

Diplomaterv

Hibrid aktív zajcsökkentő rendszer fejhallgatóban

Készítette:

Dobos László



M Ű E G Y E T E M 1 7 8 2

Budapesti Műszaki és Gazdaságtudományi Egyetem
Villamosmérnöki és Informatikai Kar

2004.

Nyilatkozat

Alulírott, **Dobos László**, a Budapesti Műszaki és Gazdaságtudományi Egyetem hallgatója kijelentem, hogy ezt a diplomatervet meg nem engedett segítség nélkül, saját magam készítettem, és a diplomatervben csak a megadott forrásokat használtam fel. Minden olyan részt, melyet szó szerint, vagy azonos értelemben, de átfogalmazva más forrásból átvettem, egyértelműen, a forrás megadásával megjelöltem.

.....
Dobos László

Tartalomjegyzék

Abstract	5
Összefoglalás	6
1 BEVEZETÉS	7
1 AZ AKTÍV ZAJCSÖKKENTÉS MÓDSZEREI	8
1.1 Zajcsökkentés analóg áramkörökkel	8
1.1.1 A késleltetés hatása	11
1.1.2 Visszacsatoló hálózatok	12
1.2 Digitális zajcsökkentés	18
1.2.1 Az előreccatolt szabályozás	18
1.2.2 A visszacsatolt szabályozás	20
1.2.3 Adaptációs algoritmusok	22
1.2.4 A normalizált LMS	28
1.2.5 A FXLMS algoritmus	28
1.3 A megvalósítandó analóg-digitális szabályozás ismertetése	29
2 AZ ANALÓG SZABÁLYOZÓ MEGTERVEZÉSE	31
2.1 A szabályozott szakasz modellje	31
2.1.1 A mozgótekerceses hangszóró	31
2.1.2 Az átviteli közeg	34
2.1.3 A mikrofon	34
2.2 A szabályozott szakasz identifikációja	35
2.3 A megvalósítás elve	40
2.4 A MATLAB szimuláció eredményei	42
2.5 Mérési eredmények	43
3 A DIGITÁLIS RENDSZER MEGTERVEZÉSE	46
3.1 A rendszer elemei	46
3.2 Az alkalmazott algoritmus	47
3.2.1 Az $\hat{S}(z)$ rendszerbecslő előállítás	47
3.2.2 A telítések figyelembevétele	48
3.3 Megvalósítás 8 kHz mintavételi frekvencián	50
3.3.1 Mérési eredmények	51

3.4	Áttérés 48 kHz mintavételi frekvenciára	52
3.4.2	Mérési eredmények	57
4	A HIBRID RENDSZER EGYÜTTMŰKÖDÉSE	63
4.1	Mérési eredmények, értékelés	63
5	ÉRTÉKELÉS	65
6	IRODALOMJEGYZÉK	66
7	FÜGGELÉK	67

Abstract

Active noise-reducing (ANR) headsets that employ analog feedback control and provide good broadband attenuation are currently available on the market. These are useful for protecting the human auditory system in a noisy environment such as inside a helicopter or near to industrial machines. However active noise control based on analog feedback technology is band-limited due to the delay between the secondary loudspeaker and the noise-detecting microphone.

Adaptive digital control is also used in many application where active noise cancellation is needed. These algorithms work well mainly in the presence of periodic noise but they can be used in the case of broadband noise, too.

To overcome the limitation of the analog feedback control it seems that a combined analog – digital ANC system works better than both systems separately.

This thesis presents an experimental study of such a combined control system which employs an analog feedback control loop and a digital feedback control algorithm. The digital algorithm was implemented on an ADSP21061 EZKIT-LITE evaluation board while the analog system consists of only cheap analog parts such as resistors, operational amplifiers, etc. First the analog controller part was designed using classical control theory. With proper stability margins the analog ANC – circuit provided a broadband attenuation of about 15-20 dB at 300 Hz and around.

After that the digital algorithm, namely the normalized FXLMS was implemented on the above mentioned DSP card. A decimating process was also implemented to reduce the delay of the sigma-delta converters. So the digital system achieved about 10 dB attenuation in the case of random noise limited to 150 Hz in bandwidth. Another problem was also solved, namely the output limitation of the headphone's speaker. For stabilizing the adaptive noise reduction it was needed to implement a rescaling mechanism which rescaled not only the output of the adaptive filter but the adaptive filter, too.

After connecting the analog and digital controller it was found that the combined control method works well, and the digital algorithm was able to extend the limited working range of the analog control. However, due to the remaining delay of the employed sigma – delta converters the digital control cancels mainly the periodic or narrow-band noises out. In the future, however, by the replacement of these slow converters better broadband attenuation can be achieved.

Összefoglalás

Aktív zajcsökkentő fejhallgatók, melyek analóg visszacsatolást alkalmaznak és jó szélessávú zajelnyomó képességgel rendelkeznek, már jelenleg is kaphatók a kereskedelmi forgalomban. Ezek az eszközök azt a célt szolgálják, hogy megvédjék az emberi hallószervet zajos környezetben, mint pl. egy helikopter belsejében vagy ipari gépek környezetében. Azonban az analóg visszacsatolást alkalmazó zajcsökkentési módszer korlátozott sáv szélességben teszi csak lehetővé a zajelnyomást a kioltó zajforrás és a zajt detektáló mikrofon közötti késleltetés miatt. Adaptív digitális szabályozást szintén alkalmaznak aktív zajcsökkentés céljából. Ezek az algoritmusok különösen periodikus zavarhatások esetében használhatóak jól, de szélessávú zaj elnyomására is képesek.

Az analóg visszacsatolásból adódó korlát megszüntetése érdekében egy kombinált analóg-digitális zajcsökkentő rendszer létrehozása azzal kecsegtet, hogy egy ilyen rendszer jobb teljesítményt nyújt, mint a kétfajta szabályozás külön-külön.

A diploma témája tehát egy olyan hibrid szabályozás tanulmányozása, amely egyrészt egy analóg visszacsatolt rendszert és egy digitális visszacsatolt rendszert tartalmaz. A digitális algoritmus egy ADSP21061 EZKIT-LITE fejlesztőkártyán került megvalósításra, míg az analóg rendszer olcsó analóg alkatrészekből (ellenállások, műveleti erősítők, stb.) épült fel.

Először az analóg szabályozás került megvalósításra a klasszikus szabályozástechnika segítségével. A stabilitási tartalékok megfelelő beállításával kb. 15-20 dB szélessávú elnyomást sikerült elérni 300 Hz környékén.

Ezután a digitális algoritmus, az FXLMS algoritmus került megvalósításra a már említett DSP kártyán. A szigma-delta átalakítók késleltetésének csökkentése érdekében decimálást kellett alkalmazni. Ezáltal a digitális rendszer kb. 10 dB-es szélessávú elnyomást hozott létre 150 Hz-es sáv szélességű véletlenszerű zaj esetén.

Problémát jelentett még a fejhallgató véges kivezérelhetősége. A probléma megoldása érdekében (az adaptív algoritmus stabilizálása érdekében) szükségessé vált egy olyan újraszámítási módszer alkalmazása, amely nemcsak az adaptív szűrő kimenetét skálázza át, hanem magát az adaptív szűrőt is.

Úgy találtuk, hogy a hibrid szabályozás valóban képes kiterjeszteni az analóg szabályozáshoz képest a működési tartományt, viszont az alkalmazott szigma-delta átalakítók nagy késleltetése miatt a digitális szabályozó inkább csak periodikus zavarjelek elnyomására volt alkalmas. A továbbiakban azonban ezeknek a lassú átalakítóknak a lecserélésével jobb szélessávú működést is el lehet majd érni.

1 Bevezetés

A környezetből az emberi fülig eljutó különféle zajok nemcsak zavaróak lehetnek, hanem nagy energiájuk révén akár fizikai sérüléshez, a hallás romlásához ill. teljes elvesztéséhez is vezethetnek. Ez a veszély az ipar, a közlekedés fejlődésével egyre inkább napi problémává válik, mivel a nagy teljesítményű munkagépek, ill. közlekedési eszközök jelentős akusztikus szennyezést termelnek. Az ezen gépek közelében dolgozó emberek hallásának védelme érdekében ezért szükségessé vált olyan módszerek létrehozása, amelyek a fül környezetében csökkentik a környező zaj energiáját. Eleinte csak passzív technikákat alkalmaztak (elkerítés, védőfal, hangelnyelő anyagok), amelyek pusztán fizikai jelenlétükkel, külső energia segítsége nélkül voltak képesek valamilyen szintű zajcsökkentésre. Ezek a módszerek azonban csak abban a frekvenciatartományban hatásosak, ahol a hang hullámhossza kisebb, vagy legalábbis összemérhető a védőeszköz fizikai méreteivel. Mivel a hang sebessége kb. 340 m/s, ezért a fenti feltétel kis frekvenciákon már nem teljesül, ezáltal a zajcsökkentés mértéke a kisfrekvenciás tartományban jelentősen lecsökken. Az ebben a frekvenciatartományban szükséges zajelnyomás passzív módszerekkel csak a védőfalak méreteinek, tehetetlenségének növelésével, ill. különleges hangelnyelő anyagok alkalmazásával érhető el. Ezek a módszerek azonban jelentős költségnövekedést vonnak maguk után, ill. eleve nem is alkalmazhatóak méretük vagy nagy tömegük miatt (pl. a fülvédő esetében sem a tömeg, sem a méret nem növelhető tetszőlegesen).

A problémák megoldására az aktív zajcsökkentő technikák alkalmazása adta meg a választ. Ezeknek a módszereknek az a lényege, hogy külső energia bevonásával (aktív módon), valamilyen beavatkozószer (pl. hangszóró) megfelelő szabályozásával ellentétes fázisú hanghullámot állítunk elő, amely a mérséklendő hanghullámmal való interferenciája révén a zaj csökkenését eredményezi. Az elektromos szabályozási technikák fejlődése egyre nagyobb mértékű elnyomást eredményez. Analóg, s később digitális elven működő megoldások is szép számban születtek, különböző szabályozási-vezérlési módszerek, struktúrák alkalmazásával.

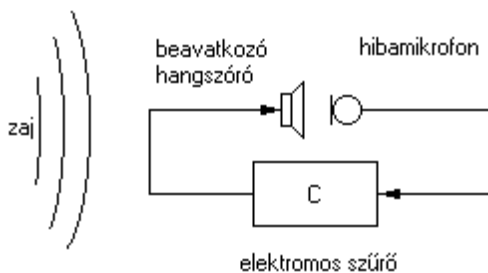
Jelen dolgozat célja egy olyan kombinált analóg–digitális zajcsökkentő rendszer kialakítása, amely az analóg áramkörök gyorsaságának és a digitális algoritmusok rugalmasságának előnyeit igyekszik egyesíteni.

1 Az aktív zajcsökkentés módszerei

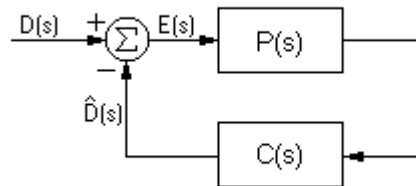
Az utóbbi néhány évtizedben (kb. 1950-től napjainkig) sokféle megoldás napvilágot látott az aktív zajcsökkentés területén. Ezek a megoldások sok mindenben különböznek, mivel az egyes alkalmazási területek más-más igényeket támasztanak. A különböző megoldások közötti hasonlóságot megragadva többféle csoportosítási szempontot is kialakíthatunk. Az első kérdés amire válaszolnunk kell, hogy milyen zajt vagyunk képesek ezekkel a rendszerekkel semlegesíteni? Ennél a kérdésnél máris kettéoszlik a megoldások köre. Az első csoportba tartoznak azok a megoldások, amelyek akár véletlenszerű (szélessávú) zajok kioltására is alkalmasak, a másodikba pedig azok, amelyek leginkább periodikus zajok kioltására képesek. Egy másik csoportosítási szempont pedig az lehet, hogy az elnyomást milyen struktúrájú rendszerrel valósítjuk meg. Itt alapvetően kétféle megoldás létezik: az előre- és a visszacsatolt szabályozás. Mivel a szabályozási struktúra és a szenzorok elhelyezése határozza meg az elnyomható zaj jellegét, ezért célszerűbb talán a különböző szabályozási struktúrák bemutatása mentén megismerkedni az aktív zajcsökkentés módszereivel, lehetőségeivel.

1.1 Zajcsökkentés analóg áramkörökkel

Az aktív zajcsökkentés területén először az analóg áramkörök kerültek alkalmazásra. Az első lépést Harry Olson és Everett May 1953-ban megjelent cikke jelentette [1], ahol egy hangszóróval és az annak szoros közelében elhelyezett mikrofonnal alkotott akusztikusan visszacsatolt aktív zajcsökkentő rendszert ismertettek (1/a. ábra).



1. ábra: Analóg zajcsökkentő felépítése



2. ábra: Az analóg zajcsökkentés hatásvázlata

A hibamikrofon jelét egy elektromos szűrővel megszűrve áll elő a beavatkozó hangszóró irányítójele. A hangszóró által kibocsátott hanghullám a mikrofon és a

hangszóró közötti akusztikus terjedés után jut el a mikrofonig, így jön létre a visszacsatolás.

Az 2. ábrán az 1. ábra blokkvázlatát láthatjuk. Itt $P(s)$ a hangszóró bemeneti feszültsége és a mikrofon kimeneti feszültsége közötti átviteli függvény, a $C(s)$ pedig a szabályozó áramkör átviteli függvényét jelöli. $P(s)$ magában foglalja a hangszóró elektromos-akusztikus, az átviteli közeg akusztikus, valamint a mikrofon akusztikus-elektromos átvitelét.

Mivel az akusztikus térben a hanghullámok összegződnek, ezért az összegző csomópontban a negatív előjel helyett pozitívet kellett volna írni. Mivel azonban a szabályozástechnikában leginkább a negatív visszacsatolást alkalmazzák, így célszerű ezt a megközelítési módot alkalmazni esetünkben is. Ez nem okoz gondot, mivel az így megtervezett szabályozó kimenetének előjelcseréjével megkapjuk a fizikailag is helyes szabályozó átviteli függvényét.

Az ábrán $E(s)$ jelöli a $D(s)$ és a $\hat{D}(s)$, azaz a zaj és az ellenzaj szuperpozícióját.

Célunk tehát a

$$Q(s) = \frac{E(s)}{D(s)} = \frac{1}{1 + H(s)} \quad (1)$$

átviteli függvény abszolút értékének minél kisebbé tétele, ahol

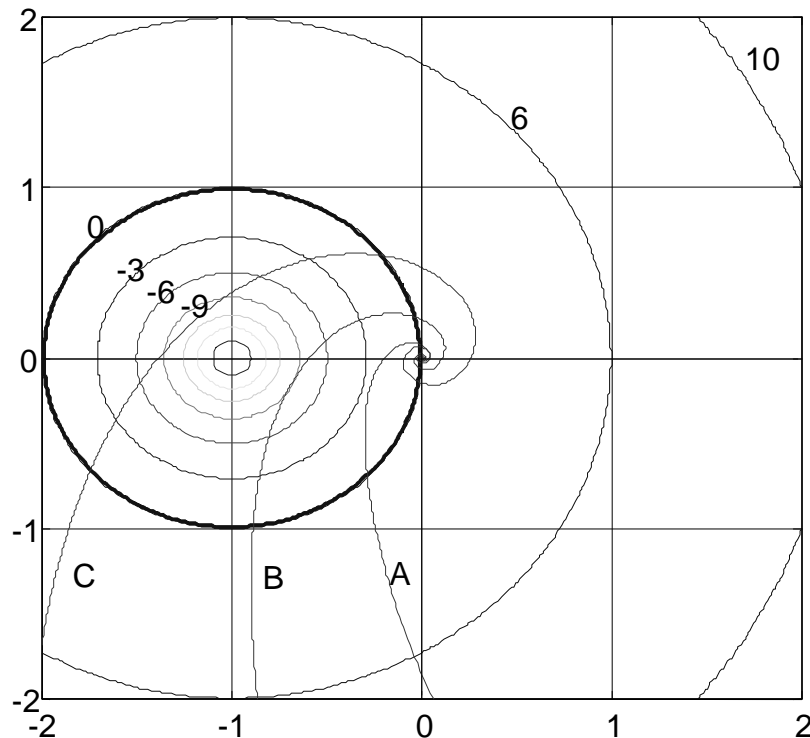
$$H(s) \equiv C(s)P(s) \quad (2)$$

a felnyitott szabályozási hurok átviteli függvénye.

A képletből látható, hogy az elnyomás mértékét $H(s)$ abszolút értékével befolyásolhatjuk: megfelelően nagy $|H(s)|$ esetén nagy elnyomás, míg kis $|H(s)|$ esetén kis elnyomás érhető csak el. Persze a dolog nem ennyire egyszerű, hiszen a stabilitásra is figyelni kell, mivel ha túl nagy a hurokerősítés, akkor a zárt hurok instabillá válhat. A stabilitási kritérium [2] pontosabban úgy hangzik, hogy a felnyitott kör Nyquist-görbéje nem kerülheti meg a $\{-1+j0\}$ komplex pontot.

Ha most az (1) képletben $H(s)$ -t független változónak tekintjük és ábrázoljuk az $|Q(H(s))|$ függvényt, akkor a 3. ábrán látható grafikont kapjuk. Az ábrán a $|Q(H(s))|$ függvény szintvonalas képe látható az s komplex sík felett. A függvény felülete egy $\{-1+j0\}$ -ba eltolt hiperboloid, az azonos abszolút értékű elnyomáshoz tartozó pontok az s -síkkal párhuzamos köröket alkotnak a függvény felületén. Látható, hogyha a Q Nyquist-görbéje belép a $\{-1+j0\}$ középpontú, 1 sugarú körbe, akkor erősítés lép fel (ezentúl

nevezzük ezt erősítési körnek), ezen kívül a rendszer csillapít. Minél közelebb van a Nyquist-görbe ($H(s)$) a Q függvény $\{-1+j0\}$ pontbeli szingularitásához, annál inkább



3. ábra: A nyitott hurok Nyquist-görbéje és a zajcsökkentés mértéke közötti kapcsolat

nő az erősítés. Az A-val jelölt Nyquist-görbe legnagyobb erősítése kb. 3 dB, a B-vel jelölté kb. 9 dB, míg a C jelű rendszert visszacsatolva instabil hálózathoz jutunk, mivel létezik egy (vagy akár több) olyan frekvencia, amelyen a visszacsatolás valós (pozitív vagy negatív attól függően, hogy negatív vagy pozitív visszacsatolást alkalmazunk-e) és a visszacsatolás abszolút értékben nagyobb egynél. Ebben az esetben pedig a kimenő jel minden határon túl növekszik.

Célunk egy olyan $H(s)$ függvény keresése, amely nem lép be az erősítési körbe, és ezen a körön kívül minél távolabb van az origótól. Egy elsőfokú aluláteresztő szűrő biztosan teljesíti ezeket a kritériumokat, hiszen annak Nyquist-görbéje a jobb félsíkon található, így elvileg tetszőlegesen nagy erősítést alkalmazva sem sértjük meg a stabilitási feltételt. Az is látszik, hogy kisfrekvenciákon nagyon nagy elnyomás érhető el, míg a nagyobb frekvenciák felé haladva ez a lehetőség fokozatosan eltűnik: az erősítés 1 lesz.

Másodfokú aluláteresztő esetén a stabilitás már feltételes, azaz bizonyos erősítésszint felett a Nyquist-görbe körülveszi a szinguláris pontot, így a visszacsatolt rendszer instabillá válik. A zajcsökkentés szempontjából is kedvezőtlenebb a helyzet, mivel a Nyquist-görbe biztosan belép az erősítési körbe, így bizonyos frekvenciákon erősítés, másokon pedig csillapítás jelenik meg. Ha növeljük a kör erősítését nemcsak a kisfrekvenciás elnyomást, hanem a nagyfrekvenciás erősítést is növeljük.

Ezek után megfogalmazhatók az analóg szabályozó tervezésénél figyelembe veendő szempontok:

- 1) A hálózat stabil legyen (a Nyquist-görbe ne vegye körül a szinguláris pontot).
- 2) Minél nagyobb legyen az elnyomás (a Nyquist-görbe az erősítési körön kívül minél távolabb legyen az origótól).
- 3) Minél kisebb legyen az erősítés (a Nyquist-görbe az erősítési körön belül minél közelebb legyen az origóhoz).

1.1.1 A késleltetés hatása

Az eddigiekben megfogalmazott állítások koncentrált paraméterű szabályozott szakaszra ($P(s)$) vonatkoztak, de mivel az elrendezésben $P(s)$ a hangszóró és mikrofon közötti hangterjedés miatt jelentős késleltetést tartalmaz, így szükséges megvizsgálni, hogy milyen következményekkel jár ennek a késleltetésnek a jelenléte a zajcsökkentés megvalósíthatósága szempontjából.

A számottevő késleltetésnek az az oka, hogy a mikrofont a hangszóró méretei miatt nem tudjuk elég közel rakni a hangszóróhoz, ugyanis ha egy pontjához közel rakjuk, akkor a többi pontjától távolabbra kerül, és ezekből a pontokból lassabban jut el a mikrofon membránjáig a hanghullám. Ha pedig a hangszóró méretét csökkentjük, akkor a leadható akusztikus teljesítmény csökken.

Legyen most

$$P(s) \equiv e^{-s\tau}, \quad (3)$$

azaz egy tiszta holtidős tag. Ennek a tagnak minden frekvencián egységnyi az abszolút értéke és fázisa a frekvencia növekedtével lineárisan növekszik (negatív irányban). Ha $P(s)$ a késleltetésen kívül koncentrált paraméterű részt is tartalmaz, akkor annak fázisához a késleltetés miatt a frekvencia növekedtével egyre nagyobb fázismennyiség adódik hozzá. Például ha a hangszóró–mikrofon távolság 3.4 cm, akkor a hang viszonylag kis sebessége (~ 340 m/s) miatt a késleltetésből származó fázistolás már 5

kHz-en eléri a 180° -ot. Ez pedig már jelentős mennyiség, ha figyelembe vesszük, hogy az általunk megcélzott frekvenciatartomány kb. 0 és 500 Hz között helyezkedik el (kb. ez az a tartomány, ahol a passzív zajcsökkentés már nem igazán működik). Ugyanis a holtidős tag a koncentrált paraméterű rész Nyquist-görbáját mintegy felcsavarja az origóra, amely csökkenti a stabilitási tartalékokat (a 3. ábrán is ezt látjuk: a görbék eredetileg másodfokú aluláteresztők voltak, de a holtidős tag miatt „befordultak” az összes síknegyedbe). Az 500 Hz törésponti frekvenciájú elsőfokú aluláteresztő szűrő esetén például az 500 Hz és a vágási frekvencia között kevesebb, mint egy dekád távolság van, így a maximális hurokerősítés kisebb mint 20 dB, amely kevesebb mint 20 dB elnyomást tesz csak lehetővé. A hurokerősítést azonban méginkább csökkenteni kell, mivel a felcsavarodás miatt a nagyobb abszolút értékű kismagyas átviteli pontok befordulnak az erősítési körbe, s emiatt jelentős lesz a zajerősítés ebben a frekvenciatartományban. A zajerősítés csökkentéséhez tehát csökkentenünk kell a zajelnyomást is.

Bode egyik tétele szerint [3], ha $H(s)$ koncentrált és holtidős tagot is tartalmaz, akkor

$$\int_0^{\infty} \ln|Q(j\omega)|d\omega = 0, \quad (4)$$

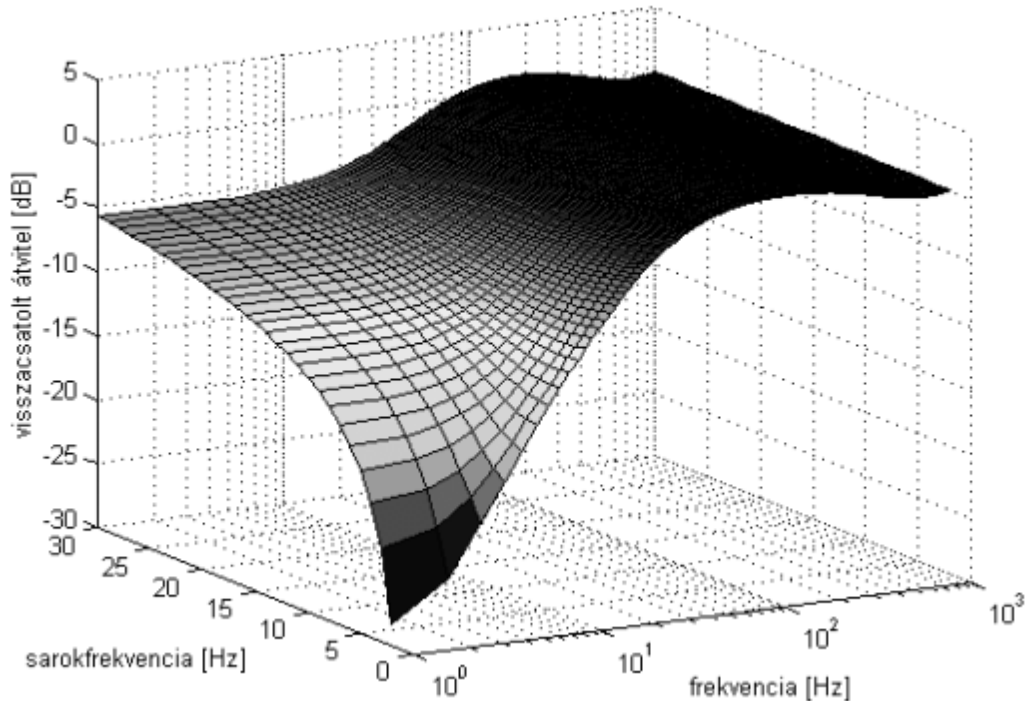
azaz összességében az elnyomás és az erősítés megegyezik (ez az ún. vízány jelenség). Amit el lehet érni egy ilyen rendszerben az az, hogy az erősítés mértékét valamilyen elfogadható szintre csökkentjük (vagyis a frekvenciatengely mentén megfelelően szétszjtjuk). Ez a csökkentés azonban csak az elnyomás nagyságának vagy az elnyomás sáv szélességének rovására hajtható végre.

1.1.2 Visszacatoló hálózatok

Ebben az alfejezetben megvizsgálunk néhány alapvető visszacsatoló hálózatot és igyekszünk olyan $C(s)$ függvényt találni, amely megfelel céljainknak ($P(s)$ -t továbbra is holtidős tagnak vesszük). A keresés intuitív, mivel sok feltétel betartására kell egyszerre figyelni, ezek a feltételek pedig kölcsönösen függenek egymástól. Az intuitív módszer miatt nem garantálható, hogy a kiadódó hálózat optimális: gyakorlottabb szakemberek talán jobb hálózatot is létre tudnának hozni.

A vizsgálódást a minimálfázisú hálózatok osztályára korlátozhatjuk, ugyanis a mindentáteresztő hálózatok csak további fölösleges fázistöbbletet okoznak.

Először is vizsgáljuk meg a legegyszerűbb elsőfokú aluláteresztő szűrőt. A 4. ábrán azt láthatjuk, hogy milyen elnyomás érhető el különböző frekvenciákon, ha a szűrő törésponti frekvenciáját változtatjuk. Az ábra úgy készült, hogy a holtidős tag fáziskarakterisztikája 1 kHz-en éri el a 180° -os fázistolást, valamint az erősítésmaximumot 3 dB-ben határoztam meg.



4. ábra A zajelnyomás lehetőségei elsőfokú aluláteresztő szűrő esetén

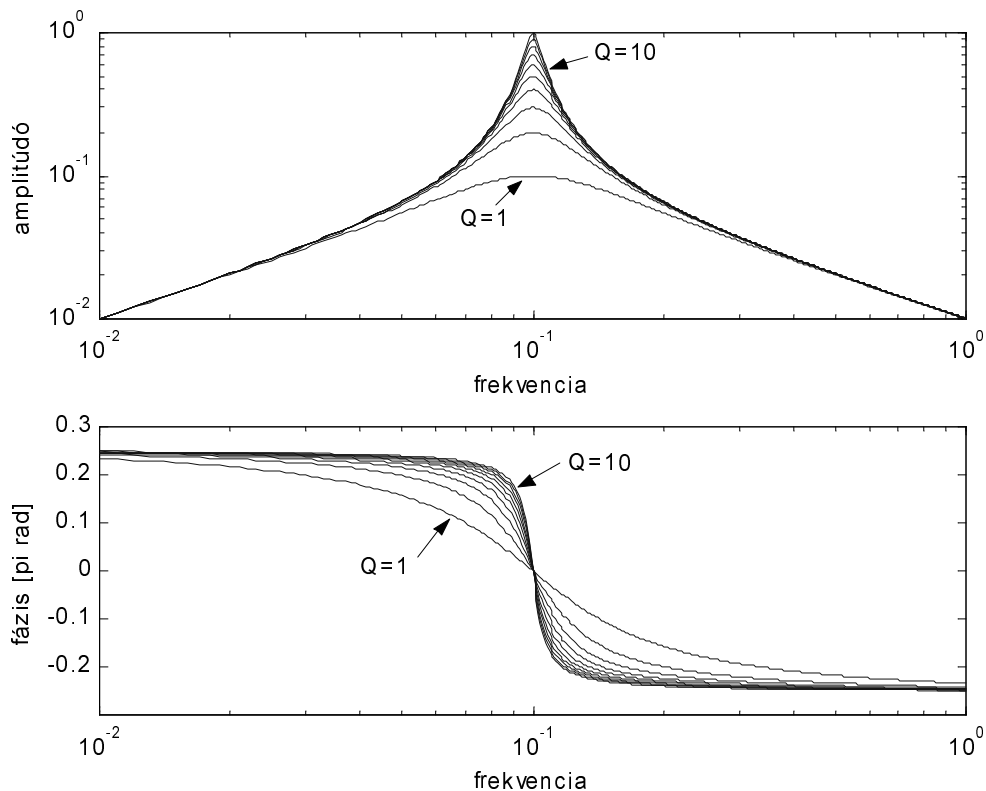
Láthatjuk hogy a helyzet igen kedvezőtlen, mivel csak nagyon kis frekvenciákon vagyunk képesek számottevő zajcsökkentésre. Például ha 20 dB-es elnyomást szeretnénk, akkor ezt csak 5 Hz-nél kisebb töréspontú szűrővel oldhatjuk meg, és az elnyomási tartomány 1-2 Hz. Ha megelégszünk 6 dB-es elnyomással, akkor is csak kb. 10 Hz-ig vagyunk képesek teljesíteni a feltételt.

Nem segít az sem, ha a zárótartományban engedményt teszünk, azaz visszatörjük a -20 dB/dekádos amplitúdókarakterisztikát 0 dB/dekádossá, mivel ekkor a zárótartománybeli nagyobb erősítés miatt kevesebb lesz az erősítéstartalék (ezt némileg kompenzálja, hogy a fáziskarakterisztikát is visszatérítjük a nullába, ezáltal pedig kicsit növekszik az erősítéstartalék, mivel a nagyobb amplitúdójú kisfrekvenciás átvitel kevésbé fordul bele az erősítési körbe).

Különböző visszacsatoló hálózatokon végzett próbálkozások után arra a következtetésre jutottam, hogy a legkedvezőbb a számlálóban elsőfokú, a nevezőben másodfokú sávszűrő, azaz a

$$H(s) = A_0 \frac{s}{1 + \frac{s}{\omega_p Q} + \frac{s^2}{\omega_p^2}} \quad (5)$$

átviteli függvényű hálózat. Ennek maximális amplitúdója az ω_p pólus- vagy más néven rezonanciafrekvenciára esik, értéke $A_0 \omega_p Q$. A karakterisztikák az 5. ábrán láthatók.



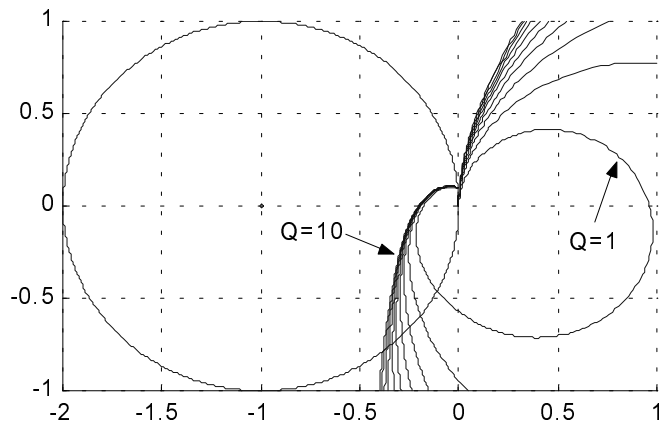
5. ábra A másodfokú sáváteresztő szűrő karakterisztikái a jósági tényező függvényében

Az ábrán $\omega_p = 0.1 \text{ rad/s}$ mellett a jósági tényező hatását láthatjuk az amplitúdóra és a fázisra.

Nagyobb jósági tényező nagyobb kiemelést jelent, és a $+90^\circ$ -ból -90° -ba történő fázistranziens is gyorsul Q növelésével.

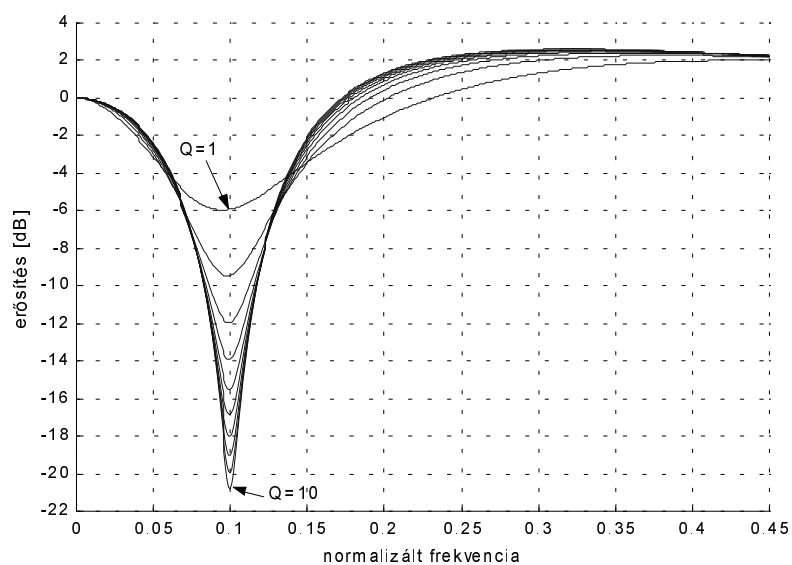
Mivel ennek a szűrőnek már 2 paramétere van, az elsőfokú aluláteresztő szűrőnél alkalmazott kétdimenziós ábrázolás nem lehetséges. Ezért külön érdemes megvizsgálni Q és ω_p hatását.

A következő ábrák úgy készültek, hogy a holtidős tag -180° -os fázisához tartozó körfrekvenciát egynek vettem. A 6. ábrán a felnyitott kör Nyquist-görbéjét láthatjuk különböző Q értékek mellett ($A_0=10$, $\omega_p=0.1$).



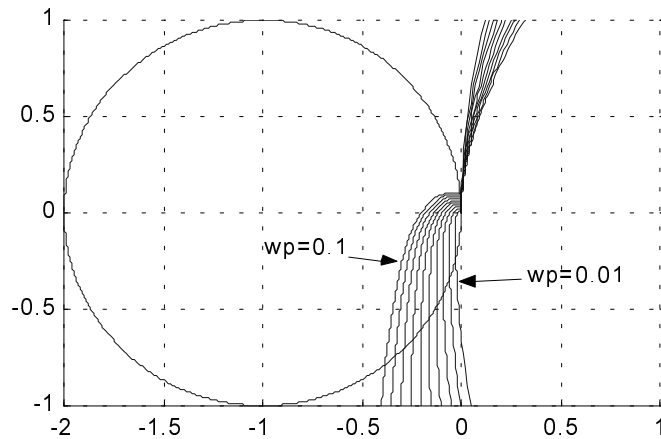
6. ábra Holtidős taggal kaszkádba kapcsolt másodfokú sávszűrő Nyquist-görbéje a Q függvényében

Láthatjuk, hogy a rendszer egyrészt stabil, mivel a Nyquist diagram nem veszi körül a szinguláris pontot, másrészt Q növelésével a stabilitási tartalékok alig csökkennek. Ezek után vizsgáljuk meg, hogy a zajcsökkentés szempontjából hogyan alakul a helyzet:



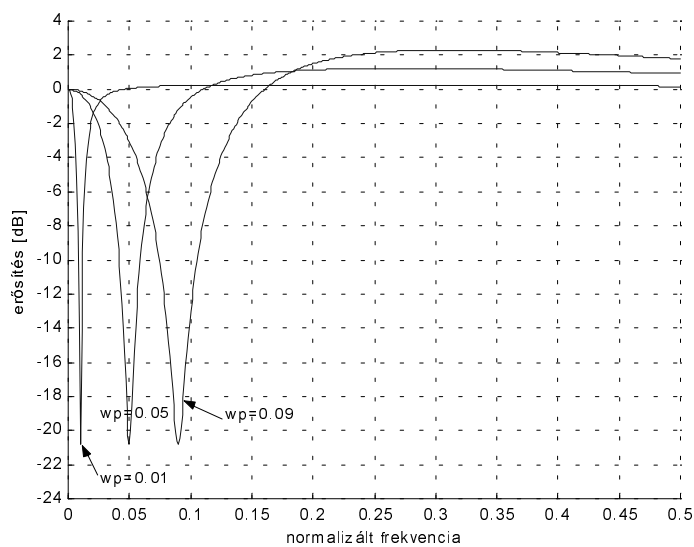
7. ábra Holtidős taggal visszacsatolt sávszűrő zajerősítése a Q függvényében

Az ábrából kitűnik, hogy Q növelésével egyrészt növelhetjük az elnyomást (és valamelyest csökkentjük az elnyomás sávszélességét), másrészt ezzel együtt nő a zajerősítés is (illetve annak sávszélessége megnő a Bode-egyenetnek (4) megfelelően). Ha most a rezonanciafrekvenciát változtatjuk, akkor a következő Nyquist-görbéket kapjuk ($Q = 10$ esetén):



8. ábra Holtidős taggal kaszkádba kapcsolt sávszűrő Nyquist-görbéje az ω_p függvényében

Világos, hogy ω_p minél közelebb kerül egyhez, annál inkább csökkennek a stabilitási tartalékok, így ügyelnünk kell arra, hogy mekkorára választjuk ω_p -t. Ugyanakkor a 9. ábra szerint a rezonanciafrekvencia növelésével növekszik a zajcsökkentés sávszélessége:



9. ábra Holtidős taggal visszacsatolt sávszűrő zajerősítése az ω_p függvényében

Ezenkívül természetesen ω_p növelése az erősítést is növeli.

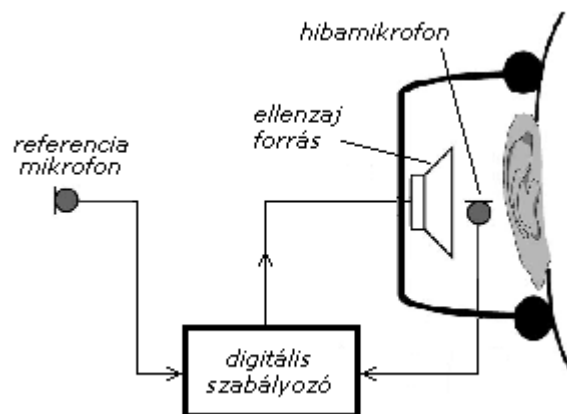
Összegzésképpen tehát azt állapíthatjuk meg, hogy ez a sávszűrő képes nagymértékű elnyomásra: az elnyomás mértéke Q - tól függ, a stabilitási tartalék és a zajerősítés mértéke pedig ω_p növelésével egyre kedvezőtlenebbé válik.

1.2 Digitális zajcsökkentés

Ebben a fejezetben áttekintjük a dolgozat szempontjából lényeges digitális zajcsökkentő megoldásokat ([1],[4]) és az alkalmazott algoritmusokat ([1],[4],[5]). Először megismerkedünk az előrecsatolt és a visszacsatolt zajcsökkentés alapvető struktúráival, majd rátérünk az adaptációt megvalósító algoritmusok tárgyalására. Bizonyos feltételezésekkel a zajcsökkentési struktúrák az adaptív prediktorhoz hasonló felépítésűek, ezért célszerűnek találtam erről is egy kisebb összefoglalót készíteni. Ezek után az adaptív digitális zajcsökkentés területén széles körben alkalmazott LMS algoritmussal ismerkedünk meg, majd rátérünk annak egy módosítására, az FXLMS algoritmusra.

1.2.1 Az előrecsatolt szabályozás

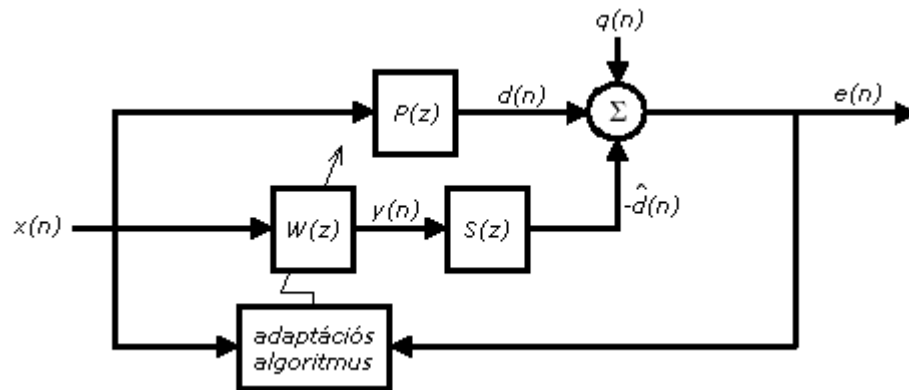
Az előrecsatolt szabályozás lényege, hogy a zaj forrásából származó információk alapján hozunk döntést a beavatkozó szerv mozgására vonatkozóan. Egy tipikus előrecsatolt zajcsökkentő rendszert láthatunk az 10. ábrán.



10. ábra Az előrecsatolt szabályozás elvi felépítése

A digitális szabályozó számára a kioltandó zaj információt a referenciamikrofon szolgáltatja. Ezt a jelet kell a digitális szabályozónak úgy módosítani, hogy az ellenzajt létrehozó hangszóró által kibocsátott hang lerombolja a fejhallgatóba bejutott hanghullámot. A rombolás sikerességéről a hangszóró közelében elhelyezett hibamikrofon szolgáltat információt, amely alapján a digitális szabályozó algoritmus

úgy tudja módosítani paramétereit, hogy a hibajel, azaz a hibamikrofon környékén lévő térrész zajszintje lecsökkenjen. A referenciamikrofont általában a zaj forrásához közel szokták elhelyezni, mivel a zajelnyomás csak akkor lesz elég nagy, ha a hibamikrofonnál megjelenő zaj és a referenzia zaj között nagy a korreláció. Az előreccsatolt szabályozási elv tehát ott alkalmazható sikeresen, ahol van egy domináns zajforrás, amelynek közelében el tudjuk helyezni a referenziajelet szolgáltató szenzort. Az előreccsatolt rendszer blokkvázlatát a 11. ábrán láthatjuk.



11. ábra Előreccsatolt zajcsökkentő rendszer hatásvázlata

Itt $x(n)$ jelöli a referenciamikrofon, $e(n)$ pedig a hibamikrofon jelét. Az ábrán $d(n)$ jelöli a hibamikrofonnál megjelenő zajnak azon részét, amely a referenzia zajjal korrelált, $q(n)$ pedig azt, amely a referenziajellel korrelálatlan. Mivel a korrelálatlan zaj statisztikailag független a referenziajeltől, ezért a referenziajeltől nem állítható elő olyan ellenzaj, amely $q(n)$ -t képest statisztikailag csökkenteni. Így $q(n)$ gyakorlatilag zavarjelként továbbra is jelen lesz az ellenzajjal szuperponált zajban.

A referenciamikrofon és a hibamikrofon közötti akusztikus átvitelt az ábrán $P(z)$, a hangszórót meghajtó feszültségnek megfelelő számérték ($y(n)$) és a hibamikrofon jelének megfelelő számérték közötti átvitelt $S(z)$ jelöli. Ez utóbbi tartalmazza az A/D-D/A átalakítások átvitelét, a hangszóró elektroakusztikus átvitelét, a hangszóró és a mikrofon közötti akusztikus jelterjedés átvitelét és a mikrofon akusztikus-elektromos átvitelét.

Ha $q(n)$ -t figyelmen kívül hagyjuk, akkor az elérendő cél az $e(n)$ jel zérussá tétele:

$$e(n) = d(n) - \hat{d}(n) = X(z)[P(z) - W(z)S(z)] = 0 \quad (6)$$

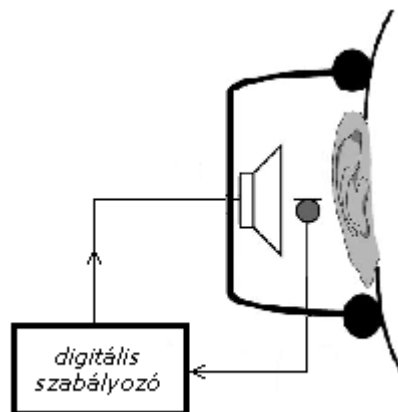
Ez csak akkor teljesülhet, ha

$$W(z) = W_{opt}(z) = \frac{P(z)}{S(z)}, \quad (7)$$

azaz az optimális $W(z)$ -nek egyszerre kell modelleznie $P(z)$ -t és $S(z)$ inverzét. Mivel $S(z)$ egyes esetekben nem invertálható (az inverz nem stabil), így előfordulhat, hogy a teljes kioltás nem érhető el. Például ha $P(z)$ -t és $S(z)$ -t is egyszerű késleltetésnek vesszük, akkor létezik csak teljes kioltás, ha $P(z)$ késleltetése nagyobb, mint az $S(z)$ késleltetése. Ha ez nem teljesül, akkor a $W(z)$ már csak jóslni tud, ami korlátozza a zajcsökkentés sáv szélességét.

1.2.2 A visszacsatolt szabályozás

Az előrecsatolt szabályozással ellentétben a visszacsatolt szabályozás nem a zavarforrásból származó referenciajelből állítja elő a beavatkozószervert irányítójelet, hanem a zajcsökkentés célhelyéről származó információkból (12. ábra).

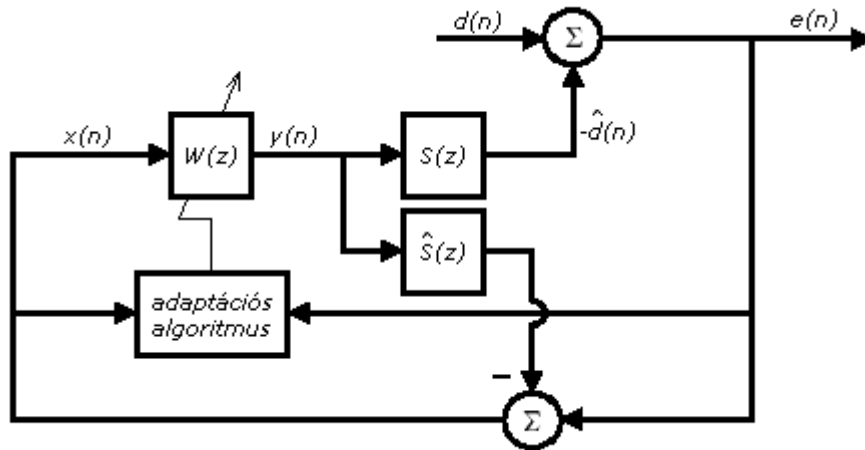


12. ábra Visszacsatolt zajcsökkentő rendszer elvi felépítése

A digitális szabályozó lehet fix vagy adaptív. Fix digitális visszacsatolt szabályozást az analóg visszacsatolt szabályozáshoz hasonlóan terveznek, ezért ez elvi újdonságot nem jelent, így erre nem térünk ki.

Az adaptív visszacsatolt szabályozók viszont a visszacsatolt analóg szabályozótól teljesen eltérő elven működnek. Míg az analóg esetben a cél olyan akusztikus visszacsatolás létrehozása volt, amely a zajcsökkentést létrehozza, addig a digitális esetben pont ennek a visszacsatolásnak a megakadályozása a cél.

Ha a digitális visszacsatolt szabályozó blokkvázlatára (13. ábra) tekintünk, akkor láthatjuk, hogy a „visszacsatolt” elnevezés eléggé félrevezető, tulajdonképpen csak az elrendezés hasonlóságával magyarázható.



13. ábra Visszacsatolt szabályozás hatásvázlata

A jelölések megegyeznek az előrecsatolt szabályozás blokkvázlatánál használt jelölésekkel. A $d(n)$ zavarjelet ismét $-\hat{d}(n)$ kioltó jellel kell összeadnunk, hogy kioltssuk a zavarjelet. Ha most az ábráról gondolatban kihagyjuk az $\hat{S}(z)$ átviteli függvényt, akkor a referencijel:

$$x(n) \equiv e(n), \quad (8)$$

tehát a mikrofon jele egy az egyben visszacsatolódik a $W(z)$ szűrő bemenetére, így az analóg visszacsatolt rendszerhez hasonló struktúra jön létre. Ez esetben ugyanazokkal a problémákkal találjuk szemben magunkat: stabilitás, korlátozott sáv szélesség, vízígy effektus, stb. Ennek elkerülése végett került az ábrára az $\hat{S}(z)$ átviteli függvény, amely az $S(z)$ átvitel becslője. Pontos becslés esetén ugyanis a visszacsatolás teljesen megszüntethető, s ekkor

$$x(n) \equiv d(n), \quad (9)$$

azaz a referencijel megegyezik a kioltandó zajjal. Ezért a kapcsolás ugyanúgy viselkedik, mint ahogyan az előrecsatolt szabályozás, azzal a különbséggel, hogy itt a referencia- és a hibamikrofon megegyezik, azaz

$$P(z) \equiv 1. \quad (10)$$

Így a (7) képletnek megfelelően

$$W_{opt}(z) = \frac{1}{S(z)}, \quad (11)$$

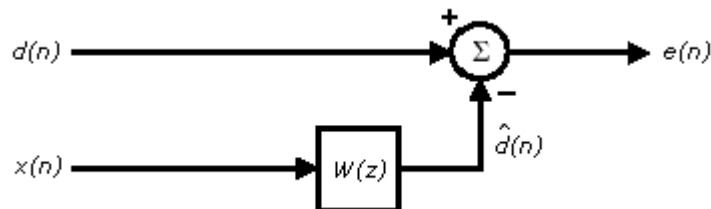
ami $S(z)$ inverz modellezését követeli meg, ha az elérendő cél a teljes kioltás. Mivel $S(z)$ általában nem invertálható (késleltetést tartalmaz), ezért tökéletes szélessávú kioltás ezzel a struktúrával nem érhető el. Keskenyebb sávú részleges elnyomás viszont elképzelhető, ha egy adott karakterű zajhoz találunk olyan $W(z)$ szűrőt, amely valamilyen, a hibajel csökkentő kritériumfüggvény szempontjából optimális.

1.2.3 Adaptációs algoritmusok

Az eddigiekben feltételeztük, hogy a rendelkezésünkre áll egy olyan adaptációs algoritmus, amely képes úgy módosítani egy (transzverzális) szűrő együtthatóit, hogy a referenciajelet ezzel a szűrővel megszűrve a hibajel lecsökkenjen. Mivel az adaptív algoritmusok családja elég tekintélyes, ezért itt csak azokkal foglalkozunk, amelyek érdemben kapcsolódnak a dolgozat témájához.

1.2.3.1 Az alapprobléma

Legyen $d(n)$ egy ismeretlen és $x(n)$ egy rendelkezésre álló (mérhető) stacionárius referenciajel (14. ábra).



14. ábra Az adaptív zajcsökkentés alapproblémája

A feladat a $W(z)$ transzverzális szűrő paramétereinek megkeresése, mégpedig úgy, hogy $x(n)$ -t a $W(z)$ szűrőn keresztül megszűrve várható értékben minél kisebb $e(n)$ hibát kapjunk. Nyilván ahhoz, hogy ezt megtehesük, feltételeznünk kell, hogy a $d(n)$ és az $x(n)$ jelek között valamilyen kapcsolat van (ha nem is determinisztikus, de statisztikus kapcsolat mindenképpen).

A hibajel a következőképpen fejezhetjük ki:

$$e(n) = d(n) - \hat{d}(n) = d(n) - \sum_{m=0}^{N_w-1} w_m(n)x(n-m) = d(n) - \mathbf{w}(n)^T \mathbf{x}(n). \quad (12)$$

Ennek a hibának a négyzetes várható értékét, tehát az

$$\varepsilon = E\{e(n)^2\} \quad (13)$$

kritériumfüggvényt, mint minimalizálandó függvényt használva az optimális szűrő a Wiener-Hopf egyenletnek megfelelően

$$\mathbf{w}_{opt}^T = \mathbf{R}^{-1} \mathbf{p}^T. \quad (14)$$

Itt az \mathbf{R} kvadratikus mátrix az ún. autokorrelációs mátrix:

$$\mathbf{R} = \begin{bmatrix} r(0) & r(1) & r(2) & \dots & r(N-1) \\ r(1) & r(0) & r(1) & & r(N-2) \\ r(2) & r(1) & r(0) & & : \\ : & & & & r(1) \\ r(N-1) & r(N-2) & \dots & r(1) & r(0) \end{bmatrix}, \quad (15)$$

ahol az $r(\cdot)$ függvény az $x(n)$ jel autokorrelációs függvénye

$$r(n, m) = E\{x(n)x(m)\} = r(n-m) = r(m-n) \quad (16)$$

stacionárius $x(n)$ jel esetén.

A (14) egyenletben a \mathbf{p}^T vektor a $d(n)$ és az $\mathbf{x}(n)$ jelek közötti keresztkorrelációs vektor:

$$\mathbf{p}^T = \begin{bmatrix} r_{dx}(0) \\ r_{dx}(1) \\ : \\ r_{dx}(N-2) \\ r_{dx}(N-1) \end{bmatrix}, \quad (17)$$

ahol az $r_{dx}(\cdot)$ függvény a $d(n)$ és az $x(n)$ jelek közti keresztkorrelációs függvény:

$$r_{dx}(n, m) = E\{d(n)x(m)\} = r_{dx}(n-m) = r_{dx}(m-n). \quad (18)$$

Az optimális szűrőbeállításnál a kritériumfüggvény minimális, s ennek értéke

$$\varepsilon_{\min} = E\{d(n)^2\} - \mathbf{p}^T \mathbf{R}^{-1} \mathbf{p} = P - \mathbf{p}^T \mathbf{R}^{-1} \mathbf{p}. \quad (19)$$

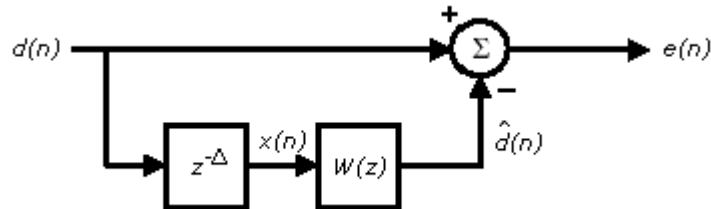
A minimum tehát függ a $d(n)$ és az $x(n)$ jelek közötti keresztkorrelációtól és az $x(n)$ jel autokorrelációjától.

1.2.3.2 Az adaptív prediktor

Ha visszagondolunk a visszacsatolt adaptív szabályozóra, az $S(z)$ átvitel jelenléte korlátozó tényezőt jelentett a zajelnyomás lehetőségét illetően, mivel ebben az átviteli függvényben késleltetés van jelen, amely lehetetlenné teszi $S(z)$ invertálását. Ha

$$S(z) \equiv z^{-\Delta}, \quad (20)$$

azaz egy egyszerű késleltetés, akkor rendszerünk az ún. adaptív prediktorral megegyező felépítéssel rendelkezik (15. ábra). Az ábrán $W(z)$ és $S(z)$ a 13. ábrához képest fel van cserélve (a linearitás miatt ez megtehető) azért, hogy az ezt megelőző alfejezetben használt jelölések és eredmények közvetlenül alkalmazhatók legyenek itt is.



15. ábra Az adaptív prediktor hatásvázlata

A zavarjel teljes kioltásához $W(z)$ -nek jóslónia kellene, innen ered az elnevezés. Mivel a jóslás nem lehetséges, ezért be kell érünk valamilyen becsléssel, ami a $d(n)$ jel statisztikájának ismeretében tehető meg. Azt, hogy adott statisztikájú jelhez mekkora elnyomás érhető el, meghatározhatjuk a (19) egyenletből. Mivel most $d(n)$ és $x(n)$ között determinisztikus kapcsolat (eltolás) van, ezért a közöttük lévő keresztkorrelációs függvény megegyezik a $d(n)$ autokorrelációs függvényének Δ -val való eltoltjával. Ez azt jelenti, hogy az \mathbf{R} mátrix változatlanul (15)-nek megfelelő, míg

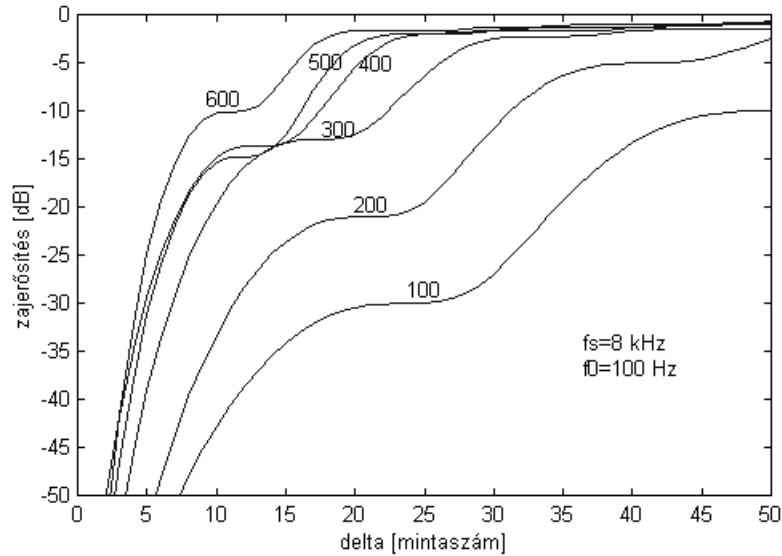
$$\mathbf{p}^T = \begin{bmatrix} r(\Delta) \\ r(\Delta+1) \\ \vdots \\ r(\Delta+N-2) \\ r(\Delta+N-1) \end{bmatrix} \quad (21)$$

szerint módosul.

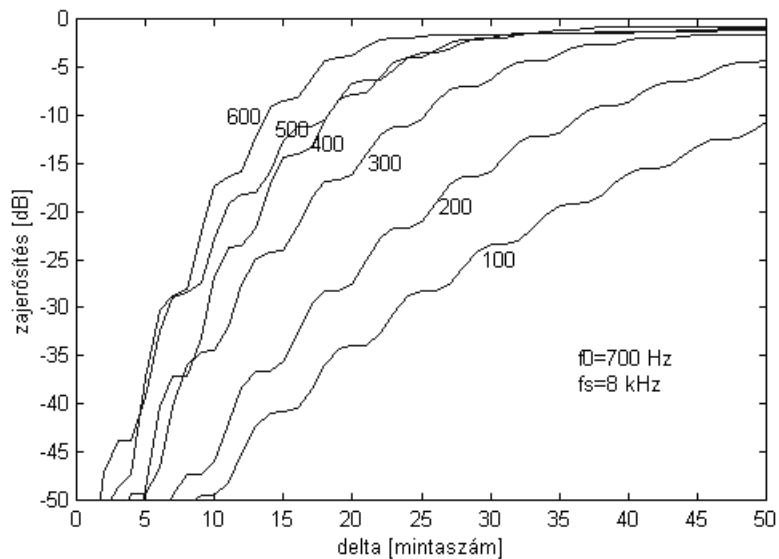
Mivel ε_{\min} értéke a bemenő jel teljesítményétől függ, ezért érdemes az (19) egyenletet a teljesítmény szerint normalizálni:

$$\varepsilon_{\min, \text{norm}} = \frac{P - \mathbf{pR}^{-1}\mathbf{p}^T}{P} = 1 - \frac{\mathbf{pR}^{-1}\mathbf{p}^T}{P}. \quad (22)$$

Az így kapott relatív hibaminimum képletet Matlab alatt különböző bemenő jelekkel szimulálva a 16. és 17. ábrát kapjuk.



16. ábra Az adaptív prediktor maradó hibája a késleltetés és a zaj sávszélessége függvényében 100 Hz alapfrekvencián



17. ábra Az adaptív prediktor maradó hibája a késleltetés és a zaj sávszélessége függvényében 700 Hz alapfrekvencián

Az ábrák azt mutatják, hogy mekkora a maradó hiba teljesítménye az egységnyi bemenőjel-teljesítményhez viszonyítva. Az alapfrekvencia (f_0) jelenti a fehérzajt áteresztő sávszűrő alsó törésponti frekvenciáját, a görbék mellé írt számok pedig a szűrő

sávszélességét (Hz-ben). Az alapprofrekvencia az első ábrán 100 Hz, a másodikon 700 Hz volt (a mintavételi frekvenciát 8 kHz-nek vettem, mert a megvalósított algoritmus is 8 kHz-en működik). Az ábrákról az olvasható le, hogy Δ növelésével az elnyomás egyre inkább eltűnik, tehát a jóslás egyre kevésbé lehetséges. Másrészt a zaj sávszélességének növelésekor is csökken a jóslás hatékonysága, mivel a zaj egyre korrelálatlanabb lesz. A két ábrát összevetve az is látszik, hogy az elnyomás mértéke nagyjából független a zaj alapprofrekvenciájától.

1.2.3.3 Az LMS algoritmuscsalád

A Wiener-Hopf egyenlet (14) egy lépésben adta meg az optimális szűrőbeállítást: a megoldás az \mathbf{R} és a \mathbf{p}^T ismeretét feltételezte. A probléma felvetésénél viszont feltételeztük, hogy a $d(n)$ jel ismeretlen, így a \mathbf{p}^T ismerete nem lehetséges, s az \mathbf{R} mátrixra is csak becslések adhatóak, ha nem ismerjük pontosan az $x(n)$ jel autokorrelációs függvényét. Ezért a gyakorlatban jobban használhatók azok az algoritmusok, amelyek iteratív módon, azaz több lépésben próbálják meg megtalálni az optimális szűrőbeállítást. Ezek az iteratív algoritmusok azt használják ki, hogy az ε hibafüggvény a \mathbf{w} paramétertér felett kvadratikus, azaz egy többdimenziós parabola. E miatt a hibafületnek csak egy minimuma van, így a hibafüggvény negatív gradiense mindig ebbe az irányba mutat, s így a gradiens becslésével a minimum felé ereszkedhetünk:

$$\mathbf{w}(n+1) = \mathbf{w}(n) - \frac{\mu}{2} \frac{\partial \hat{\varepsilon}(n)}{\partial \mathbf{w}}, \quad (23)$$

ahol μ a lépés nagyságát befolyásoló paraméter.

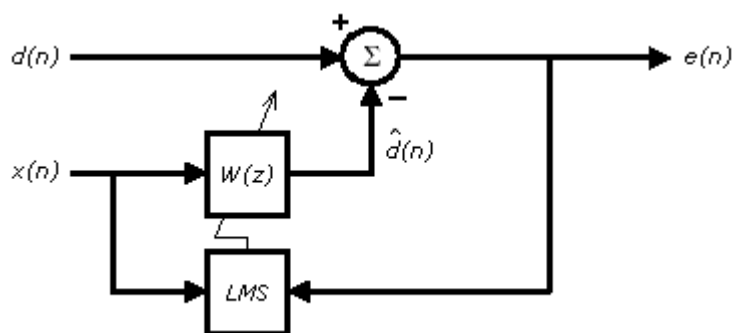
Sok, a negatív gradiens elvén alapuló algoritmus létezik, ezek közül az adaptív rendszerek legelterjedtebb algoritmusai az LMS (Least Mean Squares) algoritmus, illetve annak különböző variánsai. Népszerűségét annak köszönheti, hogy kis számítási igényéhez képest meglepően jól alkalmazható az alkalmazások széles területén.

A költségfüggvény az LMS algoritmusnál (mint azt neve is mutatja) a hiba négyzete:

$$\hat{\varepsilon}(n) = e^2(n) = [d(n) - \mathbf{w}^T(n)\mathbf{x}(n)]^2. \quad (24)$$

Ekkor

$$\frac{\partial \hat{\varepsilon}(n)}{\partial \mathbf{w}} = -2\mathbf{x}(n)e(n), \quad (25)$$



18. ábra Az LMS algoritmus hatásvázlata

ezt (23)-ba helyettesítve a

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \mu \mathbf{x}(n)e(n) \quad (26)$$

adaptációs algoritmushoz jutunk. A képletben μ jelöli az adaptációs lépésközt. Ennek a paraméternek a megválasztásán múlik tulajdonképpen az adaptáció sikere. Ha értékét túl nagyra adjuk meg, akkor az iteráció divergens lesz, tehát az algoritmus nem találja meg a minimumot. A stabilitási tulajdonságok elemzéséből az derül ki, hogy μ -nek kisebbnek kell lennie, mint az \mathbf{R} mátrix legnagyobb sajátértékének reciproka. Mivel \mathbf{R} általában nem ismert, ezért ez a módszer általában nem alkalmazható. Mivel azonban \mathbf{R} -nek csak nemnegatív sajátértékei vannak, ezért ezeknek összegével (ami megegyezik a mátrix nyomával) felülről becsülhető a legnagyobb sajátérték. Így μ maximális értékének meghatározása a gyakorlatban nagyon sokszor a

$$\mu_{\max} = \frac{1}{\mathbf{tr}\{\mathbf{R}\}} = \frac{1}{E\{\mathbf{x}^T(n)\mathbf{x}(n)\}} = \frac{1}{E\left\{\sum_{i=0}^{N-1} x_i^2\right\}} \quad (27)$$

képlet szerint történik (\mathbf{tr} a mátrix nyoma – a főátlójának összege).

Megfelelően nagy μ esetén az iteráció gyorsan eljut a minimum közelébe, de a nagy lépésköz miatt túlugrik a minimumon, tehát a minimum körül fog bolyongani. Megállapítható tehát, hogy a fennmaradó hiba a μ -vel egyenesen arányos. Ebből következik, hogy pontos adaptációhoz csak kis lépésekkel juthatunk el, ez viszont több időt vesz igénybe ugyanakkora út mellett a paramétertartományban.

1.2.4 A normalizált LMS

Kézenfekvőnek tűnik, hogy olyan alkalmazások esetén, ahol az $x(n)$ jel teljesítménye nagyságrendeket változhat, ne fix μ paraméterrel, hanem a (27) képletnek megfelelően történjen az adaptáció. Az algoritmus neve normalizált LMS, a szűrőadaptáció pedig a

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \alpha \frac{\mathbf{x}(n)e(n)}{\mathbf{x}^T(n)\mathbf{x}(n) + \beta} \quad (28)$$

képlet szerint történik. Az α paraméter veszi át a μ szerepét, a normalizálás miatt azonban a stabilitás az $x(n)$ jel teljesítményétől függetlenül a

$$0 < \alpha < 1 \quad (29)$$

feltétel betartásával biztosítható. A β paramétert a nullával való osztás elkerülése érdekében szokás alkalmazni (β egy nagyon kicsi szám).

1.2.5 A FXLMS algoritmus

Az LMS algoritmus nem alkalmazható abban az esetben, ha a referencijel és a mért hibajelnek az a része, amely a referencijelből áll elő nincsenek fázisban egymással. Ekkor ugyanis a szűrőadaptáció nem egymáshoz tartozó paraméterekkel történne, s emiatt a szűrő instabillá válna (pontosabban akkor válik instabillá, ha a fent említett két jel közötti fázis abszolút értékben nagyobb, mint 90° [4]). Az FXLMS algoritmus alapötlete, hogy ha az adaptív szűrő referencijelében éppen akkora fázistolást állítunk elő, mint amennyi az adaptív szűrő kimenete és a hibajel között van, akkor a fázishiba eltűnik.

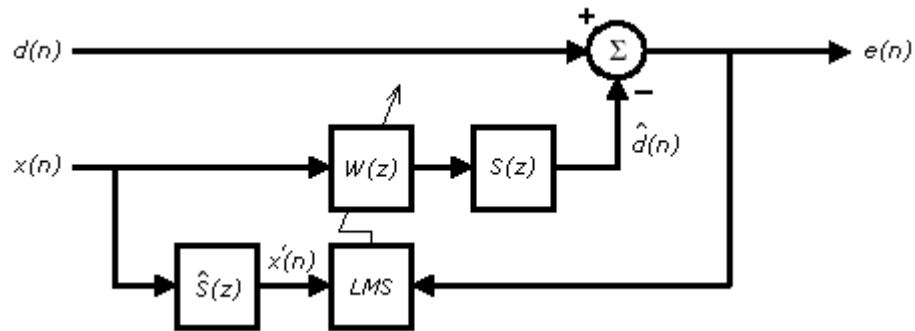
Az adaptációs algoritmus tehát a következő:

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \mu e(n)\mathbf{x}'(n), \quad (30)$$

ahol

$$x'(n) = \sum_{i=0}^{N_s-1} s_i x(n-i). \quad (31)$$

Az s_i paraméterek pedig az $\hat{S}(z)$ transzverzális szűrő együtthatói. Persze lehet IIR-szűrőt is alkalmazni, de inkább a FIR-szűrő használata terjedt el a könnyebb implementálhatóság és a feltétlen stabilitás miatt. Arra kell csupán ügyelni, hogy a szűrő hossza megfelelő nagyságú legyen, azaz $S(z)$ impulzusválaszának lényeges része beleférjen.



19. ábra Az FXLMS algoritmus hatásvázlata

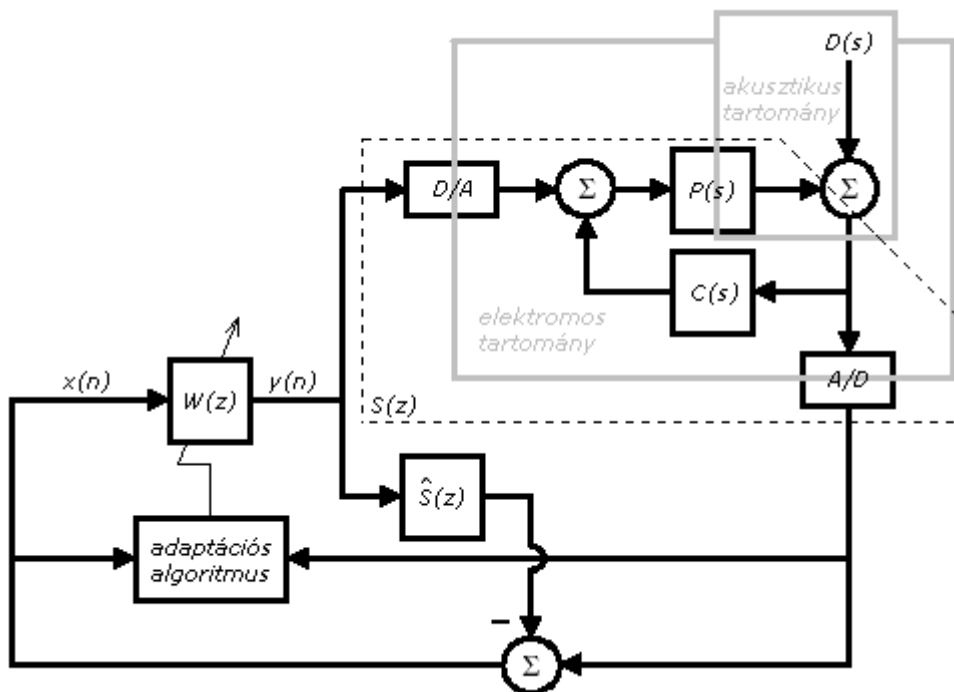
A μ maximális megengedhető értéke is módosításra szorul:

$$\mu_{\max} = \frac{1}{E\{\mathbf{x}'^T(n)\mathbf{x}'(n)\}(N + \Delta)}. \quad (32)$$

N a szűrő hosszát jelöli, Δ pedig az $S(z)$ -ben jelen lévő késleltetés mintaszámban megadva. A késleltetés, Δ paraméter megjelenése tehát lassítja az adaptációt.

1.3 A megvalósítandó analóg-digitális szabályozás ismertetése

Amint azt már a bevezetőben is említettem, a dolgozat célja egy olyan analóg-digitális hibrid szabályozás megtervezése, amely az analóg szabályozás nagy sebességének előnyeit igyekszik egyesíteni a digitális szabályozás rugalmasságából (adaptálhatóságából) eredő előnyeivel.



20. ábra A hibrid szabályozás hatásvázlata

Tulajdonképpen ez a hibrid rendszer egy olyan kétszintű szabályozás (20. ábra), ahol az alacsonyabb szinten található analóg szabályozásra épül rá a digitális szabályozás.

Az ábrán használt jelölések ugyanazok, mint az eddigiekben, így felismerhetők az előző fejezetekben ismertetett alapvető építőelemek. A szürkével bekeretezett tartomány folytonos idejű rendszert, az 1.1 alfejezetben ismertetett analóg szabályozást foglalja magába. Az ott ismertetett struktúra csak egy összegző csomópontot tartalmazott: ez reprezentálta az akusztikus szuperpozíciót. Ezen az ábrán ezt a jobb oldali összegző csomópont jelöli. A bal oldali összegző elektromos összegzést takar, amely lehetővé teszi, hogy a hangszóróra a visszacsatolt jelen kívül a digitális szabályozó által előállított vezérlőjel is kikerülhessen. Végül is a digitális algoritmus szempontjából az analóg szabályozási hurok, valamint az A/D és D/A átalakítók egy lineáris rendszert alkotnak, ezt a szaggatott vonallal bekeretezett $S(z)$ diszkrét átviteli függvény jelöli.

Az alkalmazott digitális szabályozó algoritmus az 1.2.2 alfejezetben bemutatott visszacsatolt szabályozási struktúra. A választás azért erre a típusra esett, mert az előrecsatolt szabályozás alkalmazása fejhallgató esetében általános (minden irányból érkező) zajt feltételezve nem adna jobb eredményt, mint a visszacsatolt szabályozás, ugyanakkor a visszacsatolt szabályozás kevesebb mikrofont igényel, mint az előrecsatolt (mivel nincs szükség referenciamikrofon(ok)ra).

2 Az analóg szabályozó megtervezése

Ahhoz, hogy a visszacsatolt analóg szabályozót, $C(s)$ -t meg tudjuk tervezni, szükség van a szabályozott szakasz, $P(s)$ átvitelének, amplitúdó- és fáziskarakterisztikájának ismeretére. A szükséges és elégséges modell felállítását két úton is megkezdhetjük. Az első a mérés, a második a bennünk élő fizikai kép az adott rendszerről. Ez utóbbi segít értelmezni a mérés eredményeit, és segít a modell egyszerűsítésében.

A szabályozott szakasz kimérése után következik a szabályozó megtervezése, amely a 1.1.2 alfejezetben ismertetett sávszűrő paramétereinek beállítását jelenti.

2.1 A szabályozott szakasz modellje

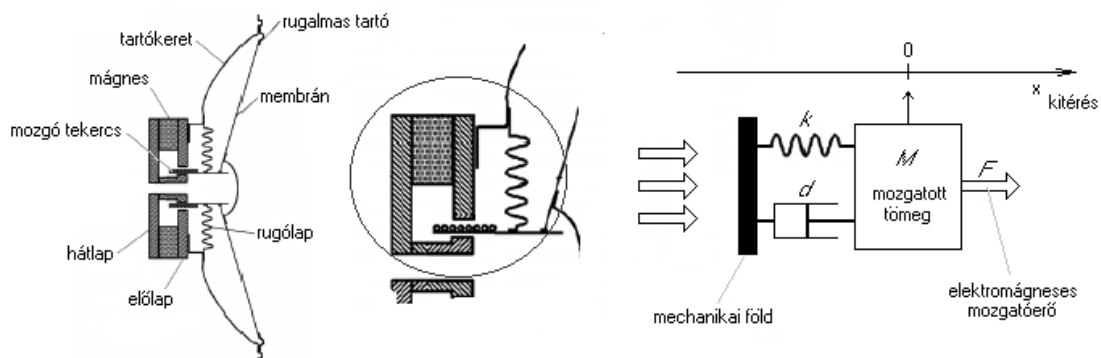
A $P(s)$ folyamat alapvetően öt fő egységre bontható szét:

- a hangszóró elektrodinamikája
- a hangszórómembrán dinamikája
- a membrán sugárzási tulajdonságai
- az akusztikus hullámterjedés
- a mikrofon akusztikus-elektromos átvitele

A következő alfejezetek ezeket az alegységeket tárgyalják.

2.1.1 A mozgótekercses hangszóró

A zajcsökkentés területén széles körben alkalmazott mozgótekercses hangszóró felépítése a 21. ábrán látható.



21. ábra A mozgótekercses hangszóró felépítése és koncentrált paraméterű mechanikai modellje [9]

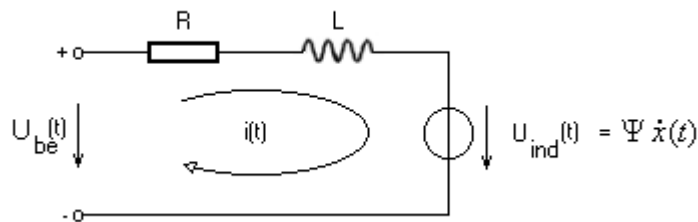
Az ábra baloldali része a hangszóró egy metszetét, a jobboldali rész pedig a hangszóró kisfrekvenciás mechanikai modelljét ábrázolja (rugalmasan rögzített, sebességgel arányosan csillapított tömeg, amelyet a hangszóró tekercsén átfolyó áram által létrehozott elektromágneses erő mozgat).

Newton mozgástörvényét alkalmazva a következő egyenletet kapjuk:

$$M\ddot{x}(t) + D\dot{x}(t) + Kx(t) = F_{EM} \equiv \Psi i(t), \quad (33)$$

ahol Ψ konstans mágneses erőtényező.

Ezek után vizsgáljuk meg az elektromos oldal felől is rendszerünk viselkedését:



22. ábra A hangszóró elektromos részének modellje

R jelöli a tekercs ohmos ellenállását, L a tekercs inductívitasát, U_{ind} pedig a mozgási indukció által létrehozott feszültséget. Láthatjuk, hogy az elektromos és a mechanikus rész között a Ψ erőtényező teremt kapcsolatot. Írjuk fel tehát Kirchhoff feszültségtörvényét:

$$u_{be}(t) = Ri(t) + L \frac{di(t)}{dt} + \Psi \dot{x}(t). \quad (34)$$

A fenti két egyenletből Laplace-transzformáció után kifejezhető a bemenő feszültség és a membrán sebessége közötti átviteli függvény:

$$\frac{V(s)}{U_{be}(s)} = \frac{s\Psi}{s^3(ML) + s^2(MR + DL) + s(RD + KL + \Psi^2) + (KR)} \quad (35)$$

Az R, D és K paraméterek általában az 1..100 értékek között találhatóak (SI rendszerben), míg az L csupán a 10^{-6} .. 10^{-5} -es intervallumban található, valamint a mozgatott tömeg (M) is csak néhány gramm. Ebből az következik, hogy a fenti átviteli függvény nevezőjében a köbös tag együtthatója nagyságrendekkel kisebb, mint az összes többi tag, így a legnagyobb pólus nagyságrendekkel nagyobb frekvenciákon helyezkedik el, mint a másik két pólus. Ha figyelembe vesszük, hogy a kisebb pólusfrekvenciák a néhány 100..néhány 1000 Hz-es intervallumba esnek, akkor a legnagyobb pólusfrekvencia legalább néhányszor 10 kHz-re adódik, amely nem érdekes

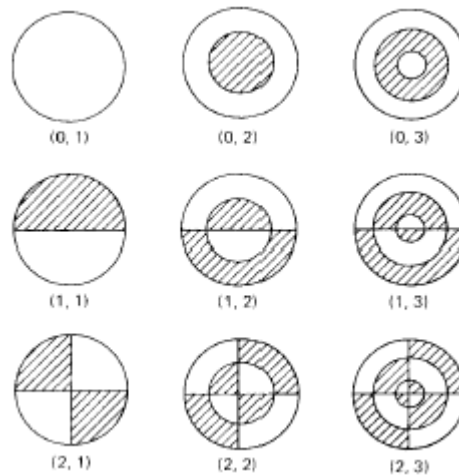
alkalmazásunk szempontjából, mert a számunkra érdekes néhány 100 Hz-ig tartó frekvenciatartományt sem amplitúdóban, sem fázisban nem befolyásolja érdemben. Így a továbbiakban a

$$\frac{V(s)}{U_{be}(s)} \approx \frac{s\Psi}{s^2(MR + DL) + s(RD + KL + \Psi^2) + (KR)} \quad (36)$$

közelítéssel élhetünk.

Ahogy azt már említettük, az előző, koncentrált paraméterekkel felépített modell nem használható csak kis frekvenciákon, ahol a hangszóró membránját merev testként kezelhetjük (akkor, ha a membránban a mechanikai hullám sebessége elég nagy ahhoz, hogy a membrán mentén a fázist állandónak tekinthessük).

Mivel a mérések során bebizonyosodott, hogy a kisfrekvenciás modell nem ad kielégítő magyarázatot a hangszóró viselkedésére már a számunkra érdekes frekvenciatartományban sem, így mindenképpen érdemes legalább kvalitatív képet alkotnunk a hangszóró membránjának mozgásáról. A következő ábrán a hangszóró membránján kialakuló első néhány módusképet figyelhetjük meg (a satírozott rész ellentétes irányú mozgást végez a be nem satírozott részhez képest). Az első sor a koncentrikus módusokat, míg az első oszlop a sugárirányú módusokat ábrázolja, a többi ezek lineáris kombinációjaként áll elő.



23. ábra A hangszóró membránján kialakuló módusképek [10]

Számunkra ez azért érdekes, mert az ellentétesen satírozott részek ellenkező irányú mozgása miatt csökkenni fog a kibocsátott hangnyomás szintje, tehát az ideális merev membránhoz képest csökken a hangszóró hatásfoka. A hatásfok csökkenése önmagában

még nem jelentene problémát, viszont a leszívások miatt a fázistolás is megnövekszik, ami csökkenti a stabilitási tartalékokat.

2.1.2 Az átviteli közeg

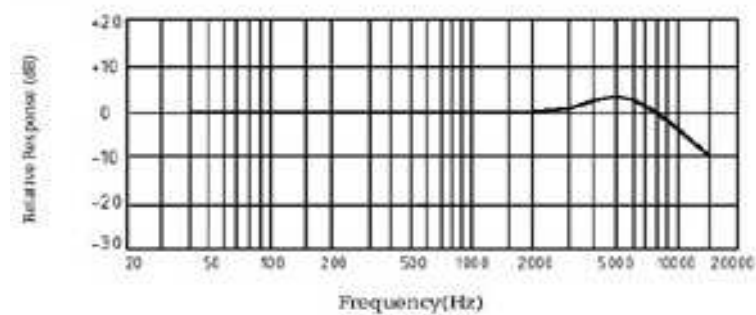
Az átviteli közeg a hangszóró membránja és a mikrofon között a levegő. Ebben a közegben a hang terjedési sebessége kb. 340 m/s, amely – mint azt korábban láttuk - számottevő holtidőt hoz be a szabályozási körbe, s ez a késleltetés lerontja a használható sáv szélességet és rontja a stabilitási tartalékokat.

A késleltetésen kívül azonban más jelenségek is fellépnek, amelyeket szintén figyelembe kell vennünk. Mivel már néhány kHz-es frekvencián a hang hullámhossza összemérhetővé válik a rendszer alkotóelemeinek méreteivel, így az olyan hullámterjedési jelenség, mint a hullámok visszaverődése, fontos szerepet játszik a szabályozott szakasz időbeli állandósága szempontjából. A hullámvisszaverődés miatt ugyanis a mikrofon membránjánál egyes frekvenciákon a nyomásszint megnő, más frekvenciákon lecsökken annak megfelelően, hogy az elsődleges és a visszavert hullámok milyen fázisban találkoznak a mikrofon membránjánál. Ezek a frekvenciák pedig attól függenek, hogy milyen a visszaverő közeg anyaga és hogy milyen az aktuális geometriai elrendezés. Így már láthatjuk, hogy a hullámjelenségek tényleg befolyásolják az analóg szabályozó megtervezését, mivel ezek a paraméterek egyáltalán nem tekinthetők állandónak egy fejhallgató esetében. A visszaverő anyagi közeg ugyanis az ember feje és füle, amelynek anyagi tulajdonságai személyenként eltérnek. Még nagyobb a paraméterváltozékonyság a geometriát illetően: itt egyrészt a viselő személyek fülének alakja nagy változékonyságot mutat, másrészt ha egyetlen személyt vizsgálnánk csak, akkor sem tudnánk a geometriai elrendezést állandósítani a viselő személy mozgása miatt (pl. a fej elfordítása, a fejhallgató igazgatása, stb. miatt). Ha ezeket a paraméterváltozásokat a szabályozó tervezése során figyelembe vesszük, akkor a kiadódó szűrőnek ezen paraméterek változása mellett is stabilan kell működnie. Ennek érdekében sajnos meg kell növelnünk a stabilitási tartalékokat, amely csökkenti a zajelnyomás mértékét.

2.1.3 A mikrofon

Mikrofonként aktív zajcsökkentő alkalmazásokban általában kondenzátormikrofonokat alkalmaznak, mivel ezek alacsony áruk ellenére elég jó minőségűek. Például a 24. ábrán

látható tipikus kondenzátormikrofon-amplitúdókarakterisztika alapján elmondható, hogy a mikrofon aluláteresztő jellegű, s kb. 2 kHz –ig egyenletes átvitelű, 5 kHz-en némi kiemelés látható, majd e fölött a karakterisztika -20 dB/dekádos meredekséggel csökken.

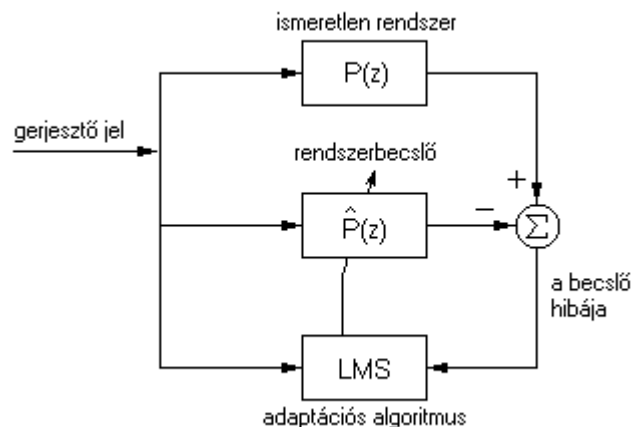


24. ábra Tipikus kondenzátormikrofon karakterisztika

Mivel kondenzátormikrofonok általában aszimmetrikus táplálást igényelnek, ezért a mikrofon kimenetének DC-leválasztása miatt felüláteresztő szűrést kell alkalmazni (további részletek a mikrofon előerősítőről a függelékben találhatóak).

2.2 A szabályozott szakasz identifikációja

Az elvégzett mérés elve a széles körben alkalmazott rendszeridentifikációs séma, amelyet a 25. ábrán láthatunk. A rendszer becslőjét a LMS algoritmus segítségével kerestem meg.

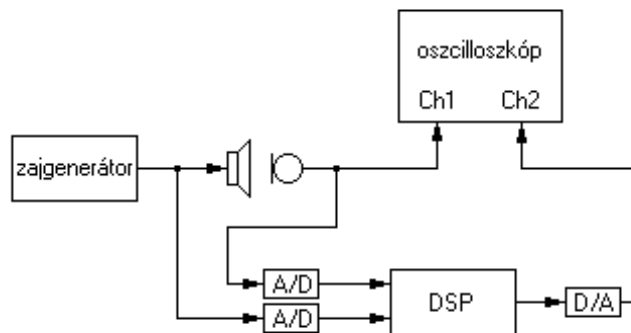


25. ábra Rendszeridentifikáció LMS algoritmussal

A kereső algoritmus addig változtatja a rendszerbecslő paramétereit (a FIR szűrő együtthatóit), míg az ismeretlen rendszer és annak becslője közötti hiba elfogadhatóvá

válí. A rendszert és becslőjét gerjesztő jel általában fehérzaj, amelynek teljesítménysűrűség spektruma egyenletes a vizsgált frekvenciatartományban.

Technikailag az identifikációt meg lehet valósítani online ill. offline módon is. Az offline identifikáció azt jelenti, hogy valamilyen digitális felvevő eszközzel felvesszük egyidőben az ismeretlen rendszer gerjesztő- valamint kimenőjelét, majd a felvett anyagon pl. Matlab környezetben elvégezzük az identifikációt. A másik, az általam is megvalósított online módszernél közvetlenül valósítjuk meg az identifikációs algoritmust pl. egy DSP-n: a DSP beolvassa a gerjesztőjelet, valamint az ismeretlen rendszer kimenő jelét, és ezekből kiszámítja a rendszer becslőjét. A következő ábra a mérési elrendezést mutatja.

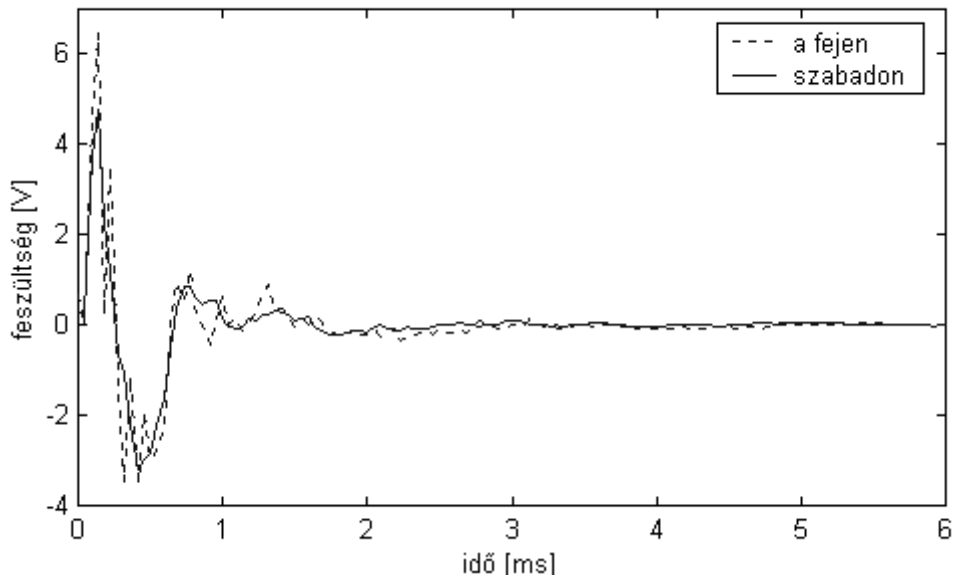


26. ábra Az identifikáció mérési összeállítása

A zajgenerátor hajtja meg a hangszórót, valamint a DSP egyik csatornáját. A mikrofon kimenete (amely a mérni kívánt rendszer kimenete is egyben) a DSP másik bemenő csatornájára kerül. A DSP-n fut a rendszeridentifikációs algoritmus, ez állítja elő a becsló hibáját, amely digitális-analóg átalakítás után az oszcilloszkópon figyelhető meg. Mivel mind a referenciajel, mind az analóg szabályozott szakasz kimenete ugyanolyan A/D átalakítón keresztül jut el az algoritmushoz, ezért ezek lineáris torzító hatása nincsen befolyással az identifikációra, úgymond kiejtik egymást.

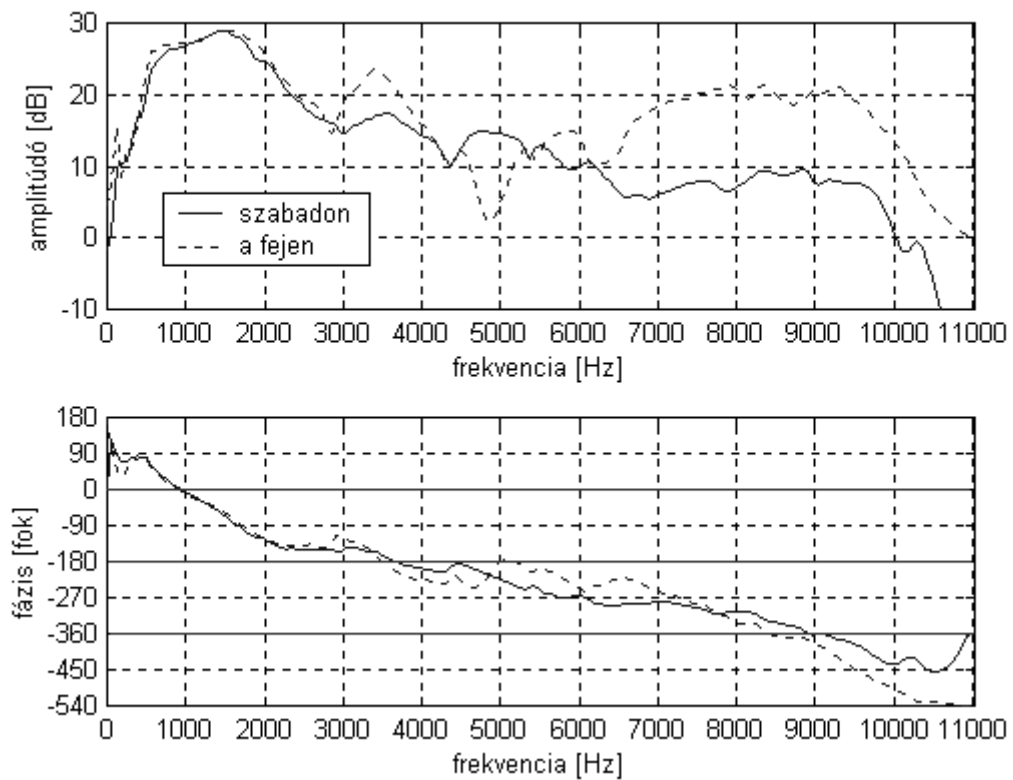
Az adaptációs programot addig kell futtatni, míg a hibajel el nem éri a már elfogadható szintet. Ez esetünkben kb. fél percnyi futtatást jelentett. A leállítást után az adaptív FIR szűrő tartalmát, azaz az analóg rendszer impulzusválaszának becslőjét a fejlesztőrendszer segítségével nyertem ki a processzor memóriájából. Méréseimet egy Sony CD250 típusú fejhallgató hangszórójára végeztem el.

A Sony fejhallgató impulzusválasza tehát a következő:



27. ábra Az szabályozott szakasz impulzusválaszának becslője

Az impulzusválasz Fourier-transzformációjából kapjuk meg a rendszer karakterisztikáit:

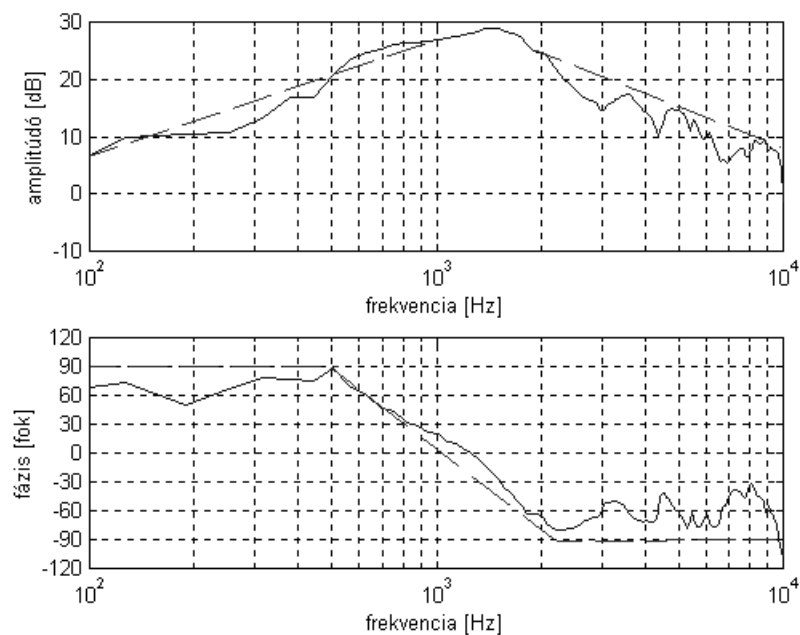


28. ábra A szabályozott szakasz karakterisztikái

Az ábra két mérés eredményeit tartalmazza: a folytonos vonallal jelzett karakterisztikák mutatják azt az esetet, amikor a fejhallgató szabadon van, a szaggatott vonallal jelöltek pedig azt, amikor a fejhallgató a felhasználó fején van. Láthatjuk, hogy a fázis- ill. az amplitúdó-karakterisztikák kb. 2 kHz-ig szinte teljesen azonosak, ezen túl viszont jelentős eltérés figyelhető meg a két eset között. Az eltérések az előző fejezetben említett hullámviszaverődéssel egészen jól magyarázhatóak, mivel a szaggatott vonal hol a folytonos vonal fölött, hol pedig az alatt halad annak megfelelően, hogy az elsődleges, ill. a visszavert hullámok építik, vagy éppen rombolják egymást. A 6 kHz-en túli „rendellenes” viselkedésre azonban ez a jelenség nem ad választ, lehet hogy a válasz az emberi fül tulajdonságaiban keresendő.

A különbségek mellett azonban közös jellemzők is felfedezhetők a két különböző esetben, ugyanis 3, 4.5 és 5.5 kHz környékén mindkét amplitúdó-karakterisztikában leszívásokat találunk, ami arra utal, hogy itt nem a hullámviszaverődésről van szó, hanem –feltételezésem szerint- a membránmódusok megjelenése miatti sugárzási hatékonyság csökkenéséről.

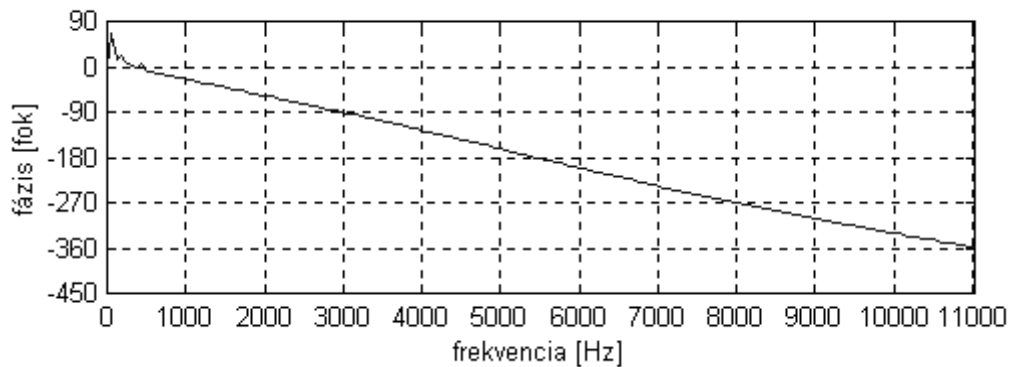
Ha a szabadon lévő fejhallgató karakterisztikájának minimálfázisú részét megvizsgáljuk (29. ábra), akkor azt vehetjük észre, hogy a membránmódusok figyelmen kívül hagyása esetén az előző fejezetben megállapított átviteli függvény elég jó közelítésnek tűnik mind az amplitúdó-, mind a fáziskarakterisztikát figyelembe véve.



29. ábra A szabályozott szakasz minimálfázisú részének összevetése a fizikai modellel

A feltételezett aszimptotikus diagramokat az ábrán szaggatott vonallal jelöltem. Az amplitúdókarakterisztikán közepén kihagytam egy részt, mivel itt valószínűleg rezonáns póluspár helyezkedik el, amely némi kiemelést okoz 1.5 kHz környékén.

Ha a rendszerünk minimálfázisú lenne, akkor a fenti ábra szerint könnyen megoldhatnánk az analóg szabályozó tervezését, mivel a fázis $\pm 90^\circ$ -on belül tartózkodik (legalábbis a vizsgált frekvenciatartományban), így elvileg egyszerű arányos taggal vissza lehetne csatolni a mikrofon jelét, a stabilitással nem is kellene foglalkoznunk, sőt erősítés sem lépne fel egyetlen frekvencián se. Azonban ha rátekintünk a vizsgált rendszer mindentátesztő részének fázisára (30. ábra), akkor az előző fejezeteknek megfelelően láthatjuk, hogy a helyzet nem ennyire egyszerű.



30. ábra A szabályozott szakasz mindentátesztő részének fázisa

Az ábra közel lineáris fázismenetet mutat, amely frekvenciafüggetlen késleltetés jelenlétére utal. A csoportkésleltetés a

$$\tau(\omega) = -\frac{d\varphi(\omega)}{d\omega} \quad (37)$$

képletből számolható ki, ez esetben

$$\tau = -\frac{\Delta\varphi}{2\pi\Delta f} = \frac{2\pi}{2\pi * 11000} s = 0.91 ms, \quad (38)$$

az ennek megfelelő térbeli távolság pedig

$$l = c\tau = 340 \frac{m}{s} * 0.91 ms = 3.1 cm. \quad (39)$$

Ez a távolság nem a hangszórómembrán közepének és a mikrofonnak a távolsága, hanem csak egy „eredő” távolság, amely nagyobb, mint a tényleges távolság. Ez azért van így, mert a membrán véges kiterjedéssel bír, így annak minden pontja sugároz a

mikrofon felé, s ezáltal nő a késleltetés, mivel a mikrofontól távolabbi pontokból később érkezik meg a nyomáshullám.

2.3 A megvalósítás elve

Az 29. ábránál megjegyeztem, hogy tetszőleges erősítést alkalmazhatnánk, ha nem lenne jelen késleltetés a rendszerben. Ekkor a legnagyobb elnyomást abban a frekvenciatartományban érnék el, ahol a legnagyobb a rendszer átvitele, ez esetben kb. 1.6 kHz környékén. Ha az amplitúdókarakterisztikát ki tudnánk úgy egyenesíteni, hogy az vízszintes legyen, akkor minden frekvencián egyforma elnyomást lehetne elérni. Ez a valóságban azonban nem vihető végbe több okból sem. Egyrészt láttuk, hogy a magasabb frekvenciatartományban változékonny a szabályozott szakasz átvitele, így hiába terveznénk egy adott átvitelhez kisimító hálózatot: amint megváltozik a szabályozott szakasz átvitele, megváltozik a kör átvitele is, s a kisimító hálózat típusától függően akár instabillá is válhat a zárt szabályozási hurok.

A másik oldalon, a kisméretű tartományban, a hangszóró kivezélhetősége állít elénk korlátot, ugyanis a hangszóró membránja nem vezérelhető ki csak egy bizonyos távolsáig: ameddig a rögzítőelemek meg tudják tartani. Persze a dolog ennél sokkal finomabb, ugyanis az sem engedhető meg, hogy a membrán kikerüljön az ún. lineáris működési tartományból, mert ha ez mégis megtörténik, akkor a nemlineáris torzítás miatt olyan komponensek is megjelennek a kibocsátott hang spektrumában, amely nincs is jelen a hangszórót vezérlő jelben. Ez pedig a hangszóró kellemetlen recsegését okozza.

Ha visszatekintünk a (36) képletre, akkor ebből a hangszóró bemenő feszültsége és a membrán kitérése közötti átvitel a Laplace tartományban $1/s$ -sel való szorzással kapható meg:

$$\frac{X(s)}{U_{be}(s)} = \frac{1}{s} \frac{V(s)}{U_{be}(s)} = A_0 \frac{1}{1 + \frac{s}{\omega_p Q_p} + \frac{s^2}{\omega_p^2}}. \quad (40)$$

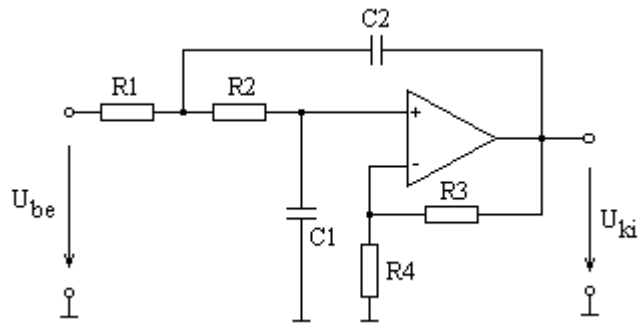
Az átvitel tehát másodfokú aluláteresztő jelleget ölt. Ebből viszont az következik, hogy ha a sebesség karakterisztikáját ki szeretnénk simítani, akkor azt csak úgy tehetnénk meg, hogy a kisebb frekvenciák felé haladva egyre növelnénk a hangszóró membránjának kitérését. Ezt nyilván csak valamilyen nullánál nagyobb frekvenciáig tehetjük meg, tehát töréspont elhelyezése válik szükségessé a kiegyenlítő hálózat amplitúdókarakterisztikájában.

A késleltetést figyelembe véve is hasonlóképpen járhatnánk el: kiegyenlítő hálózatot terveznénk abban a frekvenciatartományban, ahol ez lehetséges (így kb. egy egyszerű holtidős tagot kapnánk), majd például az (5) képletnek megfelelő hálózatot alkalmaznánk egy adott frekvenciasáv kiemelésére az ott történő zajelnyomás érdekében.

Ha az (5) és a (38) képletre tekintünk, akkor azt látjuk, hogy jellegüket tekintve ezek azonosak, így mivel a szabályozott szakasz és a megvalósítandó hurok jellege megegyezik, adott az ötlet, hogy egyszerű arányos visszacsatolással hozzuk létre a szabályozást. A 30. ábra szerint a holtidős tag 5.5 kHz-nél éri el a 180°-os fázistolást, míg a átvitel sávszűrő részének maximuma 1.6 kHz környékén van, amely az előbbi frekvenciához viszonyítva a 0.29-es relatív frekvencián helyezkedik el. A 10. ábrával összevetve ez azt jelenti, hogy jelentősebb zajelnyomás igénye esetén nincs elég stabilitási tartalék a hurokban, valamint, hogy jelentős zajerősítés lépne fel egy ilyen szabályozási hurok esetén. Elvileg (és valószínűleg gyakorlatilag is) megoldható lenne, hogy a hangszóró elektromos és mechanikai részének gondos megtervezésével a sávszűrő sávközépi frekvenciáját egy kisebb relatív frekvenciára helyezzük át, ahol az arányos visszacsatoló tag már minden gond nélkül alkalmazható, és ezáltal minimálisra csökken az alkalmazandó elektronika költsége. Ehelyett némi többletelektronikával is megoldható a kívánt sávszűrő létrehozása, s én ezt az utat követtem.

A 11. ábra szerint ha kb. 15-20 dB elnyomást szeretnénk elérni, akkor a sávszűrő relatív frekvenciájának kb. 0.1-nél kisebbnek kell lennie, hogy a magasabb frekvenciákon kicsi legyen a zajerősítés mértéke, így elhatároztam, hogy a sávszűrőt 300 Hz-es sávközépi frekvencián valósítom meg. Persze választhattam volna ennél kisebbet, vagy akár valamivel nagyobbat is, de ez a hibrid rendszer kipróbálása szempontjából nem volt lényeges szempont, így éltem a szabad választás jogával.

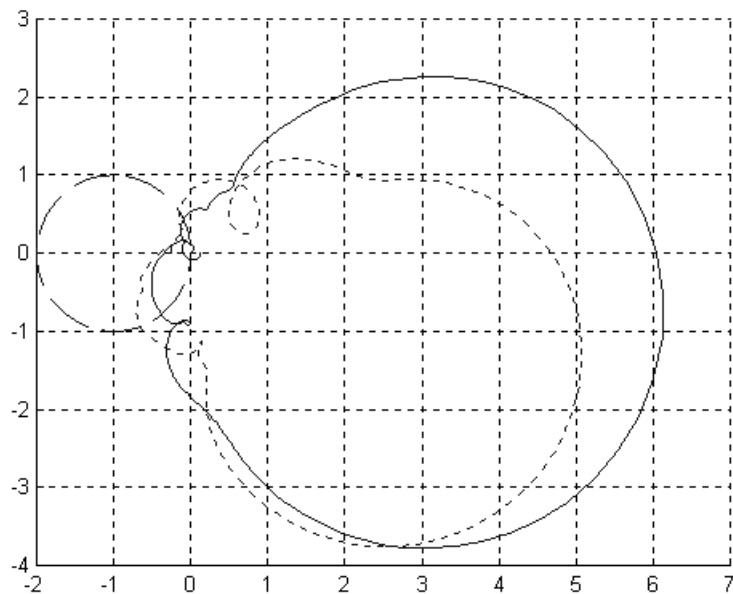
Mivel a szabályozott szakasz átvitelében kb. 600 Hz-ig a deriváló tag, azaz a számláló dominál, ezért a sávszűrő számlálóját ez a szakasz valósította meg, a nevezőt pedig egy műveleti erősítőtől, ellenállásokból és kondenzátorokból álló Sallen-Key másodfokú rezonáns aluláteresztő szűrőtag (31. ábra).



31. ábra Sallen-Key másodfokú aluláteresztő szűrő kapcsolási rajza [6]

2.4 A MATLAB szimuláció eredményei

Mielőtt összeraktam volna a szabályozási hurkot, célszerűnek tartottam szimulációt végezni annak megállapítása érdekében, hogy az előző alfejezet végén ismertetett szabályozó megfelelő stabilitási tartalékokkal rendelkezik-e, ill. hogy mekkora elnyomás érhető el a megcélzott frekvenciatartományban.

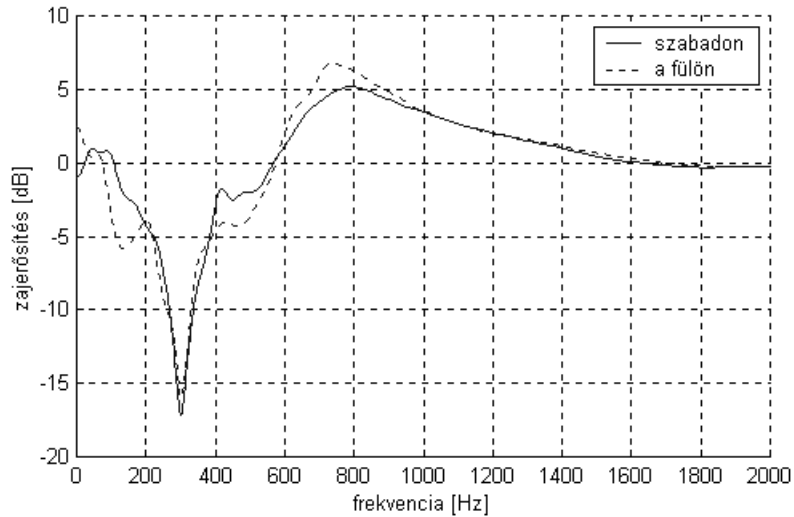


32. ábra A felnyitott kör Nyquist-diagramja (MATLAB-szimuláció)

Először is elhatároztam, hogy kb. 20 dB elnyomást szeretnék 300 Hz-en, s ennek megfelelően a szűrő jósági tényezőjét 10-nek vettem. Ezután megkerestem azt a körerősítést, amelynél a hurok Nyquist-görbéje megfelelő stabilitást mutatott (32. ábra), ez az erősítés 0.15-re adódott ki. Az ábrán jól látható, hogy a fejhallgató viselésekor a

stabilitási tartalékok leromlanak, így az erősítéstartalék kb. 6 dB, a fázistartalék kb. 55-75 fok annak megfelelően, hogy a fejhallgató a fejen van-e vagy sem.

Ezekkel a paraméterekkel a 33. ábrán látható elnyomás- ill. erősítésértékek adódtak.



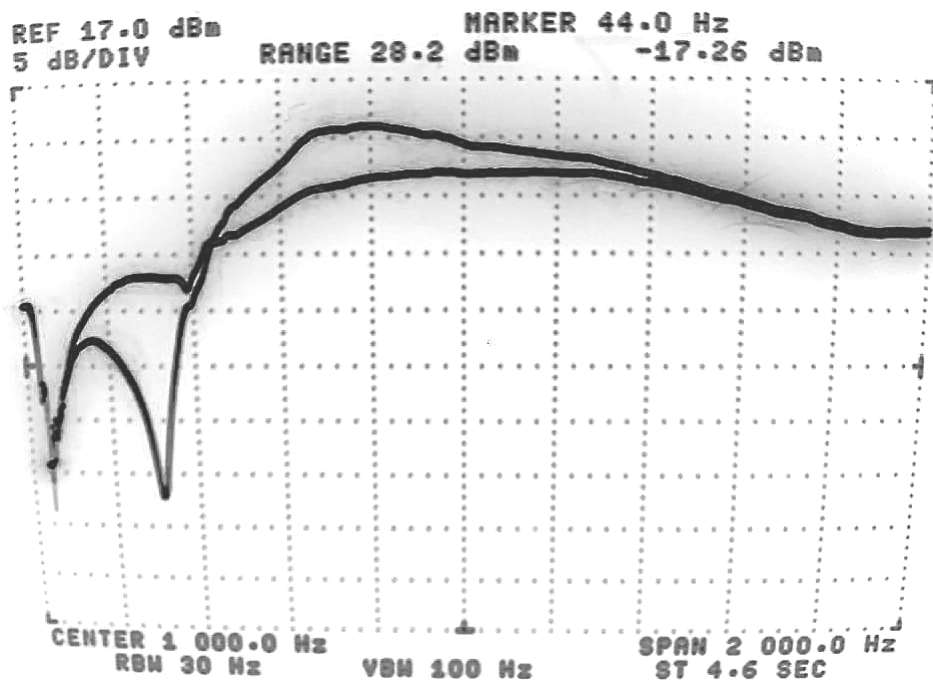
**33. ábra A zajerősítés mértéke a szabályozási hurok zárásakor
(MATLAB-szimuláció)**

A maximális elnyomás 300 Hz-en 17 dB körül van, amely viseléskor 1.5 dB-lel csökken, a maximális erősítés 7 dB körül van 700 Hz tájékán. Viseléskor az elnyomás sáv szélessége nagyobbá válik, ami kedvező jelenség, viszont csökken a maximális elnyomás, ami kedvezőtlen.

2.5 Mérési eredmények

Mivel a szimulációs eredmények biztatóak voltak, ezért a másodfokú aluláteresztő szűrő megvalósítása, majd a szabályozási hurok zárása után elkezdtem tesztelni az áramkört. A 20. ábrán ismertetett hibrid szabályozásnál a digitális rendszer egy összegző áramkör segítségével hajtja meg a hangszórót, ezért ezt is elkészítettem. Az összegző egyik bemenetére a visszacsatolt jel került, a másikra pedig a zárt rendszert meghajtó jel.

Első mérésem arra kívánt választ adni, hogy mi a különbség a nyitott és a zárt rendszer átvitele között. Ennek érdekében a spektrumanalizátor sweepelő kimenetét rákötöttem az analóg szabályozott kör bemenetére, a mikrofon jelét pedig rákötöttem az analízátor bemenetére. A mérést nyitott és zárt körre is elvégeztem (34. ábra).



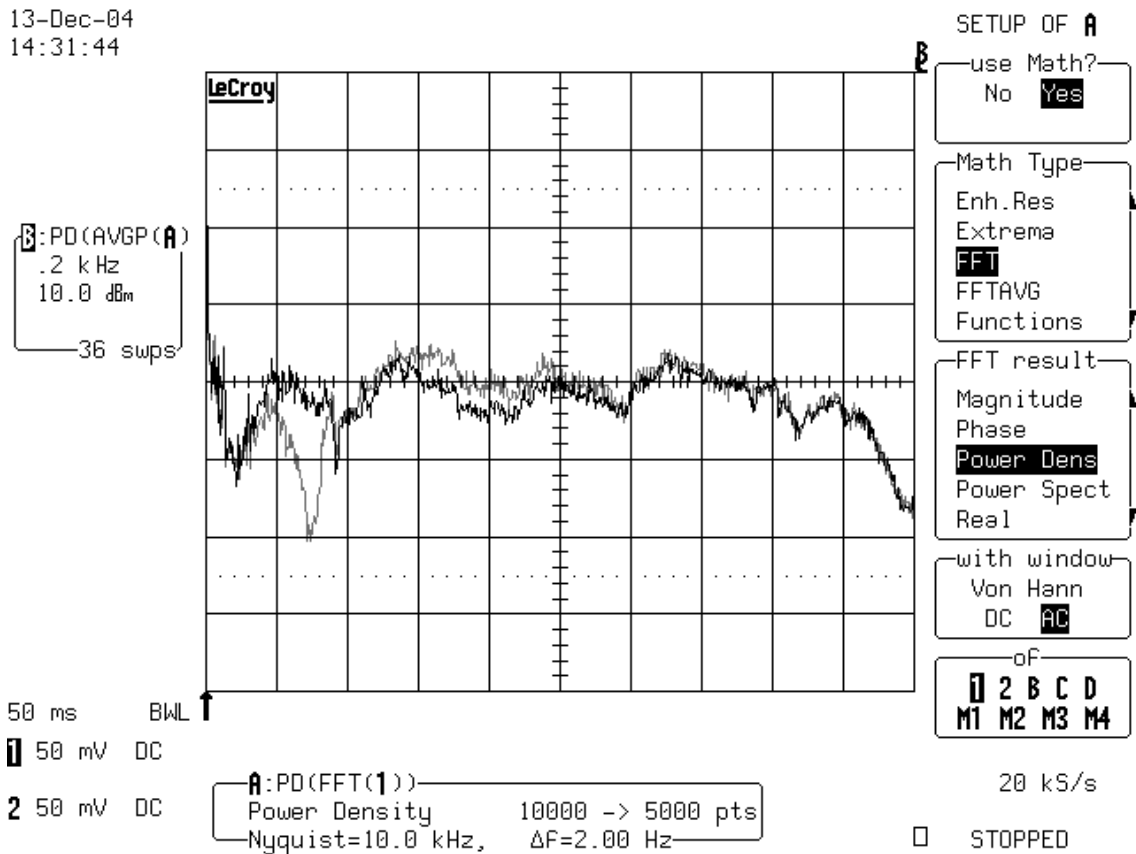
34. ábra A nyitott és a zárt rendszer átvitele sweepelő szinusszal mérve

A mérés tulajdonképpen azzal egyenértékű, mintha lenne egy másik hangszóró ugyanazon a helyen, és megmérnénk, hogy sweepelő szinuszt mennyire képes elnyomni a zajelnyomó rendszer. Ha az itt látható ábrát összevetjük a szimulációs eredményekkel, akkor láthatjuk, hogy az ott kiszámolt eredmények összhangban vannak a mérés során kapott eredményekkel.

Az előbbi mérés sweepelő szinusszal, tehát determinisztikus zajjal gerjesztette a rendszerünket, ráadásul a zaj és az ellenzaj forrása ugyanaz a hangszóró volt, így joggal merülhet fel a kérdés, hogy vajon valós környezetben is megfelelően működik-e a rendszer. Ezért elvégeztem még egy kísérletet.

A vezérlő bemenetet most leföldeltem, azaz a hangszórót csak a visszacsatolt jel gerjesztette. Külső zajforrásként egy hangdobozt alkalmaztam, amelyet fehérzajjal gerjesztettem. A mikrofon kimenetét rákötöttem egy digitális oszcilloszkóp bemenetére és a regisztrátumokon FFT-t hajtottam végre, amelyet azután átlagoltattam. A 35. ábra 36 FFT átlagát mutatja.

13-Dec-04
14:31:44



35. ábra A nyitott és zárt rendszer átvitele véletlenszerű zaj esetén

Láthatjuk, hogy az ábra nagyon hasonlít az előző mérés, valamint a szimuláció eredményeire, így az analóg elven működő zajelnyomó megfelelően működik véletlenszerű jelekre is.

Ha visszaemlékezünk a vízagy effektusra, akkor ott említettem, hogy ha az egész frekvenciatartományt vizsgáljuk, akkor az elnyomás és az erősítés összességében megegyezik, így szélessávú gerjesztés esetén a zaj effektív értéke nem változik. Ezt sikerült kísérletileg is igazolni: az előző kísérletben a mikrofon jelét RMS voltmérővel megmérve nem tapasztaltam különbséget a zajcsökkentő be- ill. kikapcsolt állapotában. Ezután ugyanezt a mérést megismételtem keskenyebb sávú zajra is, ahol a zaj számottevő teljesítménye a 0-400 Hz-es tartományba esett (ahol az elnyomó rendszer ténylegesen csökkenti a zajt), és ekkor már a voltmérő is mutatott elnyomást a zajcsökkentő kikapcsolt állapotához képest.

Végül a saját fülemlen is kipróbáltam a rendszert, s azt tapasztaltam, hogy a mély hangok jelentősen halkabbak lettek, míg a magasak némileg erősödtek. Az érzés olyan, mint amikor egy hangszínszabályozónál letekerik a basszust és föltekerik a közép frekvenciasávot.

3 A digitális rendszer megtervezése

Ebben a fejezetben bemutatom a digitális visszacsatolt adaptív szabályozó megtervezésének főbb lépéseit. Először ismertetem az ADSP 21061 EZKIT LITE fejlesztőkártyának a dolgozat szempontjából releváns alkotóelemeit, illetve ezeknek az elemeknek számunkra érdekes paramétereit. Az először elkészült digitális szabályozó az A/D–D/A átalakítók numerikus telítése miatt nagy környezeti zajszint esetén rendszerint kikerült a stabil működési tartományból, ennek megoldását taglalja egy külön alfejezet. Mint láttuk azt az adaptív prediktorról szóló alfejezetben, a késleltetések alapvetően befolyásolják a szélessávú zajelnyomás lehetőségét. Az általam megvalósított rendszerben ennek a késleltetésnek meghatározó hányada a szigma-delta átalakítók késleltetéséből származik. Mivel az először alkalmazott 8 kHz-es mintavételi frekvencián a rendszer alig mutatott szélessávú működést, ezért át kellett térnem a fejlesztőkártya által támogatott legnagyobb mintavételi frekvenciára, 48 kHz-re. Így a késleltetést némileg sikerült lecsökkenteni, de az eredményekből kitűnik, hogy jobb szélessávú működés eléréséhez sokkal gyorsabb átalakítókra van szükség.

3.1 A rendszer elemei

A megvalósított digitális szabályozó az Analog Devices cég által gyártott ADSP 21061 EZKIT LITE fejlesztőrendszer köré épült, ezért néhány szót érdemes ennek ismertetésére fordítani.

A fejlesztőkártya tulajdonképpen két fő alkotóelemből áll, az első a 21061 típusszámú DSP, a második pedig az AD1847 típusszámú sztereo audio codec IC.

A 21061-es DSP olyan 32 bites lebegőpontos aritmetikájú processzor, amelynek az architektúráját úgy optimalizálták, hogy a leggyakrabban használt digitális jelfeldolgozási algoritmusok (pl. szűrés, FFT) minél hatékonyabban megvalósíthatóak legyenek. Ennél a processzornál az ún. módosított Harvard-architektúra került megvalósításra, ahol a memória két részre van felosztva, az egyik rész a programmemória, a másik pedig az adatmemória. Míg az adatmemória csak adatokat tartalmaz, addig a programmemória a program kódja mellett adatokat is tartalmazhat. Ez a memóriakezelés lehetővé teszi, hogy egy utasítás alatt két adatszóhoz is hozzáférjünk: ez tipikusan a transzverzális szűrők végrehajtásának gyorsaságát növeli.

A másik optimalizálás az ún. MAC-jellegű (Multiply and ACcumulate) utasítás, amely lehetővé teszi a szorzatintegrálás egy utasításciklusban történő végrehajtását: míg az előző szorzat eredményét hozzáadjuk az eredményhez, addig elkészíthető a következő szorzat is. Ennek is nagy hasznát vehetjük a szűrési algoritmusoknál, de más algoritmusok is gyorsíthatóak használatával.

Ezen kívül még sok olyan architektúrális módosítás van, amely növeli a számítási teljesítményt, ezekről az Olvasó a processzor gépkönyvéből [7] kaphat képet.

A fejlesztőkártya említett másik fontos építőeleme a kétcsatornás A/D-D/A átalakítást megvalósító codec IC. A bemenő feszültségjeleket egy-egy 16 bites szigma-delta analóg-digitál átalakító konvertálja át 16 bites kettes komplementes digitális kóddá, ezt a számkódot a DSP soros kommunikáció útján kérheti el a codectól. A DSP által a codec felé elküldött 16 bites számérték pedig mint feszültség jelenik meg a szintén szigma-delta elven működő digitális-analóg konverzió után.

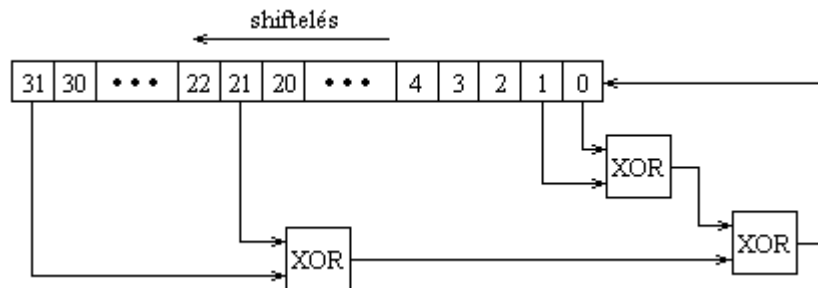
A codec számunkra legfontosabb paramétere a csoportkésleltetés, mivel az az adaptív prediktorról szóló résznek megfelelően jelentős mértékben befolyásolja az elérhető legnagyobb zajnyomás mértékét. Ennek az A/D-D/A párosnak együtt összesen 30 minta késleltetése van a szigma-delta konstrukcióhoz szükséges decimáló és interpoláló szűrők miatt. Mivel a specifikáció szerint ez a késleltetés a teljes frekvenciatartományban ugyanakkora, ezért ezek a szűrők biztosan nem minimálfázisúak. Ebből viszont az következik, hogy olyan többletfázissal rendelkezik, amely számunkra kedvezőtlen, mivel növeli a késleltetést. Ezen némiképpen lehet javítani magasabb mintavételi frekvenciára való áttéréssel, erről a dolgozat későbbi részében még esik szó.

3.2 Az alkalmazott algoritmus

3.2.1 Az $\hat{S}(z)$ rendszerbecslő előállítása

Az 1.2.5 alfejezetben láttuk, hogy az FXLMS algoritmus megvalósításához szükség van az $S(z)$ átviteli függvény becslőjére, $\hat{S}(z)$ -re. $\hat{S}(z)$ tulajdonképpen a hangszóra kiadott számérték és a mikrofon feszültségének megfelelő számérték közötti átviteli függvény, ezért az identifikálást úgy végezhetjük el, hogy a processzoron belül előállítunk valamilyen gerjesztőjelet (tipikusan fehérzajt), és a mikrofonról beolvasott érték alapján meghatározzuk a rendszer átvitelét. Az identifikációs séma megegyezik az

analóg rendszer identifikálásánál elmondottakkal, a véletlenszámok generálását 32 bites visszacsatolt shiftregiszteres álvéletlenszám-generátorral valósítottam meg (36. ábra).



36. ábra 32 bites álvéletlenszám generálása

Az algoritmus egy 32 bites regiszter megfelelő bitjeinek képezi a KIZÁRÓ-VAGY kapcsolatát, melynek eredményét shifteléskor a legkisebb helyiértékre rakja be. A regiszterbe kezdeti értéként bármilyen számot meg lehet adni, kivéve a nullát. Az előállított véletlenszám-sorozat hossza:

$$L = 2^M - 1, \quad (41)$$

ahol M a bitek száma, jelen esetben 32. Az alkalmazott 8 kHz-es frekvencia mellett ez kb. egy hetes periódusidőt jelent.

3.2.2 A telítések figyelembevétele

Mivel az A/D és D/A korlátos értéktartománnyal rendelkezik, ezért a rendszer tervezésekor ezt is figyelembe kellett venni.

A rendszerben az A/D-átalakító a mikrofon feszültségét konvertálja digitális számértékké, így az A/D telítődésének elkerülése könnyen megoldható a mikrofon előerősítőjének megfelelő beállításával. Ügyelni kell azonban arra, hogy ha túl kicsire vesszük az erősítést, akkor nem tudjuk kihasználni a 16 bites felbontást, ha pedig túl nagyra, akkor impulzusszerű gerjesztés (pl. a fejhallgató koccanása) esetén az A/D-t túlvezérelhetjük. Ezért érdemes a mikrofon jelére aluláteresztő szűrőt tenni, mivel ez csökkenti az impulzus energiáját, így a feszültségtüskék amplitúdóját is. A szűrőt a mikrofon előerősítőjében (Függelék) valósítottam meg, mivel az analóg szabályozás szempontjából is kedvező a feszültségtüskék simítása.

A D/A-átalakítónál a helyzet kicsit bonyolultabb, mivel itt az adaptív algoritmus könnyen telítésbe tudja vinni a D/A-átalakítót. Tegyük fel ugyanis, hogy a hangszóró erősítője úgy van beállítva, hogy kihasználhassuk a teljes lineáris működési

tartományát, s a linearitási határnak a D/A maximális kivezérelhetősége felel meg, tehát a D/A maximális kivezérlésénél még éppen nem kezd el a hangszóró torzítani. Az is világos, hogy a zajcsökkentő rendszer maximum akkora zajszintet képes kioltani, amekkorát a beavatkozó hangszóró képes magából kiadni. Tegyük fel azonban, hogy a mikrofonunk a maximálisan kioltható zajnál nagyobb szintű zajt detektál. Ekkor az adaptációs algoritmus igyekszik teljesíteni küldetését, tehát a mikrofon zajának minimalizálása érdekében egyre inkább kivezérlí a D/A átalakítót. Mivel a D/A maximális kivezérlésénél a zaj teljes kioltását még nem értük el, ezért az adaptációs algoritmus még nagyobb jelszintet próbál kiadni, ami a D/A telítését okozza. Ez pedig azt idézi elő, hogy a hangszóró a telített vezérlés következtében nemcsak olyan frekvenciakomponenseket sugároz, amelyek a zavarjel kioltását eredményeznék, hanem más komponenseket is.

Láttuk a digitális visszacsatolt szabályozással kapcsolatban, hogy a hibajelből (a mikrofon jeléből) akkor állítható elő pontosan a referenciajel, ha $S(z) = \hat{S}(z)$, azaz a becslés tökéletes. Ez nemcsak lineáris S -re igaz, hiszen bárhogyan is viselkedik a modellezendő rendszer, megfelelő méréseket elvégezve viselkedése kideríthető és valamilyen \hat{S} operátorral modellezhető. A D/A telítése nyilván könnyen modellezhető, s az \hat{S} -ban meg is kell valósítani a pontos referencia érdekében (és hogy ne jöjjön létre pozitív visszacsatolás).

Ez viszont azzal a következménnyel jár, hogy míg a hibajel tartalmazza a nemlinearitásból eredő frekvenciakomponenseket, addig a referenciajelben ezek a komponensek egyáltalán nincsenek jelen. Ez azt eredményezi, hogy az adaptív szűrő együtthatói beláthatatlan módon elkezdnek bolyongani a paramétersíkon, és általában nem a kedvező irányba tartanak, tehát divergálnak, lebegőpontos túlcsoportulást okoznak. A problémára többféle megoldás is létezik [8]. Az egyik megoldás az ún. leaky-LMS alkalmazása, ennek az LMS-variánsnak az együtthatók frissítése a

$$\mathbf{w}(n+1) = \beta \mathbf{w}(n) + \mu e(n) \mathbf{x}(n), \quad 0 < \beta \leq 1 \quad (42)$$

képlet szerint történik. Az algoritmus elnevezése onnan ered, hogy együtthatók mintegy „szivárognak” a $\beta < 1$ esetben, azaz az algoritmus minden időpillanatban a paraméterter origója felé mozdítja el a paramétervektort. Megfelelően kis β az együtthatók gyors szivárgása miatt megoldást jelenthet a D/A telítésének elkerülésére, viszont ez a megoldás nem ideális abból a szempontból, hogy a gyors felejtés miatt a minimális elérhető maradékhiba megnő, a zajcsökkentés kevésbé lesz hatásos.

A probléma megkerülhető olyan módon [8], hogy csak abban az esetben szivárogtatjuk az együtthatókat, amikor az feltétlenül szükséges. Ez az ún. újraszkalázási mechanizmus akkor lép életbe, ha

$$|y(n)| > A_{\max}, \quad (43)$$

vagyis amikor az adaptív szűrő kimenete ($y(n)$) telítésbe vinné az A/D-t. Ekkor

$$\beta = \frac{A_{\max}}{|y(n)|}, \quad (44)$$

egyébként pedig $\beta = 1$. Ez a megoldás tulajdonképpen azt a gondolatot fejezi ki, hogy abban az esetben, amikor a szűrő kimenete túl nagy, akkor úgy kell csökkenteni az együtthatókat, hogy a szűrő kimenete a következő ütemben változatlan bementi teljesítménnyel számolva is a telítési szint alatt legyen. Ha ez mégse történne meg, akkor a következő ütemben megint újra kell skálázni az adaptív szűrőt, stb.

3.3 Megvalósítás 8 kHz mintavételi frekvencián

A digitális szabályozó algoritmust először 8 kHz-es mintavételi frekvencia mellett valósítottam meg, az *anc8k.asm* program vázlata az eddig elhangzottakkal összhangban a következő:

```

reset:
    {általános inicializáció}
    {az AD1847 codec felprogramozása}
    jump mainprog;

mainprg:
{inicializáció}
{A/D - D/A megszakítások engedélyezése}

mainloop:
    jump mainloop; // végtelen ciklus

receive:          // ez fut le a codec megszakításoknál
    {x(n) -> x'(n)}
    {a hibamikrofon jelének beolvasása az A/D-ról}
    {a referenciajel x(n) előállítás}
    {mu(n) kiszámítása az NLMS-nek megfelelően}
    {x(n) -> y(n)}
    {y(n) alapján újraszkalázás, ha |y(n)|>1}
    {y(n) -> y'(n)}
    {NLMS szűrőadaptáció}
rts;              // visszatérés mainloopba

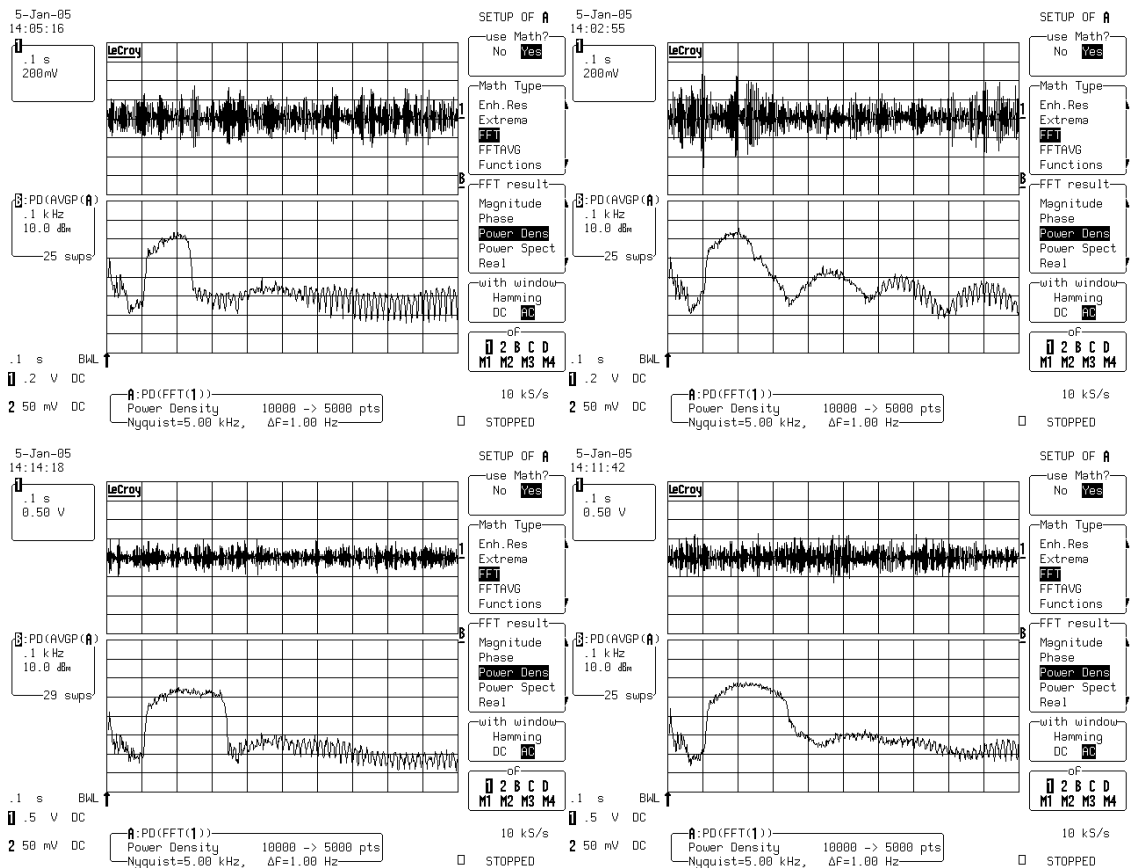
```

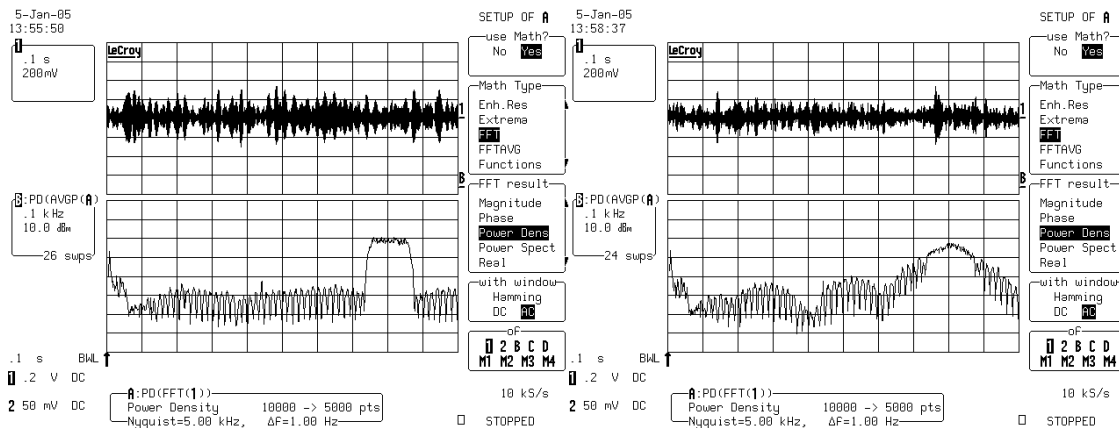
Az $\hat{S}(z)$ szűrő előállítását offline módon tehető meg az *ident8k.asm* program futtatásával.

3.3.1 Mérési eredmények

Mivel a 8 kHz-en megvalósított program nem a végleges verzió volt, ezért itt csak azok a mérési eredmények kerülnek közlésre, amelyek a véglegessel való összehasonlítás szempontjából érdekesek. Nevezetesen a szélessávú működés eredményeiről lesz szó.

Az első mérés úgy történt, hogy 4 kHz-nél nagyobb sávzélességű, ebben a tartományban egyenletes spektrális eloszlású zajjal hajtottam meg egy hangszórót, és spektrumanalizátorral figyeltem, hogy mi történik a zajelnyomó rendszer ki- ill. bekapcsolt állapotában. Mivel a zaj spektruma túl széles volt, ezért semmilyen lényeges különbséget nem találtam a két eset között. Ezután vizsgálódásomat keskenyebb sávú (kb. 100 Hz-es sávzélességű) zajokkal folytattam (a zajt úgy állítottam elő, hogy egy másik DSP-kártyán futtatott algoritmussal megsűrtem a szélessávú zajt). Három mérést végeztem: az első a 100-tól 250 Hz-ig terjedő intervallumban gerjesztette a rendszert, a második 100-350 Hz-ig, a harmadik pedig 750-850 Hz-ig. Az eredmények a következő ábrákon láthatók.





37. ábra Szélessávú elnyomás 8 kHz mintavételi frekvencia mellett

A baloldali oszlop képei a digitális zajcsökkentő algoritmus kikapcsolt állapotában, a jobboldali oszlop képei pedig annak bekapcsolt állapotában készültek.

Látható, hogy az $\hat{S}(z)$ nagy késleltetése (kb. 35 minta) miatt alig tapasztalható zajcsökkentés, sőt az első két kísérletben inkább zajerősítésről beszélhetünk, amely a zaj és az ellenzaj nagy mértékű korrelálatlanságára utal.

Mivel ezek az eredmények nem voltak megfelelőek számunkra, ezért finomítani kellett a megoldáson.

3.4 Áttérés 48 kHz mintavételi frekvenciára

Mivel a 8 kHz-en mintavételező rendszer a szigma-delta átalakítók nagy (30 minta) késleltetése miatt nem mutatott jelentősebb szélessávú elnyomást, ezért szükségessé vált ennek a késleltetésnek a redukciója. A csökkentés alapötlete az volt, hogy térjünk át a A/D-D/A átalakítók legnagyobb mintavételi frekvenciájára, 48 kHz-re, ahonnan egy kisebb frekvenciával működő zajelnyomó algoritmus decimálás után kapja meg a mintavett értékeket. Pl. a 8 kHz-en működő algoritmus 6-szoros decimálást jelentene, amely hatodára, azaz 5 mintára redukálná a 8 kHz-es algoritmus szempontjából az A/D-D/A késleltetését. Sajnos ilyen mértékű csökkenés nem érhető el, mivel a mintavételi frekvencia csökkentése miatt decimáló szűrő alkalmazása válik szükségessé, amely újabb késleltetést visz a rendszerbe. A másik oldalról a hangszórónak kiadandó mintákat 8 kHz-en állítjuk elő, de mivel az A/D és D/A egyaránt 48 kHz-en dolgozik, ezért interpoláló szűrőre is szükség van a hasonmások megjelenésének elkerülése érdekében. Ez a szűrő szintén késleltet valamennyit.

Ezek a problémák azonban némileg enyhíthetők. A szigma-delta átalakítókat általában úgy tervezik, hogy alakhú átvitelt tegyenek lehetővé, ez egyrészt lineáris fázist, másrészt egyenletes átvitelt követel meg mind az interpoláló, mind a decimáló szűrő esetében. Mivel a lineáris fázisú szűrők nem minimálfázisúak, ezért olyan fázistöbblettel rendelkeznek, amely rendszerünk szempontjából káros késleltetés-többletet hoz létre. A másik követelmény, az egyenletes átvitel szintén káros a szempontunkból, hiszen a Nyquist-frekvenciánál történő nagyon meredek letörés nagyon megnöveli még kis frekvenciákon is a fázist. A kellően kis mértékű késleltetés érdekében tehát ezeknek a szűrőknek a specifikációján kellene változtatni. Mivel ezek bele vannak égetve codec áramkörbe, ezért közvetlenül a szűrőcsere nem végezhető el. A közvetett módszer az, amiről már szó volt a fejezet elején, azaz decimálás, interpoláció. Enyhített specifikációjú decimáló és interpoláló szűrőkkel tehát a 8 kHz-es algoritmus gyorsabb A/D-D/A átalakítást lát.

A decimáló szűrő az átlapolódásmentes mintavételezés miatt szükséges, azaz hogy a mintavételi frekvencia felénél nagyobb frekvenciájú spektrumkomponenseket kiszűrje. Rendszerünkben ez háromféleképpen érhető el:

- 1.) még a 48 kHz-es mintavételezés előtt 4 kHz-nél kisebb töréspontú analóg aluláteresztő szűrővel,
- 2.) nincs analóg szűrő, hanem csak 4 kHz-nél kisebb töréspontú digitális aluláteresztő szűrő,
- 3.) semmilyen szűrőt nem alkalmazunk, hanem feltételezzük hogy a szűrő már eleve benne van valahol a rendszerben, tehát a káros spektrumkomponensek nincsenek jelen a digitalizálni kívánt jelben.

Esetünkben a 3.) megoldás nagyon kézenfekvőnek tűnik, hiszen a fejhallgató passzív zajcsökkentése révén a nagyfrekvenciás zaj komponensek nem jutnak el számottevő mértékben. Korábban már említettem, hogy a fejhallgató viselése közben a nem akusztikus kapcsolódás útján magasfrekvenciás komponensek is gerjeszthetik a mikrofont, ezért érdemes egy analóg aluláteresztőt beépíteni a rendszerbe. Ez a szűrő további segítséget nyújthat tehát a káros komponensek kiszűréséhez.

Az interpoláció azért szükséges, hogy a hasonmásokat, azaz az $n \cdot 8$ kHz ($n > 1$) körüli spektrumismétlődéseket megszüntesse. Ez alkalmazásunk szempontjából fontos kritérium, hiszen azt a zajt, amelyet ki szeretnénk oltani, nem szeretnénk magasabb frekvenciákon a fejhallgatóban meghallgatni. Az interpoláció is háromféleképpen valósítható meg:

- 1.) analóg szűrőt alkalmazunk a D/A kimenetén,
- 2.) digitális szűrőt alkalmazunk,
- 3.) vagy ha a rendszerben elfogadható, hogy a hasonmások jelen legyenek, akkor nem alkalmazunk interpolációt.

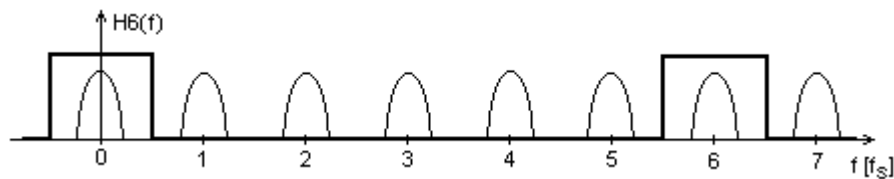
A 3. módszer sajnos ez esetben nem alkalmazható, mivel a hangszóró a 4 kHz-en túl is jól hallható hangokat ad ki.

Az analóg szűrő alkalmazása többletköltséget támaszt, mivel pontos és időtálló alkatrészek kellenek a jó minőségű szűrők létrehozásához.

A fentieket figyelembe véve úgy döntöttem, hogy az interpolációt digitális szűrővel valósítom meg.

3.4.1.1 Az interpoláló szűrő megtervezése

Az ideális 8 kHz-ről 48 kHz-re interpoláló szűrő amplitúdókarakterisztikája a 38. ábrán látható.



38. ábra Ideális hatszorozó interpoláló szűrő

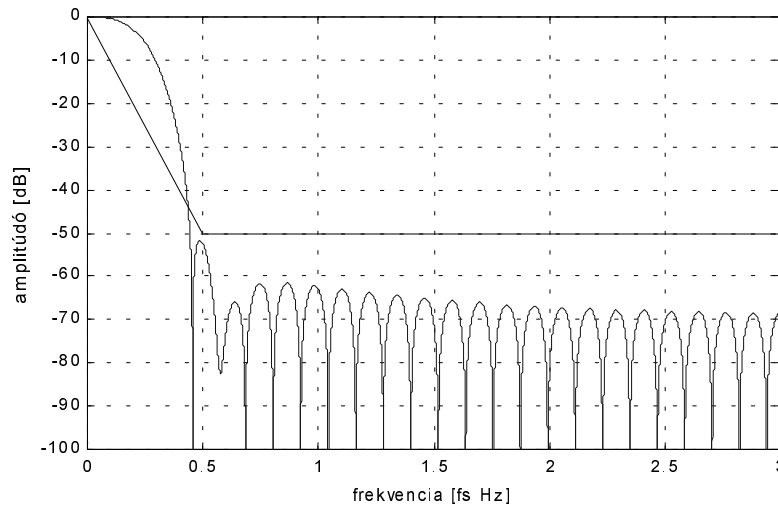
Láttuk azonban, hogy az ideális specifikáció alapján tervezett szűrő alkalmazásunk szempontjából káros, így az ideális szűrő helyett gyengített specifikációt kell megfogalmaznunk. A minimális fázis feltétel könnyen teljesíthető úgy, hogy a bármilyen módszerrel megtervezett szűrő impulzusválaszát minimálfázisú jellé alakítjuk a matlab *rceps* függvényével.

Az amplitúdómenet specifikációja úgy enyhíthető, hogy egyrészt kisebb törésponti frekvenciát írunk elő, mint a Nyquist-frekvencia, másrészt pedig enyhítünk a zárótartománybeli elnyomás mértékén.

A törésponti frekvencia meghatározásánál figyelembe vettem az analóg rész azon tulajdonságát, hogy a nagyobb frekvenciák felé haladva (kb. 2 kHz-en túl) a hangszóró-mikrofon elrendezés átvitele időben változhat, ami $\hat{S}(z)$ pontatlanságához, instabil működéshez vezethet. Az interpolációs szűrő töréspontját ezért 2 kHz-ben határoztam meg, hogy a bizonytalanságot elnyomjam (ennél kisebb frekvencia, azaz enyhébb

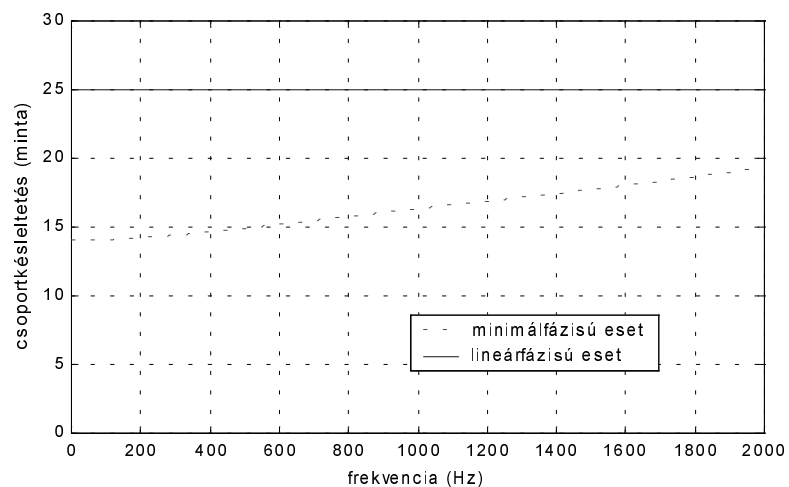
specifikáció is megadható, ekkor azonban le kell mondanunk a 2 kHz-ig való kioltásról).

A szűrő zárótartománybeli specifikációjának enyhítésére az ad módot, hogy az emberi fül bizonyos teljesítményszint alatt nem érzékeli a hangot, s ez a nagyobb frekvenciák felé haladva egyre inkább igaz [11]. Az elfogadható elnyomást 50 dB-ben állapítottam meg. Az interpoláló szűrő specifikációja és az elkészült szűrő tehát a következőképpen néz ki:



39. ábra Az interpoláló szűrő specifikációja és a megtervezett szűrő

A specifikáció tulajdonképpen rendkívül egyszerű volt: 0dB DC-átvitel és -50 dB elnyomás a Nyquist-frekvenciánál. A szűrő a Matlab *fir1* függvénye segítségével készült el, fokszáma 51.



40. ábra Az interpoláló szűrő csoportkésettetése

Mivel az említett függvény lineáris FIR-szűrőt állít elő, ezért csoportkéselettetése minden frekvencián 25 mintának felel meg (40. ábra). A szűrő minimálfázisú változatát az *rceps* utasítással kapjuk meg, a javulás kisfrekvencián jelentős, a csoportkéselettetés mintegy 40 %-al csökkent. A 8 kHz-es program szempontjából azt jelenti, hogy az ideális 5 minta mellett még kb. $16 \text{ minta} * 48 \text{ kHz} / 8 \text{ kHz} = 2 \text{ minta}$ többletkéselettetés jelenik meg az interpoláció miatt. Így az összkéselettetés 30 mintáról 7 mintára csökkent.

3.4.1.2 A decimáló-interpoláló keretrendszer

Annak érdekében, hogy a tényleges alkalmazás, tehát az FXLMS algoritmus, és a decimálás-interpolálás különváljon, érdemesnek tartottam egy olyan keretrendszer létrehozását (*frame48k.asm*), amely a mintavételi frekvencia váltását elrejtí az alkalmazás elől. A keretrendszer és az alkalmazás az egymás közötti kommunikációt megosztott változókon keresztül oldja meg, a változók tartalma szemaforral védett.

A megosztott változók:

```
left_in_data_8k2
right_in_data_8k2
left_out_data_8k2
right_out_data_8k2
```

Az *in* változók tartalmazzák a 48 kHz-en mintavételezett és 8 kHz-re decimált A/D-k tartalmát, az *out* változókba pedig azt az értéket kell beírni, amelyen a 8 kHz-ről 48 kHz-re történő interpolálást el akarjuk végeztetni, s az eredményt a D/A-kra ki akarjuk rakni. Mind a négy változó 16 bites kettes komplementes kódban értelmezendő ill. kiadandó.

Az alkalmazásban tulajdonképpen annyit kell megtenni, hogy a fájl elején egy `#include "frame48k.asm"` parancs kiadásával a keretrendszert be kell vonni a fordításba. Ezután mindjárt a főprogram legelején meg kell hívni a keretrendszer „konstruktorát”, azaz a *initframe48k* inicializációs szubrutint. Végül a *wait8kdata* szubrutin meghívása biztosítja az alkalmazás és a keretrendszer közötti szinkronizációt és kommunikációt.

A főprogram felépítése tehát a következő:

```
{egyéb include-ok}
#include "frame48k.asm"
{egyéb include-ok}
```


mainprg:

```
call initframe48k; // a keret inicializációja
{az alkalmazás inicializációja}
{A/D-D/A megszakítások engedélyezése}
```

mainloop:

```
call wait8kdata;
call alkalmazas;
jump mainloop;
```

alkalmazas:

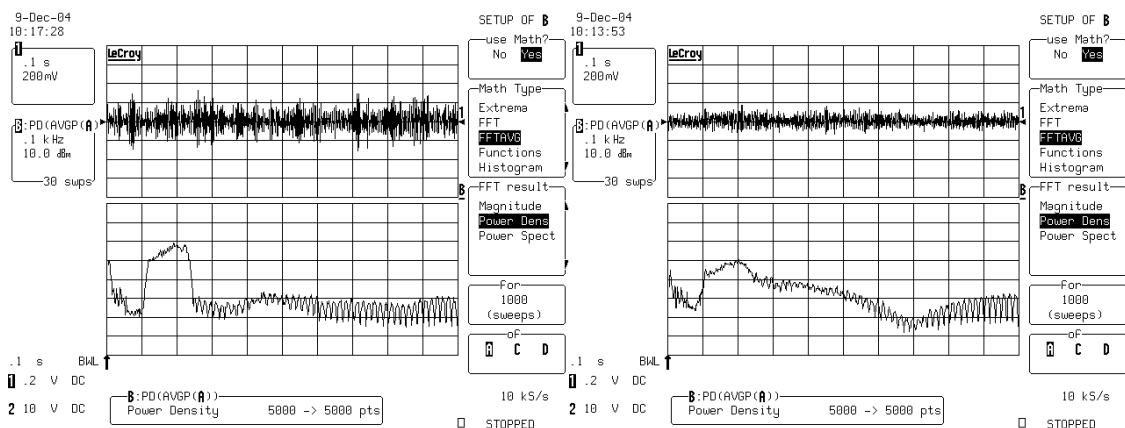
```
{alkalmazás}
// az alkalmazas szubrutinban
// a megosztott változók bárhol
// szabadon használhatóak, változtathatóak
{az alkalmazás folytatása}
```

rts; // visszatérés mainloopba

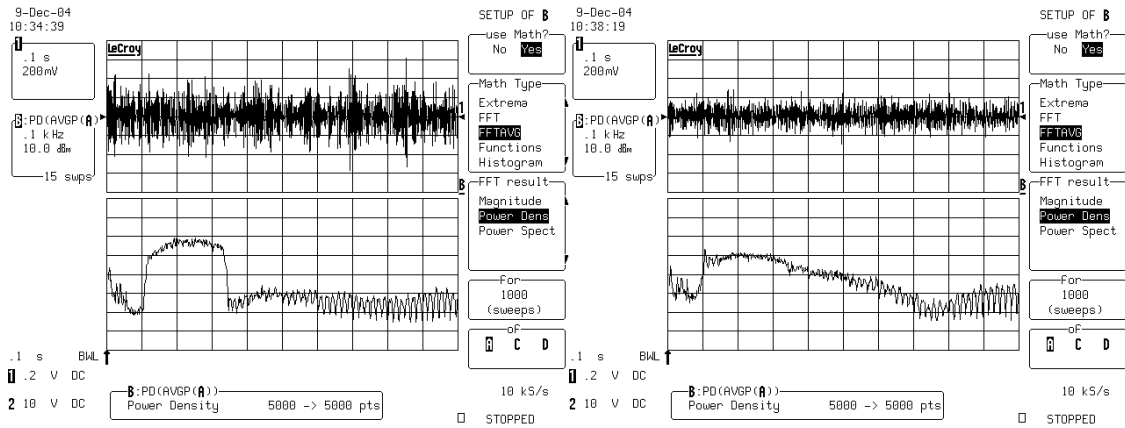
A keretrendszer alkalmazásánál egyetlen korlátozásra kell csak ügyelni, mégpedig arra, hogy a gyors kontextusváltás érdekében a keretrendszer a processzor másodlagos regiszterkészleteket használja, így az alkalmazás csak az elsődleges regiszterkészletet használhatja.

3.4.2 Mérési eredmények

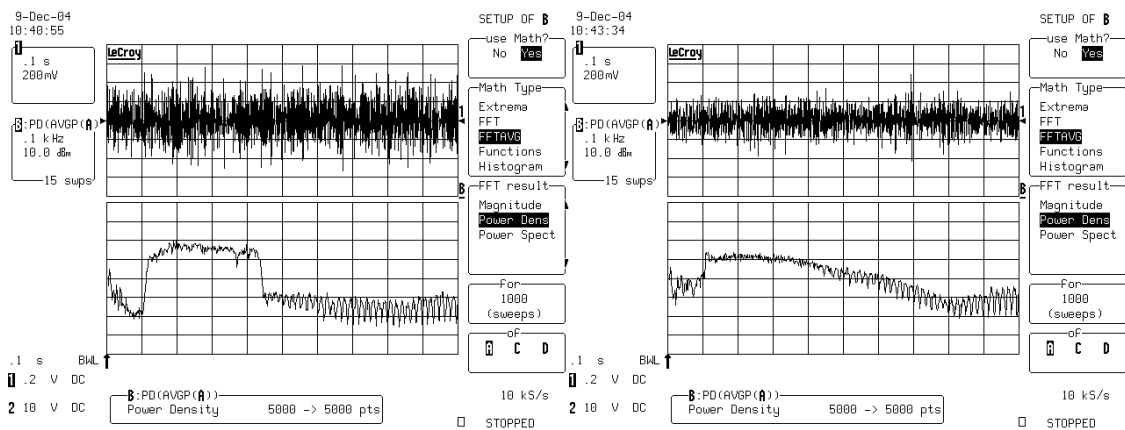
Először is vessük össze a szélessávú zajelnyomást a 8 kHz-es mérésekkel:



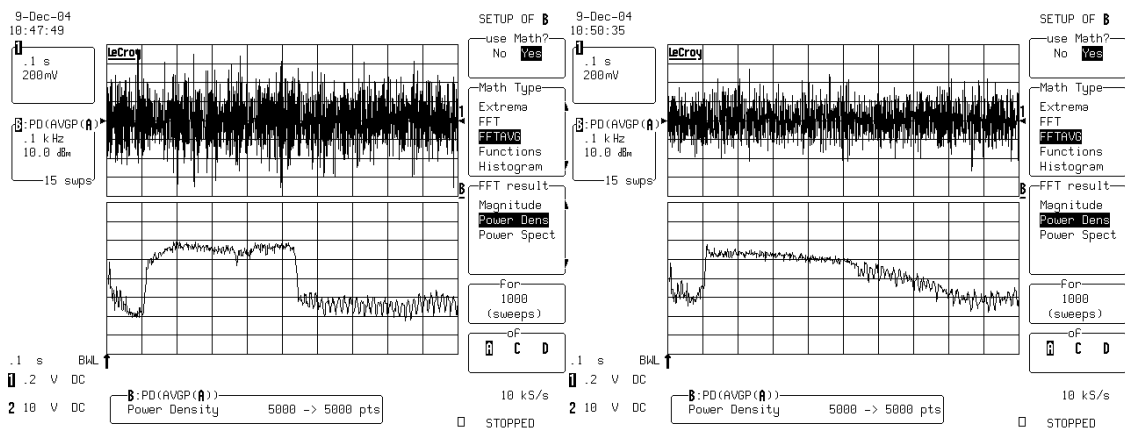
41. ábra Szélessávú elnyomás 150 Hz-es zajszálesség esetén a zajscökkentő bekapcsolása előtt (bal) és után (jobb)



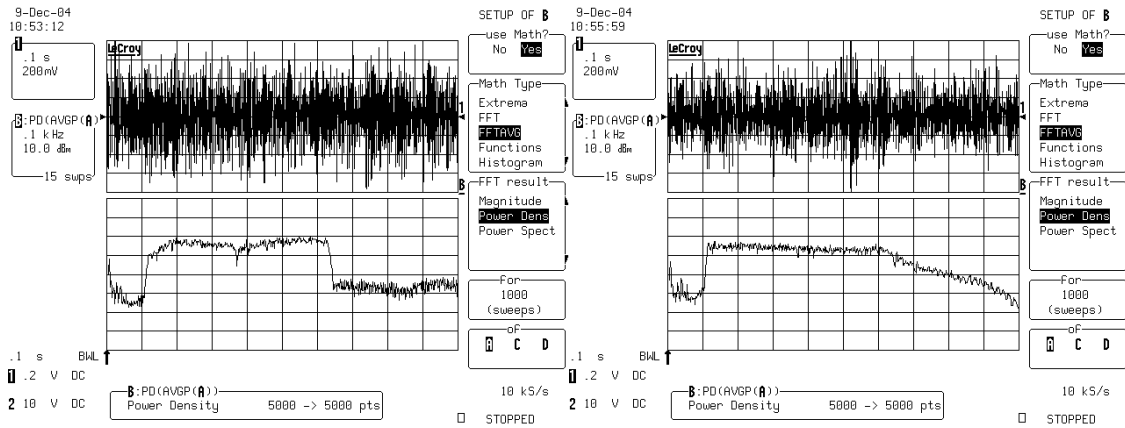
42. ábra Szélessávú elnyomás 250 Hz-es zajsválesség esetén a zajcsökkentő bekapcsolása előtt (bal) és után (jobb)



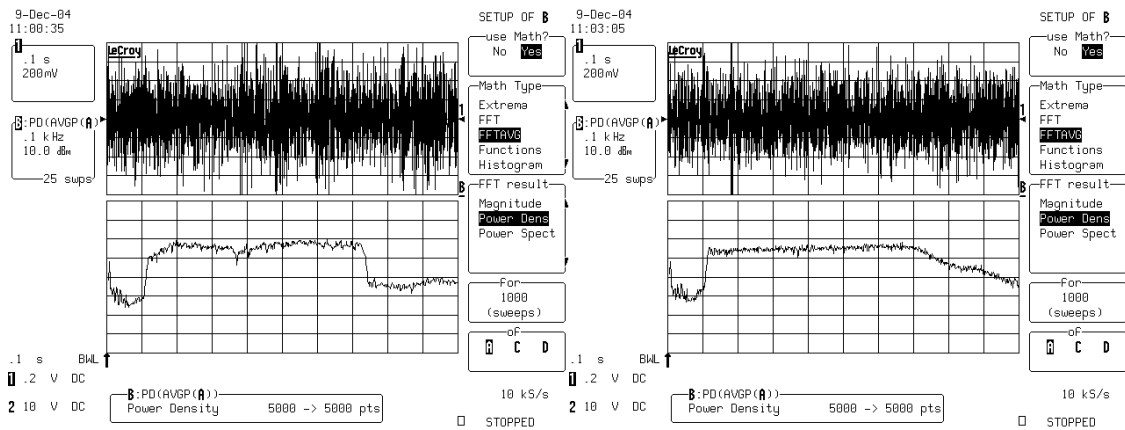
43. ábra Szélessávú elnyomás 350 Hz-es zajsválesség esetén a zajcsökkentő bekapcsolása előtt (bal) és után (jobb)



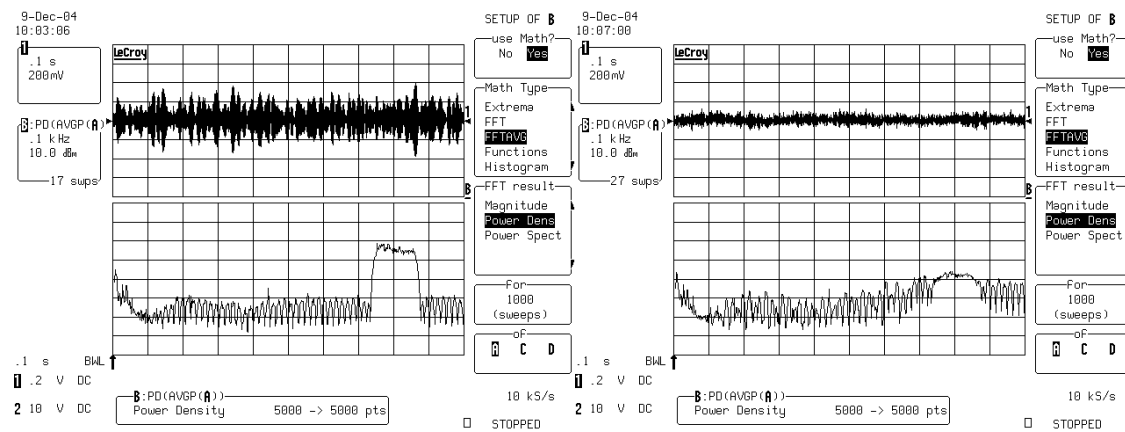
44. ábra Szélessávú elnyomás 450 Hz-es zajsválesség esetén a zajcsökkentő bekapcsolása előtt (bal) és után (jobb)



45. ábra Szélessávú elnyomás 550 Hz-es zajsvzálesség esetén a zajcsökkentő bekapcsolása előtt (bal) és után (jobb)



46. ábra Szélessávú elnyomás 650 Hz-es zajsvzálesség esetén a zajcsökkentő bekapcsolása előtt (bal) és után (jobb)



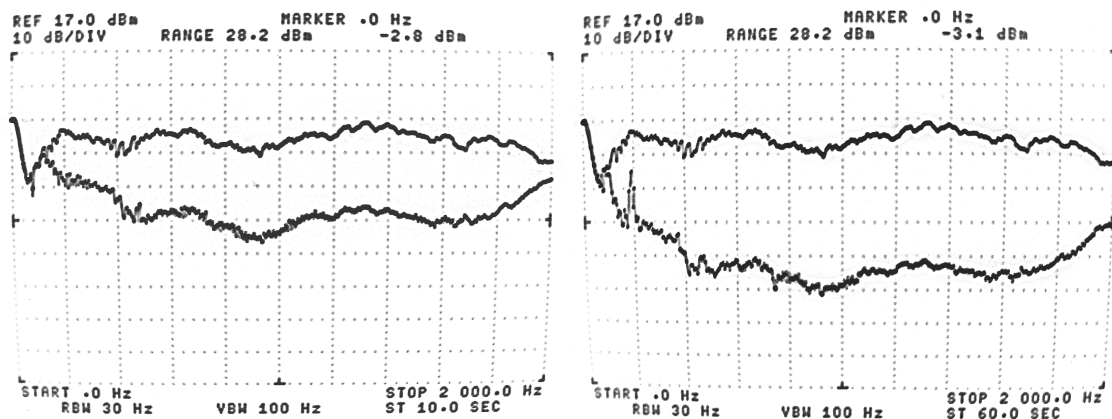
47. ábra Szélessávú elnyomás 150 Hz-es zajsvzálesség esetén a zajcsökkentő bekapcsolása előtt (bal) és után (jobb)

A 41-47. ábrák úgy készültek, hogy a zajforrásként funkcionáló hangdobozt sávkorlátozott fehérzajjal gerjesztve megvizsgáltuk, hogy a digitális zajelnyomó rendszer bekapcsolása előtt és után milyen a hibamikrofon feszültségének

teljesítménysűrűség spektruma. Az egyes kísérletek abban különböztek egymástól, hogy más és más volt a gerjesztőjel sávszélessége. Mint azt az adaptív prediktorról szóló fejezetben is láthattuk, az elnyomás mértéke a gerjesztő zaj sávszélességének növekedésével egyre kisebb lesz, s ez a fenti eredményekből is kitűnik.

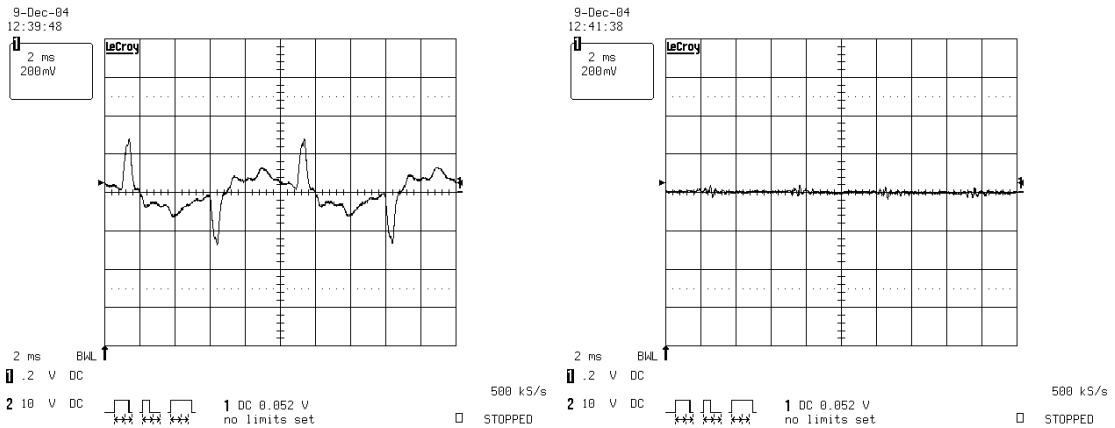
Az eredmények tehát megfeleltek várakozásainknak, mivel sikerült jelentősebb elnyomást elérni, mint 8 kHz mintavételi frekvencia mellett. Megjegyzendő azonban, hogy a szélessávú elnyomás mértéke még nagyobb lenne, ha tovább tudnánk csökkenteni a digitalizálásból adódó késleltetést más típusú A/D-D/A átalakítók alkalmazásával.

A további kísérletek arra vonatkoztak, hogy vajon mennyire sikerül a rendszernek követnie egy változó frekvenciatartalmú jelet, pl. egy sweepelő szinuszt. A következő ábrán azt láthatjuk, hogy az eredmény attól függ, hogy milyen gyors a változás: a baloldali ábrán 6-szor gyorsabban történt meg a 0-2000 Hz-ig történő sweep, mint a jobboldalin. A 10 másodperces sweep kb. 20 dB átlagos elnyomást mutat 200 és 1800 Hz között, míg egy perces, lassú sweep esetén a kioltás már eredményesebb: kb. 30-40 dB.

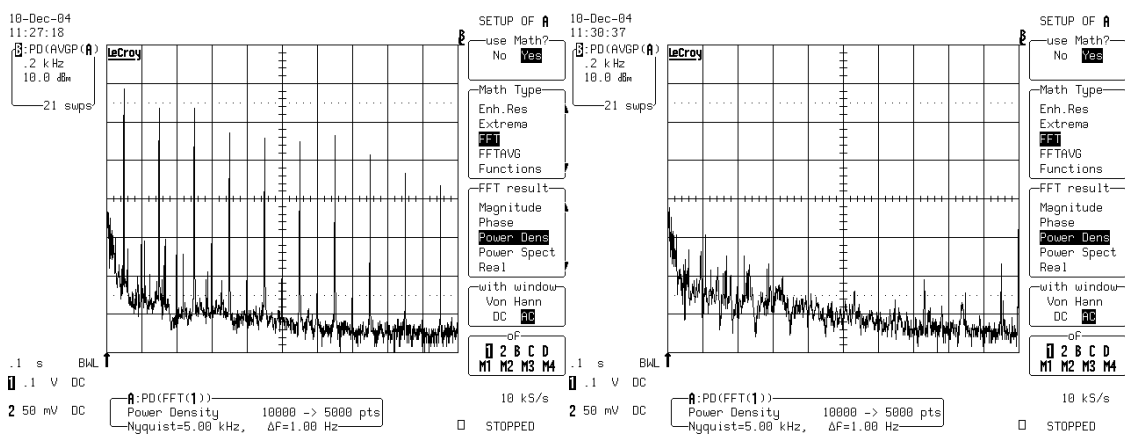


48. ábra Zajelnyomás sweepelő szinusz esetén

A következő kísérlet arra adott választ, hogy milyen mértékű elnyomást képes létrehozni a rendszer periódikus zaj esetén. A mérés maga egyszerű: egy hangdobozt függvénygenerátorral 100 Hz-es négyzögjellel meghajtva megvizsgáltuk, hogy hogyan alakul a hibamikrofon jele a zajcsökkentő rendszer be- és kikapcsolt állapotában (49. ábra).

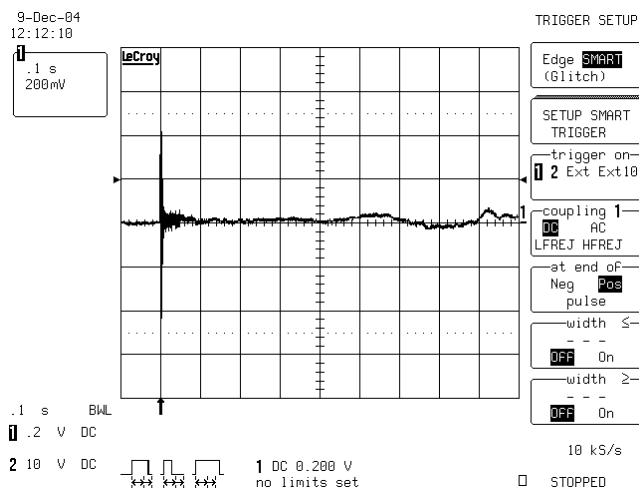


49. ábra A hibamikrofon feszültsége a digitális zajcsökkentő bekapcsolása előtt és után



50. ábra A hibamikrofon feszültségének spektruma a bekapcsolás előtt és után

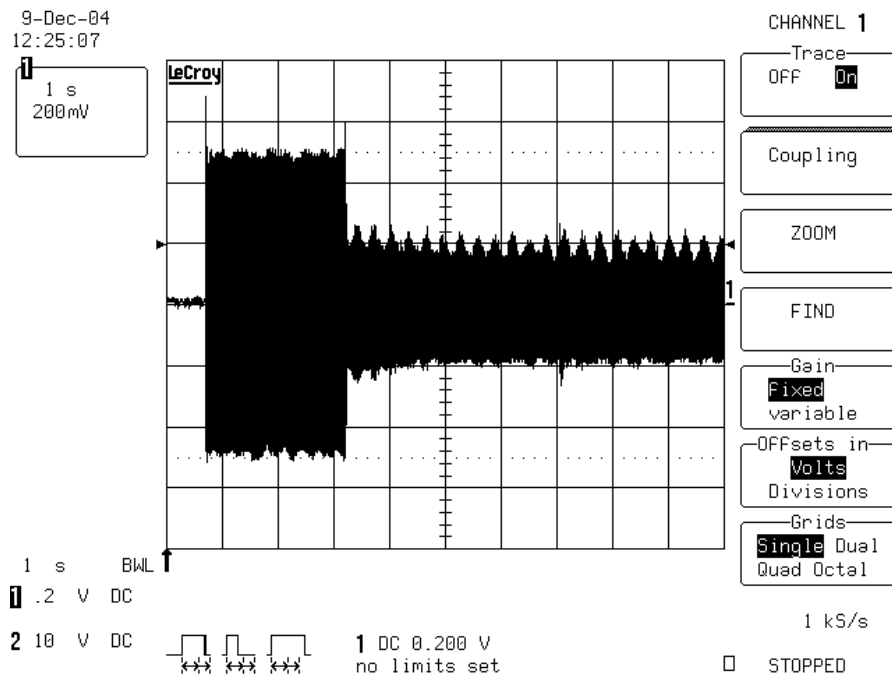
A frekvenciatartományban látszik igazán jól az eredmény: az alapharmonikus 60 dB-lel, a többi harmonikus 60..40 dB-lel csökkent a zajelnyomó rendszer bekapcsolásakor, ami jó eredménynek tekinthető.



51. ábra Kiotlási folyamat 500 Hz-es zavarjel megjelenésekor

Végül pedig nézzük meg, hogy hogyan alakul a hibamikrofon feszültsége a bekapcsolás utáni tranziens állapotban, azaz vizsgáljuk meg a beállítás gyorsaságát (51. ábra). A gerjesztőjel 500 Hz-es szinuszos jel volt, s az ábra tanúsága szerint a beállítás kb. 0.1 s alatt megtörtént.

A következő ábra azt az esetet mutatja, amikor a beavatkozó hangszóró már nem volt képes további elnyomást elérni azért, mert jobban már nem lehetett kivezérelni.



52. ábra Zajcsökkentés a hangszóró telítődési szintjéig lehetséges

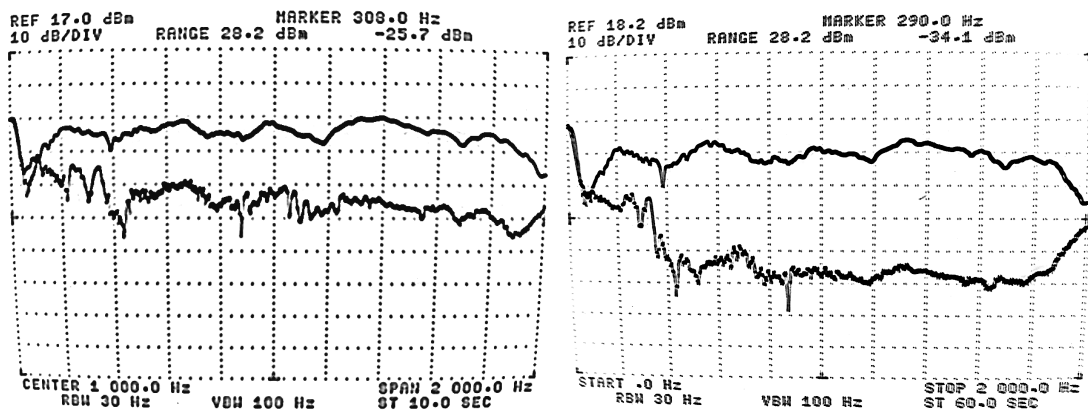
A gerjesztő jel ez esetben 100 Hz-es szinuszos zaj volt. A zajelnyomó algoritmus bekapcsolása után az képes volt egészen a telítődési szintnek megfelelő zajelnyomást produkálnia, tehát az újraskálázási mechanizmus sikeresen stabilizálta az FXLMS algoritmus működését.

4 A hibrid rendszer együttműködése

A hibrid rendszer összeillesztése úgy történik, mint ahogy az a 20. ábrán látható: az analóg részt egy összegző áramkör segítségével illesztjük a DSP-kártya D/A kimenetéhez, ezután identifikáljuk az *ident48k.asm* programmal $S(z)$ -t, amelyből előáll az $\hat{S}(z)$, amelyet azután az elnyomást végző FXLMS algoritmus használ fel.

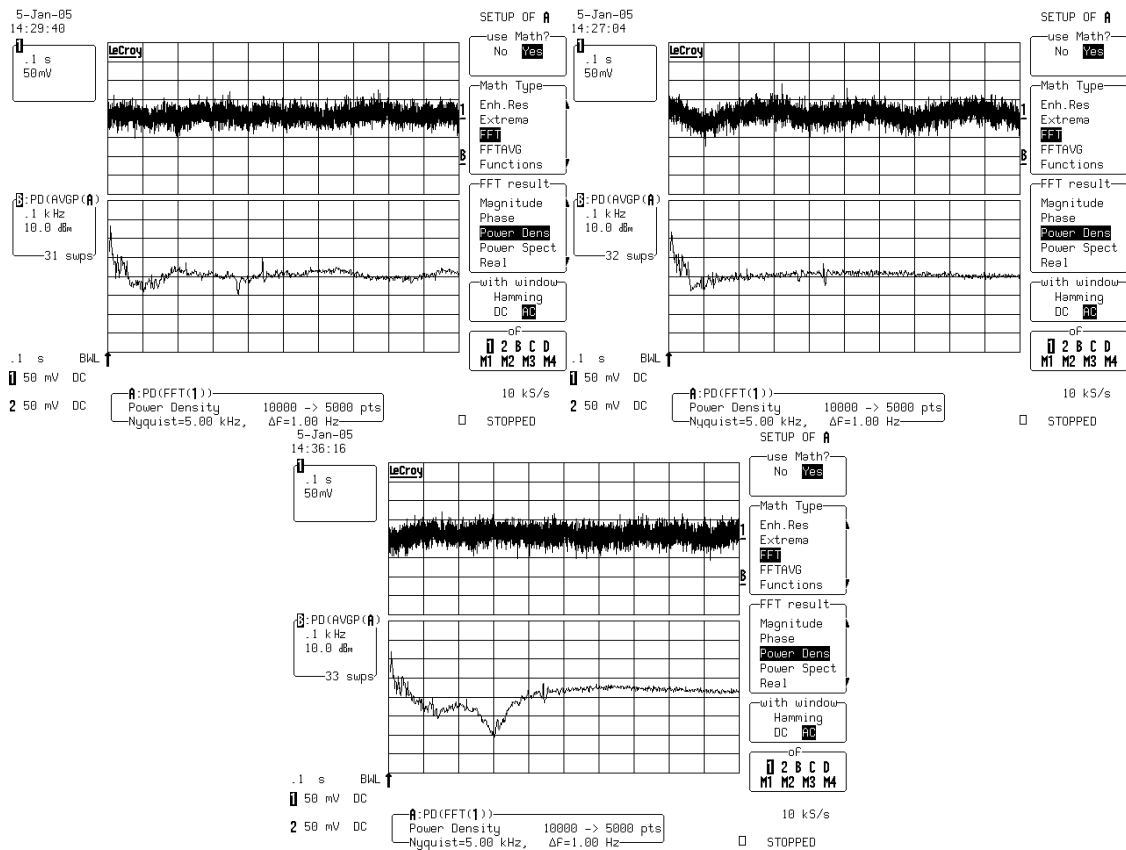
4.1 Mérési eredmények, értékelés

A hibrid rendszer mérései az előzőekben ismertetett mérésekhez hasonlóan történtek. Az első mindjárt a sweepelő szinusz esete. Láthatjuk, hogy az ábrák hasonlítanak a 48. ábrán látottakhoz, viszont jelentős különbségek is felfedezhetőek, ugyanis abban a frekvenciatartományban, ahol az analóg szűrő működik (300 Hz környékén), a digitális szabályozás olyan mértékben lelassult, hogy az ábrákon gyakorlatilag a sweepelési idő növelése sem vonta maga után az elnyomás további növekedését. Így tulajdonképpen arra a következtetésre juthatunk, hogy ahol az analóg szabályozó működik, ott helyettesíti a digitális rendszert, így ebben a frekvenciatartományban gyorsabb kioltás jöhet létre.



53. ábra A hibamikrofon feszültsége a hibrid rendszer bekapcsolása előtt és után

Az analóg szabályozó előnye akkor mutatkozik meg tehát igazán, ha szélessávú gerjesztőjelet alkalmazunk zajforrásként. Ezt mutatja a következő ábra is, ahol a gerjesztő zaj a 0-5 kHz frekvenciasávba eső sztochasztikus zaj volt.



54. ábra A hibrid szabályozás együttműködése szélessávú zaj esetén

A bal felső ábra mutatja azt az esetet, amikor mindkét szabályozó ki volt kapcsolva, a jobb felső esetben csak a digitális, míg az alsó ábrának megfelelő esetben mind az analóg, mind a digitális szabályozás be volt kapcsolva. Látható, hogy a digitális szabályozó önmagában nem volt képes ilyen széles sávú zaj mérséklésére, az analóg szabályozó bekapcsolása viszont 300 Hz környékén zajcsökkentést eredményezett.

5 Értékelés

Dolgozatom célja egy olyan analóg-digitális hibrid zajcsökkentő rendszer létrehozása volt, amely mind az analóg, mind a digitális szabályozás előnyeit igyekszik egyesíteni. Ennek érdekében először megterveztem egy analóg, majd egy digitális zajcsökkentő szabályozást, majd összeillesztettem őket egy kétszintű hibrid szabályozással.

Az analóg szabályozó viszonylag keskeny sávban képes kb. 15-20 dB elnyomás létrehozására, s erre szélessávú zaj esetén is képes. A kereskedelmi forgalomban kapható zajcsökkentő fejhallgatók egy része is hasonló szabályozással épülhet, mivel hasonló elnyomási paraméterekkel rendelkeznek, mint az itt megtervezett analóg zajcsökkentő.

A digitális szabályozó periodikus jelekre akár 60 dB-es elnyomást is létre tud hozni, viszont szélessávú zajok esetén a digitalizálásból eredő késleltetések miatt korlátozott a rendszer elnyomóképessége. Ezen valamelyest lehet javítani a szigma-delta konverterek más típusú (gyorsabb) átalakítókra való cseréjével.

A két alrendszer összekapcsolása teljesítette a hozzá fűzött reményeket, mivel az analóg szabályozó működési tartományában leginkább a gyorsabb analóg szabályozó hatása jutott érvényre, míg a magasabb frekvenciákon a digitális elven működő zajcsökkentés dominált.

A jövőt illetően a digitális rendszer továbbfejlesztése gyors A/D-D/A konverterek alkalmazásával, az analóg rész továbbfejlesztése pedig a fejhallgatónak a zajcsökkentést jobban támogató felépítésével lehetséges.

6 Irodalomjegyzék

- [1] S. J. ELLIOTT - P. A. NELSON: *Active Noise Control*.
IEEE Signal Processing Magazine, October, 1993. 12-35. o.
- [2] TUSCHÁK R.: *Szabályozástechnika*
Műegyetemi Kiadó 2003., 167-173. o.
- [3] GOODWIN, GRAEBE, SALDAGO: *Frequency domain limitations*.
Prentice Hall 2000. 9.fejezet
- [4] S. M. KUO – R. MORGAN, *Active Noise Control: A tutorial Review*.
Proceedings of the IEEE, Vol.87, No. 6, June 1999, 944-954. o.
- [5] KOLLÁR I. - PÉCELI G. - SUJBERT L.: *Digitális jelfeldolgozás (Hallgatói segédlet)* BME-MIT, 2001. 7.fejezet
- [6] HERPY M. - BERKA J-C.: *Aktív RC szűrők*.
Műegyetemi Kiadó, 1993. 212. o.
- [7] ANALOG DEVICES: *ADSP-2106x SHARC™ Users Manual*.
Second edition, May 1997
- [8] XIAOJUN Q.- HANSEN C.H.: *A study of time-domain FXLMS algorithms with control output constraint*.
Journal of Acoustical Society of America, Vol. 109, No. 6, June 2001, 2815-2823.o
- [9] KLIPPEL, W.: *Identification and Compensation of Transducer Nonlinearities*.
Meeting on „Nonlinearities in Electroacoustic Transducer“, 13rd December 2002,
Paris
- [10] WHITE, A. D.: *On Loudspeaker Implementation for Feedback Control, Open-Air, Active Noise Reduction Headsets*. 41.o.
November 19, 1999 Blacksburg, Virginia
- [11] GRÁNÁT J.: *Stúdiótechnika a műsorszórásban*.
Egyetemi jegyzet, KHVM-BME HT Budapest, 2000. 7. o.

7 Függelék

Felhasznált műszerek:

HAMEG HM8040-2 hármastápegység

HAMEG HM8030-5 függvénygenerátor

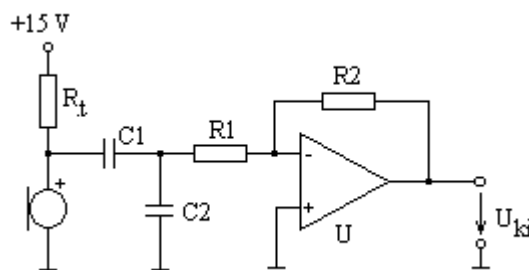
EMG1378 TR-1700 teljesítményerősítő

HEWLETT-PACKARD 3585B spektrumanalizátor

LeCroy WaveRunner LT342 digitális oszcilloszkóp

Maxwell MX-25 303 digitális multiméter

Mikrofon előerősítő:



$$R_t = 28 \text{ k}\Omega$$

$$R_1 = 470 \text{ }\Omega$$

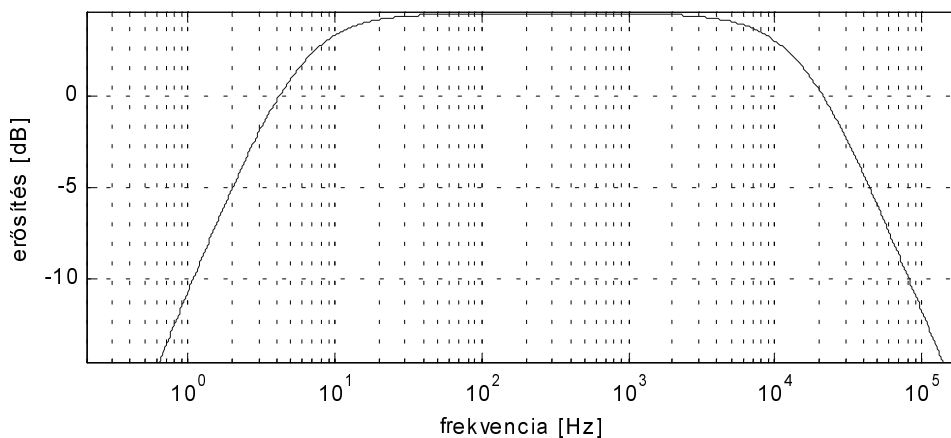
$$R_2 = 47.8 \text{ k}\Omega$$

$$C_1 = 1 \mu\text{F}$$

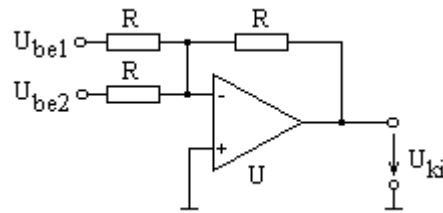
$$C_2 = 22 \text{ nF}$$

$$U = \text{OPA606KP}$$

Az R_t a mikrofon belső FET-es erősítőjének munkapontbeállításához szükséges terhelőellenállás, nyugalmi árama 0.5 mA. C_1 végzi a DC szint leválasztását, C_2 pedig a nagyfrekvenciás zavarokat söntöli. Az R_1, R_2, U kapcsolás invertáló erősítő kapcsolást valósít meg, kb. -100 -as erősítéssel. A mikrofon és a kimenő feszültség közötti átvitel amplitúdókarakterisztikája az alábbi ábrán látható:

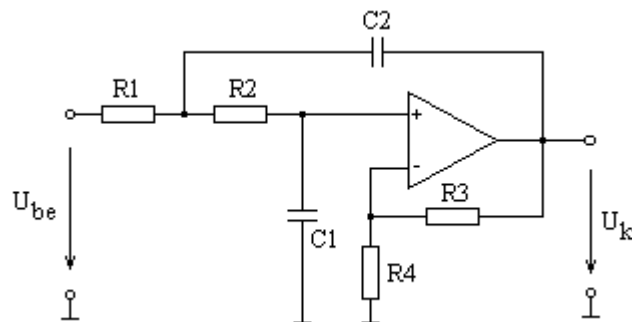


Összegző áramkör:



$R = 1 \text{ k}\Omega$
 $U = \text{uA741}$

Szabályozó másodfokú aluláteresztő szűrő [6]:



$R1 = 11.2 \text{ k} +$
 $+ 4.7 \text{ k potméter}$
 $R2 = 10 \text{ k}$
 $R3 = 920$
 $R4 = 13.1 \text{ k}$
 $C1 = 10 \text{ nF}$
 $C2 = 200 \text{ nF}$
 $U = \text{OPA606KP}$

sysIDrand.asm:Az analóg szabályozó tervezéséhez szükséges identifikációs program

```
#include "ittblfv35serial.asm"
#include "framev35serial.asm"

#define N 350
#define mu 0.005;

.SEGMENT/DM seg_dmda;
.var Input_Datas[N];
.var ref;
.ENDSEG;

.SEGMENT/PM seg_pmda;
.var filter[N];
.ENDSEG;

.SEGMENT/PM seg_pmco;
mainprg:
    b8=filter;
    l8=N;
    m8=1;
```

```

b9=b8;
l9=l8;
m9=1;
b0=Input_Datas;
l0=N;
m0=-1;
m1=1;
f7 = mu;
f0 = 0.0;
r10 = 1;

bit set imask SPR0I;          /* sport0 rx IT unmasked*/
bit set model IRPTEN;        /*IT`s enabled*/

mainloop:
  idle;
  jump mainloop;
receive:
  call data_in;
  /* [25,26,30] */
  r1 = r10;
  r2 = lshift r1 by 25;
  r2 = r1 xor r2;
  r1 = lshift r10 by 26 ;
  r2 = r1 xor r2;
  r1 = lshift r10 by 30;
  r2 = r1 xor r2;

  r10 = rot r2 by 1;
  r5 = lshift r10 by 18;
  r5 = lshift r10 by -18;

  r2 = -15;
  f5 = float r5 by r2;
  f2 = 0.25;
  f5 = f5 - f2;
  f2 = 0.5;
  f5 = f5 * f2;
  r2 = 15;
  r5 = fix f5 by r2;

  dm(right_out_data) = r5;          // ..ki a hangszorora
  r1 = dm(right_in_data);          // sys output

  r2 = -15;

  f0 = float r5 by r2;
  f1 = float r1 by r2;
  dm(ref) = f0;                    // ref jel --> kesobb szamitasi eredmeny

  f8 = 0.0;
  f13 = 0.0;
  f0= dm(i0,m1);
  f4 = pm(i8,m8);
  lcntr = N-1, do FIR until lce;
FIR:      f13 = f0 * f4, f8 = f8 + f13, f0 = dm(i0,m1), f4 =
pm(i8,m8);
          f13 = f0 * f4, f8 = f8 + f13;
          f8 = f8 + f13;

  f15 = f1 - f8;                    // error = sysout - model out

```

```

// adaptive filter coef. update
f2 = f7 * f15; // error*mu
f8 = pm(i8,m8), f0 = dm(i0,m1);
f3 = f0 * f2;
lcntr = N-1, do UPDATE until lce;
    f4 = f8 + f3, f0 = dm(i0,m1), f8 = pm(i8,m8);
UPDATE:    f3 = f0 * f2, pm(i9,m9) = f4;
    f4 = f8 + f3;
pm(i9,m9) = f4;
modify(i0,m0); // for next FIR

r14 = 15; // error out
r15 = fix f15 by r14;
dm(left_out_data) = r15;
call data_out;
rts;
.ENDSEG;

```

PsRand.asm:Álvéletlenszám-sorozat előállító modul

```

.SEGMENT/DM seg_dmda;
.var PsRandSeed = 1;
.ENDSEG;

.SEGMENT/PM seg_pmco;
// modositva: r1, r2, r5, r10
// eredmeny: f5

PsRand:
    // [25,26,30]
    r10 = dm(PsRandSeed);
    r1 = r10;
    r2 = lshift r1 by 25;
    r2 = r1 xor r2;
    r1 = lshift r10 by 26 ;
    r2 = r1 xor r2;
    r1 = lshift r10 by 30;
    r2 = r1 xor r2;

    r10 = rot r2 by 1;
    dm(PsRandSeed) = r10;
    r5 = lshift r10 by 18;
    r5 = lshift r10 by -18;

    r1 = -15;
    f5 = float r5 by r1; // by -15
    f2 = 0.25;
    f5 = f5 - f2;
    f2 = 0.5;
    f5 = f5 * f2;

    rts;
.ENDSEG;

```

Recip.asm: Reciprok-képző modul

```

.SEGMENT/PM seg_pmco;

```

```

recip:
    f11 = 2.0;
    F0=RECIPS F12, F7=F0;           //{Get 8 bit seed R0=1/D}
    F12=F0*F12;                   //{D' = D*R0}
    F7=F0*F7, F0=F11-F12;         //{F0=R1=2-D', F7=N*R0}
    F12=F0*F12;                   //{F12=D'-D'*R1}
    F7=F0*F7, F0=F11-F12;         //{F7=N*R0*R1, F0=R2=2-D'}
    F12=F0*F12;                   //{F12=D'=D'*R2}
    F7=F0*F7, F0=F11-F12;         //{F7=N*R0*R1*R2, F0=R3=2-D'}
    F0=F0*F7;                      //{F7=N*R0*R1*R2*R3}

    rts;
.ENDSEG;

```

ident8k.asm: A 8 kHz-es megvalósítás identifikációs része

```

#include "ittblfv35serial.asm"      // interrupt table
#include "framev35serial.asm"
#include "PsRand.asm"

#define Nw          350
#define muident     0.005

.SEGMENT/DM seg_dmda;
.var x_for_wfilter[Nw];
.var ref;
.var e;
.ENDSEG;

.SEGMENT/PM seg_pmda;
.var wfilter[Nw];
.ENDSEG;

.SEGMENT/PM seg_pmco;
mainprg:

// DAG1 setup
m0=1;
m1=-1;
b0=x_for_wfilter;
l0=Nw;
// DAG2 setup
m8=1;
b8=wfilter;
l8=Nw;
b9=b8;           // needed for fast update
l9=l8;

// register file setup
// reg file global variables
// reg file constants
f7 = muident;
r14 = -15;
r15 = 15;
bit set imask SPR0I; // sport0 rx IT unmasked
bit set model IRPTEN; //IT`s enabled

////////////////////////////////////
mainloop:
    idle;

```

```

                jump mainloop;
//////////
receive:
    call data_in;
    f7=muident;

    call PsRand;

    f0 = 1.0;
    f5 = f5 * f0;
    r5 = fix f5 by r15;                // by 15
    dm(right_out_data) = r5;          // ..ki a hangszorora
    r1 = dm(right_in_data);           // sys output

    f0 = float r5 by r14;              // by -15
    f1 = float r1 by r14;              // by -15
    dm(ref) = f0;                       // ref jel
    f8 = 0.0;
    f12 = 0.0;
    f0= dm(i0,m0), f4 = pm(i8,m8);
    lcntr = Nw-1, do FIR until lce;
FIR:                f12 = f0 * f4, f8 = f8 + f12, f0 = dm(i0,m0), f4 =
pm(i8,m8);
                f12 = f0 * f4, f8 = f8 + f12;
                f8 = f8 + f12;
                f5 = f1 - f8;                // error = sysout - model out

                // adaptive filter coef. update
                f2 = f7 * f5;                // error*mu
                f0 = dm(i0,m0), f8 = pm(i8,m8);
                f3 = f0 * f2;
                lcntr = Nw-1, do IDUPD until lce;
                f4 = f8 + f3, f0 = dm(i0,m0),f8 = pm(i8,m8);
IDUPD:            f3 = f0 * f2, pm(i9,m8) = f4;
                f4 = f8 + f3;
                pm(i9,m8) = f4;

                modify(i0,m1);                // for next FIR
                f0 = dm(ref);
                dm(i0,0) = f0;

                r0 = fix f5 by r15;            // by 15

                dm(left_out_data) = r0;
                call data_out;

    rts;
.ENDSEG;

```

anc8k.asm: A 8 kHz-es zajcsökkentő program

```

#define NLMS 1
#define KORL 1

#include "ittblfv35serial.asm" // interrupt table
#include "framev35serial.asm"

#if NLMS | KORL
    #include "Recip.asm"
#endif

```



```

#define Ns          350
#define Nw          350

#if KORL
    #define ykorlat      0.8;
#endif

#if NLMS
    //alfa = 1.5
    #define alfa          1.5;
    #define sigma        0.001;
    #define delta        35; // > S(z) kesleltetese (kb. 34)
    #define mumax        10.0;
    #define Enszorzo    30.5 // > 1+delta/Nw
#endif

#define mu          1.5

.SEGMENT/DM seg_dmda;
.var x_for_wfilter[Nw];
.var x_for_sfilter[Ns];
.var y_for_sfilter[Ns];
#if NLMS
    .var xx2[Nw];           // xx negyzetei
    .var Energia=0.0;      // sum{xx2}, ennek reciproka kell
    .var Ecnt=Nw;
#endif
.var xx_vect[Nw];
.var yy;
.var e;
.var ref;
.ENDSEG;

.SEGMENT/PM seg_pmda;
.var sfilter[Ns]="sfilter.dat";
.var wfilter[Nw];
.ENDSEG;

.SEGMENT/PM seg_pmco;
mainprg:

// DAG1 setup
m0=1;
m1=-1;
b0=x_for_wfilter;
l0=Nw;
b1=x_for_sfilter;
l1=Ns;
b2=y_for_sfilter;
l2=Ns;
b3=xx_vect;
l3=Nw;
#if NLMS
    b5=xx2;
    l5=Nw;
#endif

// DAG2 setup
m8=1;
b8=wfilter;

```

```

        l8=Nw;
        b9=b8; // needed for fast update
        l9=l8;
        b10=sfilter;
        l10=Ns;

// register file setup
// reg file global variables
#if NLMS
        f7 = 0.0; // mu(n)
#endif
// reg file constants
#if !NLMS
        f7 = mu;
#endif
        r14 = -15;
        r15 = 15;

        bit set imask SPR0I; // sport0 rx IT unmasked
        bit set model IRPTEN; //IT`s enabled

////////////////////////////////////
mainloop:
        idle;
        jump mainloop;
////////////////////////////////////
receive:
        call data_in;
        f7=mu;

// x(n) -> xx(n)
        f8 = 0.0;
        f12 = 0.0;
        f0 = dm(i1,m0), f4 = pm(i10,m8);
        lcntr = Ns-1, do XSXX until lce;
XSXX:    f12 = f0 * f4, f8 = f8 + f12, f0 = dm(i1,m0), f4 =
pm(i10,m8);
        f12 = f0 * f4, f8 = f8 + f12;
        f8 = f8 + f12;
        modify(i1,m1);
        dm(i3,0) = f8;

        r0 = dm(right_in_data); // error signal from mic
        f0 = float r0 by r14; // by -15

        dm(e) = f0; // beolvassuk a hibaerteket
(e(n)) a mikrofonrol

        f1 = dm(yy);
        f1 = f1 + f0; // es eloallitjuk belole es
yy(n)-bol a referenciajelet
        dm(i0,0) = f1; // ezt szurjuk w-n keresztul
        dm(i1,0) = f1; // es s-en keresztul

#if NLMS
////////////////////////////////////
// energia ujraszamolasa
////////////////////////////////////

```

```

        f4 = f8;                // xx legujabb erteket
negyzetre emeljuk
        f4 = f4*f8;            // itt
        f0 = dm(i5,0);         // elovesszuk a legreggebbi
xx^2-et
        f5 = dm(Energia);      // es az Energia elozo
erteket
        f5 = f5 - f0;          // a legreggebbi elvetele
        f5 = f5 + f4;          // a legujabb hozzaadasa
        dm(Energia) = f5;      // megvan az uj Energia
        dm(i5,m1) = f4;        // es a legreggebbi helyre
kerul xx^2(n)
        f12 = Enszorzo;
        f5 = f5 * f12;
        f9 = sigma;           // 0 Energianal is kell vmi
        f12 = f5 + f9;        // hogy reciproka ne INF
legyen
        // most jon a f12 reciprokanak kepzese
        f0 = alfa;             // f0: szamlalo, f7: nevezo
        call recip;           // f0 = f0 / f7
        f7 = f0;
        f1 = mumax;           // nagyobb-e mint mumax
        f2 = f0 - f1;
        if lt jump kisebb;
        f7 = mumax;           // ha igen korlatozzuk
kisebb:                               // ha nem ugy hagyjuk

        // most megvizsgaljuk, hogy az Energia becslo kezdeti
tranziens
        // szakaszan tulvagyunk-e
        r0 = dm(Ecnt);
        r0 = pass r0;
        if lt jump tranzvege;
        r0 = r0 - 1;
        dm(Ecnt) = r0;
        f7 = mu;               // a becslo torzitott, mu0-t
hasznaljuk

tranzvege:                             // hasznalható a mu(n)
#endif

//    x(n) -> y(n)
        f8 = 0.0;
        f12 = 0.0;
        f0 = dm(i0,m0), f4 = pm(i8,m8);
        lcntr = Nw-1, do XWY until lce;
XWY:    f12 = f0 * f4, f8 = f8 + f12, f0 = dm(i0,m0), f4 =
pm(i8,m8);
        f12 = f0 * f4, f8 = f8 + f12;
        f8 = f8 + f12;
        modify(i0,m1);         // a kov egygyel visszabb van

#if KORL
// y-t korlatozo algoritmus
        f0 = abs f8;
        f1 = ykorlat;
        comp(f0,f1);
        if le jump no_saturation; // abs(y)<ykorlat
// abs(y)>=ykorlat

        f12 = f0;

```

```

        f0 = ykorlat;
        call recip;
        f8 = f8 * f0;           // y=ykorlat*sign(y)
        f12 = f0;
        f1 = 1.0;
        f12 = f12 * f1;

        f9 = pm(i8,m8);
        lcntr = Nw-1, do end_sat until lce;
            f13 = f9 * f12;    f9 = pm(i8,m8);
end_sat:    pm(i9,m9)=f13;
            f13 = f9 * f12;
            pm(i9,m9)=f13;

no_saturation:
#endif
        f8 = -f8;
        r0 = fix f8 by r15;           // by 15
        dm(right_out_data) = r0;

        f8 = -f8;
        dm(i2,0)=f8;           // yvect <- y

//      y(n) -> yy(n)
f8 = 0.0;
f12 = 0.0;
f0 = dm(i2,m0), f4 = pm(i10,m8);
lcntr = Ns-1, do YSYY until lce;
YSYY:      f12 = f0 * f4, f8 = f8 + f12, f0 = dm(i2,m0), f4 =
pm(i10,m8);
            f12 = f0 * f4, f8 = f8 + f12;
            f8 = f8 + f12;
            dm(yy) = f8;
            modify(i2,m1);
            // adaptive filter coef. update

f0 = dm(e);
f2 = f0 * f7 ;           // error*mu
f8 = pm(i8,m8), f0 = dm(i3,m0);
f3 = f0 * f2;
lcntr = Nw-1, do UPDATE until lce;
            f4 = f8 + f3, f0 = dm(i3,m0),f8 = pm(i8,m8);
UPDATE:    f3 = f0 * f2, pm(i9,m8) = f4;
            f4 = f8 + f3;
            pm(i9,m8) = f4;
            modify(i3,m1);

        call data_out;
        rts;
.ENDSEG;

```

ident48k.asm: A 48 kHz-es identifikációs program

```

#include "ittblfv35serial.asm"           // interrupt table
#include "framev35serial.asm"
#include "frame48k.asm"
#include "PsRand.asm"

#define Nw          350

```

```

#define muident    0.005

.SEGMENT/DM seg_dmda;
.var x_for_wfilter[Nw];
.var ref;
.var e;
.ENDSEG;

.SEGMENT/PM seg_pmda;
.var wfilter[Nw];
.ENDSEG;

.SEGMENT/PM seg_pmco;
mainprg:
    call initframe48k;
// DAG1 setup
    m0=1;
    m1=-1;
    b0=x_for_wfilter;
    l0=Nw;
// DAG2 setup
    m8=1;
    b8=wfilter;
    l8=Nw;
    b9=b8; // needed for fast update
    l9=l8;
// register file setup
    // reg file global variables
    // reg file constants
    f7 = muident;
    r14 = -15;
    r15 = 15;

    bit set imask SPR0I; // sport0 rx IT unmasked
    bit set model IRPTEN; //IT`s enabled

////////////////////////////////////
mainloop:
    call wait8kdata;
    call ident;
    jump mainloop;
////////////////////////////////////
ident:
    f7=muident;

    call PsRand;

    f0 = 1.0;
    f5 = f5 * f0;
    r5 = fix f5 by r15; // by 15
    dm(right_out_data_8k2) = r5; // ..ki a hangszorora
    r1 = dm(right_in_data_8k2); // sys output

    f0 = float r5 by r14; // by -15
    f1 = float r1 by r14; // by -15
    dm(ref) = f0; // ref jel

    f8 = 0.0;
    f12 = 0.0;
    f0= dm(i0,m0), f4 = pm(i8,m8);
    lcntr = Nw-1, do FIR until lce;

```

```

FIR:          f12 = f0 * f4, f8 = f8 + f12, f0 = dm(i0,m0), f4 =
pm(i8,m8);
          f12 = f0 * f4, f8 = f8 + f12;
          f8 = f8 + f12;
          f5 = f1 - f8;          // error = sysout - model out

          // adaptive filter coef. update
          f2 = f7 * f5;          // error*mu
          f0 = dm(i0,m0), f8 = pm(i8,m8);
          f3 = f0 * f2;
          lcntr = Nw-1, do IDUPD until lce;
          f4 = f8 + f3, f0 = dm(i0,m0),f8 = pm(i8,m8);
IDUPD:      f3 = f0 * f2, pm(i9,m8) = f4;
          f4 = f8 + f3;
          pm(i9,m8) = f4;

          modify(i0,m1);          // for next FIR
          f0 = dm(ref);
          dm(i0,0) = f0;
          r0 = fix f5 by r15;          // by 15
          dm(left_out_data_8k2) = r0;
          rts;
.ENDSEG;

```

anc48k.asm: A 48 kHz-es zajcsökkentő program

```

#define NLMS 1
#define KORL 1

#include "ittblfv35serial.asm" // interrupt table
#include "framev35serial.asm"
#include "frame48k.asm"

#if NLMS | KORL
#include "Recip.asm"
#endif

#define Ns 350
#define Nw 350

#if KORL
#define ykorlat 0.8;
#endif
#if NLMS
#define alfa 1.5;
#define sigma 0.001;
#define delta 35; // > S(z) kesleltetese (kb. 34)
#define mumax 10.0;
#define Enszorzo 30.5 // > 1+delta/Nw
#endif
#define mu 1.5

.SEGMENT/DM seg_dmda;
.var x_for_wfilter[Nw];
.var x_for_sfilter[Ns];
.var y_for_sfilter[Ns];
#if NLMS
.var xx2[Nw]; // xx negyzetei
.var Energia=0.0; // sum{xx2}, ennek reciproka kell
.var Ecnt=Nw;

```

```

#endif
.var xx_vect[Nw];
.var yy;
.var e;
.var ref;
.ENDSEG;
.SEGMENT/PM seg_pmda;
.var sfilter[Ns]="sfilteran.dat";
.var wfilter[Nw];
.ENDSEG;

.SEGMENT/PM seg_pmco;
mainprg:
    call initframe48k;
// DAG1 setup
    m0=1;
    m1=-1;
    b0=x_for_wfilter;
    l0=Nw;
    b1=x_for_sfilter;
    l1=Ns;
    b2=y_for_sfilter;
    l2=Ns;
    b3=xx_vect;
    l3=Nw;
#if NLMS
    b5=xx2;
    l5=Nw;
#endif
// DAG2 setup
    m8=1;
    b8=wfilter;
    l8=Nw;
    b9=b8; // needed for fast update
    l9=l8;
    b10=sfilter;
    l10=Ns;
// register file setup
    // reg file global variables
#if NLMS
    f7 = 0.0; // mu(n)
#endif
// reg file constants
#if !NLMS
    f7 = mu;
#endif
    r14 = -15;
    r15 = 15;
    bit set imask SPR0I; // sport0 rx IT unmasked
    bit set model IRPTEN; //IT`s enabled

////////////////////////////////////
mainloop:
    call wait8kdata;
    call ANC;
    jump mainloop;
////////////////////////////////////
ANC:
    f7=mu;

// x(n) -> xx(n)

```

```

f8 = 0.0;
f12 = 0.0;
f0 = dm(i1,m0), f4 = pm(i10,m8);
lcntr = Ns-1, do XSXX until lce;
XSXX:   f12 = f0 * f4, f8 = f8 + f12, f0 = dm(i1,m0), f4 =
pm(i10,m8);
       f12 = f0 * f4, f8 = f8 + f12;
       f8 = f8 + f12;
modify(i1,m1);
dm(i3,0) = f8;

r0 = dm(right_in_data_8k2); // error signal from mic
f0 = float r0 by r14; // by -15

dm(e) = f0; // beolvassuk a hibaerteket
f1 = dm(yy);
f1 = f1 + f0; // referencia eloallitasa
dm(i0,0) = f1; // ezt szurjuk w-n keresztul
dm(i1,0) = f1; // es s-en keresztul

#if NLMS
////////////////////////////////////
// energia ujraszamolasa
////////////////////////////////////
f4 = f8; // xx(n)^2 szamitasa
f4 = f4*f8; // itt
f0 = dm(i5,0); // a legregebbi xx^2
f5 = dm(Energia); // es az Energia elozo erteket
f5 = f5 - f0; // a legregebbi elvetele
f5 = f5 + f4; // a legujabb hozzaadasa
dm(Energia) = f5; // megvan az uj Energia
dm(i5,m1) = f4; // es a legregebbi helyre
kerul xx^2(n)
f12 = Enszorzo;
f5 = f5 * f12;
f9 = sigma; // 0 Energianal is kell vmi
f12 = f5 + f9; // hogy reciproka ne INF
legyen

// most jon a f12 reciprokanak kepzese
f0 = alfa; // f0: szamlalo, f7: nevező
call recip; // f0 = f0 / f7
f7 = f0;
f1 = mumax; // nagyobb-e mint mumax
f2 = f0 - f1;
if lt jump kisebb;
f7 = mumax; // ha igen korlatozzuk
kisebb: // ha nem ugy hagyjuk

// most megvizsgaljuk, hogy az Energia becslo kezdeti
tranziens
// szakaszan tul vagyunk-e
r0 = dm(Ecnt);
r0 = pass r0;
if lt jump tranzvege;
r0 = r0 - 1;
dm(Ecnt) = r0;
f7 = mu; // a becslo torzitott, mu0-t
hasznaljuk

tranzvege: // hasznalhato a mu(n)
#endif

```



```

//      x(n) -> y(n)
          f8 = 0.0;
          f12 = 0.0;
          f0 = dm(i0,m0), f4 = pm(i8,m8);
          lcntr = Nw-1, do XWY until lce;
XWY:      f12 = f0 * f4, f8 = f8 + f12, f0 = dm(i0,m0), f4 =
pm(i8,m8);
          f12 = f0 * f4, f8 = f8 + f12;
          f8 = f8 + f12;
          modify(i0,m1);
#if KORL
// y-t korlatozo algoritmus
          f0 = abs f8;
          f1 = ykorlat;
          comp(f0,f1);
          if le jump no_saturation; // abs(y)<ykorlat
                                   // abs(y)>=ykorlat

          f12 = f0;
          f0 = ykorlat;
          call recip;
          f8 = f8 * f0; // y=ykorlat*sign(y)
          f12 = f0;
          f1 = 1.0;
          f12 = f12 * f1;

          f9 = pm(i8,m8);
          lcntr = Nw-1, do end_sat until lce;
          f13 = f9 * f12; f9 = pm(i8,m8);
end_sat:  pm(i9,m9)=f13;
          f13 = f9 * f12;
          pm(i9,m9)=f13;

no_saturation:
#endif
          f8 = -f8;
          r0 = fix f8 by r15; // by 15
          dm(right_out_data_8k2) = r0;

          f8 = -f8;
          dm(i2,0)=f8; // yvect <- y

//      y(n) -> yy(n)
          f8 = 0.0;
          f12 = 0.0;
          f0 = dm(i2,m0), f4 = pm(i10,m8);
          lcntr = Ns-1, do YSYX until lce;
YSYX:      f12 = f0 * f4, f8 = f8 + f12, f0 = dm(i2,m0), f4 =
pm(i10,m8);
          f12 = f0 * f4, f8 = f8 + f12;
          f8 = f8 + f12;
          dm(yy) = f8;
          modify(i2,m1);
          // adaptive filter coef. update

          f0 = dm(e);
          f2 = f0 * f7; // error*mu
          f8 = pm(i8,m8), f0 = dm(i3,m0);
          f3 = f0 * f2; // f3 = xx(n)*error*mu
          lcntr = Nw-1, do UPDATE until lce;
          f4 = f8 + f3, f0 = dm(i3,m0),f8 = pm(i8,m8);

```

```

UPDATE:      f3 = f0 * f2, pm(i9,m8) = f4;
             f4 = f8 + f3;
             pm(i9,m8) = f4;
             modify(i3,m1);
rts;
.ENDSEG;

```

noisegen.asm: Zajgenerátor a szélessávú működés teszteléséhez

```

#include "ittblfv35serial.asm"           // interrupt table
#include "framev35serial.asm"          // codec frame
#include "PsRand.asm"                  // pseudo random generator

#define Nn          501

.SEGMENT/DM seg_dmda;
.var noise[Nn];
.ENDSEG;

.SEGMENT/PM seg_pmda;
.var nfilter[Nn]="nfilter.dat";
.ENDSEG;

.SEGMENT/PM seg_pmco;
mainprg:
// DAG1 setup
m0=1;
m1=-1;
b4=noise;
l4=Nn;
// DAG2 setup
m8=1;
b11=nfilter;
l11=Nn;
// reg file constants
r14 = -15;
r15 = 15;
bit set imask SPR0I;           // sport0 rx IT unmasked
bit set model IRPTEN;         // IT`s enabled

mainloop:
idle;
jump mainloop;
receive:
call PsRand;
dm(i4,0)=f5;

// noise(n) -> nfilterednoise(n)
f8 = 0.0;
f12 = 0.0;
f0 = dm(i4,m0), f4 = pm(i11,m8);
lcntr = Nn-1, do nfilt until lce;
nfilt:      f12 = f0 * f4, f8 = f8 + f12, f0 = dm(i4,m0), f4 =
pm(i11,m8);
           f12 = f0 * f4, f8 = f8 + f12;
           f8 = f8 + f12;
modify(i4,m1);
f9 = 10.0;
f8 = f8 * f9;

```

```
    r0 = fix f8 by r15;
    dm(left_out_data) = r0;           // ..ki a D/A - kra
    dm(right_out_data) = r0;
    call data_out;
  rts;
.ENDSEG;
```