

# Aktív zajcsökkentő rendszerek megvalósítása szenzorhálózattal

Az aktív zajcsökkentő rendszerek az elnyomandó zajt mikrofonok segítségével érzékelik, majd ez alapján a megfelelő algoritmus által előállított „ellenzajt” hangszórók segítségével kiadják, mely a szuperpozíció elv alapján kioltja a zajt. A bonyolult algoritmusok implementálása a nagy számítási teljesítményt biztosító jelfeldolgozó processzorok (DSP) alkalmazásával lehetséges. A DSP-hez az érzékelő és beavatkozó elemeket analóg-digitális (AD) és digitális-analóg (DA) átalakítókkal csatlakoztatjuk. A számítás során gyakran felhasználunk egy olyan külső referenciajelet, melyből információt nyerünk az elnyomandó zajról (pl. frekvencia). Az aktív zajcsökkentő rendszereket sikeresen alkalmazták például forgógépek által generált zajok kioltására repülőekben, autókban, légkondicionálóknál, vagy transzformátor zajának elnyomására. Az aktív zajcsökkentésről bővebb áttekintés található [1] és [2]-ben. Az aktív zajcsökkentés alapelve egyébként nem csak akusztikus, hanem például mágneses (ld. aktív mágneses árnyékolás) és mechanikus rezgések kioltására is felhasználható [3].

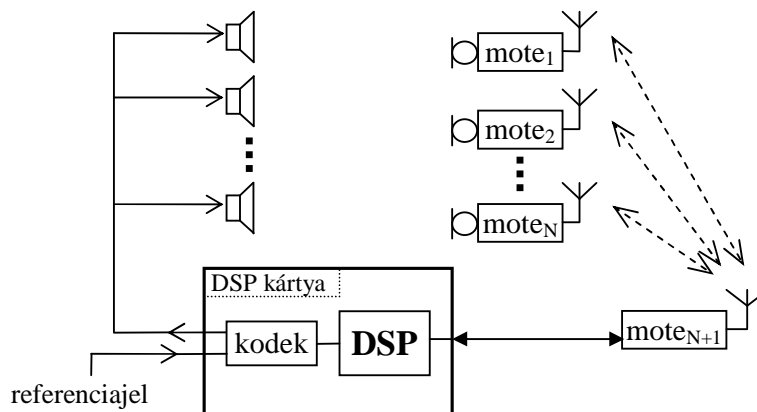
A zajcsökkentés megfelelő hatékonyságához több hangszóró és mikrofon is szükséges. A rendszer kiépítésében nagy kötöttséget jelent a mikrofonok elhelyezése, hiszen azokat mindenképpen oda kell felszerelni, ahol zajcsökkentést szeretnénk elérni. Ennek oka, hogy a zajcsökkentés csupán a mikrofonok környezetében biztosítható. A tartomány mérete az elnyomandó zaj hullámhosszának körülbelül negyede, ezen kívül a zaj és ellenzaj ekár erősítheti is egymást. A mikrofonok telepítésében nehézséget okozhat a jelvezetékek illetve a jelkondicionáló elemek elhelyezése. Ez kínál lehetőséget a szenzorhálózatok alkalmazására, ugyanis azokban az adatok továbbítása rádiós úton, vezeték nélkül történik, így az érzékelők telepítése egyszerűbb, struktúrájuk egyszerűbben átkonfigurálható.

A szenzorhálózatok tulajdonképpen olyan mérésadatgyűjtő rendszerek, melyeknek elemei (mote-ok) rádiós csatornán továbbítják az adatokat. Alkalmazásukra számos példát találhatunk az ipari, katonai és civil szférában [4]. A hagyományos szenzoroktól eltérően az egységek bizonyos alapvető feldolgozási feladatokat is el tudnak látni, képesek az összegyűjtött adatok ideiglenes tárolására is. Alkalmazásuk során problémát jelent azonban a rádiós adattovábbítás miatt fellépő bizonytalanság az adatáramlásban. Ez különösen fontos az olyan visszacsatolt struktúrák esetén, mint például az aktív zajcsökkentő rendszerek. Mivel a szenzorhálózat a visszacsatoló ágban található, ezért a rendszer instabilitását is okozhatja nem megfelelő adattovábbítás esetén. Emiatt a szenzorhálózatok szabályozási rendszerekben hibajelérzékelőként történő alkalmazása még nem igazán elterjedt [5]. Az adattovábbításra szolgáló rádiós csatorna sávzélessége ráadásul általában néhány tíz vagy száz kbps (kilobit per second), amit igen nagy hatékonysággal kell kihasználnunk, hiszen az elnyomandó akusztikus zaj pillanatértéke elég gyorsan változik. A real-time működés megköveteli az adatok igen megbízható továbbítását.

## Rendszerterv

Bár léteznek sztochasztikus jelek elnyomására alkalmas algoritmusok, a periodikus zajok elnyomása terén elért eredmények jóval biztatóbbak, ezért egy periodikus jelek elnyomására alkalmas rendszert építettünk ki. Ennek felépítése az 1. ábrán látható. Működése a következő: a szenzorhálózat  $mote_{1...N}$ -el jelölt elemei végzik a zajérzékelést. Az összegyűjtött adatokat rádióan keresztül továbbítják a hálózat és a DSP között átjáróként szolgáló  $mote_{N+1}$ -hez (bázisállomás). Az átjáró mote és a DSP között a kommunikáció soros porton keresztül történik 115.2 kbps sebességgel. A DSP a mote-ok felől a zajt leíró információk és a referenciajel alapján kiszámítja a beavatkozó jeleket és kiadja a hangszórók

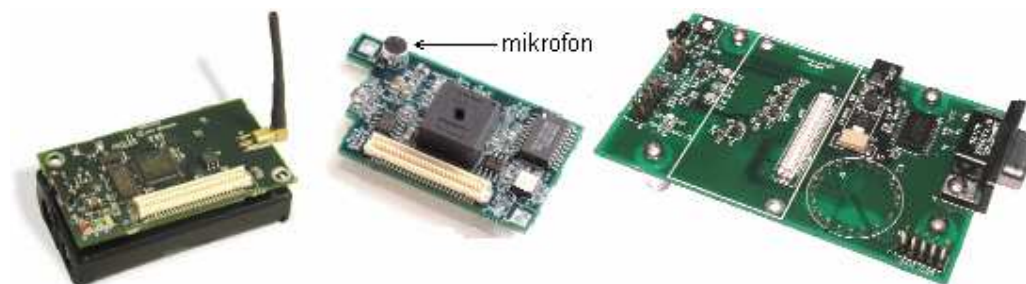
felé. A referenciajel beolvasása illetve a beavatkozó jelek kiadása a DSP-hez kapcsolódó kodeken található  $\Sigma$ - $\Delta$  sztereo AD- és DA- átalakítók segítségével történik.



1. ábra. Zajcsökkentő rendszer felépítése

A rendszer két fő hardverkomponenst tartalmaz. A jelfeldolgozó funkciót egy ADSP-21061 EZ-KIT Lite fejlesztői kártya [6] valósítja meg. A fejlesztői kártya tartalmaz egy ADSP-21061-es (SHARC) jelfeldolgozó processzort és a hozzá kapcsolódó perifériákat: UART illesztő, boot programot tartalmazó EEPROM, sztereo kodek, és az ahhoz tartozó analóg jelkondicionáló áramkörök, csatlakozók, gombok. A DSP egy 32 bites lebegőpontos processzor, mely 40 MHz-es órajel frekvenciával üzemel. A dedikált műveletvégző egységei, és párhuzamos műveletvégzést illetve memóriakezelést lehetővé tévő architektúra segítségével igen hatékonyan megvalósíthatók rajta jelfeldolgozási algoritmusok. A programfejlesztés assemblyben történt.

A szenzorhálózat építőelemeiként a Berkeley MICAz mote-okat alkalmaztuk [7]. A mote-ok moduláris felépítésűek. Központi elemük tartalmaz egy flash memóriát, egy CC2420-as, ZigBee szabvány szerint működő és 250 kbps adatátviteli sebességre képes rádiós IC-t, egy 8 bites ATmega128-as típusú mikrokontrollert, mely 7.3728 MHz-es órajel frekvencián üzemel, a tápellátást biztosító elemeket, antennát, és egy csatlakozót. A csatlakozón keresztül illeszthetünk a mote-hoz szenzorkártyát, valamint programozókártyát. A szenzorkártya többféle típusú lehet, rendszerünkben egy MTS310-es típusú, mikrofont is tartalmazó szenzorkártyát használtunk. A MIB510 típusú programozókártya segítségével soros porton keresztül programozható PC-ről a hozzá csatlakoztatott mote. A mote és a hozzá illeszthető eszközök a 2. ábrán láthatók.



2. ábra. Berkeley MICAz mote, szenzorkártya és programozókártya

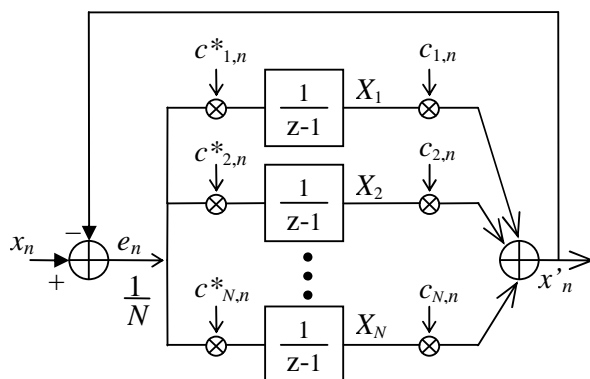
A programozókártya másik funkciója, hogy a rajta található RS-232-es csatlakozón keresztül szabványos jelszinteken hozzáférhetünk a mote-on lévő mikrokontroller UART-jának ki- és bemenetéhez. Az 1. ábrán látható rendszerben a  $mote_{N+1}$  valójában egy programozó kártyára helyezett mote, mely így szabványos RS-232-es soros porton történő

kommunikációra képes, ezért egyszerűen illeszthető a DSP kártyához. A mote-ok programozása NesC nyelven egy TinyOS nevű beágyazott operációs rendszerben történt, mely lehetővé teszi a perifériák egyszerű kezelését. Az operációs rendszer szabadon letölthető, nyílt forráskódú.

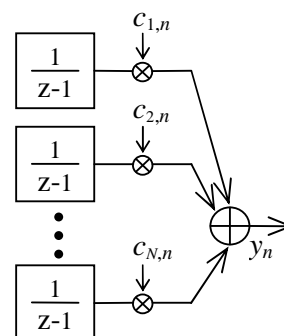
A mote-ok a zaj mintavételezését 1.8 kHz-es mintavételi frekvenciával végzik a mikrokontrolleren található AD-átalakító és időzítő segítségével. Ekkora mintavételi frekvencián még megfelelő az érzékelhető és így elnyomható zaj sáv szélessége. A zajminták egyszerű továbbítása esetén ekkora mintavételi frekvencián tapasztalatok szerint maximum 2-3 mote működtethető. Ennek oka, hogy a mote-ok számának növekedtével nő az adott idő alatt a hálózatban továbbítandó adatmennyiség, a csatorna sáv szélessége viszont korlátozott, ráadásul a keretelés és a szoftveres overhead miatt csupán valamivel több mint 60% kihasználtság érhető el. Ennek megoldásaként előfeldolgozást végzünk a jelen, melynek során a periodikus zajjel Fourier-együtthatóit határozzuk meg. Mivel a Fourier-együtthatók általában lassabban változnak, mint maga a jel, így továbbításuk ritkábban is lehetséges, tehát kevésbé jelent korlátozást a hálózat sáv szélessége. A felbontás az ún. rezonátoros Fourier-analizátor segítségével történik, melynek bemutatására a következő fejezetben, a rezonátor alapú zajcsökkentő algoritmus ismertetésénél kerül sor. Az algoritmus ezen együtthatókat közvetlenül is fel tudja használni.

### Rezonátor alapú zajcsökkentő algoritmus

Ezt a zajcsökkentő algoritmust periodikus jelek elnyomására fejlesztették ki. Alapkonceptiója az, hogy a periodikus zajjelet felbontja harmonikus komponenseire a Fourier-együtthatók rekurzív előállítása révén. Az egyes harmonikusokra épülő szabályozási körök egyenként gondoskodnak azok elnyomásáról, cél a harmonikusok nulla alapjelének követése. A Fourier-együtthatók előállítása a 3.a. ábrán látható rezonátor alapú Fourier-analizátor segítségével történik [8]



3.a. ábra. Fourier-analizátor (FA)



3.b. ábra. Szabályzó blokk (SZAB)

Az ábrán  $X_i$  jelöli a komplex Fourier-együtthatókat, valamint  $c_{i,n} = e^{j \cdot 2 \cdot \pi \cdot i \cdot f_1 \cdot n}$ , amelyek a felbontás alapjául szolgáló bázisfüggvények.  $f_1$  az alapharmonikus frekvenciát,  $i$  a harmonikus számát,  $n$  az időt, \* pedig a komplex konjugáltat jelöli.

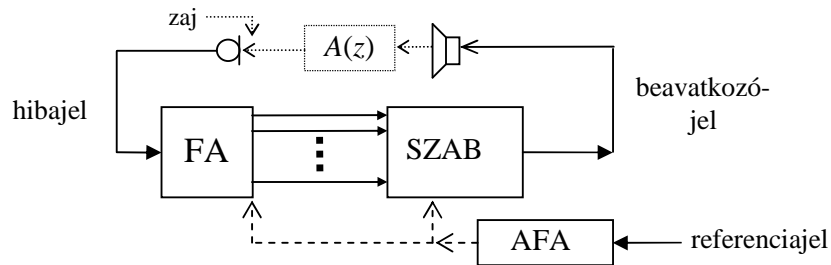
Az analizátor  $N$  db rezonátor csatornát tartalmaz. Egy csatorna működése a következő: az  $i$ . rezonátor a bemenetén lévő jelben található  $i$ . harmonikusust a  $c_{i,n}^*$  forgó egységvektor segítségével DC frekvenciára keveri. A DC jelre az  $\frac{1}{z-1}$  átviteli függvénnyel jellemezhető

integrátor átvitele végtelen. Az integrátor kimenetén egy  $X_i$  komplex szám jelenik meg, melyet a  $c_{i,n}$  forgó vektor visszakeveri az eredeti  $i$ . harmonikus frekvenciájára. Mindebből következik, hogy az  $i$ . rezonátor csatorna  $i$ . harmonikusra vett átvitele végtelen. Mivel a

rezonátorok egy visszacsatolt szabályozási körben találhatóak, így a felbontandó  $x_n$  jelben található  $i$ . harmonikus hiba nélkül képes előállítani, mivel az arra vonatkoztatott nyílthurkú erősítés végtelen. A 3.a. ábra alapján a Fourier-analizátor kimenetét az (1) egyenlet írja le:

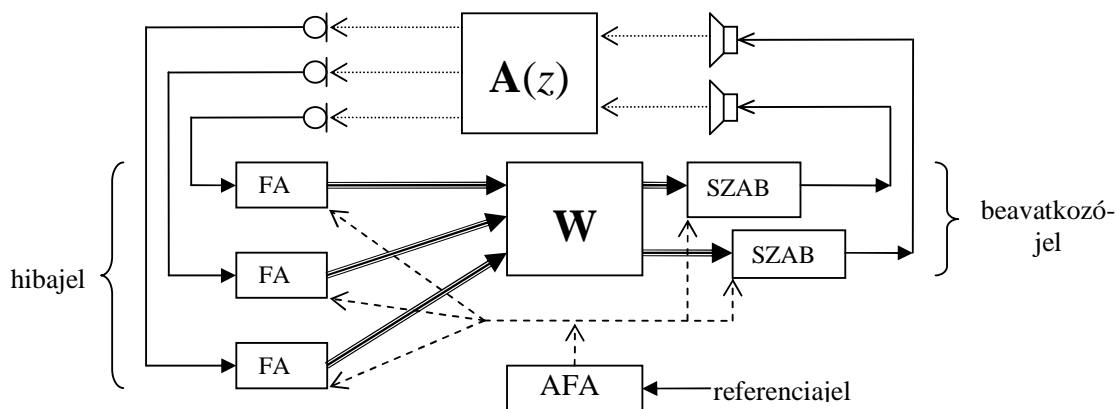
$$x'_n = \sum_{i=-N}^N X_i c_{i,n}, \text{ ahol } c_{i,n} = e^{j \cdot 2 \cdot \pi \cdot i \cdot f_1 \cdot n} \quad (1)$$

Az egyenlet a jelek Fourier-felbontásának alakját adja, melyben  $X_i$  az  $i$ . harmonikus Fourier-együtthatóját jelöli. A felbontás során fontos, hogy a  $c_{i,n}$  függvényekben  $f_1$  megegyezzen a zaj valódi alapharmonikus frekvenciájával ( $f_{1\_zaj}$ ). Erről egy adaptív Fourier-analizátoros struktúra (AFA) [9] gondoskodik. Működésében kihasználjuk, hogy amennyiben a 3.a. ábrán látható FA-ban nem teljesül  $f_{1\_zaj} = f_1$ , akkor az  $X_i$  együttható a frekvenciaeltéréssel arányos sebességgel forog. A zajcsökkentő rendszerben az  $X_i$  együtthatókat a mikrofonokkal érzékelt maradó zajból állítjuk elő. Az  $X_i$ -re épített integrátoros szabályzó blokk (SZAB) a 3.b. ábrán látható. Az  $X_i$  együtthatókat az annak megfelelő  $c_{i,n}$  csatornára vezetjük. A szabályzó blokk kimenetét hangszóróra vezetve adjuk ki az ellenzajt. Ez az elrendezés a 4. ábrán látható [10].



4. ábra. Egycsatornás rezonátoros zajcsökkentő rendszer

Az integrátoros szabályzás miatt teljes elnyomás elérhető. Láthatjuk, hogy a kiadott ellenzaj egy  $A(z)$  akusztikus átviteli függvénnyel módosítva ér a mikrofonhoz. Kimutatható, hogy ha az  $X_i$ -k átcsatolása  $\alpha \cdot w_i$  együtthatóval történik, akkor amennyiben  $w_i$  legalább  $90^\circ$ -ra megközelíti a  $A(z)$  inverzét az  $i$ . rezonátorfrekvencián, akkor  $\alpha$  csökkentésével biztosítható a stabilitás. Optimális esetben  $w_i = A^{-1}(z_i)$ , így eredőben egységnyi az átvitel.  $A(z)$  magában foglal minden, a digitális ki- és bemenetek közti tagot (hangszóró, mikrofon, ADC, DAC, jelkondicionálók). Emiatt  $A(z)$  identifikációja szükséges. A több mikrofont és hangszórót tartalmazó, ún. többcsatornás rendszerek blokkvázlata a 5. ábrán látható.



5. ábra. Többcsatornás zajcsökkentő rendszer

Az egycsatornás rendszerhez képest az jelent különbséget, hogy több FA és SZAB blokk található, melyek között egy  $W$  átcsatoló mátrix teremt kapcsolatot. Optimális esetben:

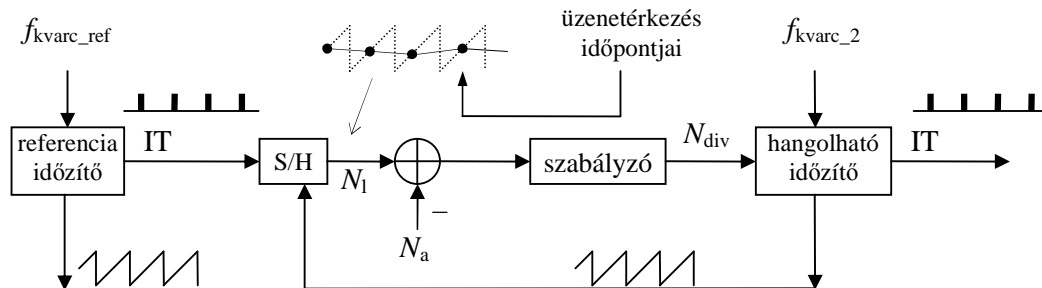
$$\mathbf{W}_i = \mathbf{A}^\#(z_i) \quad (2)$$

ahol  $\#$  a pszeudoinverzest jelöli.  $\mathbf{A}(z)$  itt egy átviteli függvény mátrix, mely a ki- és bemenetek közti kapcsolatot írja le. Minden mikronhoz tartozik egy FA blokk, és minden hangszóróhoz tartozik egy SZAB blokk. Az AFA ebben az esetben is minden FA és SZAB blokk számára biztosítja a  $c_{i,n}$  együtthatókat. Az általunk megvalósított zajcsökkentő rendszerben az FA blokkok a mote-okon találhatók, melyek a bázisállomás segítségével továbbítják az  $X_i$  együtthatókat. Az AFA virtuálisan szintén a bázisállomás segítségével kapcsolódik a mote-okon futó FA blokkokhoz. A zajcsökkentő rendszer többi része a DSP-n foglal helyet.

### Hálózati felépítés és szinkronizáció

A hálózat időosztásos (TDM) rendszerben működik. A bázisállomás által küldött szinkronüzenetet követően minden mote kap egy egyedi időszeletet, melyben az előállított  $X_i$  együtthatókat elküldheti a bázisállomásnak, mely továbbítja ezeket a DSP felé. A szinkronüzenetet a bázisállomás periodikusan küldi olyan gyakorisággal, hogy minden mote számára legyen elegendő idő a kommunikációra. A szinkronizációs üzenet a mintavételi időpontok és a FA blokkokban található  $c_{i,n}$  együtthatók szinkronizálását is szolgálja. A  $c_{i,n}$ -k szinkronizálása azért szükséges, mert ez szolgál viszonyítási alapként az egész rendszerben. A szinkronizáláshoz elegendő az alapharmonikus rezonátor fázisát és frekvenciáját ismerni, a többi ebből származtatható. A mintavétel szinkronizálása azért szükséges, mert habár a névleges mintavételi frekvenciák megegyeznek, a mintavételezésért felelős időzítőket működtető kvarcoszcillátorok által meghatározott órajel-frekvenciák eltérnek a névleges értékeiktől, így a mintavételi időpontok elcsúsznak a névleges adatokból adódó időpontokhoz képest. Az elcsúszás nem megijósolható, ennek következtében viszont nem tudjuk, hogy a mote-ok által érzékelt zaj fázisa milyen valóságos fázishelyzetnek felel meg, ami az  $A(z)$  átviteli függvény bizonytalanságát okozza, tehát instabillá teszi a rendszert. A mintavételezést ezért szinkronizálni kell. Szinkronizációs referenciaként a bázisállomás szolgál.

A szinkronizációs algoritmus a mote-okon található időzítő működését ismerve válik érthetővé. Az időzítőben egy számláló értéke növekszik az órajel ütemében, mígnem eléri a mintavételi frekvenciát meghatározó, szabadon változtatható  $N_{div}$  értéket. Ekkor a számláló nullázódik, és ismétlődik a folyamat. Ez tehát egy programozható órajelosztó. A számláló időfüggvénye egy fűrészjel, melynek lefutó élénél megszakítás (IT) generálódik, ennek hatására pedig megtörténik a jel mintavételezése. Amennyiben az egyes mote-okon létrejövő fűrészjeleket fázismerev kapcsolatba hozzuk, akkor létrejön a szinkron mintavételezés. Erre egy PLL-szerű algoritmus szolgál, melynek blokkdiagramja a 6. ábrán látható.



6. ábra. Szinkronizációs algoritmus

A PLL egyes részeit megvalósítva a PLL elvből adódóan a struktúra biztosítja a szinkronizálást. A fázisdetektor funkció az időzítő számlálójának a szinkronizációs üzenet megérkezésekor történő kiolvasásával valósul meg (S/H elem). A kiolvasott érték arányos a fűrészjelek fáziskülönbségével, mivel az üzenetet a bázisállomás a mintavételi időpontok

alkalmával küldi. Amennyiben a kiolvasott értéket állandó  $N_a$  szinten tarjuk, akkor létrejön a fáziszárt állapot. Ennek biztosítása a VCO funkciót megvalósító hangolható időzítő osztásviszonyának, tehát az  $f_s$  mintavételi frekvencia megváltoztatásával lehetséges. Ezt a szabályzó blokk végzi. Ha  $N_1 < N_a$ , akkor  $f_s$ -t növeli ( $N_{div}$ -et csökkenti), ellenkező esetben  $f_s$ -t csökkenti ( $N_{div}$ -et növeli).

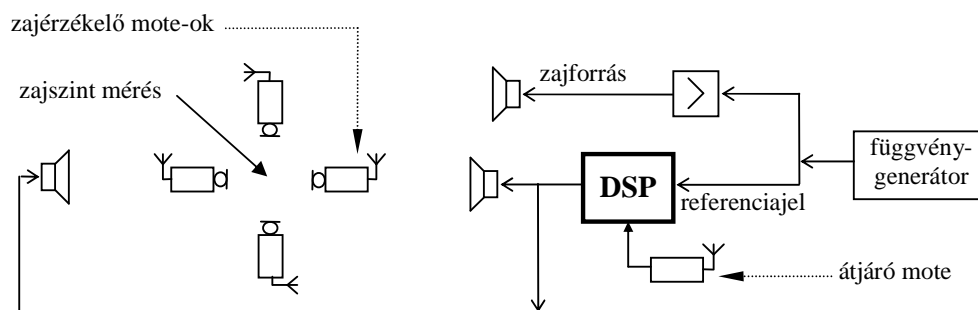
A rezonátorok szinkronizálása két szinten történik. Először a bázisállomás szinkronizálódik a DSP-n lévő rezonátorokhoz, aztán pedig szinkronizálja magához a többi mote-ot. Ez utóbbi az alapharmonikus-fázis és -frekvencia egyszerű elküldésével történik. Mivel a szinkronüzenet küldését a bázisállomás a mintavételi időpontok elején, tehát a  $c_{l,n}$  fázisának frissítésekor küldi, felhasználásuk a többi mote-on szintén a mintavételi időpontok elején történik, így az üzenet továbbítása miatt fellépő késleltetés mindig állandó, mivel a mintavételi időpontok szinkronizálva vannak. A késleltetés miatt fázistolás lép fel, ez azonban állandó, és benne foglaltatik  $\mathbf{A}(z)$ -ben, tehát  $\mathbf{W}$  mátrix kompenzálja azt. A szinkronizáció nem történhet minden mintavételi időpontban, de a frekvencia ismeretében a fázis a szinkronizációt követő időpontokban is előállítható.

A bázisállomás és DSP közötti szinkronizáció esetén ez a módszer nem működik, mivel azok mintavételi frekvenciája eltér. Ebben az esetben a bázisállomás periodikusan szinkronizációt kezdeményező üzenetet küld. A DSP kiszámítja az üzenet érkezésekor az alapharmonikus fázisát, amelyet az alapharmonikus-frekvenciával együtt továbbít a bázisállomás felé. Mivel a szinkronizációt kezdeményező üzenet elküldése és a szinkronizációs üzenet megérkezése között eltelt idő állandó, így az előző gondolatmenet szerint a kommunikáció miatti késleltetés nem okoz problémát.

## Teszteredmények

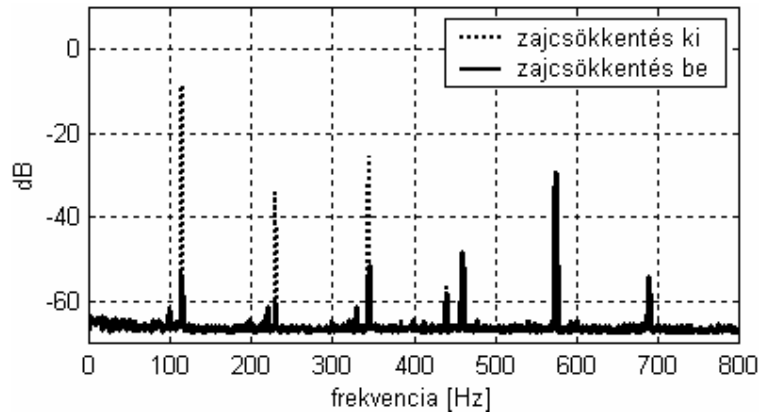
Az ismertetett elvek alapján megvalósított rendszer 2 hangszórót és 4 mikrofont tartalmaz. A hangszórók számát a DSP-hez kapcsolódó kodek sztereo kimenete korlátozza kettőre. A mote-ok száma elvileg akár a 10-20-at is elérheti, korlátot a DSP számítási kapacitása, valamint a Fourier-együtthatók változási gyorsasága szab. Utóbbi nagymértékben alkalmazásfüggő, függ az elnyomandó zaj tulajdonságaitól.

A 7. ábrán egy olyan mérési elrendezés látható, melyben a négy zajérzékelő mote egy kb. 20 cm oldalhosszúságú négyzet csúcspontjaiban helyezkedik el, a két zajlenyomó hangszóró pedig a négyzet két szemközti csúcsától 40 cm-re található. A mérést a mote-ok között elhelyezett mikrofon segítségével végeztük.



7. ábra. Mérési elrendezés

Zajként egy 105 Hz-es háromszögjelet alkalmaztunk, melyet egy függvénygenerátor állított elő, és egy hangszóró segítségével alakítottuk akusztikus zajjá. Referenciajelként a generátor kimeneti jelét használtuk. Ennek segítségével határozza meg az AFA a zaj frekvenciáját. A 8. ábrán látható spektrum alapján megállapítható, hogy alapharmonikusra 40 dB elnyomást értünk el, második és harmadik harmonikusra pedig 20 dB-t.



8. ábra. Zajelnyomás periodikus jelre

## Értékelés

Az itt bemutatott alkalmazás jó példát nyújt arra, hogy a jelek előfeldolgozása milyen előnyöket jelenthet. Esetünkben ez az érzékelők számának növelését teszi lehetővé a mintavételi frekvencia csökkentése nélkül. Előnyként jelenhet még meg, hogy az elosztott feldolgozás valamelyest tehermentesíti a központi feldolgozó egységet.

Jövőbeni tervek között szerepel egy gyakorlatban is jól használható nagyméretű rendszer kifejlesztése, mely az eddigénél több mikrofont és hangszórót tartalmaz.

## Irodalomjegyzék

- [1] Kuo, S. M., D. R. Morgan: „Active Noise Control: A Tutorial Review”, *Proceedings of the IEEE*, vol. 87. No. 6., pp. 943-973, June. 1999.
- [2] Sujbert L., „Több mikrofon és hangszóró: a periodikus zajelnyomás újabb eredményei”, *Elektronet* 1996. április
- [3] Elliott., S. J.: „Distributed Control of Sound and Vibration”, in 2004 *Int. Symp. On Active Noise Control of Sound and Vibration*, ACTIVE '04, pp 1-25, Williamsburg, Virginia, Sept. 2004
- [4] Akyildiz, I. F., W. Su, Y. Sankarasubramaniam, and E. Cayirci: „Wireless sensor networks: A survey”, *Comput. Netw.*, vol. 38, no. 4, pp. 393-422, 2002
- [5] Mathiesen, M., G. Thonet, N. Aakwaag: „Wireless ad-hoc networks for industrial automation: a current trends and future prospects”, in *Proceedings of the IFAC World Congress*, Prague, Czech Republic, July 4-8, 2005
- [6] „ADSP-2106x SHARC EZ-KIT Lite Reference Manual”, Analog devices Inc. P.O Box 9106 Norwood, 1997
- [7] [www.xbow.com/Support/Support\\_pdf\\_files/XBOW\\_Smart\\_Dust\\_ProductInfoGuide.pdf](http://www.xbow.com/Support/Support_pdf_files/XBOW_Smart_Dust_ProductInfoGuide.pdf)
- [8] Péceli, G.: „A common structure for recursive discrete transforms”, *IEEE Trans. Circuits Syst.* Vol. CAS-33, pp.1035-1036, Oct. 1986.
- [9] Nagy, F.: „Measurement of signal parameters using nonlinear observers”, in *IEEE Trans. on Instrumentation and Measurement*, vol. IM-41 No. 1. pp. 152-155, Feb.1992
- [10] Sujbert, L., G. Péceli, „Periodic noise cancellation using resonator based controller”, in *1997 Int. Symp. on Active Control of Sound and Vibration*, ACTIVE '97, pp. 905-916, Budapest, Hungary, Aug. 1997