

Aktív zajcsökkentést demonstráló rendszer továbbfejlesztése

Csofcsics Donát
Gusztáv Tamás

Előző félévben végzett munka

- Szenzorhálózattal működő többcsatornás zajcsökkentő rendszer megismerése
 - Berkeley mote-ok
 - Analog Devices ADSP21065 KIT
- DSP programozási környezet megismerése
- Kísérletezés szabadtéri zajcsökkentésre alkalmas összeállításokkal

Új célkitűzések

- A régi környezet instabilitása nehezkesse tenné a továbblépést
- Egy diplomamunka keretében született egy zajcsökkentést demonstráló eszköz, mely az egyszerű használathoz továbbfejlesztést kívánt

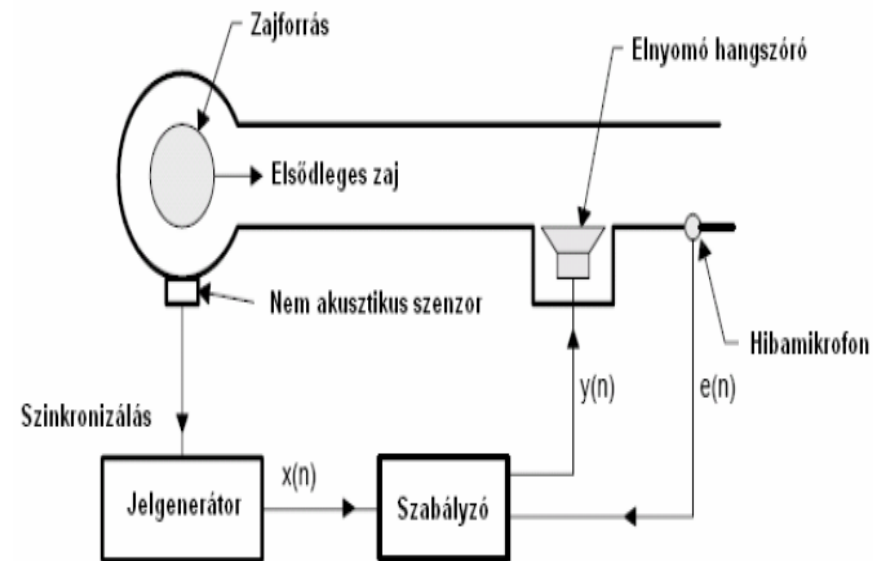
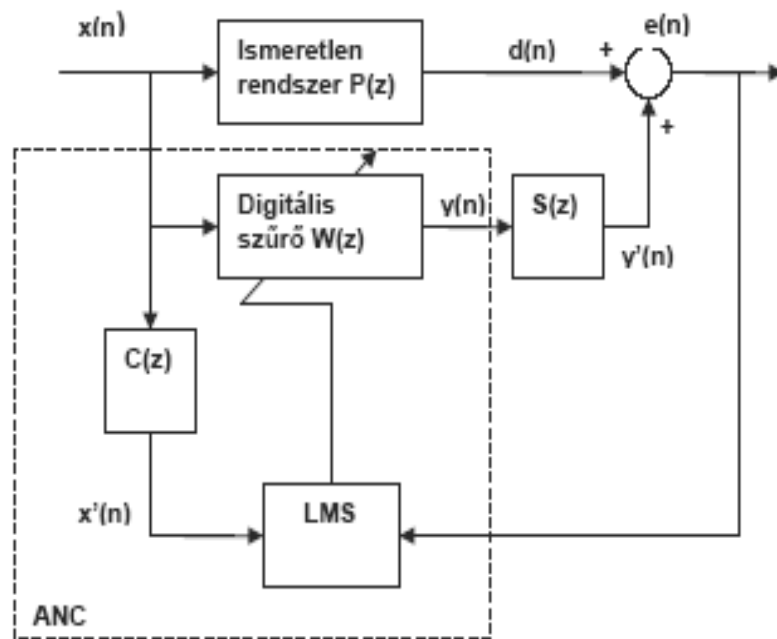
A rendszer leírása

- Egyszerű zajcsökkentő rendszer
 - Zajcsökkentés csőben
 - A zajt az egyik hangszóró biztosítja
 - Elnyomás: egy szemben elhelyezett hangszóró segítségével
 - Szellőzőrendszer, kipufogó zajcsökkentésének modellezése



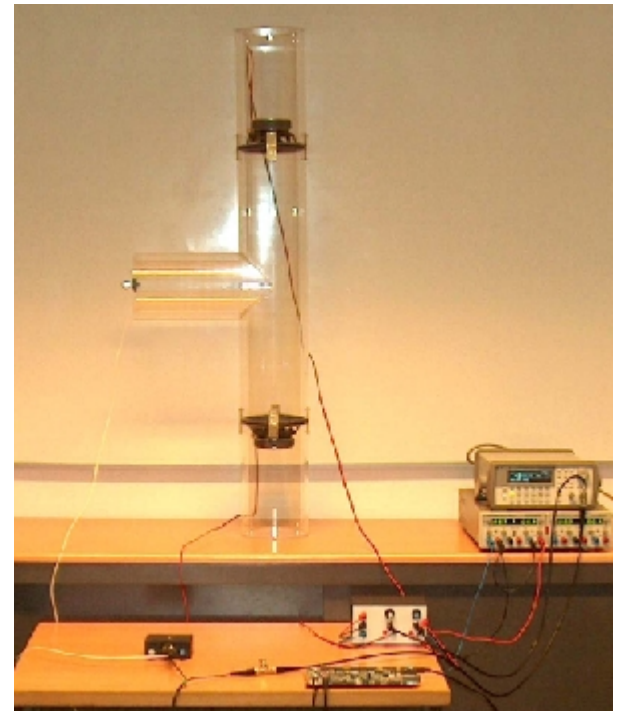
Zajcsökkentés megvalósítása

- Keskenysávú előrecsatolt zajcsökkentő rendszer FXLMS algoritmussal



Eredeti rendszer elemei

- Zajtér: a cső + két hagszoró
- Erősítő + tápegység
- Mikrofon + erősítő
- ADSP BF537 EZ-KIT Lite
- Zajforrás (függvénygenerátor)



Feladat megfogalmazása

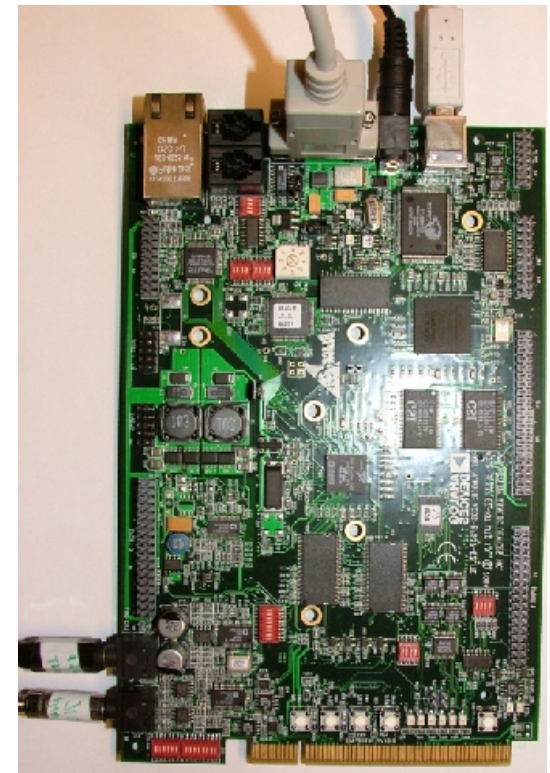
- Könnyen használható számítógépes kezelői felület biztosítása
- Teljeskörű vezérlés számítógépről
- Bootolás flash-ből, VisualDSP-s program letöltés kiküszöbölése

Félév során elvégzett fejlesztések I.

- Megoldottuk a DSP program vezérlését számítógépről

Opciók:

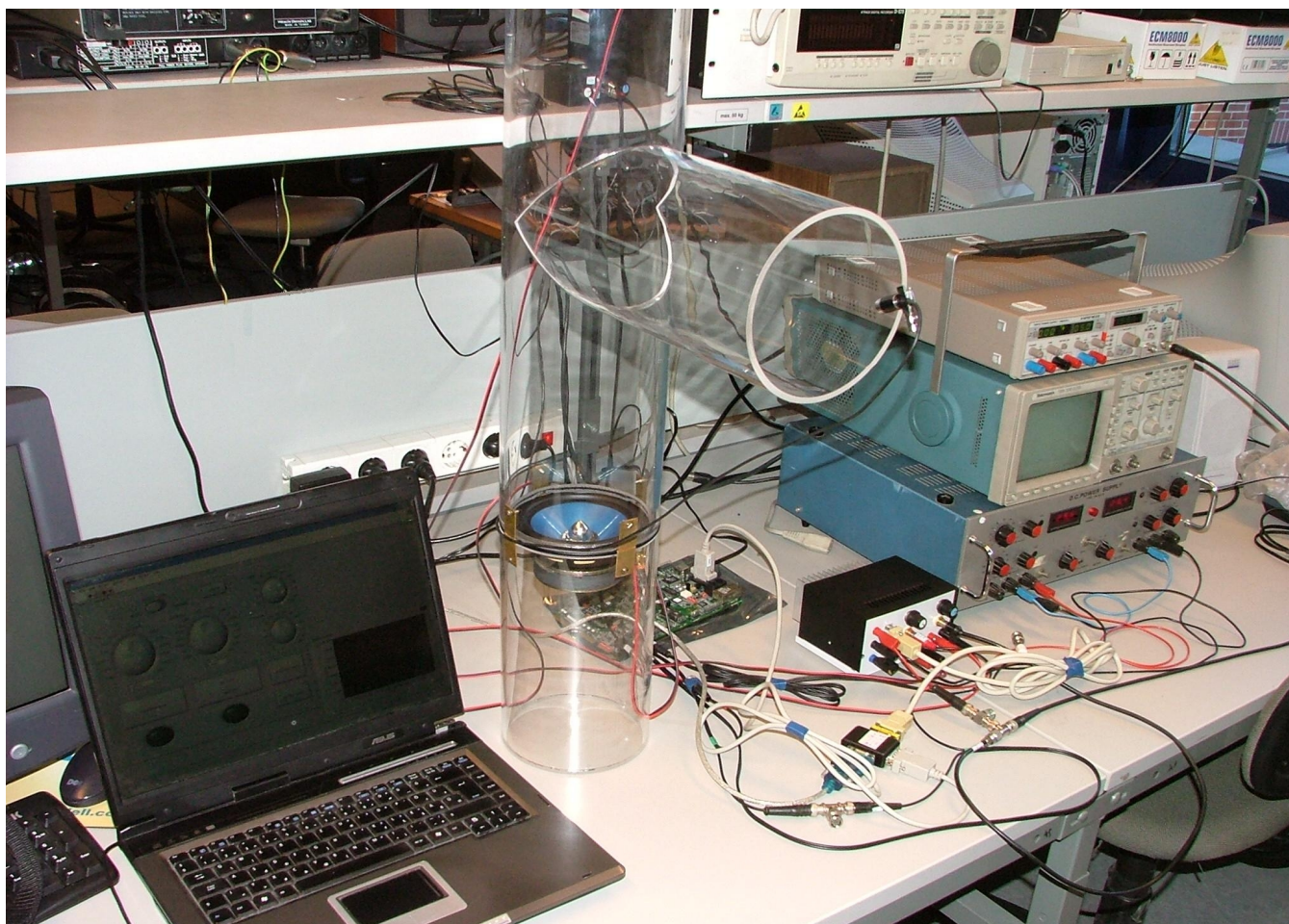
- Identifikáció be/kikapcsolása
- Identifikációs μ változtatása
- Zajcsökkentés be/kikapcsolása
- Zajcsökkentő μ változtatása
- Adaptív szűrőegyütthatók nullázása
- Másodlagos-út együtthatóinak továbbítása számítógépre és megjelenítése



Félév során elvégzett fejlesztések II.

- DSP program betöltése flashből
- Az eredeti erősítő hangerőszabályzása számítógépről
- Előbbiekhez szükséges kiegészítő eszközök elkészítése
 - Digitális potméter kártya
 - Soros port elosztó (ismétlő)

Általunk továbbfejlesztett rendszer



Felhasznált szoftverfejlesztő eszközök

- Visual DSP++
- LabView
- WinAvr

DSP program fejlesztése I.

- Eredeti funkciók
 - Identifikáció
 - Zajcsökkentés
 - Funkcióváltás: Kártyán lévő gombok segítségével, nehézkes állapotgéppel történt : kellene egy gyorsan átlátható felület

DSP program fejlesztése II.

- Új funkciók megvalósítása:

Két lehetőség:

- VisualDSP-n keresztül, JTAG interfészen változtatni a megfelelő paramétereket
 - Előnyök:
 - Labview támogatás
 - Nem kell módosítani a DSP programot
 - Hátrányok:
 - Minden beavatkozáshoz meg kell állítani a processzort
 - Mindenképpen szükség van a VisualDSP-re

DSP program fejlesztése III.

- Soros porton keresztüli vezérlés
 - Előnyök:
 - Futás közben lehet módosítani paraméterek, figyelni változókat
 - Csak egy soros port kell és a felhasználói program
 - Hátrány:
 - Ki kell egészíteni a DSP programot

Választás: Soros port

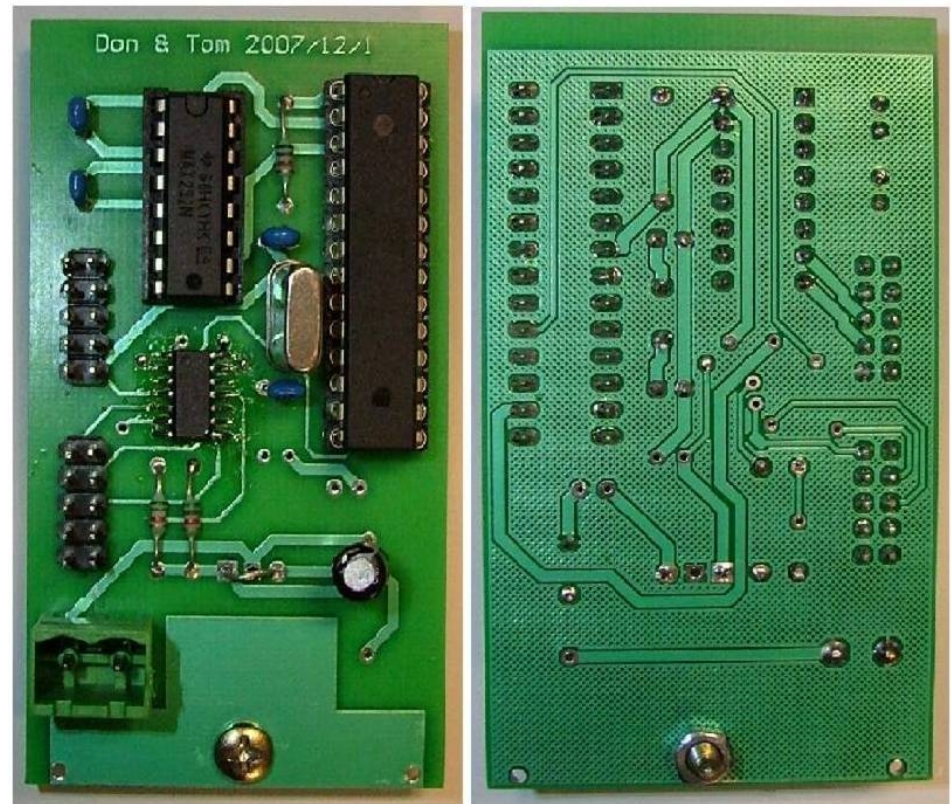
- Soros port kezelés megszakítással
 - Receive interrupt fogadja a beérkező parancsokat
 - Adatok visszaküldését a PC felé Transmit interrupt végzi

Digitális potméter kártya I.

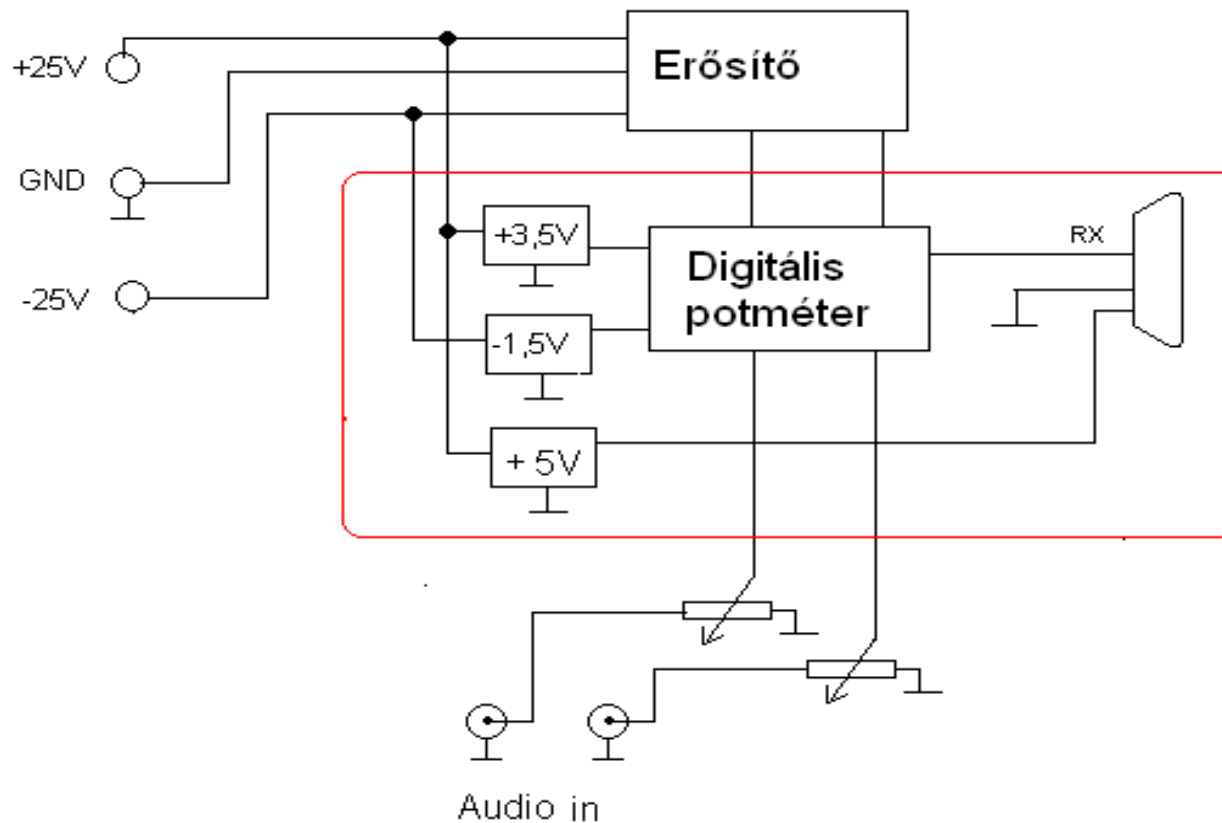
- Cél:
 - A beavatkozó hangszóró és a zajforrás hangerejének szabályozása soros porton keresztül
 - Az erősítő felépítésében a lehető legkisebb módosítás végrehajtása, az eredeti analóg potméter legyen ugyanúgy használható

Digitális potméter kártya II.

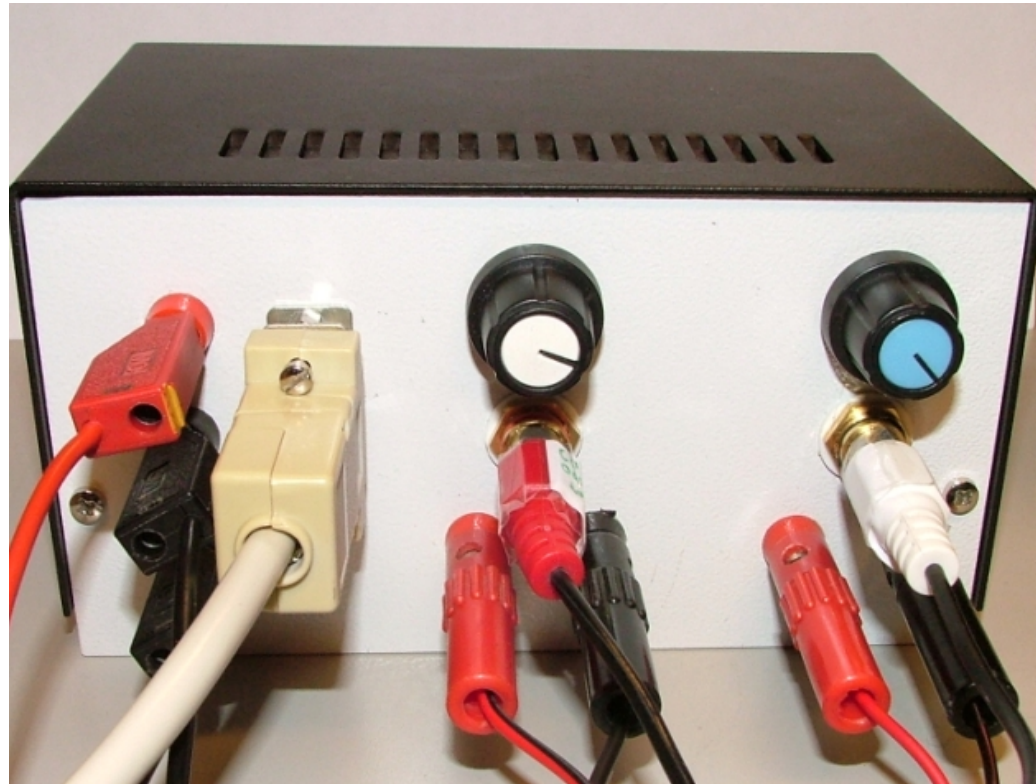
- MCP42010: két digitális potméter 3 terminállal
- ATmega8 mikrokontroller
- MAX232 soros port szintillesztő



Digitális potméter kártya III.

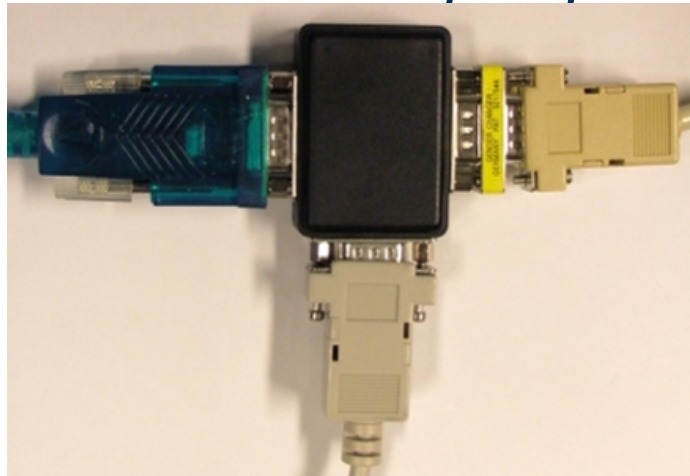


Az erősítő kiegészítve a soros porttal



Soros port ismétlő

- Az RS232 szabvány szerint 1 DTE 1db 5k Ω bemeneti impedanciájú DCE-t vagy DTE-t tud meghajtani.
- Kell egy eszköz ami továbbítja a jelet mindkét periféria felé:

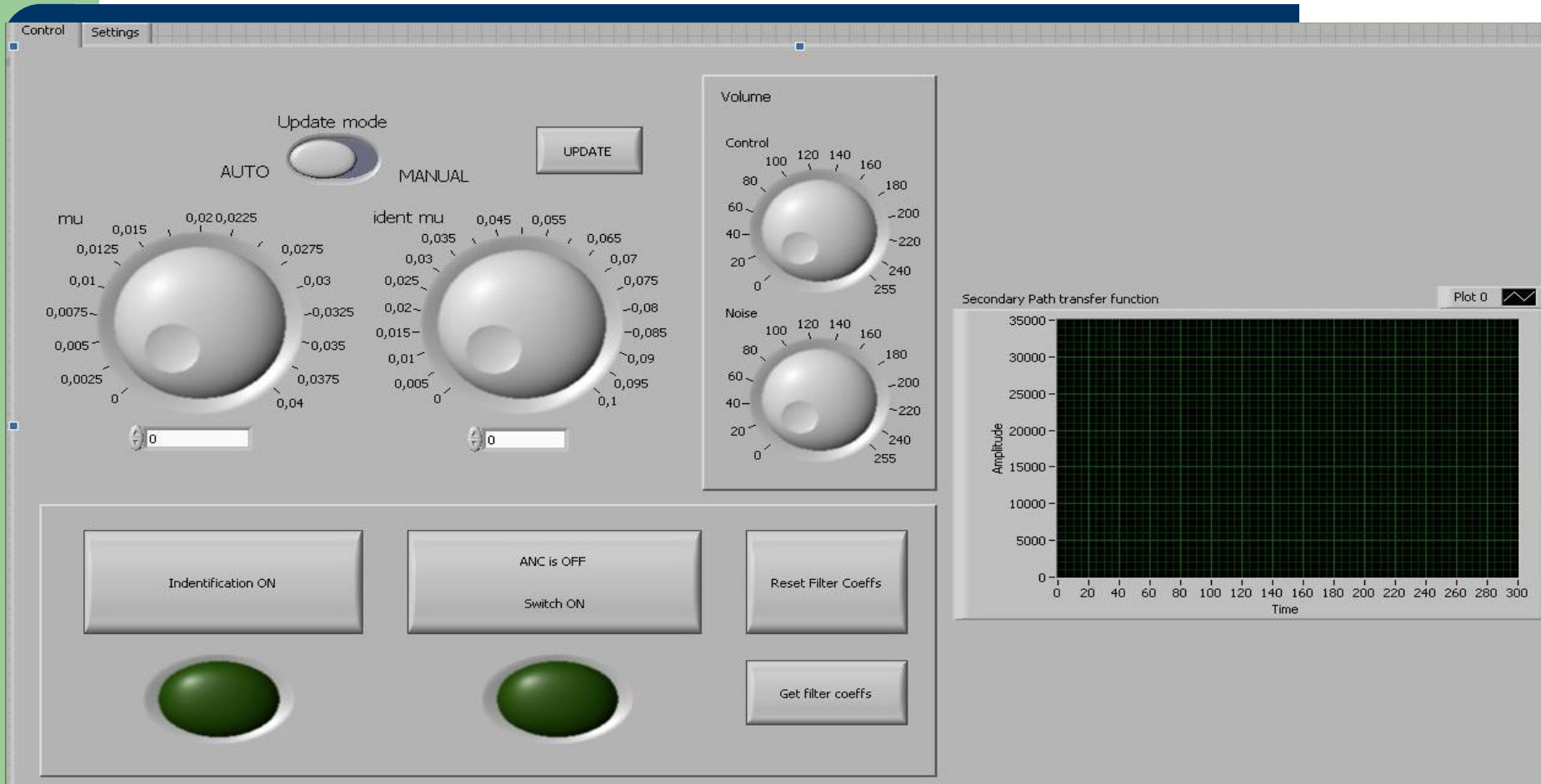


DCE: Data communication equipment DTE: Data terminal equipment

Felhasználói felület

- Szoftverválasztás: Labview
 - Egyszerű soros-port kezelés
 - Feladathoz jól illeszkedő kezelőfelület elemek
 - (VisualDSP támogatás)
- Követelmények:
 - Minden funkció legyen könnyen elérhető
 - Illegális művelet ne legyen végrehajtható

Vezérlő felület LabView-ban I.



Vezérlő felület LabView-ban II.

- Program felépítése
 - Inicializálás (soros port, belső változók)
 - Műveletvégzés (vezérlés, kommunikálás)
 - Soros port felszabadítása, alaphelyzet visszaállítása

Műveletvégző keret

- Felhasználói beavatkozásokat eseményvezérelten kezeltük le
- Esemény hatására a megfelelő parancsokat továbbítjuk az eszközök számára
- Kért adatokat fogadjuk a DSP felől

Kommunikációs protokoll

- A vezérlő program fix 4 byte hosszúságú üzenetekkel kommunikál a soros porton 9600baud sebességgel:



- A parancsbyte dönti el, hogy az adott üzenet a DSP-nek vagy az erősítőnek szól

Programletöltés a flashbe

- Probléma, hogy minden indításnál be kell tölteni a programot a DSP-be
- Megoldás: Bootoljunk az onboard flashból
Ehhez szükséges lépések:
 - Boot-Mode kiválasztása
 - A flashdriver program betöltése a DSP-be
 - A zajcsökkentő forrásfájlokból loader-fájl generálása
 - A flashdriver segítségével a fájl beírása a flash-be
 - Reset után a processzor a flashból bootol

Elért eredmények

- Gyorsan és egyszerűen összerakható, egy vezérlő programot igénylő rendszer
 - Mindössze a LabView kódból generált .exe fájlra van szükség
- Egyszerű hozzáférés a belső változókhoz futás közben is

További lehetőségek

- A hibajel amplitúdójának figyelemmel kísérése a kezelői felületen
- A mikrofon túlvezérlése esetén a hangerő automatikus állítása