez. Kapcsolódáskor az alábbi paraméterek a beállítandóak: Baudrate: 115200 b/s, paritás: nincs, stop bit: nincs.

Ha a DSP működik, az ablakba írt karakterek megjelennek. Parancsokat az alábbi rendben adhatunk. Kiválasztjuk a használni kívánt funkciót. A funkció paraméterei a funkció osztályokhoz tartozó betűjellel kezdődő parancsokkal módosíthatók.

A funkció osztályok:

- 1) jelgenerálás: G
- 2) mérés: M
- 3) torzításmérés: R
- 4) átviteli függvény mérése: A
- 5) impedanciamérés: I
- 6) hangolható szűrő: F

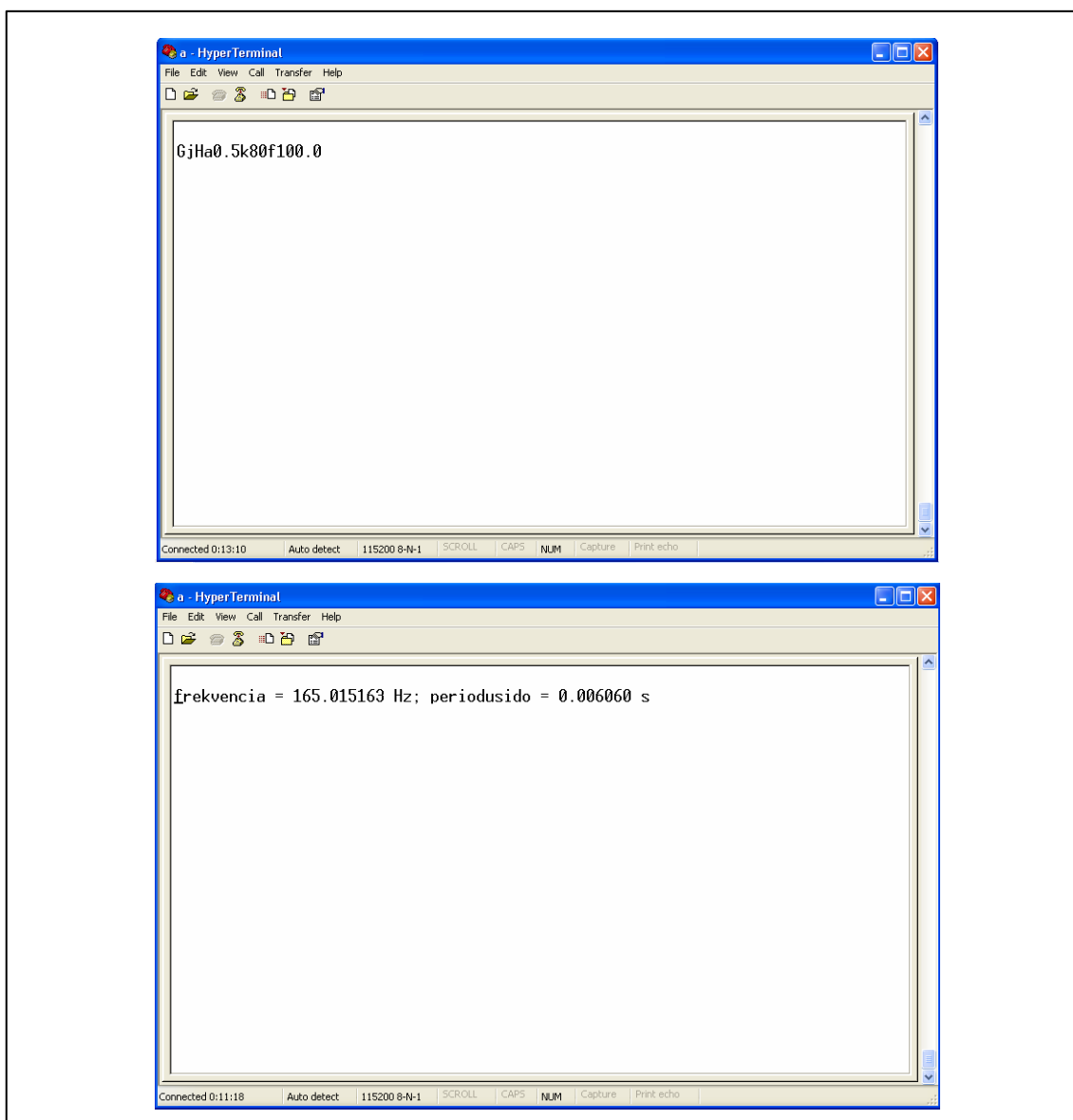
A funkciókon belül megadható paraméterek, a kapcsolódó betűjelek, és paraméter értékek:

Funkció	Paraméterkód	Paraméterérték
jelgenerálás (G)	j - jelforma	S-színuszjel, H-háromszögjel, N-négyszögjel, O-jelgenerálás leállítása
	a - amplitúdó	lebegőpontos szám
	f - frekvencia	lebegőpontos szám

	k - kitöltés	kitöltés %-ban megadva '%' jel nélkül
	n – komponensszám (Fourier- együtthatók száma)	egész szám
mérés (M)	P – csúcserték mérése	nincs
	E – effektív érték mérése	nincs
	F – frekvencia és periódusidő mérés	nincs
	t – mérési idő	lebegőpontos szám
	k – mérési idő kompenzálása	1 – van 0 – nincs
torzításmérés (R)	f – mért jel frekvenciája	lebegőpontos szám
átviteli függvény mérés (A)	k – frekvenciatartomány eleje	lebegőpontos szám
	v – frekvenciatartomány vége	lebegőpontos szám
	lin – lineáris frekvenciaosztás	nincs
	log – logaritmikus frekvenciaosztás	nincs
	d – lineáris esetén a frekvencia lépésköz logaritmikus esetén az osztópontok száma	lebegőpontos szám
	t – mérési időköz (adott frekvencián)	lebegőpontos szám
impedanciamérés (I)	ke – első kalibráció kezdése	nincs
	km – második kalibráció kezdése	nincs
	mz – mérés végzése, mérés eredménye	nincs

	komplex impedanciaként megjelenítve	
	mc – mérés végzése, mérés eredménye soros R-C helyettesítő képre számolva	nincs
	ml - mérés végzése, mérés eredménye soros R-L helyettesítő képre számolva	nincs
szűrő megvalósítás (F)	d – alsó törésponti frekvencia	lebegőpontos szám
	u – felső törésponti frekvencia	lebegőpontos szám
	n – szűrő-együtthatós szám	lebegőpontos szám
	a – aluláteresztő szűrő	nincs
	f – felüláteresztő szűrő	nincs
	s (sáv)	a – sáváteresztő szűrő z – sávzáró szűrő

Az alábbi ábrákon az interfészt használat közben láthatjuk. 8.1. felső ábra a parancsküldést jeleníti meg. A parancs értelmezése: jelgenerálás, jelforma: háromszögjel, amplitúdó: 0.5 V, kitöltési tényező: 80 %, frekvencia: 100.0 Hz. A 8.1. alsó ábrán egy frekvenciamérés üzemmódban megjelenített mérési eredményt láthatunk. A megjelenítés egyszerű, könnyen olvasható.



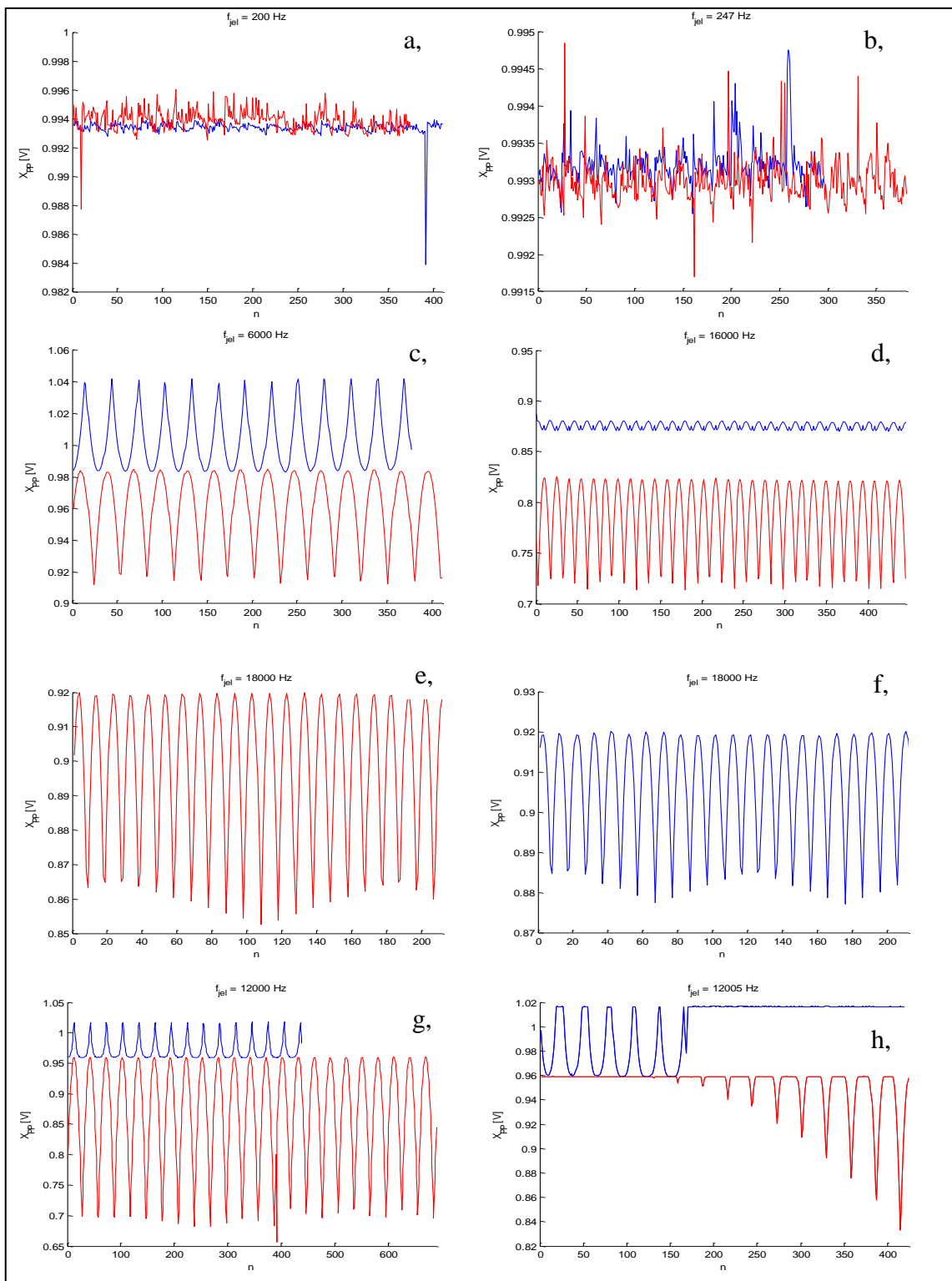
8.1. ábra (felső) Parancs megadása a HyperTerminal-on keresztül, (alsó) DSP által soros porton küldött mérési eredmény megjelenítése a HyperTerminal-on

8.2 Egyszerűbb mérések

A másodfokú interpoláció hibacsökkentő hatását szinuszjel mintavételezésénél vizsgáltam. A szinuszjel amplitúdója 1 V csúcstól csúcsig, a mérést 0,2 másodperces mérési periódusokban végeztem. A HyperTerminalon megjelenített mérési

eredményeket szöveges fájlba mentettem és Matlab-bal jelenítettem meg. A kapcsolódó ábrákon pirossal az interpolálás nélkül, kékkel az interpolálással végzett csúcsértékmérés eredményeit jelöljük.

Arra számítottam, hogy az interpolációval pontosabb csúcsértékmérést tudunk megvalósítani. Az eredmények alapján elmondható, hogy a mérési bizonytalanság csökkent. Ugyanazt a frekvenciájú jelet interpoláció nélkül és interpolációval vizsgálva, a mérésenkénti csúcsérték eredmény kisebb tartományon belül ingadozott (8.2.a és 8.2.b ábrák). Az ábrák x tengelyén n az egymást követő mérések indexét jelöli. Különösen igaz ez, ha a mintavételezett jel frekvenciája a mintavételi frekvencia osztója, mint $f = 6, 12, 16$ és 18 kHz (8.2.c, d, e, f, g ábrák). A probléma ilyenkor az lehet, hogy mivel a mintavételi frekvencia egész számú többszöröse a jel frekvenciájának, ezért a DSP a periodikus jelet mindig ugyanazokban az állapotaiban mintavételezi. Ha ezek a pontok nem a csúcsérték közelében vannak, akkor nagy hibát eredményez. Természetesen teljesen pontosan nem állíthatóak be ezek a frekvenciaértékek, ez eredményezi a mérési eredmények lassú ingadozását, mert így a perióduson belül lassan vándorolnak a mintavételi pontok. Ezt súlyosbítja, ha a mintavételihez közeli frekvenciájú (például $f = 18$ kHz, 8.2.e és f ábrák) jelet mérünk, mert ekkor az egy perióduson belül mért minták száma is jelentősen csökken. A mérés hibája 18 kHz-es jelet mintavételezve mindkét csúcsértékmérésnél nőtt, de az interpoláció itt is pontosabb. A mintavételezett jel frekvenciáját 12kHz-ről 5 Hz-cel elhangolva ismét csökken az interpolációs és a becslés nélküli mérés hibájának különbsége (8.2.g és h ábrák), de 8.2.h ábrán látható mérési eredmények második feléből az is jól látszik, hogy ha a mintavételi frekvenciával összemérhető frekvenciájú jelet a null-értékhez közeli fázisaiban mintavételezzük, a csúcsértékmérésben interpolálás nélkül jelentős hiba keletkezik.



8.2. ábra Amplitúdómérés interpoláció nélkül és interpolációval

8.3 Rezonátor alapú átviteli függvény mérés

A rezonátor alapú átviteli függvény mérését egy Krohn Hite 3322 típusú negyedfokú szűrőn ellenőriztem. Ez a szűrő programozható törésponti frekvenciával rendelkezik, és megvalósítható alul-, felüláteresztő, valamint sáváteresztő és sávzáró szűrő. A szűrő karakterisztikáját a laborban rendelkezésre álló HP3585B típusú, átviteli függvény mérésére is alkalmas spektrumanalizátorral is lemértem.

A szűrő azonos beállításai mellett a DSP-vel és a spektrumanalizátorral végzett eredményeket bemutató ábrákat egy csoportba. A HP3585B típusú spektrumanalizátorral mért ábrákat is közlöm, hogy ellenőrizhetőek legyenek az eredményeink egy precíziós műszer segítségével is. Az analizátor képernyőjéről készített ábrák esetén legfontosabb paraméterek alsó sorában megjelenő START és STOP feliratok, melyek a vizsgált frekvenciatartományt adják meg. A frekvenciatengely 10 osztást tartalmaz. Az amplitúdótengely (y tengely) 10 dB/osztású.

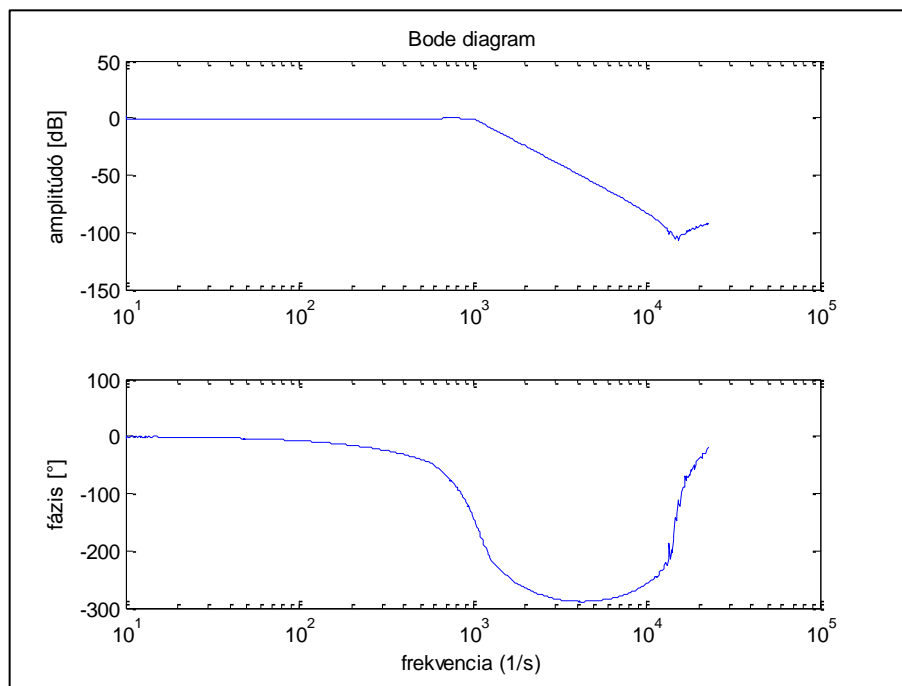
A mért szűrőfajták:

- 100 Hz-en levágó aluláteresztő szűrő
- 1 kHz-en levágó aluláteresztő szűrő
- 100 Hz határfrekvenciájú felüláteresztő szűrő
- 1 kHz határfrekvenciájú felüláteresztő szűrő
- 1 és 5 kHz közötti frekvenciasávban áteresztő szűrő
- 1 és 5 kHz közötti frekvenciasávban elnyomó szűrő

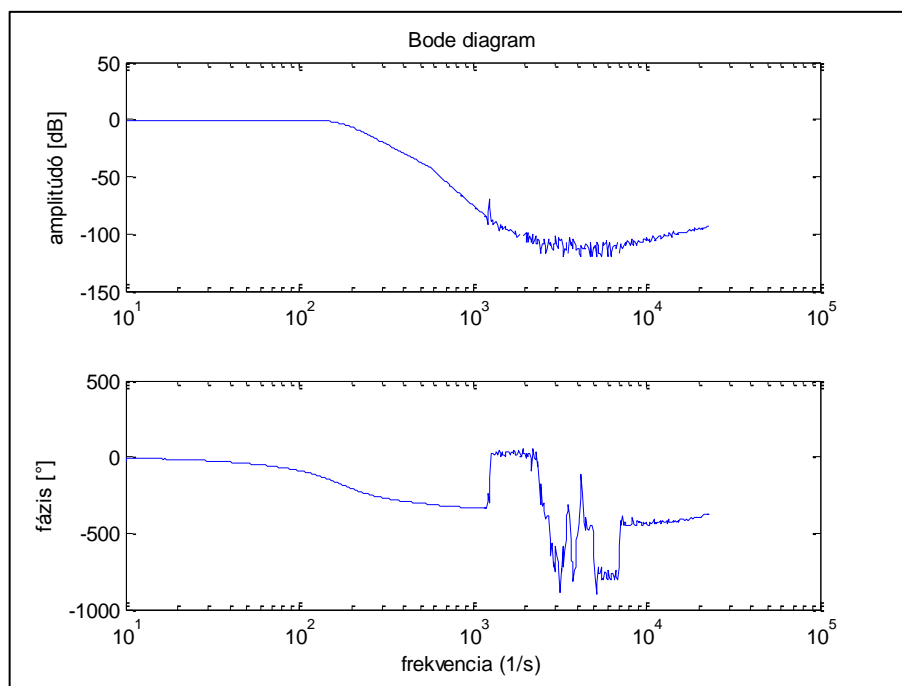
A DSP-vel logaritmikus frekvencia-léptetéssel mértünk, ennek megfelelően a legtöbb ábra a frekvencia logaritmikus felbontásával készült. A spektrumanalizátor lineárisan léptetett frekvenciával mért.

A 8.3.-8.10. ábrák a DSP méréseiből a Matlab-bal ábrázolt szűrőkarakterisztikák. Láthatóan az átvitel mérés sikeres volt, mert a szűrőkről a beállításnak megfelelő karakterisztikát mértünk a DSP-n. A negyedfokú szűrőnek megfelelően, az amplitúdó karakterisztika az elnyomó tartományokon -80 dB/dekád-dal csökken.

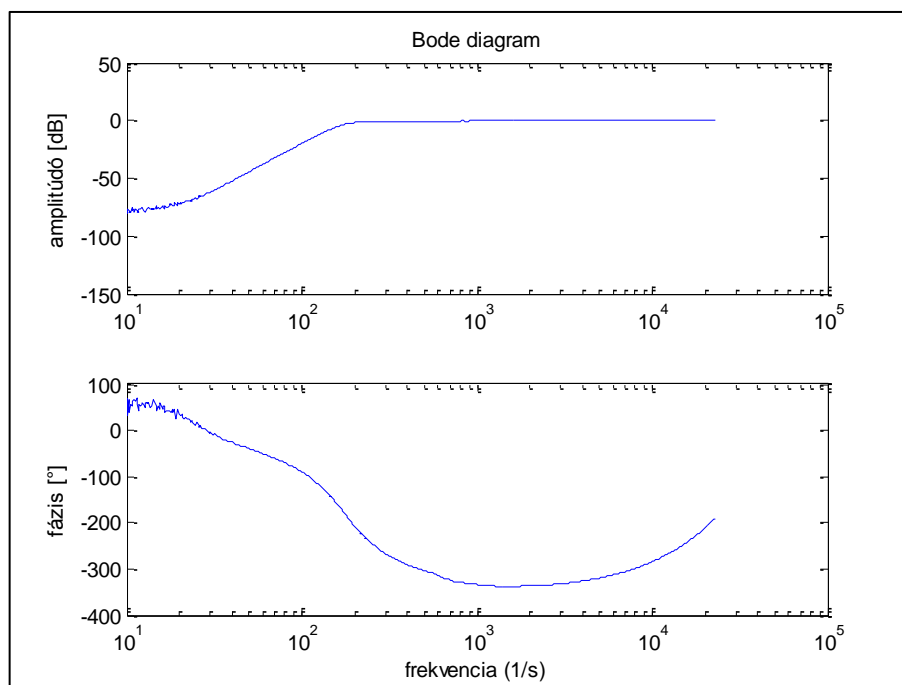
Annak az oka, hogy nagy elnyomású frekvenciákon a fáziskarakterisztika nem ad értelmezhető eredményt az, hogy a fázist a komplex átvitel valós és képzetes részének argumentumaként számoljuk. Ezek az értékek nagy elnyomásnál 10^{-5} nagyságrendűek, és a mérés és a számábrázolás miatt ebben a tartományban a mérés hibája már összemérhető a mért értékekkel, valamint a mérési zaj is jelentős. Ezekben a tartományokban szerencsére a fázismenet azonban nem is játszik jelentős szerepet, hiszen a nagy elnyomás miatt nem számíthatunk jelentős teljesítményű kimeneti értékre.



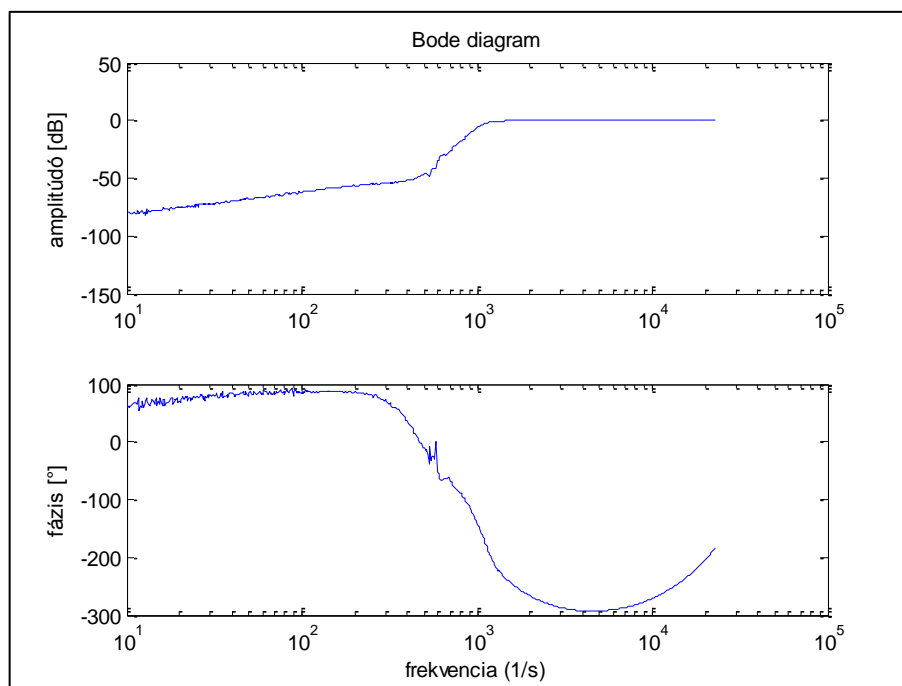
8.3. ábra 1 kHz-en levágó aluláteresztő szűrő amplitúdó és fáziskarakterisztikája



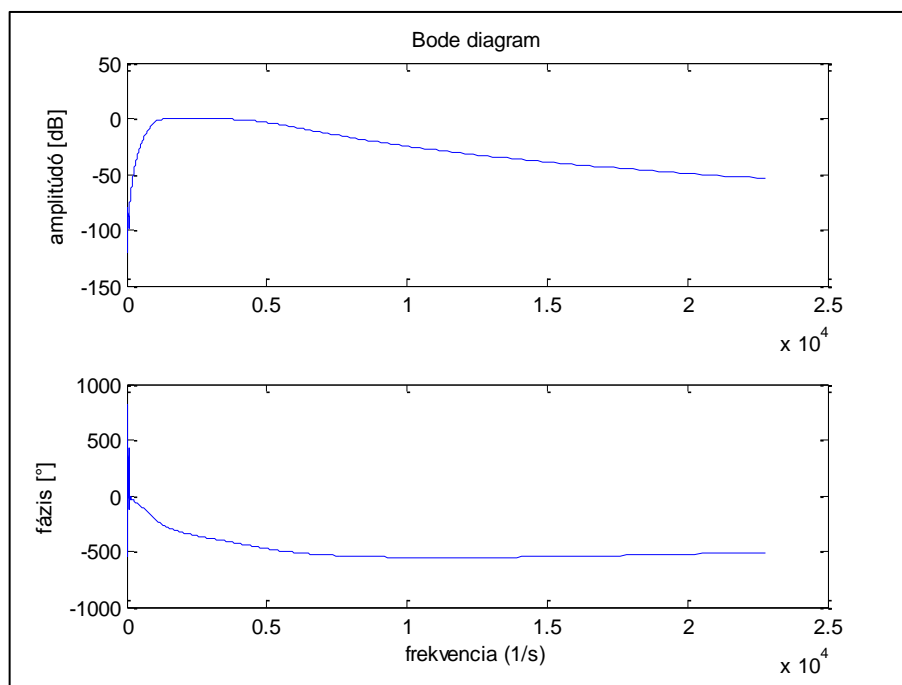
8.4. ábra 100 Hz-en levágó aluláteresztő szűrő amplitúdó és fáziskarakterisztikája



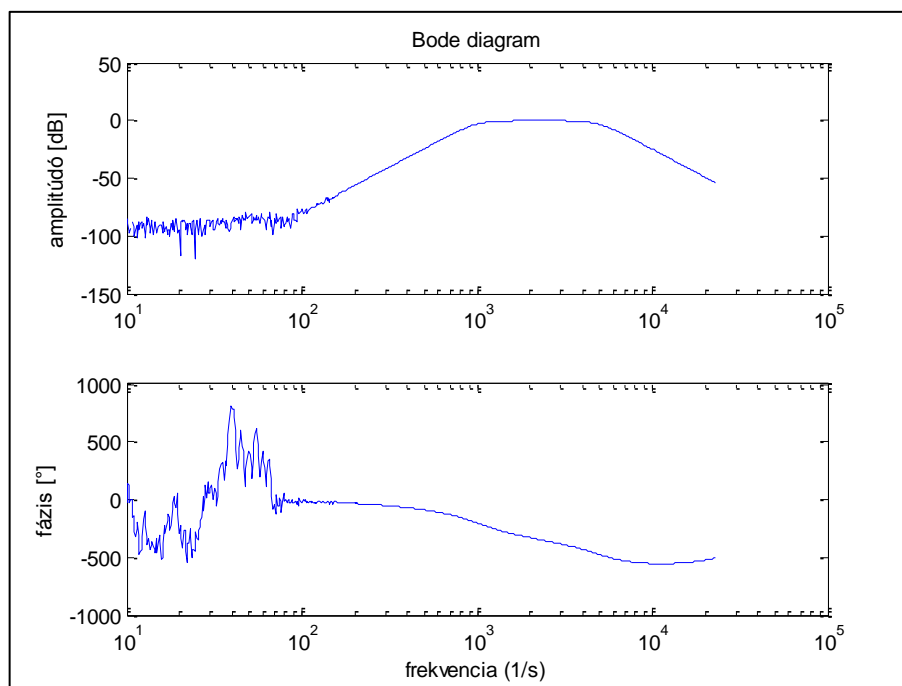
8.5. ábra 100 Hz-es felüláteresztő szűrő amplitúdó és fáziskarakterisztikája



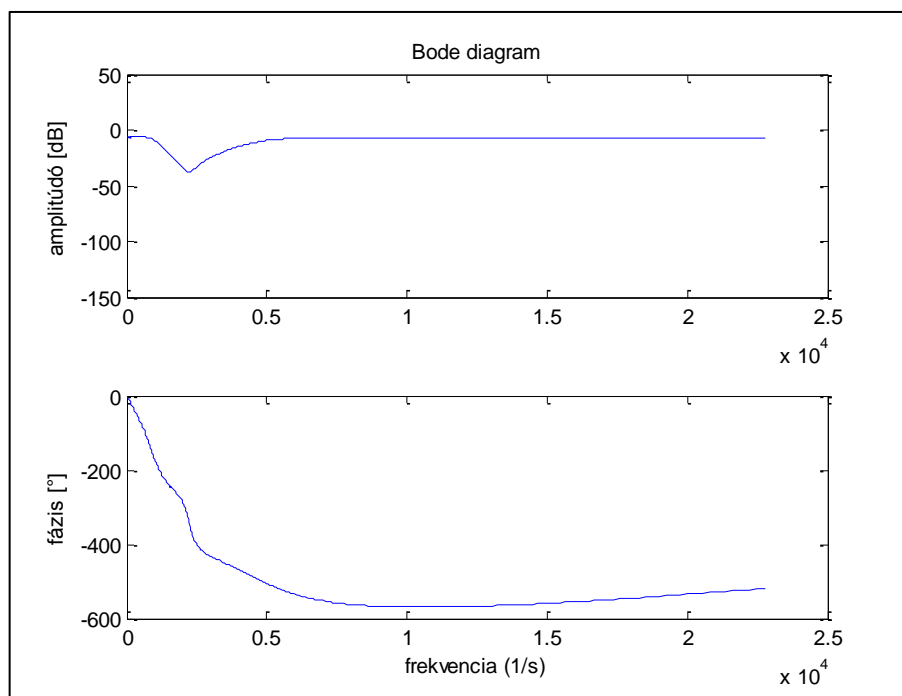
8.6. ábra 1kHz-es felüláteresztő szűrő amplitúdó és fáziskarakterisztikája



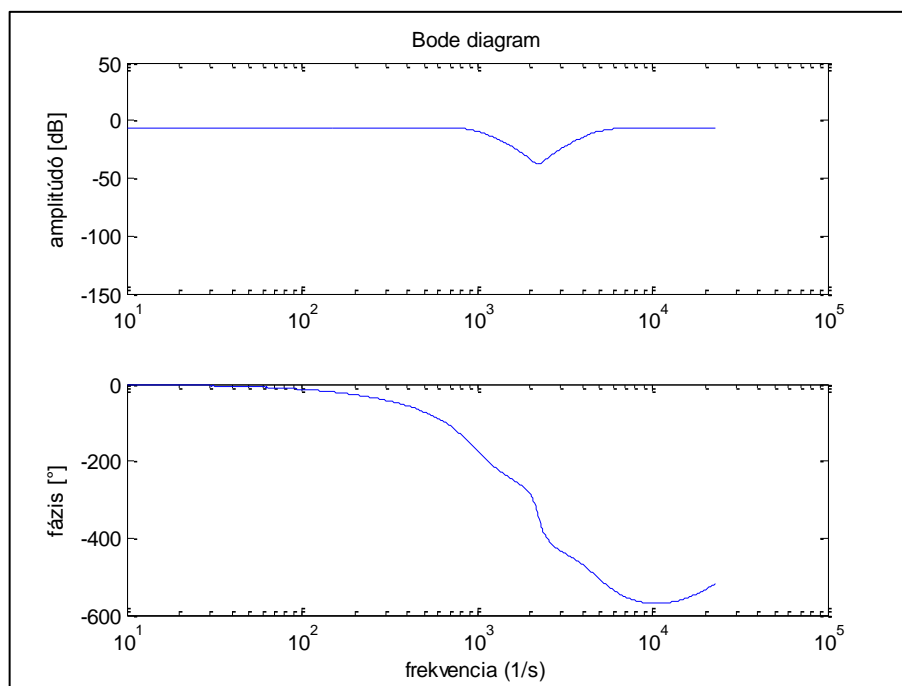
8.7. ábra 1-5 kHz-en sáváteresztő szűrő lineáris frekvenciamenet mellett



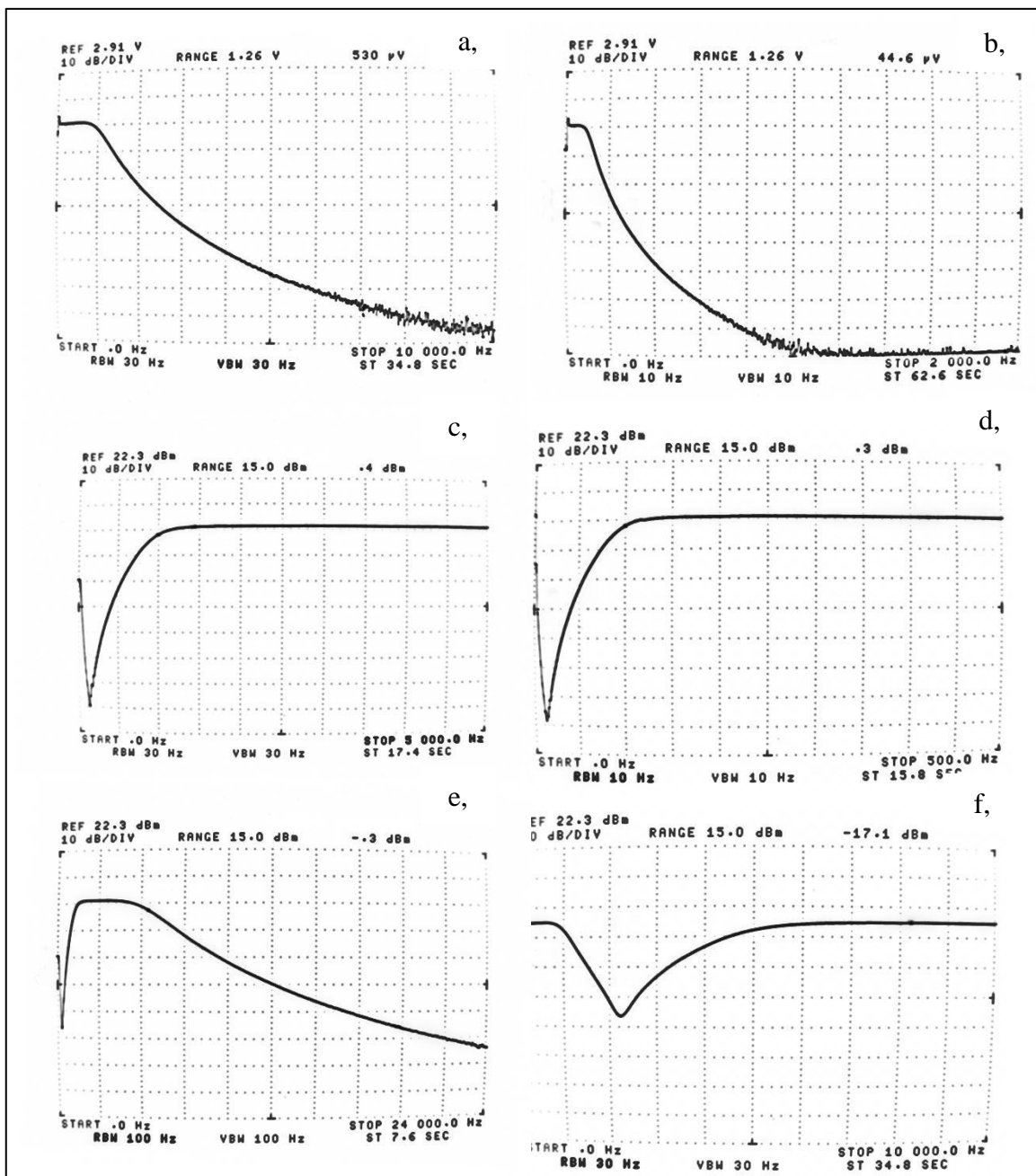
8.8. ábra 1- 5 kHz-en sáváteresztő szűrő logaritmikus frekvenciaosztás mellett



8.9. ábra 1-5 kHz-en sávzáró szűrő lineáris frekvenciaosztás mellett



8.10. ábra 1-5 kHz-en sávzáró szűrő logaritmikus frekvenciaosztás mellett



8.11. ábra Szűrők átviteli karakterisztikája spektrumanalizátorral mérve (b) 1 kHz-es aluláteresztő szűrő; (a) 100 Hz-es aluláteresztő szűrő; (c) 1 kHz-es feluláteresztő szűrő; (d) 100 Hz-es feluláteresztő szűrő; (e) 1-5 kHz-en sáváteresztő szűrő; (f) 1-5 kHz-en sávzáró szűrő

8.4 Impedanciamérés

A következőkben mérési eredményeket közlök arra vonatkozóan, hogy a megvalósított impedanciamérő milyen pontossággal képes méréseket végezni.

Az impedanciamérés eredményét a 8.1.-8.3.- táblázatok tartalmazzák.

Igyekeztem széles skálán mérni ellenállásokat, kondenzátorokat és tekercseket. Az első oszlop a mért kétpólusok valódi értékét tartalmazza, melyet nagy pontosságú impedanciamérővel mértem meg, hogy ellenőrzést nyújtson a DSP-n mért értékekhez. A referenciaként használt impedanciamérő típusa Wayne Kerr 6425. A legtöbb kétpólust értékét 2000 Hz-en vizsgáltam. Ezek alól kivétel a két nagyon kis értékű kondenzátor, azok értékét a mérési határnak is tekinthető 23 kHz-en mértem. A kis értékű kondenzátor alacsony frekvencián vizsgálva túl kicsi impedanciát képvisel, nem mérhető jól a valós és képzetes rész közötti különbség. Kis impedanciák esetén ráadásul a mérővezetékek impedanciája is hibát okozhat a mérésben. A pirossal jelölt számok a mérés közben ingadozó értékeket jelölik, tehát ezek már nem tekinthetők pontos eredményeknek, és jellemzik a mérés zaját. Alapvetően várható, és a mérések is azt mutatják, hogy minél jobban eltér a mért impedancia abszolút értéke a DSP körülbelül 600 Ohmos kimeneti ellenállásától, akkor a mérés hibája is nő. Ez az impedanciamérés hibaanalíziséből analitikusan is kiszámítható. A mérési eredmények alapján látható, hogy amennyiben a mért impedancia abszolút értéke néhány száz Ohm, a mérési hiba egy százalék alatt marad, néhány 10 Ohm és néhány kOhm tartományban néhány százalék. Ezen tartományokon kívül pedig több tíz százalék is lehet. Ezen mérések segítenek abban, hogy be tudjuk határolni a műszer alkalmazhatóságának tartományát.

Ellenállás	Impedancia (f = 2000 Hz)	Z	hiba [%]
19.439 Ohm + 307.9 uH	19.326019 + 3.930738j	19.721706	18.2361
99.705 Ohm	99.735420 + 0.153396j	99.735540	0.1568
149.26 Ohm	149.261963 + 0.227675j	149.262125	0.1525
199.49 Ohm	199.422013 + 0.347497j	199.422313	0.1775
471.25 Ohm	470.970150 + 1.088930j	470.971400	0.2386
1.4847 kOhm	1483.102125 + 6.408764j	1483.116000	0.4449
9.9705 kOhm	9939.730000 + 186.094538j	9941.473000	1.8918
21.979 kOhm	21851.342500 + 885.139800j	21869.262500	4.0689
99.805 kOhm -0.05°	95361.800000 + 17185.860000j	96898.020000	17.8701

8.1. táblázat

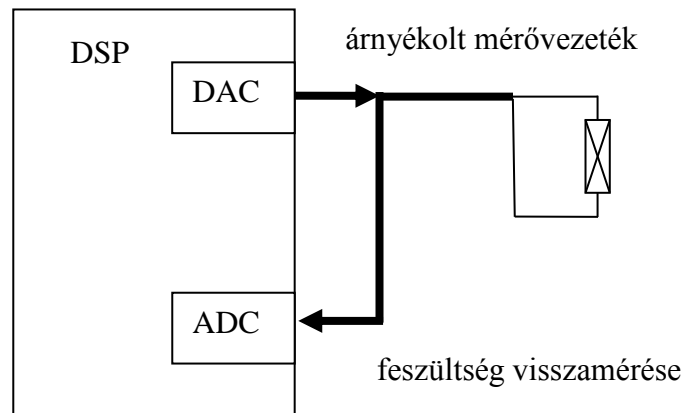
Tekercs	Impedancia	Z	hiba [%]
10.47 uH + 452.5 mOhm	0.548388 Ohm + 0.011101 mH	0.565853	20.4175
93.57 uH + 2.3682 Ohm	2.469352 Ohm + 0.089946 mH	2.715742	4.1955
1.0213 mH + 5.7270 Ohm	5.739303 Ohm + 0.945609 mH	13.196293	6.7685
1.5018 mH + 956 mOhm	0.135386 Ohm + 1.217273 mH	15.297298	4.6770
2.2536 mH + 10.15 Ohm	10.175061 Ohm + 2.203478 mH	29.500043	2.0953
9.6805 mH + 57.42 Ohm	57.749750 Ohm + 10.084845 mH	139.267725	3.7852
22.120 mH + 129.17 Ohm	129.287688 Ohm + 22.350673 mH	309.194925	0.9465

8.2. táblázat

Kondenzátor	Impedancia (f = 2000 Hz)	Z	hiba [%]
4.606 nF + 87 Ohm	226.741275 Ohm + 4.474679 nF	17785.395000	3.0441
33.478 nF + 17.3 Ohm	21.841430 Ohm + 32.226120 nF	2469.443750	3.8893
356.54 nF + 1.2 Ohm	1.591432 Ohm + 356.607575 nF	223.157050	0.1764
979.55 nF + 413 mOhm	0.612223 Ohm + 979.986000 nF	81.204965	0.2492
(f = 23000 Hz)			
22.74 pF -0.2 kOhm	58357.295000 Ohm + 0.366255 nF	61339.490000	95.7449
381.7 pF + 104 Ohm	4568.466500 Ohm + 0.409925 nF	17487.861250	25.5703

8.3. táblázat

8.4.1 Ajánlott mérési elrendezés impedanciaméréshez



8.12. ábra Impedanciamérés elrendezése

Az impedanciamérés gyakorlati kialakítását a 8.12. ábra mutatja. A következőkben a mérés során elvégzendő műveleteket mutatom be.

A rendszert a mérés előtt kalibrálni kell. A kalibrálás két lépésből áll, első lépésként üresjárásban, a mérő zsinór végét üresen hagyjuk, második lépésként a zsinór végére

egy kitüntetett, ellenállást megvalósító impedanciát kapcsolunk. Ez az impedancia a kalibrációs ellenállás, a későbbiekben csak így utalok rá. A tesztek során használt kalibrációs ellenállás nagy mérőpontosságú impedanciamérőn lement értéke 1484,7 Ohm.

A DSP kimenetét T csatlakozóval ketté osztjuk, az egyiket a DSP bemenetére csatlakoztatjuk, a másikra két csipeszben végződő vezetékét kapcsolunk, ezekre fogjuk csíptetni a mérni kívánt impedanciákat.

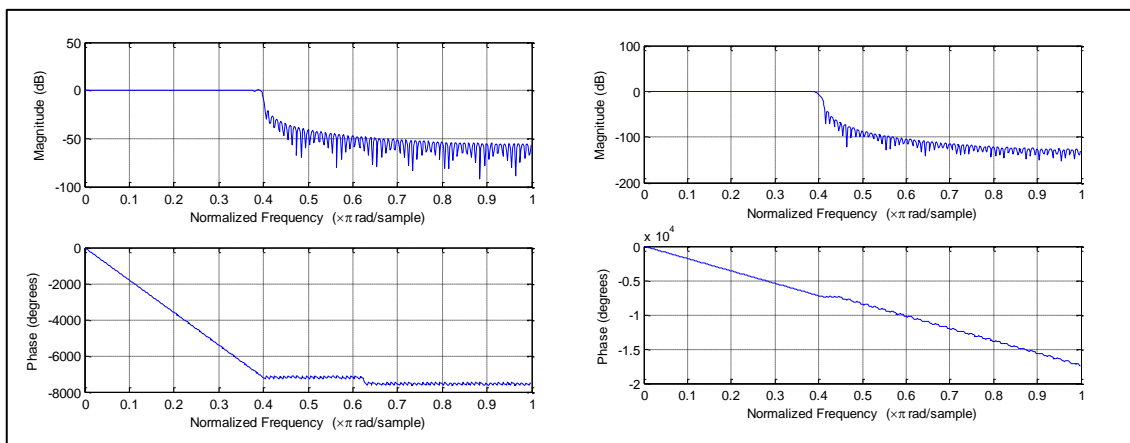
Első kalibrációt üresjárásban végezzük, a mérőszinór végét üresen hagyjuk.

A második kalibrálás során a kalibrációs ellenállást a mérőszinóra kapcsoljuk.

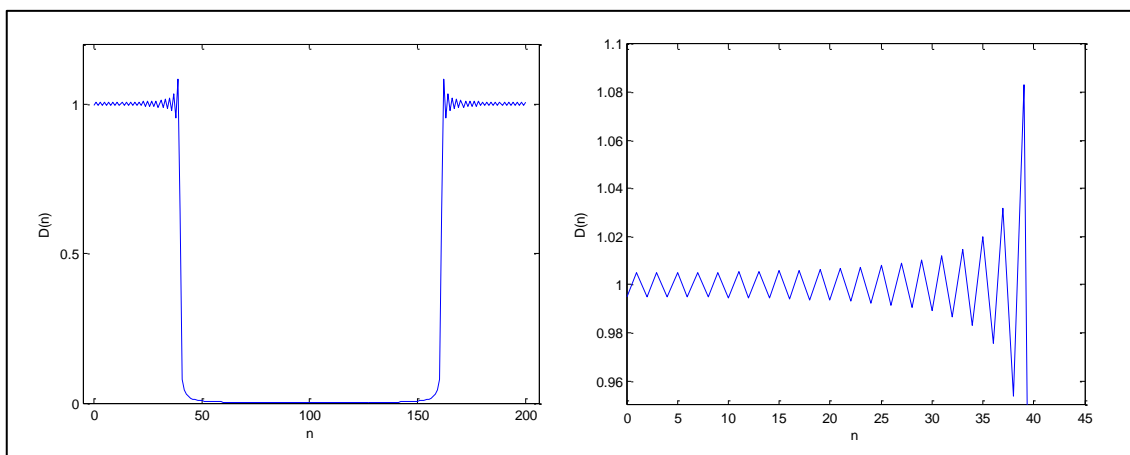
A mérés pontosságának javulását várjuk, ha az első kalibrálás során a csipeszes mérőszinórt nem használjuk, mert a mérővezeték által képviselt kapacitás is befolyásolhatja az áramkör működését.

8.5 Hangolható szűrő

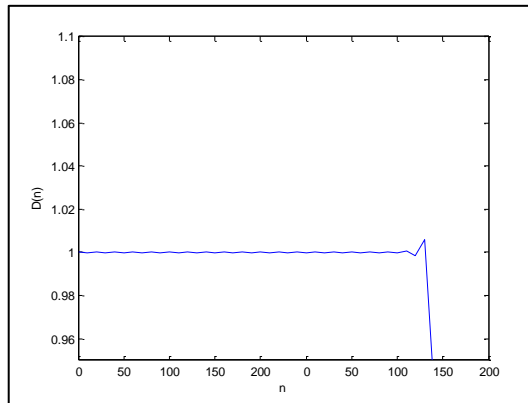
A DSP-n megtervezett szűrőegységűtthetőkát először tesztelés céljából kiolvastam a DSP memóriájából, és Matlab-ban ábrázoltam, hogy megvizsgáljam, helyes karakterisztika jött-e ki. A példaként vizsgált szűrő beállításai: 201 pontos aluláteresztő 9600 kHz-ig. A 8.13.a ábra a kész szűrő átviteli karakterisztikáját ábrázolja Hanning ablak nélkül. Az 8.14.a ábra a szűrő impulzusválaszának DFT-jét (átviteli karakterisztikáját) ábrázolja. Kinagyítva (8.14.b ábra) látszik, hogy a minták erősen hullámosak, 0.08 maximum kitéréssel a levágás közelében. Ahogy vártuk, Hanning ablakozással a hullámosság kisimul (8.15. ábra), az amplitúdó karakterisztika nagyobb elnyomást tesz lehetővé (8.13.b ábra).



8.13. ábra Szűrő amplitúdó és fázis karakterisztikája
négyzet-ablakkal és Hanning-ablakkal



8.14. ábra A szűrő átviteli karakterisztikájának mintái lineáris skálán teljes tartományra,
és az áteresztő tartomány határán (ablakozás nélkül)



8.15. ábra A szűrő átviteli karakterisztikájának mintái lineáris skálán az áteresztő tartomány határán (Hanning-ablak)

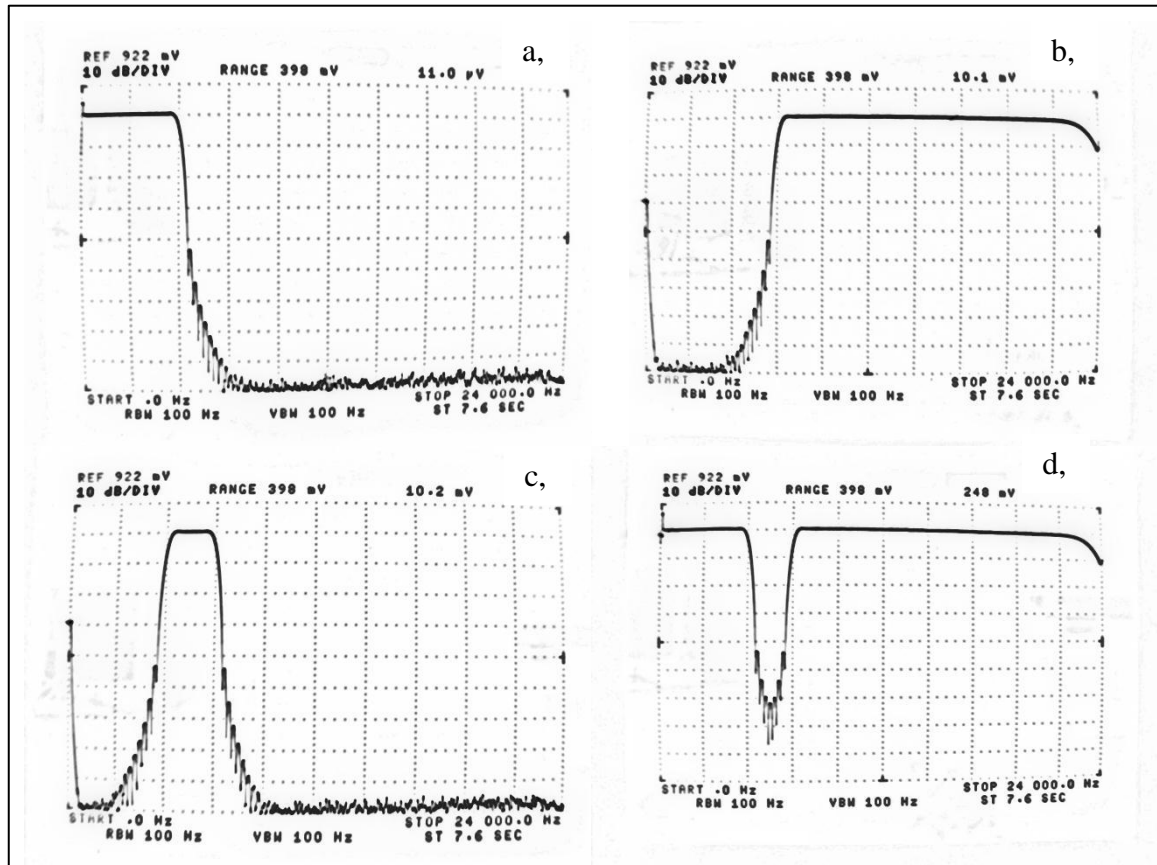
A DSP-n kialakított programozható szűrővel alul-, felüláteresztő, sávzáró és sáváteresztő karakterisztika valósítható meg. Tesztelés céljából mindegyik típusból megvizsgáltam egy-egy 201 pontos szűrőt, és a karakterisztikát ellenőriztem mérésel.

A megvalósított szűrők:

- 4,8 kHz-ig aluláteresztő szűrő
- 7,2 kHz-től felüláteresztő szűrő
- 4,8-7,2 kHz-en sáváteresztő szűrő
- 4,8-7,2 kHz-en sávzáró szűrő

A szűrőknek egy HP3585B típusú, átviteli függvény mérésére is alkalmas spektrumanalizátorral visszamért eredményeit a 8.16.a-d ábrák mutatják be. A kijelölt karakterisztikát szépen követik. A mérések során a 0...24 kHz-es tartományt vizsgáltam, ugyanis a DSP 48 kHz-es mintavételi frekvencián üzemel. Ennek megfelelően a felüláteresztő jellegű szűrők esetén a 24 kHz körül látható letérés már a rendszer átlapolásgátló szűrőjének a hatása; a szűrőtervezés ezt nem befolyásolja. Az ábrákon az x, tehát frekvenciatengely a beállításoknak megfelelően 0...24 kHz tartományt fed le, tehát egy osztás 2,4 kHz-nek felel meg. Az y skálán a műszer beállításai alapján egy osztás 10 dB-nek felel meg.

Tapasztalatként elmondható, hogy a megtervezett szűrők jól követik a specifikációt. Megfelelő, többször 10 dB-es elnyomást biztosítanak zárótartományban. A legszerűnyebb elnyomás sáváteresztő szűrő esetén tapasztalható, mely azonban így is sávközépi frekvencián 50-60 dB.



8.16. ábra Szűrő karakterisztikák (a) 4,8 kHz-ig aluláteresztő szűrő; (b) 7,2 kHz-től felüláteresztő szűrő; (c) 4,8-7,2 kHz-en sáváteresztő szűrő; (d) 4,8-7,2 kHz-en sávzáró szűrő

9. Összefoglaló értékelés és kitekintés

A feladat egy olyan műszer elkészítése volt, amely integrálja az elterjedten használt mérőműszerek és jelgenerátorok funkcióit. A feladat megvalósítására egy DSP-s kártya állt rendelkezésre, ami a felhasználó által, PC felhasználásával soros porton vezérelhető.

A célkitűzés teljesült, az eszköz alkalmas az alapvető műszerfunkciók ellátására, valamint olyan komplexebb feladatok elvégzésére, mint átvitel mérés, komplex impedanciamérés és szűrőtervezés. Mindezt a digitális műszerek előnyeivel: az eredmények a PC-n megjeleníthetők, és további vizsgálatok végezhetők rajtuk. Az eszköz hangfrekvenciás tartományban működik, tehát pl. akusztikai mérésekre alkalmas, de egyéb mérés technikai felhasználás során is dolgozunk ebben a tartományban.

A mérési eredmények alapján nagy pontosságú eszközökkel összehasonlítva látható, hogy habár nem precíziós műszer, de elfogadható pontosságot biztosít, mely legalább tájékoztató mérések elvégzésére alkalmas.

A feladat megoldása során megismerkedtem különböző alapvető jelfeldolgozási problémákkal és megoldásukkal, és az elméleti eredményeket gyakorlatban is sikerült alkalmazni.

A műszerhez kapcsolódó továbbfejlesztési lehetőség például egy grafikus PC-s interfész készítése, mely a hetedik fejezetben bemutatott felhasználói interfészre épülve a rendszer laikusként történő felhasználását segítené. A komplexebb műveletek, például átviteli függvény mérésénél PC-vel ábrázolhatnánk, már a mérés során az átviteli függvény frekvencia- és fázismenetét.

A PC-n való irányítás felváltására hardveres felhasználói interfész, kijelző tervezésére is lehetőség van. Ekkor a mérés beállításait és az egyszerűbb mért értékeket is a kijelzőn lehetne megjeleníteni.

Irodalomjegyzék

- [1] Scherer Balázs, dr. Tóth Csaba, Molnár Károly, Rendszerarchitektúrák laboratórium - Virtuális műszerek mérés sorozat, BME MIT, Budapest, 2009.
- [2] S. Sumathi, P. Surekha, „*LabVIEW based Advanced Instrumentation Systems*”, Japan: Springer, Április, 2007.
- [3] M. Santori, „An Instrument that Isn't Really” (Laboratory Virtual Instrument Engineering Workbench), *IEEE Spectrum*, Volume: 27 8, pp. 36-39, Aug, 1990
- [4] Goldberg, H.: What is virtual Instrumentation? *IEEE Instrumentation and Measurement Magazine* 3(4) , pp. 10-13, Dec, 2000
- [5] National Instruments Magyarország [Online] [Hivatkozva: 2011. december 15.] <http://hungary.ni.com>
- [6] Schnell László, „*Jelek és rendszerek mérés technikája*”
- [7] Koller István, Fiala Péter, „Mérési útmutató 2. éves fizika alapszakos hallgatók számára”, online elérhető:
http://www.hit.bme.hu/~koller/Merestechnika_Lab/7.pdf
- [8] Zoltán István, „Méréstechnika”, Műegyetemi Kiadó, Budapest, 1997.
- [9] G. Péceli, „A Common Structure for Recursive Discrete Transforms”, *IEEE Trans Circuits Syst.*, Vol. CAS-33, No. 10, Oct. 1986., pp. 1035-1036.
- [10] L. Sujbert, G. Péceli, Gy. Simon, „Resonator Based Non-parameteric Identification of Linear Systems”, *IEEE Trans. on Instrumentation and Measurement*, Vol. 54., No. 1., Feb. 2005., pp. 386-390.
- [11] Dr. Simonyi Ernő, „*Digitális szűrők*”, Műszaki Könyvkiadó, Budapest, 1984.
- [12] Fodor György, „*Hálózatok és rendszerek*”, Műegyetemi Kiadó, Budapest, 2006.
- [13] BME-MIT tanszéki munkaközösség, Digitális jelfeldolgozás jegyzet, Műegyetemi kiadó, 2003