



M Ű E G Y E T E M 1 7 8 2

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

Somogyi Kristóf

**JELSZINTÉZIS ÉS JELANALÍZIS  
SZOFTVERDEFINIÁLT  
ALAPOKON**

KONZULENS

Krébesz Tamás István

BUDAPEST, 2019

# Tartalomjegyzék

<b>Összefoglaló .....</b>	<b>5</b>
<b>Abstract.....</b>	<b>6</b>
<b>1 Bevezetés .....</b>	<b>7</b>
<b>2 LabVIEW és a grafikus programozás .....</b>	<b>8</b>
2.1 Adatfolyam-programozás .....	8
2.2 Függvény és VI.....	8
2.3 Pozitívumok .....	10
2.4 Negatívumok.....	11
<b>3 A műszervezrlés bemutatása.....</b>	<b>12</b>
3.1 A távvezérelt műszerekről általánosan .....	12
3.2 A DAQ kártya vezérlése .....	13
3.2.1 Az eszköz specifikációja.....	13
3.2.2 A bemenet vezérlése .....	15
3.2.3 A kimenet vezérlése.....	16
3.3 Az oszcilloszkóp képernyőképének mentése.....	17
<b>4 A DAQ kártya vizsgálata .....</b>	<b>19</b>
4.1 Az eszköz megismert határai .....	19
4.2 Összehasonlítás .....	22
<b>5 A mintavételezés vizsgálata.....</b>	<b>24</b>
5.1 Koherens és inkohere ns mintavételezés vizsgálata.....	24
5.1.1 A koherens mintavételezés feltétele .....	24
5.1.2 Az inkohere ns mintavételezés következménye: a spektrális szivárgás .....	25
5.2 A mintavételezési információk megadása .....	27
<b>6 Az aluláteresztő szűrő.....</b>	<b>30</b>
6.1 A szűrő modell.....	30
6.2 Szűrő tervezés .....	31
6.3 Szűrő realizálás .....	32
6.4 Hibaanalízis .....	34
6.4.1 $A_0$ worst case hibája.....	34
6.4.2 $\omega_0$ worst case hibája.....	35
6.5 Stabilitásvizsgálat .....	36

6.5.1 A zárt rendszer pólusai alapján.....	36
6.5.2 A Nyquist stabilitás vizsgálata.....	37
6.5.3 A fázistartalék számításával.....	38
6.6 Szimuláció .....	39
<b>7 Az átviteli függvény automatizált felvétele.....</b>	<b>41</b>
7.1 Hasonlóságok és különbségek .....	41
7.2 Szinuszos gerjesztés.....	42
7.3 Négyszög gerjesztés.....	43
7.4 Multitone gerjesztés .....	46
7.5 Összegzés.....	47
<b>8 A tervezett és implementált szűrő összehasonlítása.....</b>	<b>48</b>
8.1 Prognózis .....	48
8.2 Az implementált szűrő átviteli függvénye .....	48
8.3 Értékelés.....	48
<b>9 Kitekintés: Frekvencia és fázis becslés.....</b>	<b>51</b>
<b>10 Függelék.....</b>	<b>54</b>
10.1 Az oszcilloszkóp képernyőképének mentése – blokk diagram.....	54
<b>Irodalomjegyzék.....</b>	<b>55</b>
<b>Ábrajegyzék.....</b>	<b>56</b>

# HALLGATÓI NYILATKOZAT

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

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

Kelt: Budapest, 2019. 12. 02.

.....  
Somogyi Kristóf

# Összefoglaló

A szoftverdefiniált műszerezés magában hordozza a digitális megközelítés azon előnyét, hogy egyetlen hardver felhasználásával akár több alkalmazás is megvalósítható. Tehát egyetlen univerzális eszközzel több, tetszőleges funkciókkal ellátott műszert is realizálhatunk, kiváltva ezzel a hagyományos műszereket.

Szakedolgozatomban bemutatom a virtuális műszerek létrehozásához szükséges programozói elveket és a virtuális műszerek felépítését. Ismertetem továbbá, hogyan kell szoftveresen műszert vezérelni egy általános célú mérési adatgyűjtő (DAQ) kártya és egy oszcilloszkóp alkalmazásával. Megvizsgálom mire képes az előbb említett kártya jelgenerálás és jelanalízis szempontjából, illetve milyen mintavételezési problémákba ütközhetünk. Bebizonyosodik, hogy bizonyos frekvenciatartományon túl már problémás a jelgenerálás, ezért ennek javítása érdekében egy szűrő áramkörrel egészítem ki a rendszert. Végül, a kimeríthetetlen lehetőségek tárházának demonstrációja képpen, megalkotok egy egyedi műszerfunkciót, amely képes automatizáltan felvenni a megépített szűrő átviteli karakterisztikáját.

## **Abstract**

Software defined instrumentation holds the advantage of the digital approach, that multiple applications can be implemented by using only one hardware. Thus, with one universal device, many measurement instruments can be realized with freely chosen functions, to replace the conventional measurement instrumentation.

In my thesis work the programming paradigm and basic elements of creating a virtual instrument will be introduced. I will inspect the method of controlling an instrument by software applying a data acquisition (DAQ) device and an oscilloscope. Later, the capabilities of the aforementioned DAQ device in the aspects of signal generation and analysis, together with the problems of sampling will be examined. It will be turn out, that the signal generation is problematic over a certain frequency range, so in order to overcome this a filter circuit will be applied. Finally, as a demonstration of the vast number of opportunities, a unique function, which can determine the transfer characteristics of the previously built filter will be implemented.

# 1 Bevezetés

Mára villamosmérnöki körökben is lehetővé vált a szoftver tervezés rétegzett kezelhetősége. Ennek köszönhetően csupán a szoftver réteg kicserélésével több, különböző funkcionalitást is biztosíthatunk egyetlen hardver felhasználásával.

Ezzel párhuzamosan a jelfeldolgozás területén egyes analóg módszereket felváltottak a digitális megfelelők, mivel ezek pontosabbak, flexibilisebbek és az analóg rendszerekkel ellentétben nem igényelnek rendszeres kalibrálást [1].

A szoftverdefiniált műszerezés koncepciója magába foglalja mind a digitalizálás, mind a komponens szétválasztás előnyeit. Egy hardvert, mint műszermagot felhasználva valósítunk meg digitális műszereket szoftveres úton. Egyetlen eszközzel és a megfelelő fejlesztőkörnyezettel, a hardver korlátjain belül mozogva képesek vagyunk a hagyományos, fizikai műszerekkel azonos, vagy akár teljesen tetszőleges funkcionalitású műszereket létrehozni és azok működését automatizálni.

Az elképzelés sok lehetőséget rejt magában, mivel több műszert is helyettesíthetünk egyetlen eszközzel, ezzel pénzt és helyet spórolva. Továbbá rajtunk áll, milyen szoftvert készítünk, így teljesen egyéni, a feladathoz testre szabott műszert állíthatunk elő, amely akár olyan funkciókkal is rendelkezhet, amilyenekkel egyetlen másik, a piacon megtalálható társa sem.

Valamint alkalmunk nyílik a hagyományos műszerek távvezérlésére és automatizálására. Ennek köszönhetően például szoftveres kezelőfelületet biztosíthatunk fizikailag nehezen hozzáférhető műszerek számára vagy akár többlépéses mérési folyamatokat végezhetünk el egyetlen gombnyomásra.

## 2 LabVIEW és a grafikus programozás

A LabVIEW szoftver, azaz a Laboratory Virtual Instrument Engineering Workbench egy rendszer tervező platform és fejlesztői környezet a „G” grafikus programnyelv számára. Első verziója 1986-ban jelent meg és azóta piacvezető szoftverré vált többek között a mérésautomatizálás, adatgyűjtés és műszervezrlés terén [2].

### 2.1 Adatfolyam-programozás

A LabVIEW-ban használt programozási paradigma az adat hozzáférhetőségén alapszik, a programnyelv elemeit VI-nak (Virtual Instrument) nevezzük. Ha egy VI vagy függvény számára megfelelő mennyiségű adat áll rendelkezésre, akkor az a VI vagy függvény le fog futni. A végrehajtás folyamatát az határozza meg, hogy a programozó miként hozza létre a grafikus blokkdiagramok struktúráját. Az egyes blokkokat a blokkok különböző funkciójú csomópontjai között húzódó vezetékek kötik össze. Ezeken a vezetékeken keresztül haladnak az adatok és amint egy VI-ba beérkezik az összes szükséges adat, az adott blokk képes elvégezni feladatát. Ezt nevezzük az adatfolyam-programozás elvének. Ha jobban belegondolunk a valóságban használt műszerek is hasonlóképp működnek [2].

### 2.2 Függvény és VI

Grafikus programozás során minden feladatot ezekkel a grafikus blokkokkal végzünk el a legegyszerűbbtől a legbonyolultabbig. Egyes blokkok, mint például azok, amelyek egyszerűbb matematikai műveleteket végeznek el, logikai kapukat valósítanak meg vagy egy tömbön végeznek valamilyen műveletet, a programozó számára rejtett módon működnek, leggyakrabban egy C nyelvben megírt függvényt futtatnak le.

Más blokkoknak viszont beleláthatunk a működésébe. Egy VI három részből áll: blokk diagram, előlapi panel (továbbiakban front panel) és a csatlakozó panel. Utóbbi reprezentálja a VI-t, ha más VI-ok blokk diagrammájában szerepeltetjük. A front panelen helyezkednek el a különféle vezérlőszervek és indikátorok, mint például a generálandó hullámforma kiválasztása vagy a mintavételezett jel spektruma. A blokk diagram tartalmazza a grafikus forráskódot. Minden, a front panelen elhelyezkedő objektum megjelenik a blokk diagramban is egy terminál formájában.

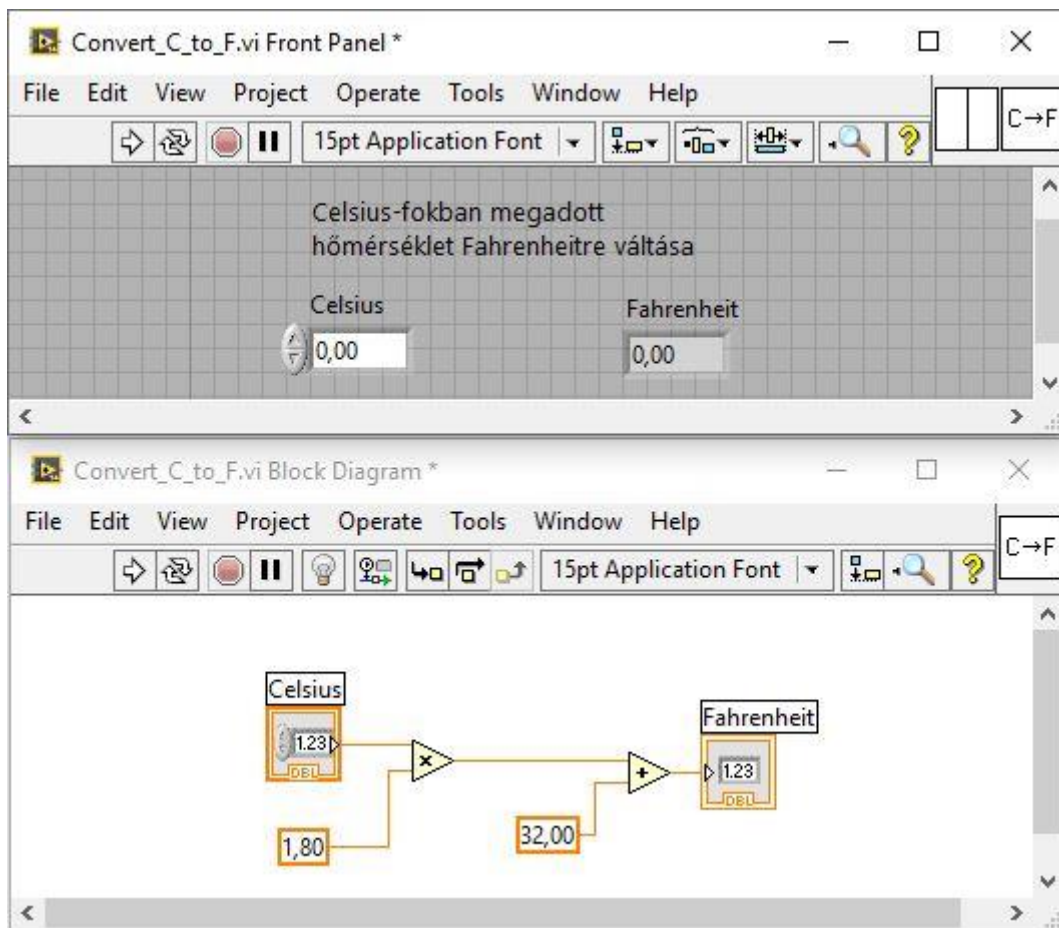


A 2.1 ábrán egy egyszerű példa szerepel, amely a Celsius-fokban megadott hőmérséklet értéket váltja át Fahrenheitre. A VI lényegében az

$$1,8\text{ }^{\circ}\text{C} + 32 = \text{ }^{\circ}\text{F} \quad (1)$$

összefüggést valósítja meg, ahol  $^{\circ}\text{C}$  az átváltandó hőmérsékletet jelenti Celsius-fokban,  $^{\circ}\text{F}$  pedig ennek Fahrenheitben megadott megfelelőjét.

A front panelen lehet megadni az átváltandó értéket és a mellette lévő indikátoron jelenik meg az eredmény. A blokk diagram tartalmazza az átváltást végző elemeket, az elvégezni kívánt műveletnek megfelelő összeköttetésben. Ezek a front panellel kapcsolatot létesítő terminálok, két konstans, illetve a szorzás és összeadás műveletét elvégző blokkok.



2.1 ábra: Celsius-Fahrenheit átváltó VI

Minden virtuális műszer futtatható önmagában a front panelt, mint felhasználói felületet használva, vagy egy másik VI részeként. Ekkor a front panelen elhelyezkedő vezérlők és kijelzők csomópontként fognak működni és ezek határozzák meg, hogy a

csatlakozó panelen milyen kimenetek és bemenetek jelennek meg. Ennek köszönhetően minden virtuális műszer önmagában is tesztelhető mielőtt egy nagyobb alkalmazás részeként beépítésre kerül. Az készített alkalmazások java része tehát különböző VI-ok és függvények összekapcsolásával és egymásba ágyazásával épülnek fel [2].

## 2.3 Pozitívumok

A grafikus megközelítés által a programozásban kevésbé jártasak is képesek programokat készíteni az általuk ismert laboratóriumi felszerelés virtuális reprezentációinak elhelyezésével és összekötésével. A fejlesztői környezet rengeteg példakóddal és széleskörű dokumentációval segíti a felhasználókat, így korábbi tapasztalat nélkül is igen gyorsan lehet elkészíteni kisebb alkalmazásokat [2].

A LabVIEW a National Instruments saját gyártású műszerei mellett rengeteg idegen gyártótól származó műszerre kínál drivereket és műszer specifikus VI-okat. Ha esetleg mégis olyan eszközre szeretnénk programot fejleszteni, amire nincs beépített támogatás, akkor is lehetőségünk van az általános VISA (virtual instrument software architecture) alkalmazásprogramozási felületet használni, de GPIB parancsokat is küldhetünk és soros porton keresztül is kommunikálhatunk eszközünkkel.

A fejlesztő környezet a műszer specifikus VI-ok mellett rengeteg más kész VI-al rendelkezik a villamosmérnöki szakma különböző területeiről. Találhatunk VI könyvtárakat többek között jelfeldolgozás, adatgyűjtés, matematikai műveletek, jelgenerálás, jel kondicionálás, statisztika és analízis feladatokhoz, illetve függvényeket, amelyek például szűrőket valósítanak meg és mátrixműveleteket vagy integrálási, differenciálási műveleteket végeznek el. Továbbá lehetőség van egy MathScript nevezetű szöveg alapú program komponens hozzáadására is. A MathScript rendelkezik a VI-okhoz hasonló kapcsolódási csomópontokkal, tehát könnyen integrálható a grafikus programkörnyezetbe. Szintaxisa nagyrészt megegyezik a MATLAB szintaxisával, így MATLAB kódokat is elhelyezhetünk az alkalmazásunkban, ezzel kihasználva mind a szöveg alapú, mind a grafikus programozás nyújtotta előnyöket [2].

További tagadhatatlan előny, hogy lehetőségünk van párhuzamos programozásra. Ez az automatizált tesztrendszerek esetében különösen fontos, ahol nem ritka, hogy több folyamat, mint például az adatfeldolgozás, a mintavételezés vagy más hardverrel történő kapcsolattartás, párhuzamosan történik [2].

## 2.4 Negatívumok

A legtöbb felhasználó a LabVIEW lassúságát és lefagyásra vagy összeomlásra való gyakori hajlamát kritizálják, akár a legegyszerűbb feladatok végzése közben is. Sok esetben az egyetlen megoldást a szoftver vagy a teljes számítógép újraindítása jelenti.

Mivel általában egy programozó először szövegalapú programnyelveket tanul meg ezért sokaknak nehéz hozzászokniuk ehhez a teljesen új, más gondolkodási formát igénylő nyelvhez. A legtöbb esetben ugyanis nem, vagy csak közvetetten alkalmazhatóak a korábban megtanult alapvető programozási alapelvek és konstrukciók.

További hátrány, hogy az elkészült kód a grafikus jellege miatt jóval nehezebben dokumentálható, mint a szöveg alapú változatok, illetve ugyanezen okból kifolyólag a kód hordozása is körülményesebb.

## 3 A műszervezélés bemutatása

### 3.1 A távvezérelt műszerekről általánosan

A távvezérelhető műszerek alatt olyan műszereket értünk, amelyeket az adott mérőeszközön található kezelő szervek nélkül is tudunk használni, azaz hozzáférünk a szolgáltatásaihoz. Különböző fizikai interfészek használatával összeköttetést biztosítunk a műszer és a vezérlő szerv - rendszerint egy számítógép - között, ahonnan különféle parancsok segítségével vagyunk képesek elérni az eszköz funkcióit.

A szoftverdefiniált műszerezésnek sokféle programba történő leképzése lehet, de felállítható egy általános lépéssorozat, amely minden vezérlési folyamatra érvényes [3]. Ezek a következők:

#### 1. Inicializálás

Elsőként fel kell állítsuk a kapcsolatot a műszer és a vezérlő szerv között. Ebben a lépésben adjuk meg a műszer buszának címét, vagy általánosabban valamilyen referenciát, amin keresztül elérjük az adott műszert. Továbbá van lehetőség resetelni vagy egyéb specifikus állapotba állítani az eszközt.

#### 2. Konfiguráció

Ezt követően fel kell konfigurálni az eszközt az általunk kívánt művelet elvégzésére. Itt kell például a mérési határokat, kimeneti impedanciát, mérési üzemmódot, mintavételezési adatokat stb. megadni. A beállítások száma a művelet komplexitásától függ. Ezután eszközünk készen áll a mérések elvégzésére.

#### 3. Tevékenység

Az eszköz a már megadott konfigurálási beállításoknak megfelelően elvégzi az általunk meghatározott tevékenységet, mint például a mérés vagy a jelgenerálás.

#### 4. Lezárás

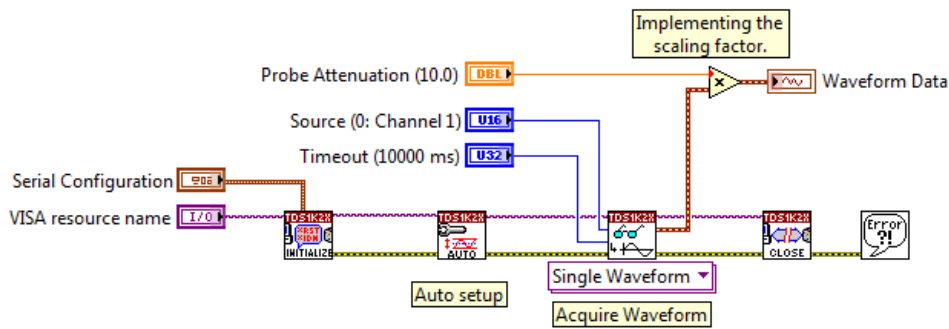
A tevékenység végeztével megszakítjuk a szoftveres kapcsolatot és felszabadítjuk a műszert és minden egyéb, a folyamat során használt erőforrást.

#### 5. Hiba információk kinyerése és megjelenítése (opcionális)

Ha a műszervezélés folyamata során valamilyen hiba történt, nem árt, ha tudunk róla. Ezek a hibák lehet, hogy a műveletnek csak egy kisebb részegységét befolyásolják, de előfordulhat, hogy a teljes folyamat idő előtti megszakítását eredményezik. Ezen

információk megjelenítése nem vitális a vezérlés szempontjából, de hasznos segítséget jelenhetnek a szoftver tesztelésekor, illetve a kész program használatakor.

A 3.1 ábrán egy Labview-specifikus példa látható a fentebb említett lépéssorra, ahol egy oszcilloszkópról olvasunk be egy hullámformát a vezérlő PC-be. A séma négy plusz egy lépése ebben az esetben egyszerűen öt VI képében jelenik meg. A VI-ok sorrendje a fenti felsorolás eleminek sorrendjével azonos.



3.1 ábra: oszcilloszkóp vezérlése

A következőkben megtekintjük, hogy pontosan hogyan folyik a szoftverdefiniált műszerezés LabVIEW-ban.

## 3.2 A DAQ kártya vezérlése

### 3.2.1 Az eszköz specifikációja

A feladat elvégzése során a National Instruments 6036E típusú DAQ (Data Acquisition) kártyája áll rendelkezésünkre. Ez az eszköz 16 darab 16 bites analóg bemenettel, két 12 bites analóg kimenettel, nyolc digitális I/O porttal, és összesen 68 pin csatlakozóval rendelkezik [5].

Az időzítéshez kapcsolódó alkalmazásokhoz az adatgyűjtő rendszer a National Instruments saját fejlesztésű időzítés vezérlőjét használja. Rendelkezik külön analóg input, analóg output és általános felhasználású időzítővel, melyek maximálisan 50 ns-os felbontást tudnak elérni. Ezzel lehetőség nyílik bufferelt impulzus generálásra, egyenlő időközönként történő mintavételezésre és a mintavételi sebesség megváltoztatására a mintavételezés megszakítása nélkül [5].

Mindemellett képes interface-t biztosítani SCXI (Signal Conditioning Extensions for Instrumentation) rendszerek számára, így a feszültség és áramforrások mellett hőelemek, hőszenzorok és nyúlásmérő bélyegek jeleit is fogadni tudja [5].

A kártya bemenetén és kimenetén is maximum  $\pm 10$  V-os feszültség szint engedhető meg. A bemeneten maximum 200 kHz-es mintavételi frekvencia állítható be.

Az eszköz vezérléséhez LabVIEW 2014-et használunk, a szükséges DAQ specifikus driverek kiegészítésével. Egy analóg bemenetre (AI1) és egy analóg kimenetre (DACOUT0) lesz szükségünk a mérés, illetve a jelgenerálás céljából, illetve a földelést biztosító csatlakozókra. A használt lábak a 3.2 ábrán látható lábkiosztáson kerültek megjelölésre.

ACH8	34	68	ACH0
ACH1	33	67	AIGND
AIGND	32	66	ACH9
ACH10	31	65	ACH2
ACH3	30	64	AIGND
AIGND	29	63	ACH11
ACH4	28	62	AISENSE
AIGND	27	61	ACH12
ACH13	26	60	ACH5
ACH6	25	59	AIGND
AIGND	24	58	ACH14
ACH15	23	57	ACH7
DAC0OUT <sup>1</sup>	22	56	AIGND
DAC1OUT <sup>1</sup>	21	55	AOGND
RESERVED	20	54	AOGND
DIO4	19	53	DGND
DGND	18	52	DIO0
DIO1	17	51	DIO5
DIO6	16	50	DGND
DGND	15	49	DIO2
+5 V	14	48	DIO7
DGND	13	47	DIO3
DGND	12	46	SCANCLK
PFI0/TRIG1	11	45	EXTSTROBE*
PFI1/TRIG2	10	44	DGND
DGND	9	43	PFI2/CONVERT*
+5 V	8	42	PFI3/GPCTR1_SOURCE
DGND	7	41	PFI4/GPCTR1_GATE
PFI5/UPDATE*	6	40	GPCTR1_OUT
PFI6/WFTRIG	5	39	DGND
DGND	4	38	PFI7/STARTSCAN
PFI9/GPCTR0_GATE	3	37	PFI8/GPCTR0_SOURCE
GPCTR0_OUT	2	36	DGND
FREQ_OUT	1	35	DGND

<sup>1</sup> Not available on the NI 6034E

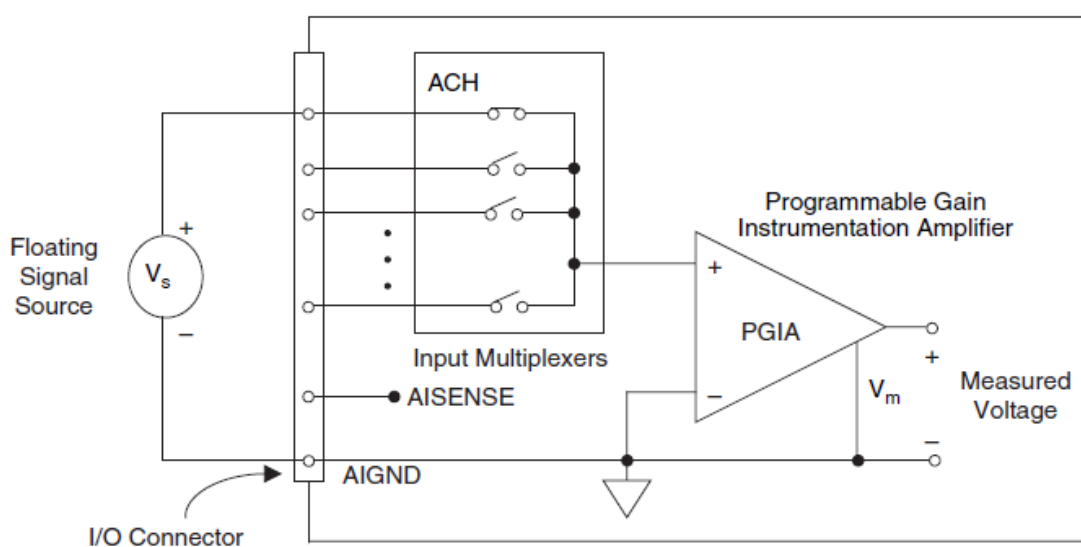
3.2 ábra: a DAQ lábkiosztása

### 3.2.2 A bemenet vezérlése

#### Inicializáció és konfiguráció:

A Create Channel VI által létrehozunk egy új csatornát a mérés elvégzéséhez. Ez a VI egyszerre valósítja meg az inicializáció és a konfiguráció lépését. A konfiguráció részeként különféle beállításokat kell végeznünk, ahhoz, hogy helyes eredményeket kapjunk a mérés során.

Először is meg kell adni, hogy milyen csatornára van szükségünk. Mivel egy analóg jel feszültségét szeretnénk mérni ezért az Analog Input Voltage módot kell választani. Emellett be kell állítani a mérési határokat ( $\pm 5V$ ), a használni kívánt PIN nevét (AI1), a használni kívánt mértékegységet (Volt), valamint a bemenet konfigurációját (RSE). RSE (Referenced Single Ended) üzemmódban, a bemeneti jelünk az eszköz programozható erősítőjének pozitív lábára csatlakozik, míg a negatív lábára a föld kerül, így ez a beállítás ideális a vizsgálandó jel fogadására [4]. A konfigurációt a 3.3 ábra mutatja.



3.3 ábra: egy csatorna RSE konfigurációja

Továbbá, a mintavételezéshez szükségünk lesz egy időzítőre is. Ezt a Timing VI-al valósíthatjuk meg. Itt meg kell adni, hogy mintavételezéshez szükséges órajelet használunk (Sample Clock), véges számú mintát szeretnénk venni (Finite Samples), a veendő minták számát és a mintavétel sebességét. Utóbbi két értéket az adott mérés igényeinek megfelelően fogjuk beállítani.

#### Tevékenység:

A felkonfigurált csatornából a Read VI segítségével olvasható ki a fogadott hullámforma. Itt is meg kell adni, hogy analóg bemeneti csatornából olvasunk ki több mintát egy hullámforma típusú adat vezetékbe [4].

Ahhoz, hogy megkezdhessük a mintavételezést még szükséges a Start Task VI beszúrása a beolvasás blokkja elő. Ez az a VI ugyanis, ami a taskunkat futó állapotba helyezi át, ezzel megkezdve a mérést. A mintavételezés végeztével pedig a Stop Task VI-t fogjuk lefuttatni, ami a futó állapotból visszaállítja taskunkat a Start Task VI előtti állapotba [4].

A kiolvasott adatokon már tudunk méréseket végezni, így meghatározható például a jel RMS-e, peak to peak amplitúdója, frekvenciája és spektruma.

#### Lezárás:

A jel vizsgálatának végeztével a Clear Task VI megszakítja a taskot, felszabadítja az általa használt erőforrásokat majd törli is a taskot [4].

#### Hiba információk kinyerése és megjelenítése

Minden, a mintavételezésben résztvevő virtuális műszer rendelkezik hiba információk továbbítására szolgáló bemeneti és kimeneti csomópontokkal. Ezeknek a csomópontoknak az összekötésével egy soros kapcsolatot hozhatunk létre a blokkok között, így a hibát leíró üzenet végigfolyik a legutolsó blokkig. A legutolsó VI kimenetére kötött hibajelző ablak elhelyezésével minden hibaüzenetről értesülni fogunk.

### **3.2.3 A kimenet vezérlése**

#### Inicializáció és konfiguráció:

A jel fogadásához hasonlóan itt is új csatornát kell létrehoznunk. A Create Channel VI beállításai nagyrészt megegyeznek az előbbiekkal, viszont itt Analog Output Voltage módra van szükség és az AO output PIN-t használjuk [4].

A Timing VI-nak ebben az esetben szüksége van a generált jelre, mivel annak  $dt$  komponense adja meg, milyen időközönként kell kiadni a mintákat (Use Waveform konfiguráció). Emellett itt is beállítjuk, hogy véges számú mintát szeretnénk megjeleníteni a kimeneten [4].

#### Tevékenység:



Az adatok kimeneten való megjelenítéséhez szolgál a Write VI. Itt beállítjuk, hogy egyetlen csatornán adunk ki egy analóg hullámforma jelet. A mintavételezéshez hasonlóan a jelgenerálásnál is használjuk a Start Task VI-t és a Stop Task VI-t, de az előző esttel ellentétben most nem fogják közre a kiírás blokkját, hanem utána helyezkednek el. Elhelyezkedik köztük viszont a Wait Until Done VI, ami biztosítja, hogy a jelgenerálás befejeződik még a Stop Task VI lefutása előtt [4].

#### Lezárás:

A kiírás befejeztével itt is lezárjuk a task-ot a Clear Task VI-al.

#### Hiba információk kinyerése és megjelenítése

A jelgenerálásban résztvevő blokkokat összekötjük és az utolsó után kivezetjük az adatokat a front panelre.

### **3.3 Az oszcilloszkóp képernyőképének mentése**

Bizonyos esetekben egyszerűbb a jeleket egy oszcilloszkópon megvizsgálni, mint szoftveres úton. Az egyes beállítások hatását rögtön láthatjuk a képernyőn, ezért, ha egy mérés során sokszor szeretnénk változtatni a konfiguráción gyorsabb és kényelmesebb lehet a fizikális kezelőszerveket használni, mint a szoftveres úton vezérelni a műszert.

Mindent, ami az oszcilloszkóp képernyőjén megjelenik lehetőségünk van elmenteni számítógépünkre, feltéve, hogy az eszköz rendelkezik a megfelelő csatlakozóval, ami a vezérlésre lehetőséget ad. LabVIEW-ban nem létezik erre külön előkészített VI, viszont a funkció megvalósítható VISA elemek segítségével.

Erre mutat példát a 10.1 alfejezet blokk diagramja. Elsőként most is kapcsolatot teremtünk a használni kívánt műszerrel a VISA Open blokk által. Majd Clear funkcióval kitöröljük az eszköz kimeneti és bemeneti bufferjeit, hogy a jelenlegi adatok ne zavarják a műveletet. Ezután elküldjük a képernyőmentést megvalósító két parancsot a Write blokkal. Ez a `SAVE:IMAGe:FILEFormat PNG` és a `HARDCopy START`. Előbbivel azt mondjuk meg a programnak, hogy PNG formátumban mentse le a képernyőképet, utóbbival pedig elindítjuk ezt a műveletet. Ezután a válaszképpen kapott adatokat kiolvassuk a Read funkcióval. A továbbiakban már nem lesz szükségünk az oszcilloszkópra, ezért lezárhatjuk a kapcsolatot a Close blokkal.

Az eszköz által küldött adatok még nincsenek olyan állapotban, hogy azokat képként meg lehessen jeleníteni. Adott elérési útvonalon létre kell hoznunk egy fájlt a

Create File elemmel, amibe ezeket az adatokat a Write to Binary File blokk segítségével beírjuk. Miután a Close File funkcióval lezártuk a fájlt, már megtekinthető a készített képernyőkép. Az esetleges hibákat egy hibajelző ablakkal kivezetjük a frontpanelre.

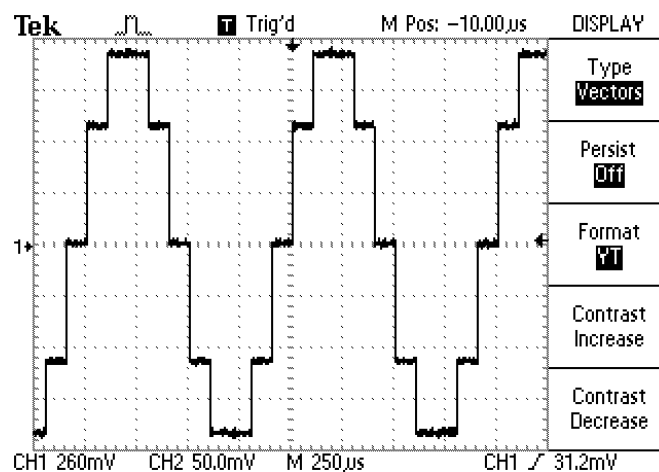
A DAQ kártya vezérléséhez hasonlóan az oszcilloszkópnál is megjelennek az automatizált műszervezélés általános lépései. Az inicializáció (VISA Open), a konfiguráció (VISA Clear, VISA Write - SAVE:IMAGE:FILEFormat PNG parancs), a tevékenység (VISA Write - HARDCopy START parancs, VISA Read) a lezárás (VISA Close) és a hibák megjelenítése mind könnyen felismerhető az alkalmazásban.

## 4 A DAQ kártya vizsgálata

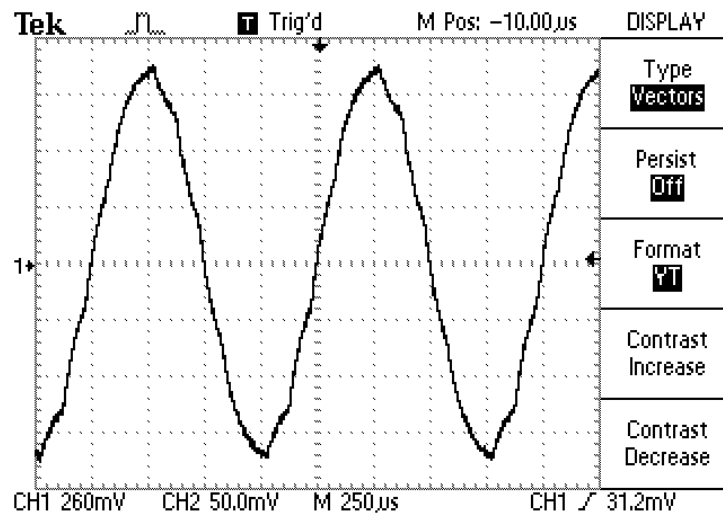
### 4.1 Az eszköz megismert határai

A DAQ kártya adatlapját megtekintve láthatjuk, hogy az analóg bemenet mintavételi sebessége 200kSample/sec [5], így a Nyquist-Shannon mintavételi elv értelmében a kártya alkalmatlan 100 kHz-es vagy annál nagyobb frekvenciájú jel torzításmentes feldolgozására. Az analóg kimenet esetén nagyobb korlátoltságot tapasztalhatunk. Ugyanis a DMA maximális frissítési sebessége 10 kHz [5], tehát ez a legnagyobb érték, amit mintavételi frekvenciának megadhatunk. A korábban említett elv miatt maximum 5 kHz-es jelet vagyunk képesek generálni az eszközzel.

A 4.1 ábrát megtekintve szembevető, hogy az 1 kHz-en generált szinusz jel lépcsős. Ennek oka, hogy a D/A átalakító nulladrendű tartón keresztül továbbítja a jelet a kimenetre és nincs semmiféle szűrő, ami a szomszédos minták közötti átmenetet javítaná. A lépcsősség azért is szembevető, mivel ezen a frekvencián már csak igen kevés mintából áll a jel egy periódusa. A frekvencia növelésével az egy periódust alkotó minták száma tovább csökken, míg 4-5 kHz magasságában már lényegében egy négyszög jelet kapunk. A jel valamelyest kisimítható egy egyszerű aluláteresztő szűrő használatával. Ezt egy breadboardon elhelyezett 10k $\Omega$ -os ellenállás és 8.2 nF-os kondenzátor segítségével könnyen megvalósítható. Az eredményt a 4.2 ábrán láthatjuk.

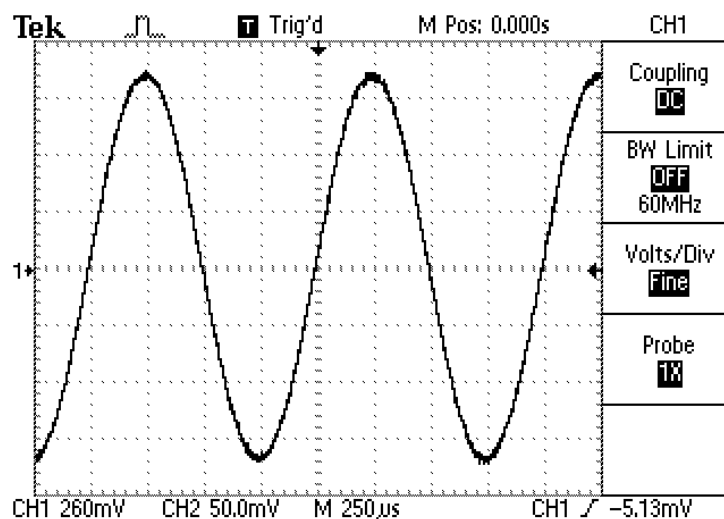


4.1 ábra: 1 kHz-es szinusz jel a DAQ kimenetén



4.2 ábra: 1 kHz-es szinusz jel az elsőfokú szűrő kimenetén

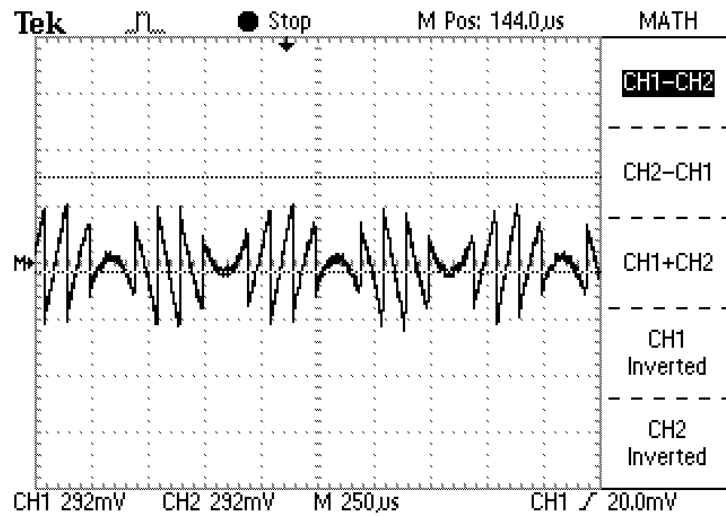
A szűrés némileg javított a jelalakon, viszont nem lehetünk teljesen elégedettek, még így is torz a jel. Kijelenthetjük, hogy nem elég az elsőfokú szűrő használata, legalább egy másodfokúra lesz szükségünk. A másodfokú szűrő használatával kapott jel a 4.3 ábrán tekinthető meg. Ez a szűrő már kellően kisimította az eredeti szinusz jelünket, és torzítás már nem tapasztalható. Maga a szűrő egy későbbi fejezetben kerül részletesebb tárgyalásra.



4.3 ábra: 1 kHz-es szinusz jel az másodfokú szűrő kimenetén

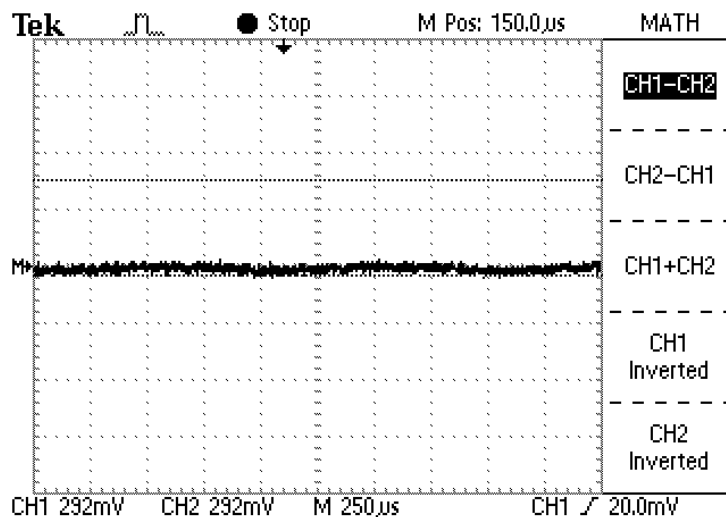
A szűrés hatékonyságát egyszerűen ellenőrizhetjük egy hibajel előállításával. Egy függvénygenerátorral előállítunk egy referencia szinusz jelet és a DAQ kártya által generált szinusz jellel együtt rákötjük egy oszcilloszkópra. Az oszcilloszkóp MATH funkciójával pedig a két jel különbségét vesszük.

A 4.4 ábrán a 4.1 ábra szinusz jelével adódó hibajel látható. Az eredmény leginkább egy kvantálási hibához hasonlít.



4.4 ábra hibajel szüretlen DAQ jel esetén:

A 4.5 ábrán az a hibajel látható, amely a referenciajel és a másodfokú szűrővel megszürt jel különbségeként adódott. A hibafüggvény közel konstans nulla amplitúdóval rendelkezik, így mondhatjuk, hogy a két jel nagyjából megegyezik. A kis eltérések betudhatóak a függvénygenerátor hibájának, tehát a szűrőnk hatékonyan simítja ki a DAQ eredeti jelét.



4.5 ábra: hibajel szürt DAQ jel esetén

## 4.2 Összehasonlítás

A DAQ kártya által generált jel egy Hameg 8030-as függvénygenerátorával került összevetésre. Az összehasonlításhoz mindkét eszközzel 10, 100 és 1000Hz-es, háromszög, négyszög és szinusz jeleket generálunk, 1 V-os amplitúdóval. Ezeket a kártya felkonfigurált bemenetén fogadjuk, majd a mintavételezést követően a jelek Peak-to-Peak amplitúdóját, illetve RMS-ét mérjük meg. Megjegyzendő, hogy a mérések az eszközök által eredetileg generált jelekre vonatkoznak. Semmiféle szűrésen vagy egyéb jel kondicionáláson nem esnek át. A mérési eredmények a következőknek adódtak:

10 Hz	háromszögjel		négyzetjel		szinuszjel	
	DAQ	fv. gen.	DAQ	fv. gen.	DAQ	fv. gen.
<b>V<sub>PP</sub></b>	2,0007 V	1,9579 V	1,9935 V	1,9339 V	2,0007 V	1,9626 V
<b>V<sub>RMS</sub></b>	577,3587 mV	566,4668 mV	1,001 V	969,2861 mV	707,1290 mV	692,1463 mV

100 Hz	háromszögjel		négyzetjel		szinuszjel	
	DAQ	fv. gen.	DAQ	fv. gen.	DAQ	fv. gen.
<b>V<sub>PP</sub></b>	2,001 V	1,9543 V	1,9934 V	1,9364 V	2,0009 V	1,9667 V
<b>V<sub>RMS</sub></b>	557.76151 mV	566,2388 mV	1,0001 V	969,6064 mV	707,1259 mV	691,8309 mV

1 kHz	háromszögjel		négyzetjel		szinuszjel	
	DAQ	fv. gen.	DAQ	fv. gen.	DAQ	fv. gen.
<b>V<sub>PP</sub></b>	1,5944 V	1,9540 V	1,9935 V	1,9418 V	1,8953 V	1,9559 V
<b>V<sub>RMS</sub></b>	565,6338 mV	575,5294 mV	1,0001 V	969,3004 mV	707,0164 mV	689,0630 mV

A fentiekből látszik, hogy alacsonyabb frekvenciákon a DAQ kártya sokkal pontosabb amplitúdójú jelet generál, mint a függvénygenerátor és az RMS eredmények is jobban megközelítik az elméleti értékeket. A kHz-es frekvencián viszont már pontatlan -eredményeket kapunk szinusz és háromszög jelek esetén.

Nézzük meg azt is, hogy mekkora a bemeneten fogadott jel hibája, a kimenetet, mint referenciát felhasználva. Ehhez a kimeneten és a bemeneten mintavételezett értékekből kiszámítjuk a jel négyzetes középértékét a

$$X^{RMS} = \sqrt{\frac{1}{n} \sum_{i=0}^{n-1} |x_i|^2} \quad (2)$$

egyenletet felhasználva, ahol  $x_i$  az adatokat tartalmazó tömb  $i$ . eleme,  $n$  pedig az elemek száma.

A hibát a következőképp számíthatjuk ki:

$$\text{hiba} = \frac{X_I^{RMS} - X_{\text{ref}}^{RMS}}{X_{\text{ref}}^{RMS}} \quad (3)$$

$X_I^{RMS}$  a bemeneten fogadott jel RMS-e,  $X_{\text{ref}}^{RMS}$  pedig a referenciaként szolgáló, kimeneten megjelenített jel.

Az alábbi táblázat mutatja, hogy az egyes frekvenciákon melyik jelalakra mekkorának adódott ez a hiba.

	<b>háromszögjel</b>	<b>négyszögjel</b>	<b>szinuszjel</b>
<b>10 Hz</b>	9,05847 ppm	123,559 ppm	30,262 ppm
<b>100 Hz</b>	28,1438 ppm	119,126 ppm	32,1171 ppm
<b>1 kHz</b>	124,91 ppm	122,885 ppm	106,707 ppm

## 5 A mintavételezés vizsgálata

### 5.1 Koherens és inkohereus mintavételezés vizsgálata

#### 5.1.1 A koherens mintavételezés feltétele

Koherens mintavételezésről akkor beszélünk, amikor a feldolgozandó jelből egész számú periódust mintavételezünk. Ez azt jelenti, hogy a  $T$  mintavételi időtartam, vagy más szóval regisztrátum hossza egész számú többszöröse a mintavételezett jel periódusidejének. Vagyis,

$$T_{\text{regisztrátum}} = kT_{\text{jel}} \quad (4)$$

ahol  $k$  egy pozitív egész szám.

Érdekes a fenti összefüggést frekvenciákra felírni. Ehhez szükség lesz a spektrum frekvenciafelbontására, amely

$$\Delta f = \frac{f_s}{N} \quad (5)$$

alakban írható fel.  $N$  a vett minták számát jelöli,  $f_s$  pedig a mintavételi frekvencia.

A regisztrátum felírható a következő képpen is:

$$T_{\text{regisztrátum}} = NT_s = \frac{N}{f_s} = \frac{1}{\Delta f} \quad (6)$$

Az (6) egyenlet eredményét visszahelyettesítve a (4) egyenletbe a megfelelő átalakításokkal

$$f_{\text{jel}} = k\Delta f \quad (7)$$

adódik. Vagyis a mintavételezett jel frekvenciája a frekvenciafelbontás egész számú többszöröse kell legyen.

Mivel a frekvenciafelbontást a mintavételi frekvencia és a mintaszám aránya adja, ezért bármelyik helytelen megválasztása inkohereus mintavételezéshez vezethet.

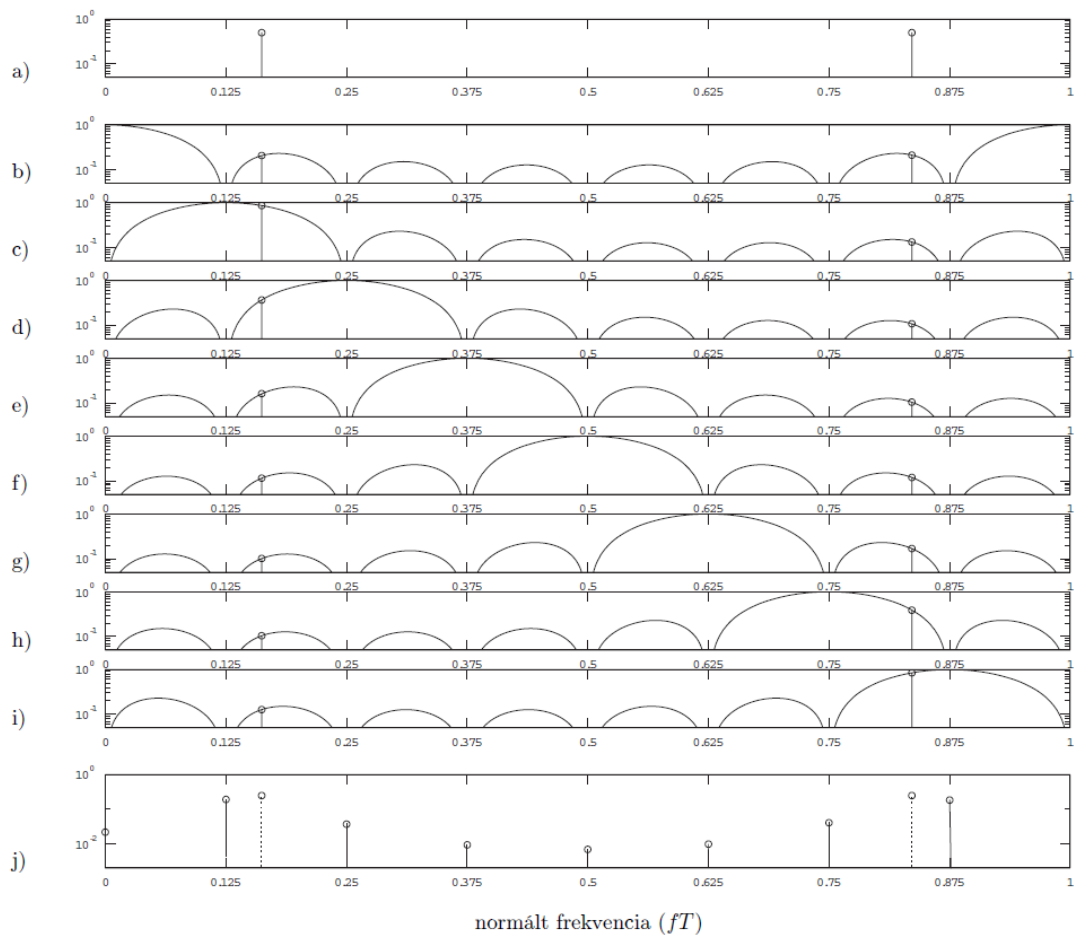


### 5.1.2 Az inkohereus mintavételezés következménye: a spektrális szivárgás

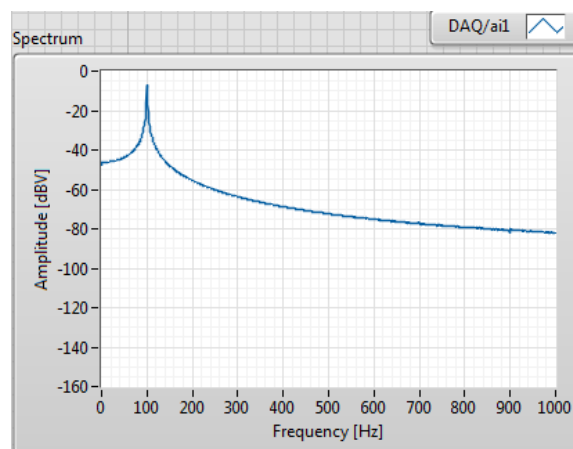
A DFT-re érdekes szűrőbankként tekinteni, ahol az egyes szűrők sávközepe a frekvenciafelbontás szerint növekedik [6]. Kohereus mintavételezés esetén a mintavételezett jel frekvenciakomponense(i) mindig vagy a szűrők fő hullámának közepére, vagy valamelyik nullhelyre esnek, így vagy szűretlenek maradnak, vagy teljesen kiszűrődnek.

Nem kohereus mintavételezés esetén viszont a mintavételezett jel frekvenciája nem egész számú többszöröse a frekvenciafelbontásnak. Ebből kifolyólag az egyes komponensek soha nem esnek nullhelyre vagy a szűrő sávközepeire. Minden szűrő a 5.1 ábrán is látható módon valamilyen véges mértékben csillapítja őket. A szűrőkarakterisztikák átfedése illetve periodikus „leszívásai” miatt tehát a spektrumon olyan komponensek is megjelennek, amelyeken nem található jel. Megjegyzendő, hogy az 5.1 ábra vízszintes tengelyét a (dimenzió nélküli)  $fT_s$  normált frekvencia képezi, ahol  $f$  a frekvencia,  $T_s$  pedig a mintavétel periódusideje [6].

A jelenséget spektrális szivárgásnak nevezzük és, mint ahogy az az 5.2 ábrán megfigyelhető, úgy jelentkezik, mintha a jelnek egy „szoknyája” lenne. Ez azért jelenthet problémát mert egyes esetekben képes a vizsgált jel kis amplitúdójú hasznos spektrumkomponenseit eltakarni.



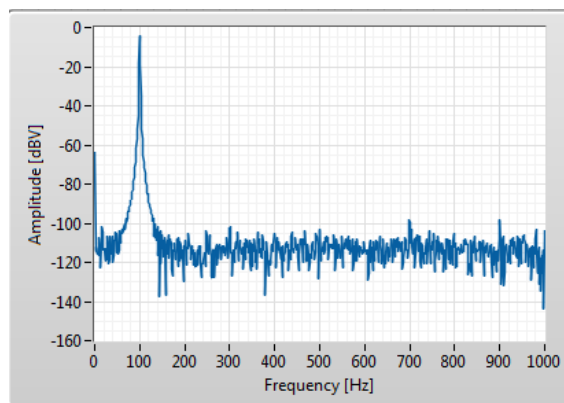
**5.1 ábra: Spektrumszivárgás és picket fence jelenség: (a) egy szinuszos jel spektruma, (b-i) a DFT-szűrőbank szűrőkarakterisztikái ( $N = 8$ ), (j) a szűrőkimenetek teljesítménye a megfelelő sávközépen ábrázolva (szaggatottal a valódi teljesítmény-spektrum) [6].**



**5.2 ábra: Spektrális szivárgás**

Továbbá megesik, hogy a kapott spektrumon hibás amplitúdót látunk. Ennek oka, hogy a jelkomponens nem az ablakfüggvény fő hullámának közepére esik és mivel a szűrőkarakterisztikák nem elég „laposak” a sávközép környezetében, így a valósnál kisebb értéket kapunk a csillapítás miatt [6].

Ezeket a hatásokat enyhíteni lehet különböző ablakozó függvények használatával. Például a Hanning ablak Fourier-transzformáltjának szélesebb főhulláma és kisebb oldalhullámai vannak, mint a rect ablaknak. Emiatt nem koherens mintavételezés esetén a főhullamba eső komponensek kisebb, míg az oldalhullamba esők nagyobb csillapítást szenvednek [7]. Az 5.3 ábrán is jól megfigyelhető, hogy használatával a spektrális szivárgás hatása jelentősen csökkenthető. Ablakozás nélkül például a 100 Hz-es jel 150 Hz-es frekvencia komponense körülbelül -50 dBV-os amplitúdóval rendelkezik, míg ablakozással ez az érték már nagyjából -110 dBV. Az ablakozás 60 dBV-al csökkentette a komponens amplitúdóját, ami azt jelenti, hogy ezredére csillapította a zavaró komponens.



5.3 ábra: Hanning ablak hatása

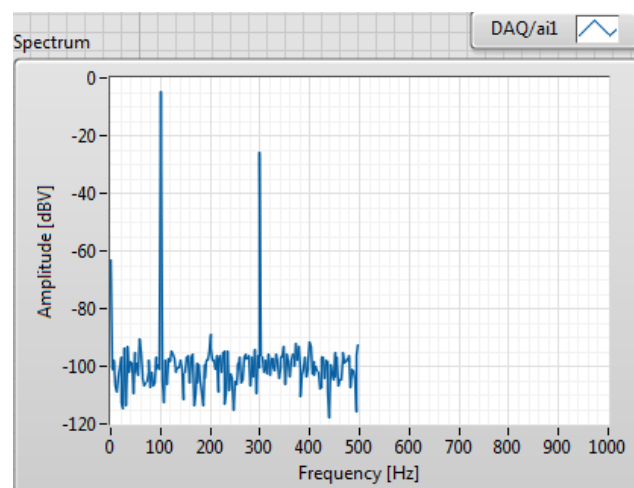
## 5.2 A mintavételezési információk megadása

LabVIEW-ban mind jelgenerálásakor, mind jel fogadásakor szükséges a mintaszám és a mintavételi frekvencia megadása. Előbbinél a hullámformát előállító VI-nak, utóbbinál az időzítést megvalósító VI-nak van rá szüksége.

A mintavételi frekvencia megválasztásakor oda kell figyelni a Nyquist-Shannon mintavételi elv teljesítésére, miszerint a mintavételi frekvenciának legalább kétszerese kell lennie a mintavételezett jel frekvenciájához képest. Amennyiben nem tartjuk be a mintavételi tételt, a fejlesztőkörnyezet egy hibaüzenettel leáll. Az alulmintavételezés

hibáját tehát nem követhetjük el, de bizonyos esetekben még így sem elég, ha éppenhogy betartjuk. Egyik eset az, amikor időtartományban szeretnénk kiértékelni a jelet. Másik eset amikor a periódikus jel sávszélessége igen nagy. Ez azért probléma, mert spektrum ábrázolásakor a mintavételi frekvencia feléig kerülnek kiszámolásra a spektrális komponensek. Ha túl kicsi a mintavételi frekvencia ez gondot okozhat egy háromszög vagy négyszög jel spektrumának ábrázolásakor, amelyek spektruma elvileg végtelen. Ezt az esetet érzékelteti az 5.4 ábra, ahol egy 100 Hz-es háromszögjel spektrumán annak alapharmónikusát és első felharmónikusát látjuk csak. Tehát bizonyos esetekben érdemes, és kell törekedni a túlmintavételezésre, hogy praktikus szempontból elegendő jelkomponenst tudjunk megjeleníteni és adott esetben további érdemi jelfeldolgozást végezni.

Továbbá a mintavételi frekvencia helyes megválasztása után az előző fejezetben leírtakat szem előtt tartva kell kiválasztani a minták számát, hogy utána koherensen mintavételezett jelet kapjunk.



**5.4 ábra: 100 Hz-es háromszögjel spektruma 1 kHz-es mintavételi frekvenciával**

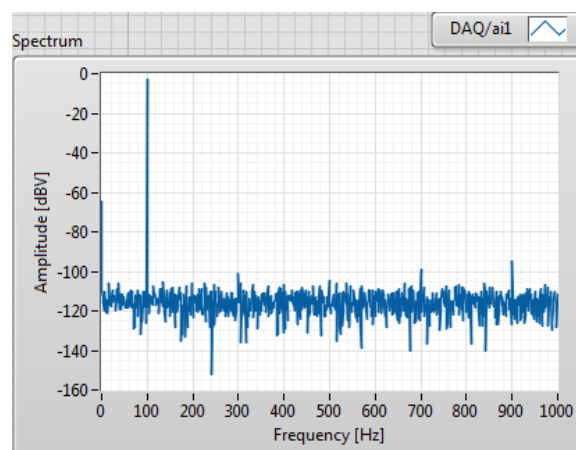
A jelgenerálás esetében a mintavételi frekvencia kissé disszonáns kifejezés, ilyenkor valójában a minták kiadásának sebességéről beszélünk. Mivel az 1 Hz – 5 kHz tartományon belül szeretnénk jelet generálni, ezért a mintavételi információkat úgy kell megadni, hogy a teljes, lefedni kívánt tartományon belül minden jel helyesen legyen generálva. Korábban láthattuk, hogy a kimeneten beállítható legnagyobb mintavételi frekvencia 10 kHz és a fentebb leírtak miatt törekedünk a túlmintavételezésre, ezért ennek megfelelő mintavételi frekvenciát be is állíthatjuk.

A regisztrátumnak tartalmaznia kell a jel legalább egy teljes periódusát, mint ahogy azt a (4) egyenlet is leírja. Valamint előfordulhat, hogy a jel periódusideje nagyságrendekkel nagyobb, mint a mintavételi periódusidő, ilyenkor pedig igen nagy mintaszámra van szükség, ahogy az a (4) és (6) egyenlet összevetéséből is látszik. Figyelembe véve, hogy az 1 Hz – 5 kHz tartományon belüli jelekre van szükségünk, tízezer minta elégnek bizonyul. Ugyanis a DAQ és a függvénygenerátor összehasonlításakor 10 Hz-es a legkisebb jel. Az RMS számítás szempontjából fontos, hogy több periódust kapjunk. 10000 minta és 10 kHz-es mintavételi frekvencia mellett az (6) egyenlet alapján 1 s lesz a regisztrátum hossza. A 10 Hz-es jel periódusideje 0.1 s, így a (4) összefüggés szerint 10 periódus is meg fog jelenni a kimeneten.

Az átviteli függvény számításakor 1 Hz-es jeleket is felhasználunk. Az előbb említett összefüggések alapján ebben az esetben a regisztrátum hossza és a jel periódusideje is 1 s, ezért a regisztrátum csak egy periódust tartalmaz, de ott ez is elég.

A DAQ kártya bemeneti portjánál szélesebb frekvenciatartományból választhatunk, mint a kimeneti esetében, ezért itt nagyobb mintavételi frekvencia is megadható. De mivel nem fogunk nagyobb frekvenciájú jeleket mintavételezni, mint amiket előállítunk ezért elég itt is 10 kHz-et beállítani. A mintaszámot szintén beállíthatjuk a jelgenerálásnál megadott értékre.

A megadott 10 kHz-es mintavételi frekvencia és 10000 mintaszám alapján 1 Hz-es frekvenciafelbontást kapunk, így minden jel, amit generálunk és beolvasunk koherensen mintavételezett lesz, mint ahogy azt az 5.5 ábra is mutatja.



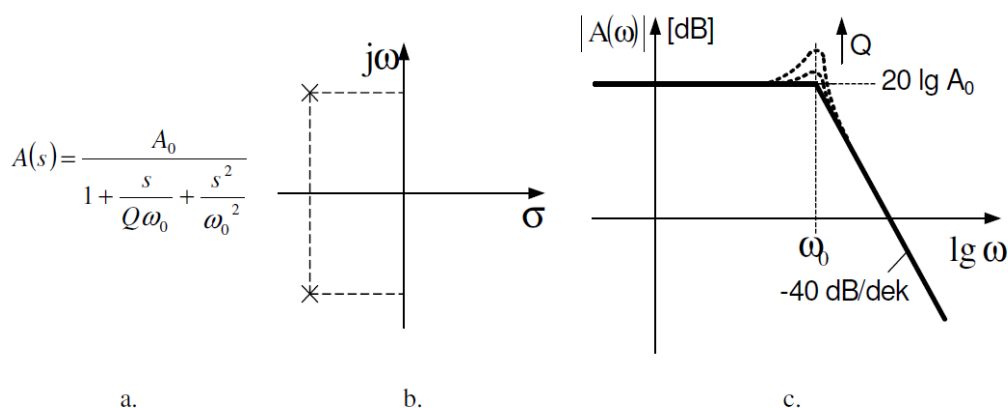
**5.5 ábra: Koherensen mintavételezett 100 Hz-es, 1V amplitúdójú szinuszos jel spektruma**

## 6 Az aluláteresztő szűrő

### 6.1 A szűrő modell

Mint ahogy azt korábban láthattuk, az elsőfokú aluláteresztő szűrő nem bizonyult elégségesnek a DAQ kártyával generált jel kisimításában, így szükség volt egy másodfokú szűrőre. Továbbá ahhoz, hogy tesztelni tudjuk az átviteli függvény automatizált felvételét szükség volt egy olyan szűrőre, amelynek ismert az átviteli karakterisztikája. Erre a célra tökéletesen megfelelt a sajátkezűleg tervezett és épített másodfokú, aktív aluláteresztő szűrő.

Az általános átviteli függvény, zérus-pólus elrendezés és közelítő amplitúdó Bode-diagram a 6.1 ábrán látható.



6.1 ábra: Másodfokú aluláteresztő szűrő [8].

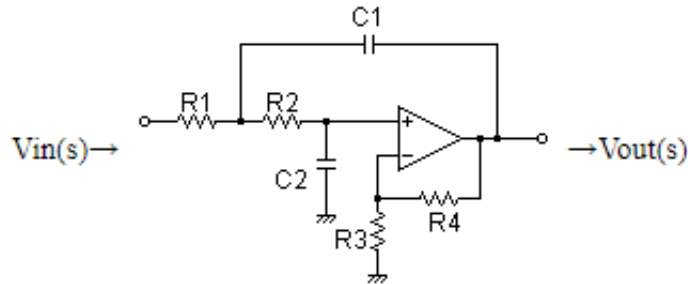
a. átviteli függvény b. zérus-pólus elrendezés c. Bode-diagram frekvenciamenete

A passzív szűrőkkel kapcsolatban számos hátrányos tulajdonság merül fel. Az induktivitás nagy helyfoglalású és viszonylag nehezen megvalósítható, drága alkatrész. Viszont induktivitás nélkül, csak R, C elemeket alkalmazva passzív hálózattal nem lehet konjugált komplex átviteli pólusokat létrehozni. További hátrány, hogy a passzív szűrők csak egyben, részekre bonthatóság nélkül tervezhetők, vizsgálhatók. A műveleti erősítők alkalmazásával fenti hátrányok kiküszöbölhetők. A műveleti erősítőt R-C hálózattal visszacsatolva, konjugált komplex pólusokat tudunk megvalósítani, induktivitás nélkül [8].

Számos olyan áramköri megoldás létezik, amikben az áramköri topológia megváltoztatása nélkül, a beépített alkatrészek típusának (R vagy C) megváltoztatásával

különböző típusú alaptagok valósíthatók meg. Ilyen áramköri kialakítás a Sallen-Key alaptag is [8].

A Sallen-Key alaptaggal megvalósított másodfokú aluláteresztő szűrő kapcsolási rajza a 6.2 ábrán látható.



6.2 ábra: Sallen-Key másodfokú aluláteresztő szűrő

A szűrő átviteli függvénye meghatározható az erősítő bemeneteire eső feszültségekre alkalmazott szuperpozíció tételével:

$$\frac{U_{ki}}{U_{be}} = A_0 \frac{1}{1 + s[R_1 C_1 (1 - A_0) + C_2 (R_1 + R_2)] + s^2 R_1 R_2 C_1 C_2} \quad (8)$$

$$\text{ahol } A_0 = \frac{R_3 + R_4}{R_3} \quad (9)$$

Az általános alakkal összevetve az alábbi megfeleltetések adódnak:

$$\omega_0 = \frac{1}{\sqrt{R_1 R_2 C_1 C_2}} \quad (10)$$

$$\frac{1}{Q\omega_0} = R_1 C_1 (1 - A_0) + C_2 (R_1 + R_2) \quad (11)$$

## 6.2 Szűrő tervezés

A gerjesztő jelet előállító DAQ kártya korlátait figyelembe véve a szűrő törésponti frekvenciáját választhatjuk  $f_0 = 800$  Hz-nek. Az alkalmazás szempontjából szükségtelen, hogy az erősítés a törésponti frekvenciánál kiemeljen, így kézenfekvő a Butterworth vagy más néven maximálisan lapos karakterisztikájú szűrőtípust választani, amely jósággi tényezője  $Q = 0,707$ .

A tervezést valamelyest egyszerűsítette  $R_1$  és  $R_2$ , illetve  $C_1$  és  $C_2$  azonos értékűként való kezelése, így a (10) és (11) egyenletek a következőképp módosultak:

$$ha R_1 = R_2 \text{ és } C_1 = C_2 \quad (12)$$

$$\omega_0 = \frac{1}{(R_1 C_1)} \quad (13)$$

$$\frac{1}{Q\omega_0} = R_1 C_1 (1 - A_0) + 2C_1 R_1 \quad (14)$$

A (13) és (14) egyenletből pedig levezethető, hogy

$$Q = \frac{1}{3 - A_0} \quad (15)$$

Így  $A_0 = 1,5859$ -re adódik.

### 6.3 Szűrő realizálás

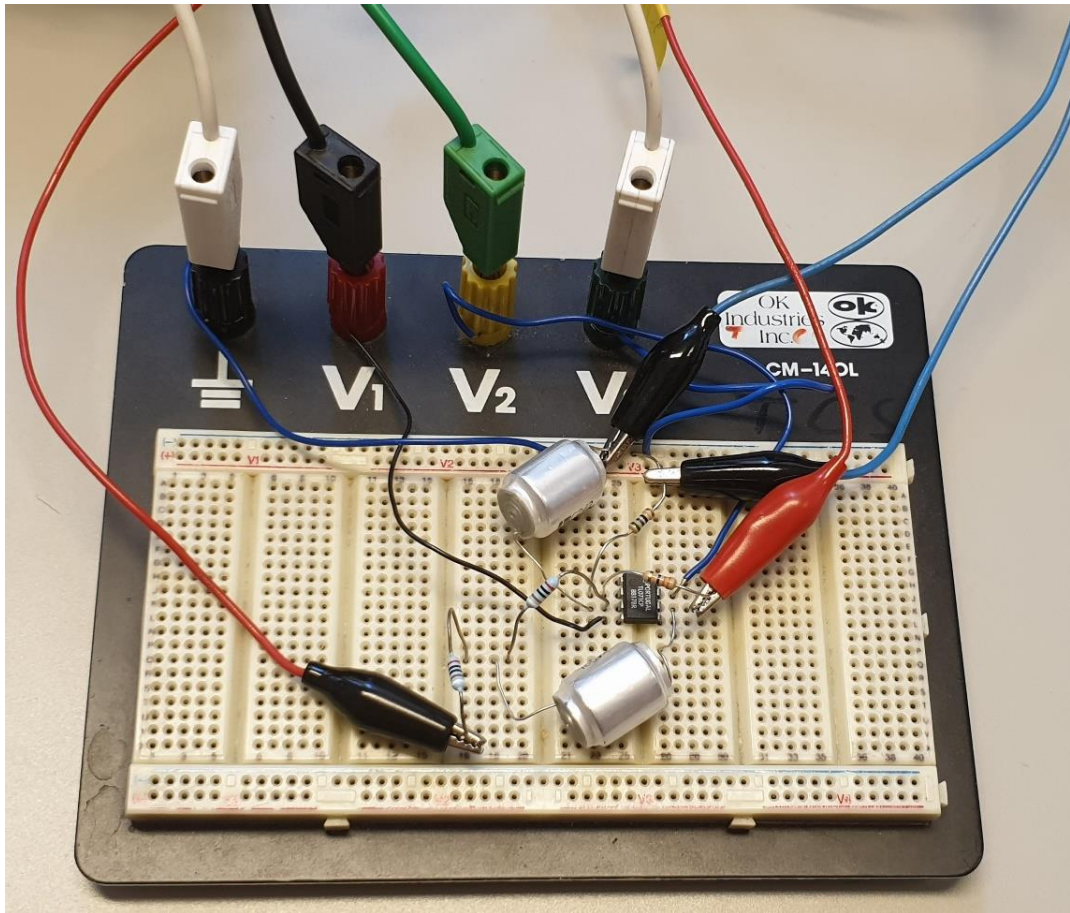
Szabványos alkatrészekkel nem lehet olyan szűrőt készíteni, amely tulajdonságai a tervezéskor specifikált értékekkel pontosan megegyeznének, de igyekeznünk kell azokat minél jobban megközelíteni.  $C_1 = C_2 = 220 \text{ nF}$ ,  $R_1 = R_2 = 910 \text{ } \Omega$  értékűre választásával a törésponti frekvencia  $f_0 \approx 795 \text{ Hz}$ -re esett. Az egyenáramú erősítés pedig  $A_0 = 1,589$ -nek adódott,  $R_3 = 56 \text{ } \Omega$  és  $R_4 = 33 \text{ } \Omega$  értékű ellenállás beépítésével.

A műveleti erősítők közül a Texas Instruments TL071-es integrált áramkör ideális választásnak bizonyul. Ez egy alacsony zajtényezővel, alacsony offset feszültséggel és bias árammal rendelkező eszköz, amely a 3 MHz-es sáv szélességével megfelel a kívánt alkalmazásra.

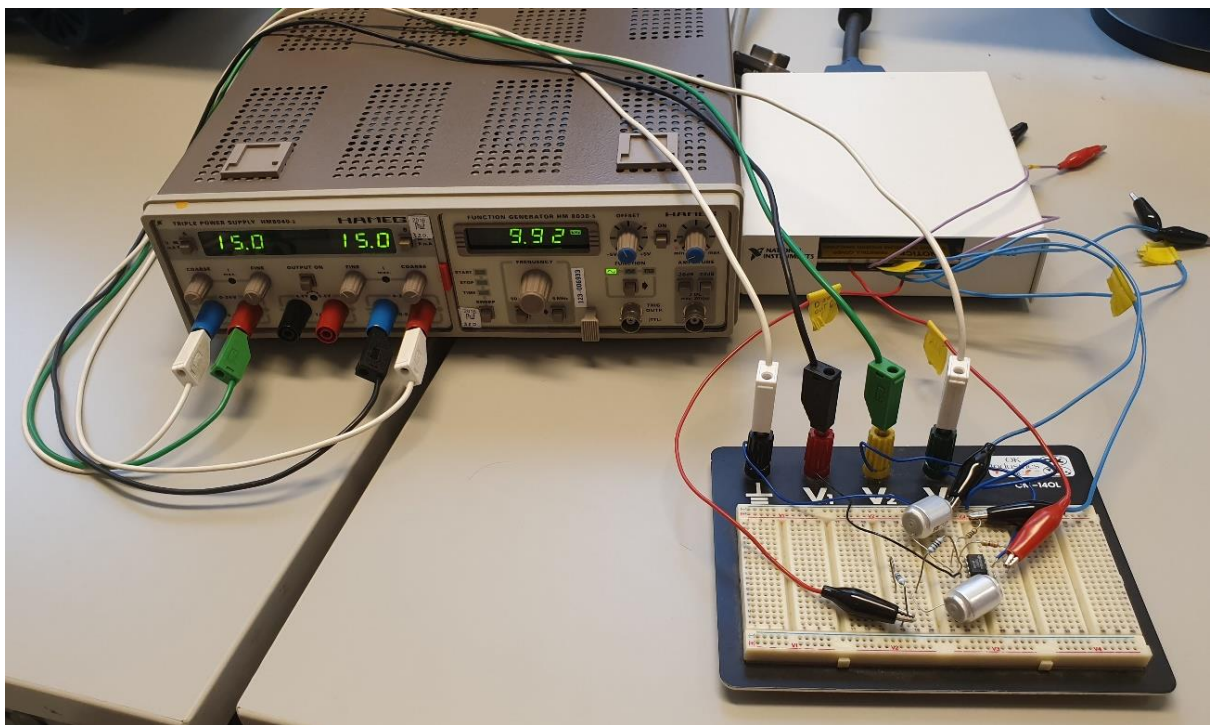
Mivel semmi nem indokolta, hogy az alkatrészek nyomtatott huzalozású lemezre legyenek felszerelve, így a szűrő egy breadboardon épült meg, ahol könnyebb és gyorsabb volt a beépítés és a későbbiekben könnyebben hozzá lehetett férni az egyes elemekhez. Szimmetrikus tápfeszültségként pedig az IC adatlapján ajánlott  $\pm 15 \text{ V}$  került beállításra.

A 6.3 ábrán és a 6.4 ábrán a megépített szűrő, illetve a szűrő vizsgálatokor használt műszer konstrukció tekinthető meg.





6.3 ábra: a realizált szűrő



6.4 ábra: a szűrő összekötve a táppal és a DAQ kártyával

## 6.4 Hibaanalízis

Az alkatrészek kiválasztása és a szűrő megépítése után érdemes hibaanalízist végezni az egyenáramú erősítésre és a törésponti körfrekvenciára. Ennek ismeretében a későbbi vizsgálati eredménykehez lesz viszonyítási alapunk. Mindkét paraméter worst case hibáját fogjuk számítani.

### 6.4.1 $A_0$ worst case hibája

A (9) egyenletből tudhatjuk, hogy  $A_0$  csak  $R_3$  és  $R_4$  értékétől függ. Mindkét ellenállásnak azonos toleranciával rendelkezik, ez a rájuk festett színekódokról leolvasható. A hiba értéke:

$$h_{R_3} = h_{R_4} = \frac{\Delta R_3}{R_3} = \frac{\Delta R_4}{R_4} = \pm 5\% \quad (16)$$

A worst case hiba kiszámításához szükséges előbb az érzékenység meghatározása. A két ellenállásra ezek a következők:

$$c_{R_3} = \frac{\partial A_0}{\partial R_3} = -\frac{R_4}{R_3^2} \quad (17)$$

$$c_{R_4} = \frac{\partial A_0}{\partial R_4} = \frac{1}{R_3} \quad (18)$$

Az érzékenységek ismeretében meghatározható a két ellenállásra vonatkoztatott abszolút hiba, amelyek:

$$\Delta A_{0R_3} = c_{R_3} \Delta R_3 = -\frac{R_4}{R_3^2} \Delta R_3 \quad (19)$$

$$\Delta A_{0R_4} = c_{R_4} \Delta R_4 = \frac{\Delta R_4}{R_3} \quad (20)$$

Ezek után már kiszámítható a worst case hiba:

$$\begin{aligned} \left| \frac{\Delta A_0}{A_0} \right|_{w.c.} &= \left| -\frac{R_4 R_3^2}{\Delta} R_3 \frac{R_3}{R_3 + R_4} \right| + \left| \frac{\Delta R_4}{R_3} \frac{R_3}{R_3 + R_4} \frac{R_4}{R_4} \right| \\ &= \frac{\Delta R_3}{R_3} \frac{R_4}{R_3 + R_4} + \frac{\Delta R_4}{R_4} \frac{R_4}{R_3 + R_4} = \frac{R_4}{R_3 + R_4} 2h_{R_3} \\ &= 0,0371 \end{aligned} \quad (21)$$

A (21) egyenlet alapján  $A_0 = 1,5859 \pm 0,0371$ , azaz  $A_{0\min} = 1,5488$  és  $A_{0\max} = 1,623$ .

## 6.4.2 $\omega_0$ worst case hibája

A (10) egyenlet szerint az  $\omega_0$  törésponti körfrekvencia az  $R_1$ ,  $R_2$ ,  $C_1$  és  $C_2$  paraméterektől függ. Az ellenállások és kondenzátorok páronként azonos értékűek és típusúak, így a toleranciájuk is megegyezik. A két 910  $\Omega$ -os ellenállás ugyan azt a színkódot használja, mint a 33 és 56  $\Omega$ -os, ezt leolvastva a hiba:

$$h_{R_1} = h_{R_2} = \frac{\Delta R_1}{R_1} = \frac{\Delta R_2}{R_2} = \pm 2\% \quad (22)$$

A kondenzátorokon is szerepel egy tolerancia kód, ezeknek hibája:

$$h_{C_1} = h_{C_2} = \frac{\Delta C_1}{C_1} = \frac{\Delta C_2}{C_2} = \pm 10\% \quad (23)$$

Ezúttal is elsőként az érzékenységeket kell meghatározni. Ezek a következők:

$$c_{R_1} = \frac{\partial \omega_0}{\partial R_1} = -\frac{1}{2} (R_1 R_2 C_1 C_2)^{-\frac{3}{2}} R_2 C_1 C_2 \quad (24)$$

$$c_{R_2} = \frac{\partial \omega_0}{\partial R_2} = -\frac{1}{2} (R_1 R_2 C_1 C_2)^{-\frac{3}{2}} R_1 C_1 C_2 \quad (25)$$

$$c_{C_1} = \frac{\partial \omega_0}{\partial C_1} = -\frac{1}{2} (R_1 R_2 C_1 C_2)^{-\frac{3}{2}} R_1 R_2 C_2 \quad (26)$$

$$c_{C_2} = \frac{\partial \omega_0}{\partial C_2} = -\frac{1}{2} (R_1 R_2 C_1 C_2)^{-\frac{3}{2}} R_1 R_2 C_1 \quad (27)$$

Az érzékenységek alapján megállíthatóak az abszolút hibák, melyeket a következő összefüggések írnak le:

$$\Delta \omega_{0R_1} = -\frac{1}{2} (R_1 R_2 C_1 C_2)^{-\frac{3}{2}} R_2 C_1 C_2 \Delta R_1 \quad (28)$$

$$\Delta \omega_{0R_2} = -\frac{1}{2} (R_1 R_2 C_1 C_2)^{-\frac{3}{2}} R_1 C_1 C_2 \Delta R_2 \quad (29)$$

$$\Delta \omega_{0C_1} = -\frac{1}{2} (R_1 R_2 C_1 C_2)^{-\frac{3}{2}} R_1 R_2 C_2 \Delta C_1 \quad (30)$$

$$\Delta \omega_{0C_2} = -\frac{1}{2} (R_1 R_2 C_1 C_2)^{-\frac{3}{2}} R_1 R_2 C_1 \Delta C_2 \quad (31)$$

Mindezek ismeretében felírható a worst case hiba is:

$$\begin{aligned}
\left| \frac{\Delta\omega_0}{\omega_0} \right|_{w.c.} &= \left| -\frac{1}{2} (R_1 R_2 C_1 C_2)^{-\frac{3}{2}} R_2 C_1 C_2 \Delta R_1 (R_1 R_2 C_1 C_2)^{\frac{1}{2}} \right| + \dots & (32) \\
&+ \left| -\frac{1}{2} (R_1 R_2 C_1 C_2)^{-\frac{3}{2}} R_1 R_2 C_1 \Delta C_2 (R_1 R_2 C_1 C_2)^{\frac{1}{2}} \right| \\
&= \frac{1}{2} \frac{R_2 C_1 C_2 \Delta R_1}{R_1 R_2 C_1 C_2} + \dots + \frac{1}{2} \frac{R_1 R_2 C_1 \Delta C_2}{R_1 R_2 C_1 C_2} \\
&= \frac{1}{2} \frac{\Delta R_1}{R_1} + \frac{1}{2} \frac{\Delta R_2}{R_2} + \frac{1}{2} \frac{\Delta C_1}{C_1} + \frac{1}{2} \frac{\Delta C_2}{C_2} = \frac{1}{2} 2h_{R_1} + \frac{1}{2} 2h_{C_1} \\
&= h_{R_1} + h_{C_1} = 12\%
\end{aligned}$$

A (32) egyenlet eredménye alapján  $\omega_0 = 4995 \text{ rad/sec} \pm 12\%$ , azaz  $\omega_{0\min} = 4395,6 \text{ rad/sec}$  és  $\omega_{0\max} = 5594,4 \text{ rad/sec}$ . Az előbbi értékeket frekvenciára átírva  $f_0 \approx 795 \text{ Hz} \pm 12\%$ ,  $f_{0\min} \approx 700 \text{ Hz}$  és  $f_{0\max} \approx 890 \text{ Hz}$ . Tehát a megépített szűrő törésponti frekvenciája 700 és 890 Hz között helyezkedhet el. Ez egy igen nagy tartomány, amely a kondenzátorok jelentős pontatlanságából adódik.

## 6.5 Stabilitásvizsgálat

### 6.5.1 A zárt rendszer pólusai alapján

A szűrő megépítése előtt szükséges különböző stabilitási vizsgálatok elvégzése, hogy bebizonyosodjon a rendszer megfelelő működése.

A kiválasztott alkatrészek értékeiből az áramkör átviteli függvénye a következőnek adódott:

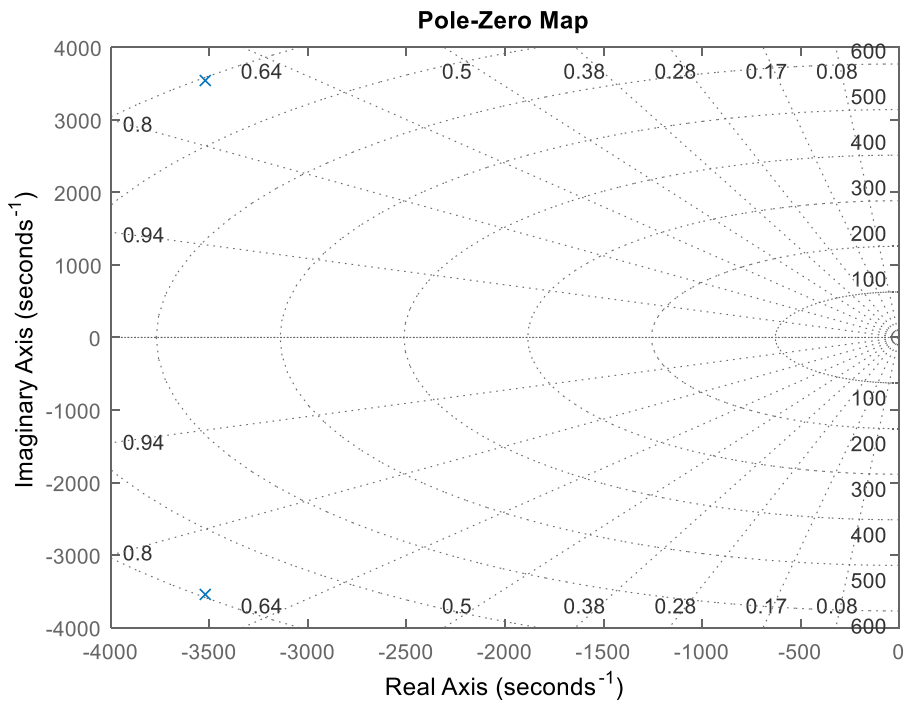
$$A(s) = \frac{39670619,09}{249500774,9 + 7042,9s + s^2} \quad (33)$$

Az átviteli függvény ismeretében már meghatározható a rendszer zérus-pólus elrendezése. MATLAB-ban az  $A(s)$  függvény modellezése után a `pzmap` utasítással mind az elrendezést, mind a zérusok és pólusok pontos értékét megkapjuk.

Mivel a másodfokú aluláteresztő szűrőről beszélünk, így azt várjuk, hogy nem lesznek zérusok és egy, komplex konjugált póluspár lesz. A 6.5 ábrán látszik, hogy ez valóban, így van. A póluspár és az átviteli függvény gyöktényezős alakja a következőnek adódott:

$$p_{(1,2)} = -3521,5 \pm 3542,5j \quad (34)$$

$$A(s) = \frac{39670619,09}{(s + 3521,5 + 3542,5j)(s + 3521,5 - 3542,5j)} \quad (35)$$

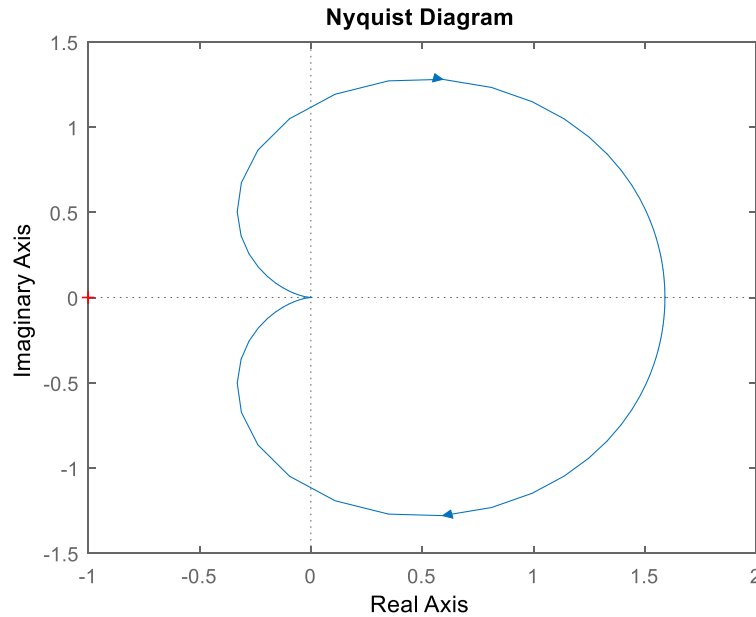


6.5 ábra: Pólus-zérus elrendezés

A komplex konjugált póluspár a bal félsíkra esik, azaz a pólusok valós része negatív, tehát ez alapján a rendszer stabilis [9].

### 6.5.2 A Nyquist stabilitás vizsgálata

A MATLAB nyquist utasításával elkészíthető a Nyquist diagram, aminek segítségével ellenőrizhető a Nyquist stabilitási kritérium. Mivel az előző pontban kiderült, hogy a rendszernek nincs labilis pólusa, ezért alkalmazható az egyszerűsített kritérium is. Eszerint a zárt rendszer stabilis, ha a felnyitott rendszer Nyquist diagramja nem veszi körül a  $(-1+0j)$  pontot [9]. Mint ahogy az a 6.6 ábrán látható ez nem történik meg, így a kritérium teljesül.



6.6 ábra: Nyquist diagram

### 6.5.3 A fázistartalék számításával

A fázistartalék kiszámítása azért hasznos, mert ennek ismeretében nem csak arra a kérdésre tudunk választ adni, hogy stabil-e a vizsgált rendszerünk, hanem azt is megtudjuk mondani, hogy milyen messze van a stabilitás határhelyzetétől. A fázistartalék kvantitatív módon megadja azt, hogy a vágási körfrekvencián működő rendszer fázisszögét mennyivel lehet még növelni, ahhoz, hogy a stabilitás határhelyzetét jelentő  $-180^\circ$ -os fázisszöget elérjük. Meghatározása a

$$\varphi_t = \varphi(\omega_c) + 180^\circ \quad (36)$$

egyenlet szerint történik, ahol  $\varphi(\omega_c)$  a vágási körfrekvencián felvett fázisszög [9].

Mindehhez először a vágási körfrekvenciát kell meghatározni. Ezen a körfrekvencián a rendszer erősítése 0 dB, azaz az átviteli függvény abszolútértéke egy.

$$|A(j\omega)| = 1 \quad (37)$$

$$\frac{39670619,09}{\sqrt{3521,5^2 + (\omega_c + 3542,5)^2} \sqrt{3521,5^2 + (\omega_c - 3542,5)^2}} = 1 \quad (38)$$

A (38) egyenlet megoldásaként  $\omega_c = 5566,9484$  rad/s adódik, ami  $f_c = 886$  Hz. A vágási frekvencián felvett fázisszöget, úgy kaphatjuk meg, hogy vesszük ezen a

frekvencián az átviteli függvény argumentumát. Ezek után már kiszámítható a fázistartalék is.

$$A(j\omega_c) = -0,1523 - 0,9883j \quad (39)$$

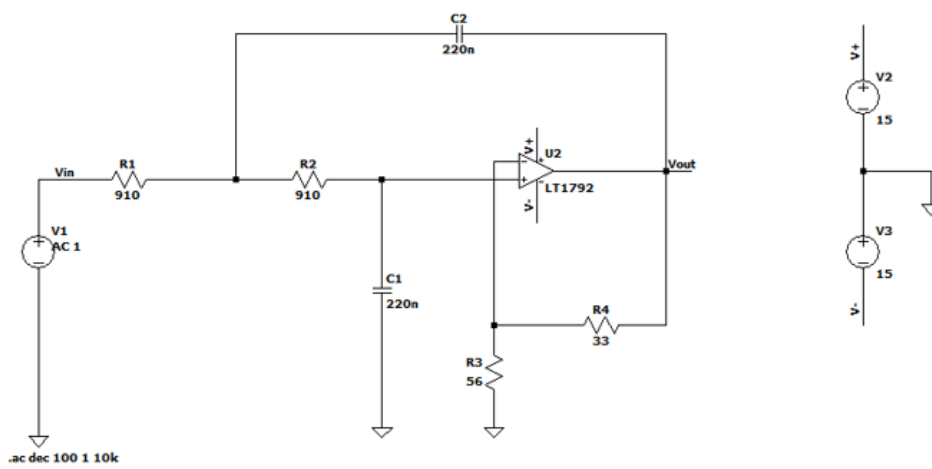
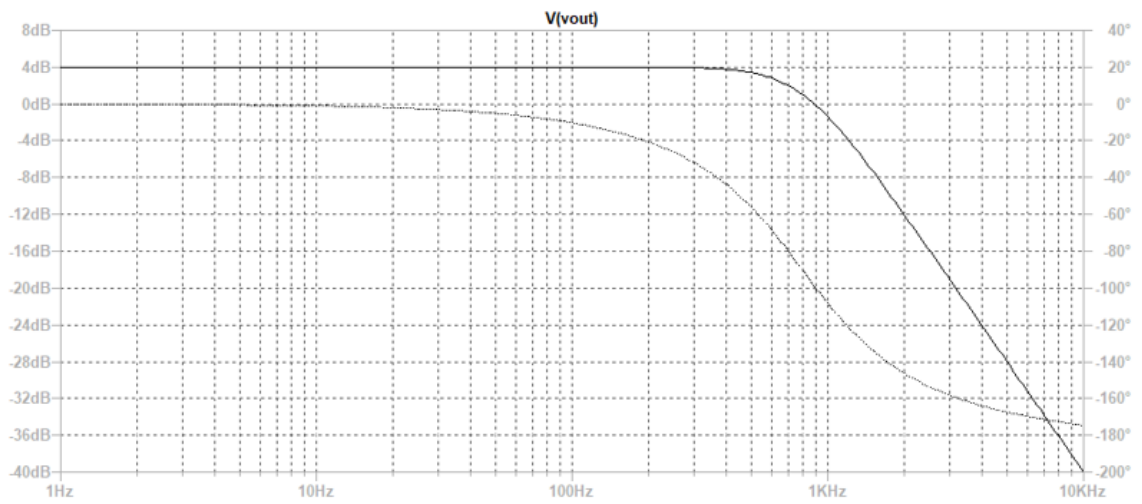
$$\varphi(\omega_c) = \arg(A(j\omega_c)) = -98,7585^\circ \quad (40)$$

$$\varphi_t = \varphi(\omega_c) + 180^\circ = 81,24^\circ \quad (41)$$

## 6.6 Szimuláció

A szűrő működéséről készült egy szimuláció is az LTspice szimulációs szoftverben. A kapcsolat megrajzolását követően többek között a váltakozó áramú analízis elkészítésére is lehetőség nyílik. Ezután pedig a szimulációban ellenőrizhető a kiszámított törésponti frekvencia és fázistartalék helyessége is. A szoftver sajnos nem tartalmazta a kiválasztott műveleti erősítő modelljét, így azt egy másik, hasonló tulajdonságú erősítővel helyettesítettem.

A szimulált Bode-diagramról leolvasott törésponti frekvencia és fázistartalék értékek megegyeznek a korábban kiszámítottokkal.



6.7 ábra: a kapcsolási rajz és a szimulált Bode-diagram



## 7 Az átviteli függvény automatizált felvétele

### 7.1 Hasonlóságok és különbségek

Az átviteli függvény meghatározását megvalósító LabVIEW szoftverre több verzió is született. Általános felépítésüket tekintve az egyes verziók megegyeznek. Mindegyik kirajzolja az amplitúdó Bode-diagrammot, kiszámítja a törésponti frekvenciát, meghatározza az egyenáramú erősítést és ezek alapján megadja az átviteli függvényt.

Az amplitúdókarakterisztika kirajzolásához egy kétdimenziós tömbre van szükségünk, amelyben az első index a vizsgált frekvenciákat tartalmazza, a második pedig az ezeken mért amplitúdókat decibelben. A kívánt diagrammot úgy kaphatjuk meg, hogy ábrázoljuk a tömb elemeit, úgy, hogy az első indexet a koordináta-rendszer abszcisszatengelyéhez, a másodikat pedig az ordinátatengelyhez rendeljük.

A törésponti frekvencia meghatározása a zérushely keresés elvét követi. Adott a vizsgált frekvenciák és mért amplitúdók tömbje és ebben szeretnénk megtalálni azt a frekvenciát, ahol a diagram eléri a -3dB-es pontot. Mivel nem lehetünk abban biztosak, hogy nincs a rendszerben erősítés és a -3 dB-es pont valóban -3 dB-nél van ezért előbb szükséges megkeresni a tömb legnagyobb amplitúdó értékét és az annál hárommal kisebb értéket kutatni. Mivel aluláteresztő szűrőt vizsgálunk ezért tudjuk, hogy a zárótartomány csak negatív meredekségű lehet, tehát teljesülni kell annak, hogy az amplitúdó értékek tömbjének  $(i-1)$ . -ik eleme nagyobb vagy egyenlő legyen a -3 dB-es pontnál, az  $i$ .-ik eleme pedig kisebb vagy egyenlő. Ha megvan a -3 dB-es pont, a kétdimenziós tömbből megkaphatjuk a hozzá tartozó törésponti frekvenciát is.

Az erősítés és a törésponti frekvencia ismeretében már felírható az átviteli függvény. Feltételezzük, hogy a vizsgált szűrő Butterworth típusú és így ismert a jósági tényezője. Az adatokat a megfelelő VI-nak megadva előállítható az átviteli függvény modellje, egy másik ezt megkapva pedig képes ezt nekünk megjeleníteni.

Az egyes változatok közötti különbséget a használt gerjesztés jelenti. Ez esetenként befolyásolja az egyenáramú erősítés meghatározását és a Bode-diagram amplitúdó értékeinek kiszámítási módját. A feladatot mindegyik elvégzi, azt, hogy ezt melyik, milyen hatékonysággal teszi meg, a későbbiekben láthatjuk.

## 7.2 Szinuszos gerjesztés

Mint ahogy az ismert, a szinuszos jelnek egyetlen hasznos frekvenciakomponense van. Ezért, ha ezt a jelet szeretnénk használni az átviteli függvény felvételére minden egyes vizsgálni kívánt frekvencián rá kell adnunk egy szinuszos jelet a rendszerre. Mivel a hasznos komponensen kívül zavaró komponensek is lehetnek, ezért a kimeneti oldal olvasásakor meg kell határozni a legnagyobb amplitúdóval rendelkező tagot, amely a lényegi információt hordozza. Ha ez megvan, a kimeneten mért amplitúdó és a bemenetre adott jel amplitúdójának hányadosát a

$$20 \log \left( \frac{U_{ki}}{U_{be}} \right) = dB \quad (42)$$

összefüggés által átszámítjuk decibelbe és ezt elmentjük az aktuális frekvenciával együtt a tömbünkbe.

Az  $A_0$  erősítést szintén a bemeneti és kimeneti amplitúdók hányadosából kaphatjuk meg. Ezt elég csak egyszer kell kiszámolni és értelemszerűen olyan frekvencián, ahol a szűrő nem csillapít. Mivel aluláteresztő szűrőrőt vizsgálunk, ezért érdemes ezt az első vizsgált frekvencián megtenni.

A szinuszos jel használata ugyan egyszerűnek és logikusnak tűnik, azonban egy súlyos problémát hordoz magában. Nagyon lassú. Minden vizsgálandó frekvencián újra és újra ráadni a rendszerre a gerjesztést, beolvasni a szűrő kimenetén megjelenő értékeket és utána feldolgozni az adatokat időigényes feladat a DAQ kártya számára. Például, ha az 1 – 4000 Hz-es tartományt 1 Hz-es felbontásban szeretnénk átfésülni, akkor négyezer iteráció lefutását kéne megvárjuk, ami csak hosszú-hosszú percek eltelével történne csak meg.

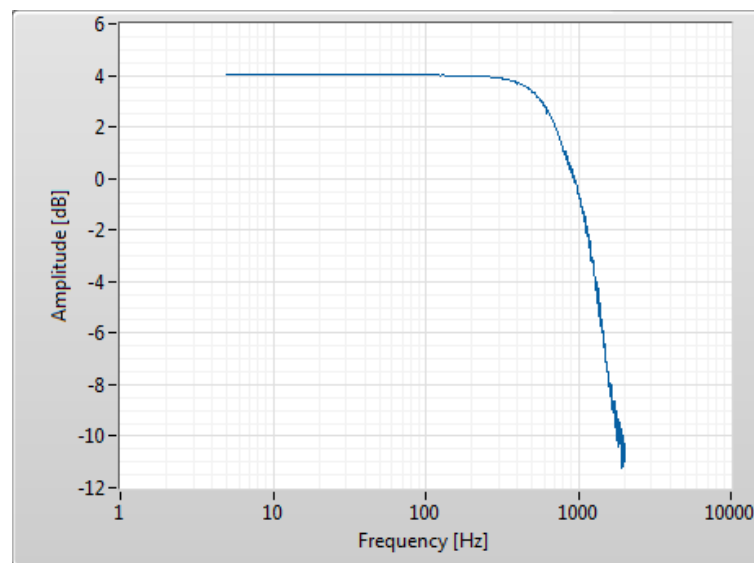
A felbontás csökkentése, azaz a frekvencialépés növelése felmerülhet, mint lehetséges megoldás a futási idő csökkentésére de következésképpen ez az eredmény pontosságának romlásával jár. Minél kisebb a felbontás, annál hamarabb lefut a program és annál pontatlanabb lesz az eredmény. Másik mód a vizsgált frekvenciatartomány szűkítése, de ez leginkább akkor hatásos, ha van valami fogalmunk arról merre keressük a törésponti frekvenciát és az amplitúdókarakterisztika se biztos, hogy teljes képet fog adni.

Továbbá át lehetne alakítani a szoftvert, úgy, hogy az a törésponti frekvencia megtalálása után rögtön megálljon. A -3 dB-es pont keresése minden ciklusban

megnövelné a számításigényt és az amplitúdó Bode-diagramm megjelenítését is elvesztenénk, de mivel a kártya portjainak használata erőforrásigényesebb feladat és így ezek használatára kevesebbszer kerülne sor, ezért valamelyest csökkenne a program futási ideje.

Mivel a következőként bemutatott, másik két verzió jóval hatékonyabban látja el a feladatát és nem szenved a fentiekben leírt problémákban, ezért a fenti problémákkal terhelt megoldások nem kerültek implementálásra.

A 7.1 ábrán látható a szinuszos gerjesztés használatával készült amplitúdó Bode-diagram.



7.1 ábra: szinuszos gerjesztéssel készült amplitúdó Bode-diagramm

### 7.3 Négyszög gerjesztés

A négyszög jellel való gerjesztésnél kissé másfajta megközelítés használatos, mint a szinuszosnál. Ebben az esetben kihasználjuk azt, hogy az 50%-os kitöltési tényezőjű négyszögjel spektrumában az alapharmonikuson kívül az egész számú páratlan felharmonikusok jelennek meg. Ezért a jelet egy multiszínuszhoz hasonlóan lehet kezelni, vagyis egyetlen jelet használunk, több hasznos frekvenciakomponenssel. A különbség az, hogy az egyes komponensek nem egyforma amplitúdóval rendelkeznek, hanem  $1/x$ -es burkoló szerint csökkennek. Ennek köszönhetően nincs szükség arra, hogy újra meg újra gerjesszük a rendszert, elég ezt egyszer megtenni és a bemenő és kijövő jel spektrumát összehasonlítani.

LabVIEW-ban lehetőségünk van arra, hogy egy jel spektrumát alkotó elemeire bontsuk. A megfelelő beállítással az egyes frekvenciakomponensek decibelben kifejezett amplitúdóértékeit egy tömb elemeiként kapjuk meg. Ha a szűrt amplitúdó tömb elemeiből frekvencia helyesen kivonjuk a szűretlen tömb elemeit, akkor a

$$\log x - \log y = \log \left( \frac{x}{y} \right) \quad (43)$$

logaritmikus azonosságot kihasználva az amplitúdó karakterisztikához juthatunk.

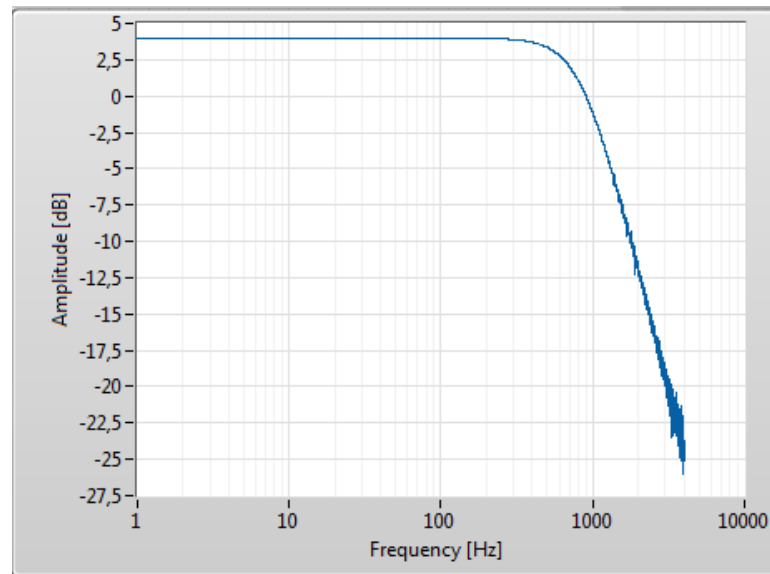
Az erősítés kiszámításához a szűrt amplitúdó tömb és a szűretlen amplitúdó tömb első elemének különbségét kell venni majd ebből visszszámítani az eredeti arányt a

$$10^{\frac{\text{Szűrt} - \text{Szűretlen}}{20}} = A_0 \quad (44)$$

kifejezés szerint.

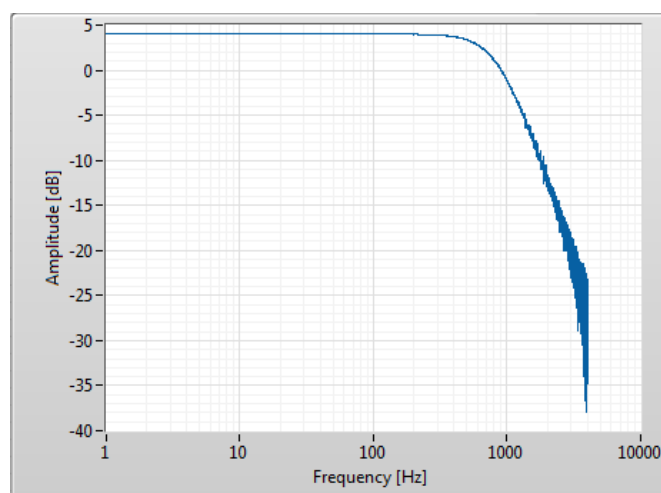
A 7.2 ábrán látható amplitúdókarakterisztika 1 Hz frekvenciájú négyszögjellel készült, így a vizsgált frekvenciatartományban minden második frekvencia értéken van egy hasznos komponensünk. Megfigyelhető, hogy 1 kHz fölött egyre inkább megnő a spektrum varianciája. Mivel a négyszögjel frekvenciakomponenseinek amplitúdója  $1/x$ -es burkoló szerint csökken, így ezeken a frekvenciákon a szűretlen jel komponenseinek amplitúdója már alapvetően igen kicsi, nagyjából -60 dBV. A szűrő karakterisztikáját tekintve már a záró tartományban vagyunk, ahol a szűrő már -40 dB/dekád-os meredekséggel csillapít. Ennek következtében szűrt jelnél ugyan ezek az amplitúdók még kisebbek, körülbelül -90 dBV-esek lesznek. Ezek a nagyon kicsi értékek pedig a variancia növekedését okozzák.

Mindez elkerülhető lenne a négyszögjel frekvenciájának növelésével, hiszen ekkor a kHz-es tartományba nagyobb amplitúdójú komponensek esnének és a szűrés után nem kapnánk ennyire kicsi amplitúdókat. Ámbár ennek következtében az átviteli függvény meghatározása is pontatlanabbá válna.



7.2 ábra: négyzög gerjesztéssel készült amplitúdó Bode-diagramm

További kísérleteket lehet végezni a jel kitöltési tényezőjének változtatásával. Ha 50% helyett 25%-os vagy 75%-os kitöltési tényezőt használunk a jel spektrális képében az egész számú páros felharmonikusok helyett csak minden negyedik harmonikus hiányzik [10]. Ez azt jelenti, hogy finomabb lehet a frekvencia felbontásunk hiszen másfélszer több hasznos frekvenciakomponensünk van így a lefedett frekvenciatartományban. Csakhogy, amint ez a 7.3 ábrán is látható, ennek következtében a variancia növekedés jóval jelentősebbé válik, így, ha tiszta jelleggörbét szeretnénk kapni ajánlatos inkább a jól megszokott szimmetrikus négyzög jelet használni.



7.3 ábra: 25%-os kitöltési tényezőjű négyzög gerjesztéssel készült amplitúdó Bode-diagramm

Mivel a gerjesztést csak egyszer kell ráaduk a rendszerre, így ezt elég a cikluson kívül megtennünk. A cikluson belül csak az adatok feldolgozásával kell foglalkozni. Mindez jelentősen gyorsabbá teszi a szoftvert a szinuszos gerjesztéshez képest. Ennek dacára a kHz-es tartományba tapasztalt problémák miatt inkább csak az 1 kHz-nél kisebb törésponti frekvenciájú szűrők vizsgálatára alkalmas.

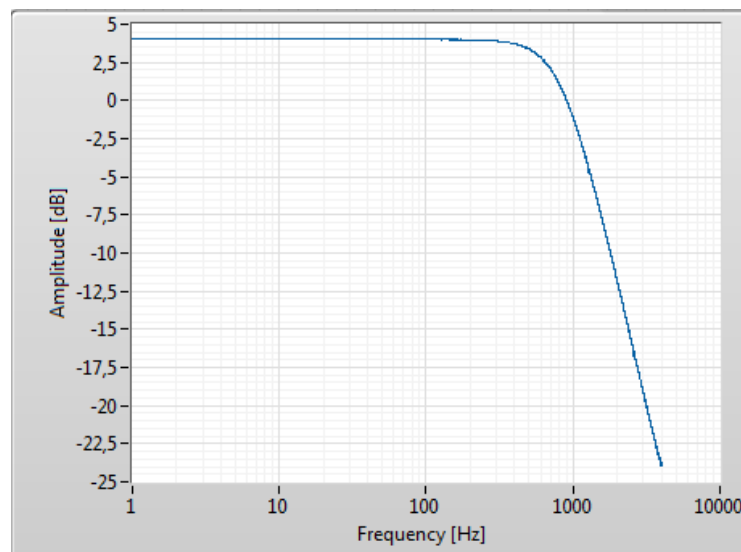
## 7.4 Multitone gerjesztés

A harmadik használt gerjesztésünket a multitone jel képezi. Ebben az esetben különböző frekvenciájú szinuszjelek összeadásával állítunk elő gerjesztő jelet, így a jel spektrumában az adott frekvenciákon azonos amplitúdójú komponenseket láthatunk.

Az eljárás nem sokat változott az előző esethez képest. Itt is a szűrt és a szűretlen amplitúdó értékek tömbjeivel dolgozunk és a szinuszos gerjesztéshez hasonlóan itt is csak a számunkra hasznos információval szolgáló komponenseket szabad figyelembe venni. Ez nem is meglepő tekintve, hogy jelünk sok szinusz jel összegéből tevődik össze.

A gerjesztés természetéből adódóan elég ezt is egyszer ráadni a rendszerre. Ennek következtében a négyzög jelhez hasonlóan, rövid időn belül jutunk az eredményhez.

A multitone jelet előállító VI igen kicsi frekvenciaközzel képes előállítani a hullámformát. 3 Hz-es lépésközzel a 7.4 ábrán is látható amplitúdó karakterisztikát kapjuk. A törésponti frekvenciát ugyanekkora pontossággal tudhatjuk meg. Az erősítés számítása pedig a négyzögjelnél leírtakkal megegyező módon történik.



7.4 ábra: multitone gerjesztéssel készült amplitúdó Bode-diagramm

## 7.5 Összegzés

Mind a három gerjesztés megismerése után biztosan kijelenthetjük, hogy a szinuszos gerjesztés használata a legkevésbé hatékony módszer az átviteli függvény meghatározására. Igen finom frekvenciafelbontással képes működni, viszont több különböző frekvencián is rá kell adni a vizsgálandó rendszerre, ami már-már kiváráthatatlan futásidőt eredményez. Emiatt pedig erősen alulmarad a másik két módszerrel szemben.

A további két gerjesztés, a négyszög és a multitone jel használata sok tekintetben hasonlít egymáshoz. Mindkettő jel több, számunkra hasznos frekvenciakomponenst tartalmaz, így lehetséges az átviteli függvény egy lépésben való meghatározása a szűrt és szűretlen jel spektruma alapján. Következésképpen pedig a futási idő is rövid lesz.

1 Hz-es négyszög jel alkalmazásával elég sok komponens áll rendelkezésre, így pontos eredményhez juthatunk. Azonban a hullámforma spektrumának természetéből adódóan a kHz-es frekvenciákon egyre kevésbé kapunk pontos eredményeket, így az amplitúdó Bode-diagramm egy részén megnő a variancia, illetve magasabb törésponti frekvenciájú szűrők esetén sem fogunk helyes eredményt kapni.

A kitöltési tényező megváltoztatásával finomítható a frekvenciafelbontás és ezzel pontosabban kiszámíthatjuk az átviteli függvényt amennyiben a vizsgált szűrő töréspontja nem a kilohertzes tartományban helyezkedik el. Ebben a tartományban ugyanis a szimmetrikus négyszögjelnél tapasztaltakhoz képest pontatlanabb eredményt kapunk és a spektrum varianciája is sokkal jobban megnő.

A multitone jel a többi gerjesztésnél bemutatott problémákban nem szenved és a teljes vizsgált frekvenciatartományban helyes karakterisztikát és pontos eredményt szolgáltat, amennyiben elvégezzük a feldolgozandó adatok szükséges előszűrését. Mindezek miatt pedig ez lesz a feladat elvégzésére leginkább alkalmas jel.

Elvi lehetőségként felmerülhet még a zajjal történő gerjesztés is, viszont a fehér zaj generálásának nehézségei és a használatával kapott eredmény feltételezhető pontatlansága miatt ez nem képezi jelen dolgozat részét.

## 8 A tervezett és implementált szűrő összehasonlítása

### 8.1 Prognózis

A szűrő tervezésekor kitűztük célként, hogy egy Butterworth típusú, másodfokú aluláteresztő szűrőt szeretnénk elkészíteni, amelynek törésponti frekvenciája 800 Hz-nél helyezkedik el és egyenáramú erősítése 1,59. Mint ahogy azt láthattuk szabványosított alkatrész elemekkel nem lehet megvalósítani egy pontosan ilyen áramkört, így igyekeztünk ezt minél jobban megközelíteni. A kiválasztott alkatrészek alapján a törésponti frekvenciának nagyjából 795 Hz-nek, az egyenáramú erősítésnek pedig 1,589-nek kellene lennie. Ez alapján pedig azt várjuk, hogy az átviteli függvény körülbelül a következőnek adódjon:

$$A(s) = \frac{1,589}{1 + 2,8228 \times 10^{-4} s + 4,008 \times 10^{-8} s^2} \quad (45)$$

Persze tapasztalhatunk ettől kisebb nagyobb eltéréseket, de ezeknek is a hibaanalízis során kiszámított határok között kell lenniük.

### 8.2 Az implementált szűrő átviteli függvénye

Az előző fejezetben leírtak alapján a szűrőnk átviteli függvényét multitone gerjesztéssel és - mivel elviekben törésponti frekvenciája a kilohertzes tartomány alatt helyezkedik el - négyszög gerjesztéssel is fel lehet venni. Mindkét esetben azonos tartományon belül mozognak a mért eredmények. A törésponti frekvencia 808-812 Hz körül mozog, az egyenáramú erősítés pedig nagyjából 1,585 és 1,592 között helyezkedik el. Ha 75%-os kitöltési tényezőjű négyszög jelet használunk az elméleti értékhez pár Hertz-el közelebbi értéket kapunk, nagyjából 805 Hz-et.

$$A(s) = \frac{1,5912}{1 + 2,7791 \times 10^{-4} s + 3,86 \times 10^{-8} s^2} \quad (46)$$

A (46) egyenletben egy multitone-nal kapott átviteli függvényt láthatunk, ekkor a szoftver szerint 810 Hz volt a törésponti frekvencia.

### 8.3 Értékelés

A kapott értékeket a tervezéskor számítottokkal összehasonlítva nem tapasztalunk nagy eltérést. Az erősítés közel megegyezik az elméleti értékkel, a törésponti frekvencia nagyjából 15 Hz-el csúszott feljebb és a két átviteli függvény is csaknem ugyanannyi.



A különbség nagyrészt a felhasznált alkatrészek pontatlanságával magyarázható. Az egyes elemek ellenállásának megmérése után kiderült, hogy a 910  $\Omega$ -os ellenállások valójában 905  $\Omega$ -osak, a 33  $\Omega$ -os inkább 33,5  $\Omega$  és az 56  $\Omega$  a valóságban 56,7  $\Omega$ . Az előbbi tűrése  $\pm 2\%$  míg az utóbbi kettőé  $\pm 5\%$ , így a mért értékek a hibahatáron belül helyezkednek el. A két kondenzátor pedig 220 nF helyett cirka 217 nF.

Ha ezekkel az új ellenállásokkal és kapacitásokkal újra számoljuk a törésponti frekvenciát valóban 810 Hz-et kapunk és az erősítés is 1,59-nek adódik. A (46) egyenletben szereplő átviteli függvény pedig jóformán megegyezik a program által számított

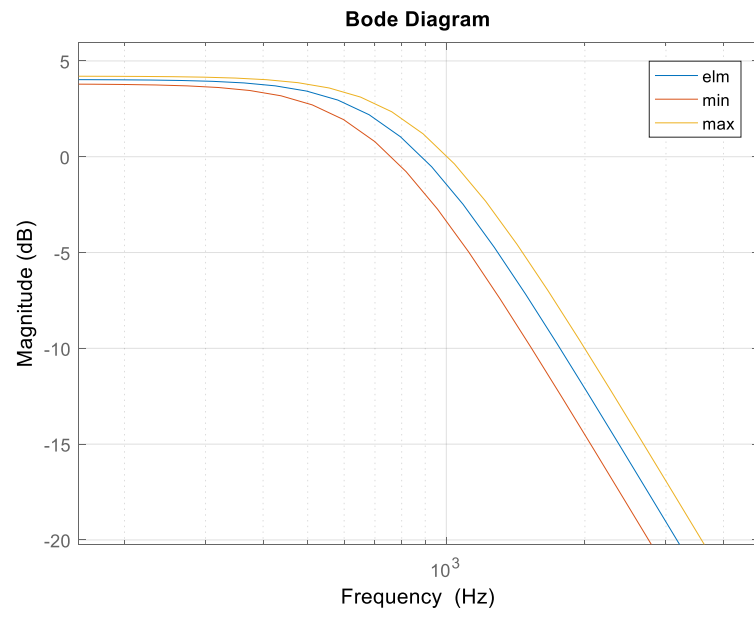
$$A(s) = \frac{1,5908}{1 + 2,769 \times 10^{-4} s + 3,8567 \times 10^{-8} s^2} \quad (47)$$

függvénnyel.

Elmondható, hogy a kimért értékek mind bőven a meghatározott hibahatárokon belül helyezkednek el. A rendelkezésre álló alkatrészekből sikerült olyan szűrőt készíteni, ami az eredetileg specifikált értékeket igen jól megközelíti.

A készített szoftver képes elég pontosan kimérni a szükséges adatokat és azokból előállítani az átviteli függvényt. A korábbi fejezet amplitúdókarakterisztikáiból az is látható, hogy nincs a töréspont környékén ugrás az átvitelben, tehát a kívánt Butterworth jelleg is teljesült.

A 8.1 ábrán az elméleti (*elm*), valamint a felső (*max*) és alsó (*min*) határt képező átviteli függvények Bode-diagramjai láthatóak. A *min* és *max* függvények a 346.4 fejezetben tárgyalt hiba analízisben kapott értékekből adódtak. Értelemszerűen a *min* függvény esetében az aluláteresztő függvény általános képletébe (6.1 ábra/a)  $A_{0min}$  és  $\omega_{0min}$ , a *max* függvény esetében  $A_{0max}$  és  $\omega_{0max}$  került behelyettesítésre.



**8.1 .ábra: Az elméleti, illetve a két határt képező átviteli függvények Bode-diagramjai**

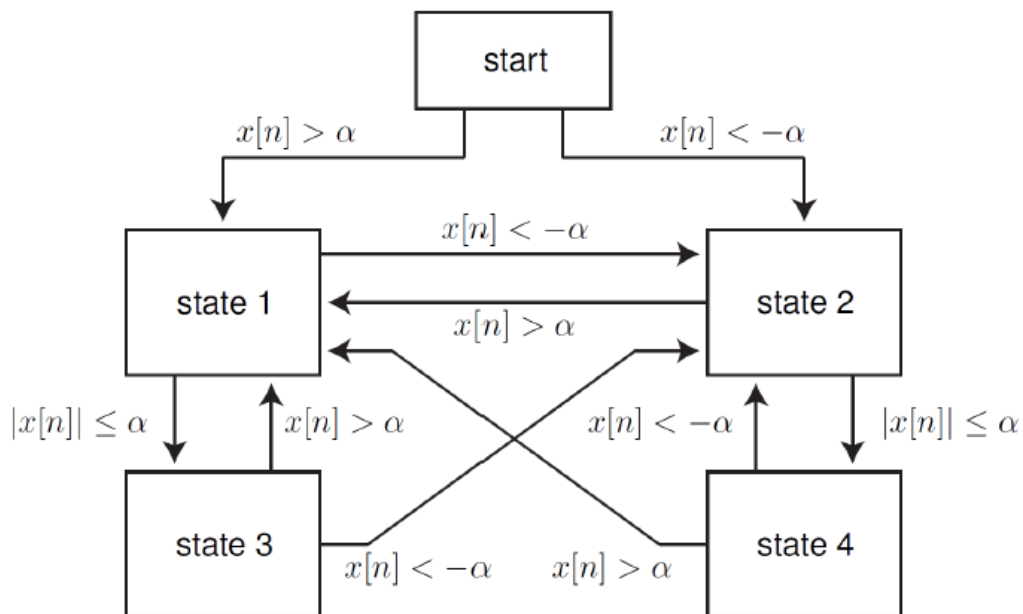
## 9 Kitekintés: Frekvencia és fázis becslés

Kitekintésként nézzünk meg egy frekvencia és fázis becslő algoritmust, amely az irodalomjegyzékben megjelölt dolgozat által javasolt nullátmenet detektáláson alapul. Segítségével bármilyen, zajjal terhelt periodikus jel frekvenciája és fázisa becsülhető. Működése a következőképp néz ki.

A jelet az alábbi alakban vizsgáljuk

$$z[n] = b_0 \exp(j(\omega_0 n T + \theta_0)) \quad (48)$$

ahol  $n$  a minta indexét,  $\omega_0$  a körfrekvenciát,  $b_0$  az amplitúdót,  $T$  a mintavételezés periódusidejét,  $\theta_0$  pedig a fázistolást jelöli. A továbbiakban  $z[n]$ -t valós és képzetes részre bontjuk és ezeken figyeljük a nullátmeneteket. Ezek detektálását két állapotgép végzi, melyek működését az alábbi ábra demonstrálja.  $X[n]$  jelöli  $z[n]$  valós vagy képzetes részét,  $\alpha$  pedig megadja a küszöbértéket, ami alatt a nullátmeneteket detektáljuk. A két állapotgép párhuzamosan fut és ha  $1 \rightarrow 2$  vagy  $3 \rightarrow 2$  állapotátmenetet észlelünk, akkor negatív meredekségű, míg, ha  $2 \rightarrow 1$  vagy  $4 \rightarrow 1$  állapotátmenetet észlelünk, akkor pozitív meredekségű nullátmenet történt [11].



9.1 ábra: A nullátmenet detektálás állapotgépe

Minden észleléskor elmentünk egy becsült idő és egy fázis értéket. A lehetséges fázis értékek ismertek, az alábbi táblázatban láthatóak.

nullátmenet típusa	fázis
imag rész, pozitív	$k2\pi$
valós rész, negatív	$k2\pi + \pi/2$
imag rész, negatív	$k2\pi + \pi$
valós rész, pozitív	$k2\pi + 3\pi/2$

Az átmenet időpontjára egy becslést végzünk a következőképpen. Ha nullátmenet detektálás történt pozitív meredekség esetén az 1-es, negatív esetén a 2-es állapotban vagyunk, tehát a  $\pm\alpha$  által meghatározott sávon kívül. Ekkor lineáris interpolációt végzünk az aktuális és a legutóbbi, a sávon kívül eső pont között. Tehát az egyes  $t_i$  értékek a következőképp számíthatóak ki: [11]

pozitív meredekségű nullátmenet esetén:

$$t_i = t_{n1} - \frac{x[n_1]}{\frac{x[n_2] - x[n_1]}{t_{n2} - t_{n1}}} = t_{n2} - \frac{x[n_2]}{\frac{x[n_2] - x[n_1]}{t_{n2} - t_{n1}}}, \quad (49)$$

negatív meredekségű nullátmenet esetén:

$$t_i = t_{n1} - \frac{x[n_1]}{\frac{x[n_{s1}] - x[n_{s2}]}{t_{s1} - t_{s2}}} = t_{n2} - \frac{x[n_2]}{\frac{x[n_{s1}] - x[n_{s2}]}{t_{s1} - t_{s2}}}, \quad (50)$$

ahol  $x[n_1]$  pozitív meredekség esetén az 1-es állapot negatív meredekség esetén a 2-es állapot utolsó értéke,  $x[n_2]$  pozitív meredekség esetén az 2-es állapot negatív meredekség esetén az 1-es állapot utolsó értéke. Ezek az értékek valós vagy képzetes számok, attól függően, hogy melyik állapotgépben történt a nullátmenet. Továbbá  $t_{n1}=i_1T$  és  $t_{n2}=i_2T$ , ahol  $i_1$  és  $i_2$  a minták indexei,  $T$  pedig a mintavételezés periódusideje.

Amennyiben már van kettő vagy annál több pont páruunk, elvégezhető a fázis és frekvencia becslése. Ehhez lineáris regressziót hajtunk végre a ponthalmazon, hogy meghatározzuk a legkisebb négyzetes hibával illeszkedő egyenest. Ennek az egyenesnek a meredeksége fogja adni a becsült frekvenciát, eltolása pedig a becsült fázist [11].

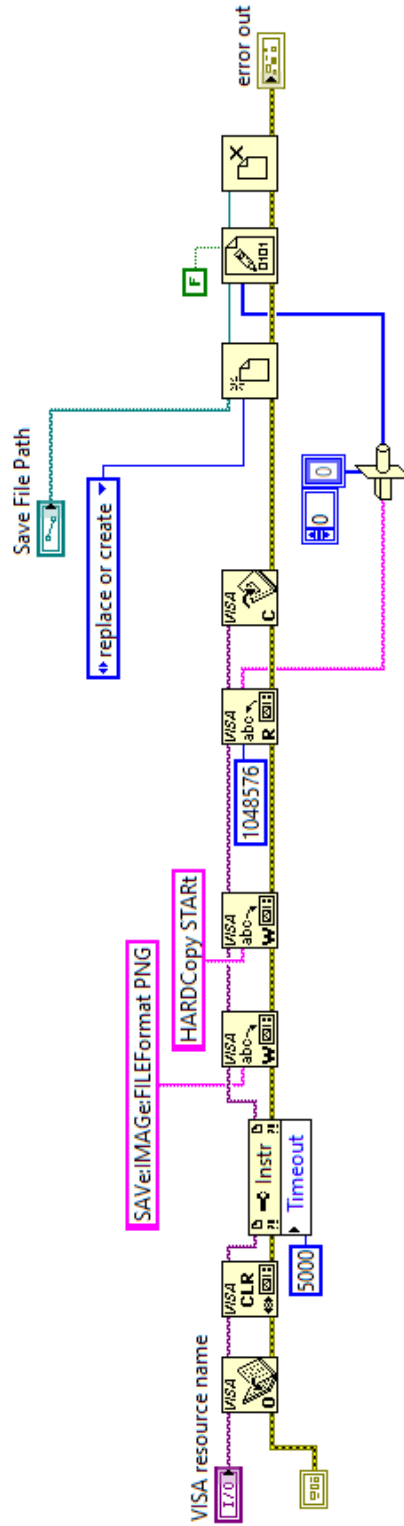
Az algoritmus előnye, hogy nem kell az egész jelet ismernie, minden új minta érkezésével frissíthető az eredmény, így a feldolgozási késleltetés kicsi. Tehát alkalmas

arra, hogy akár valós idejű alkalmazásokban is felhasználjuk. Továbbá külön  $\alpha_1$  és  $\alpha_2$  megadásával, az offsettel rendelkező jeleket is kezelni tudja.

Az algoritmus LabVIEW-ban való implementálásával akármilyen zajjal terhelt jel vizsgálata lehetőségessé válik. Mivel az algoritmus kicsi feldolgozási késleltetést ígér, és a folyamat során kiszámított frekvencia és fázis értékek bármikor újra elérhetők a tömbök alapján, így arra is lehetőség nyílik, hogy nyomon kövessük egy változó frekvenciájú és/vagy fázisú periodikus jel fázisának és frekvenciájának alakulását. A fent említett algoritmus implementálása, tesztelése, kiértékelése és az eredmények összehasonlítása jelen dolgozatban alkalmazott eljárások eredményivel jövőbeni feladat maradt.

# 10 Függelék

## 10.1 Az oszcilloszkóp képernyőképének mentése – blokk diagram



## Irodalomjegyzék

- [1] Kolumbán, G., Krébesz, T. I., C. M. Lau, F.: *Theory and Application of Software Defined Electronics: Design Concepts for the Next Generation of Telecommunications and Measurement Systems*, IEEE Circuits And Systems Magazine, Second Quarter 2012, pp. 8-9
- [2] Wikipedia: *LabVIEW*, <https://en.wikipedia.org/wiki/LabVIEW> (revision 01:58, 22 September 2019)
- [3] National Instruments: *Ismertető a műszervezélésről*, <http://www.ni.com/product-documentation/3382/en/> (2019. okt.)
- [4] National Instruments: *Ismertető a DAQ vezérléséről*, <http://www.ni.com/product-documentation/2835/en/> (2019. okt.)
- [5] DAQ NI 6034E/6034E/6034E User Manual (A DAQ kártya használati utasítása)
- [6] Balogh L., Kollár I., Németh J., Péceli G., Sujbert L.: *Digitális jelfeldolgozás*, BME jegyzet, Műegyetemi kiadó, Budapest, 2008
- [7] Sujbert L.: *Méréstechnika segédlet*, BME jegyzet, 2017
- [8] Varjasi I., Balogh A., Futó A., Gájász Z., Hermann I., Kárpáti A.: *Elektornika 2*, BME jegyzet, 2014
- [9] Keviczky L., Bars R., Hetthéssy J., Barta A., Bányász Cs.: *Szabályozástechnika*, BME jegyzet, 2004
- [10] Electronic Music Interactive V2: Square and Rectangle waves, <https://pages.uoregon.edu/emi/14.php/> (2019 okt.)
- [11] Yizheng Liao: *Phase and Frequency Estimation: High-Accuracy and Low-Complexity Techniques*, Worcester Polytechnic Institute, 2011

# Ábrajegyzék

2.1 ábra: Celsius-Fahrenheit átváltó VI .....	9
3.1 ábra: oszcilloszkóp vezérlése .....	13
3.2 ábra: a DAQ lábkiosztása .....	14
3.3 ábra: egy csatorna RSE konfigurációja.....	15
4.1 ábra: 1 kHz-es szinusz jel a DAQ kimenetén .....	19
4.2 ábra: 1 kHz-es szinusz jel az elsőfokú szűrő kimenetén.....	20
4.3 ábra: 1 kHz-es szinusz jel az másodfokú szűrő kimenetén.....	20
4.4 ábra hibajel szűretlen DAQ jel esetén: .....	21
4.5 ábra: hibajel szűrt DAQ jel esetén .....	21
5.1 ábra: Spektrumszivárgás es picket fence jelenség: (a) egy szinuszos jel spektruma,26	
5.2 ábra: Spektrális szivárgás.....	26
5.3 ábra: Hanning ablak hatása .....	27
5.4 ábra: 100 Hz-es háromszögjel spektruma 1 kHz-es mintavételi frekvenciával.....	28
5.5 ábra: Koherensen mintavételezett 100 Hz-es, 1V amplitúdójú szinuszos jel spektruma .....	29
6.1 ábra: Másodfokú aluláteresztő szűrő [8].....	30
6.2 ábra: Sallen-Key másodfokú aluláteresztő szűrő.....	31
6.3 ábra: a realizált szűrő .....	33
6.4 ábra: a szűrő összekötve a táppal és a DAQ kártyával .....	33
6.5 ábra: Pólus-zérus elrendezés .....	37
6.6 ábra: Nyquist diagram.....	38
6.7 ábra: a kapcsolási rajz és a szimulált Bode-diagram .....	40
7.1 ábra: szinuszos gerjesztéssel készült amplitúdó Bode-diagramm .....	43
7.2 ábra: négyszög gerjesztéssel készült amplitúdó Bode-diagramm.....	45
7.3 ábra: 25%-os kitöltési tényezőjű négyszög gerjesztéssel készült amplitúdó Bode-diagramm .....	45
7.4 ábra: multitone gerjesztéssel készült amplitúdó Bode-diagramm .....	46
8.1 .ábra: Az elméleti, illetve a két határt képező átviteli függvények Bode-diagramjai	50
9.1 ábra: A nullátmenet detektálás állapotgépe .....	51