



EFOP-3.4.3-16-2016-00014

SZÉCHENYI 2020

Mérésadatgyűjtés, jelfeldolgozás

tankönyv

Sarcevic Péter
Szegedi Tudományegyetem
Mérnöki Kar

Szeged
2019

Szegedi Tudományegyetem
Cím: 6720 Szeged, Dugonics tér 13.
www.u-szeged.hu
www.szechenyi2020.hu

SZÉCHENYI 2020



MAGYARORSZÁG
KORMÁNYA

Európai Unió
Európai Szociális
Alap



BEFEKTETÉS A JÖVŐBE



EFOP-3.4.3-16-2016-00014



Jelen tananyag a Szegedi Tudományegyetemen készült az Európai Unió támogatásával.
Projekt azonosító: EFOP-3.4.3-16-2016-00014

Szerző:

Dr. Sarcevic Péter PhD Szegedi Tudományegyetem Mérnöki Kar

Lektorálta:

Dr. Odry Péter PhD Dunaújvárosi Egyetem Informatika Intézet

ISBN: 978-963-306-671-3

© Sarcevic Péter 2019

Szegedi Tudományegyetem
Cím: 6720 Szeged, Dugonics tér 13.
www.u-szeged.hu
www.szechenyi2020.hu

1



MAGYARORSZÁG
KORMÁNYA



Európai Unió
Európai Szociális
Alap



BEFECTETÉS A JÖVŐBE

Tartalomjegyzék

Előszó	6
1 Jelek és rendszerek alapjai	8
1.1 Jelek	8
1.1.1 Jelek csoportosítása	8
1.1.2 Fontosabb jelek	9
1.1.3 Jelek tulajdonságai	14
1.2 Rendszerek.....	15
1.2.1 Rendszerek osztályozása	16
1.3 Példák	18
1.4 Ellenőrző kérdések	22
2 A mérési és jelfeldolgozási eljárások alapjai	23
2.1 Modellezés.....	23
2.2 A jelfeldolgozás modellje.....	24
2.2.1 A digitális technika előnyei.....	26
2.2.2 Digitális jelek	27
2.2.3 Jelfeldolgozó eszközök	28
2.3 Zajok.....	30
2.4 Példák	30
2.5 Ellenőrző kérdések	36
3 A/D és D/A átalakítók.....	37
3.1 A/D átalakítás	37
3.1.1 Mintavételezés.....	37
3.1.2 Kvantálás.....	45
3.2 D/A átalakítás	46
3.2.1 Dekódolás.....	47
3.2.2 Tartás.....	47
3.3 Példák	49
3.4 Ellenőrző kérdések	51

4	Konvolúciós és korrelációs függvények	52
4.1	Konvolúció	52
4.1.1	Konvolúció folytonos időben	52
4.1.2	Konvolúció diszkrét időben	54
4.1.3	A konvolúció tulajdonságai	55
4.2	Keresztkorreláció	56
4.3	Autokorreláció	57
4.4	Példák	58
4.5	Ellenőrző kérdések	64
5	Fourier-sor és Fourier transzformáció	65
5.1	Fourier-sor	65
5.2	Fourier transzformáció	68
5.2.1	A Fourier transzformáció tulajdonságai	69
5.3	Példák	71
5.4	Ellenőrző kérdések	78
6	Diszkrét Fourier transzformáció (DFT) és gyors Fourier transzformáció (FFT).....	79
6.1	Diszkrét Fourier transzformáció (DFT).....	79
6.1.1	Spektrum	81
6.2	Gyors Fourier transzformáció (FFT)	81
6.3	Példák	85
6.4	Ellenőrző kérdések	91
7	LTI rendszerek és vizsgálatuk.....	92
7.1	Súlyfüggvény és átmeneti függvény.....	92
7.1.1	Súlyfüggvény	92
7.1.2	Átmeneti függvény	93
7.2	LTI rendszerek soros és párhuzamos kapcsolása	93
7.2.1	Soros kapcsolat	93
7.2.2	Párhuzamos kapcsolat	94
7.3	Időállandó és sajátfrekvencia.....	95
7.4	Stabilitás	97

7.5	Laplace transzformáció.....	97
7.5.1	Laplace transzformáció tulajdonságai.....	98
7.6	z-transzformáció.....	99
7.6.1	A z-transzformáció tulajdonságai.....	100
7.7	Átviteli függvény.....	100
7.7.1	Diszkrét idejű átviteli függvény.....	101
7.8	Frekvenciafüggvény.....	101
7.8.1	A frekvenciafüggvény ábrázolása.....	102
7.9	Példák.....	104
7.10	Ellenőrző kérdések.....	105
8	Analóg szűrők.....	106
8.1	Szűrők sávjai.....	106
8.2	Szűrők típusai.....	107
8.2.1	Aluláteresztő szűrő.....	107
8.2.2	Felüláteresztő szűrő.....	107
8.2.3	Sáváteresztő szűrő.....	108
8.2.4	Sávzáró szűrő.....	108
8.2.5	Mindentáteresztő szűrő.....	108
8.3	Szűrők tervezése.....	109
8.4	Közelítési eljárások.....	110
8.4.1	Legismertebb közelítési eljárások.....	110
8.5	Passzív elemekkel megvalósított szűrők.....	114
8.6	Ellenőrző kérdések.....	119
9	Digitális szűrők.....	120
9.1	FIR szűrők.....	120
9.1.1	FIR szűrők tervezése ablakozással.....	121
9.2	IIR szűrők.....	123
9.2.1	Bilineáris transzformáció.....	126
9.3	FIR és IIR szűrők összehasonlítása.....	126

9.4	Analóg és digitális szűrők összehasonlítása	127
9.5	Példák	128
9.6	Ellenőrző kérdések	132
10	Adatgyűjtő rendszerek	133
10.1	Protokollok.....	133
10.2	NI alapú adatgyűjtő rendszerek.....	137
10.2.1	Egyszerű adatgyűjtő eszközök	138
10.2.2	Valós idejű rendszerek	139
10.3	Vezetéknélküli szenzorhálózatok.....	142
10.4	Ellenőrző kérdések.....	144
	Irodalomjegyzék.....	145

Előszó

A tankönyv a Szegei Tudományegyetem mechatronikai mérnöki és gépészmérnöki szakokon oktatót Mérésadatgyűjtés, jelfeldolgozás című tárgy tananyagát tartalmazza.

A tárgy és egyben a tankönyv célja a hallgatók megismertetése az alapvető mérésadatgyűjtési és jelfeldolgozási technikákkal. A tárgy tematikájának megfelelően a tankönyv tárgyalja a jelek és rendszerek alapjait, a mérési eljárások alapjait, az analóg-digitális és digitális-analóg átalakítást, alapvető jelfeldolgozási módszereket és az adatgyűjtő rendszereket. Az anyagrészek könnyebb megértése érdekében tartalmaz Scilab, Xcos és LabVIEW segítségével megoldott példákat is. Az anyagrészek elsajátításához nélkülözhetetlen, hogy a hallgatók rendelkezzenek alapvető ismeretekkel programozás, grafikus programozás, matematika, mérés technika, elektronika és digitális technika témakörökben.

Ez a tananyag szem előtt tartja a megfelelést a tanulási eredmény alapú szemlélet megvalósításának, az előírt, illetve elvárt szakmai kompetenciáknak, kompetencia-elemeknek, amelyek kialakításához a tantárgy jellemzően és érdemben hozzájárul, így a hallgató:

a) tudása tekintetében elmondható, hogy

- ismeri a mechatronikai, elektromechanikai, informatikai, mozgásszabályozási rendszereket, szenzorokat és aktuátorokat, valamint azok szerkezeti egységeit, alapvető működésüket mind gépészeti, mind elektrotechnikai, mind irányítástechnikai megközelítésből.
- ismeri az alapvető mechatronikai tervezési elveket, módszereket ezen belül a gépészeti és finommechanikai konstrukciók, valamint az analóg és digitális áramkörök tervezésének alapjait.
- ismeri az alapvető gépészeti, villamos- és irányítástechnikai rendszerekkel kapcsolatos számítási, modellezési, szimulációs módszereket.
- ismeri a számítógépes irányítás, mérésadatgyűjtés, beágyazott rendszerek, optikai érzékelés, képfeldolgozás eszközeit, részegységeit, alapvető tervezési és programozási módszereit.
- ismeri a gépészetben és az elektronikában használatos alapvető mérési eljárásokat, azok eszközeit, műszereit, mérőberendezéseit.

b) képességei fejlődésének eredményeképpen

- alkalmazni tudja mechatronikai, elektromechanikai, mozgásszabályozási termékek és technológiák tervezéséhez kapcsolódó alapvető számítási, modellezési elveit, módszereit, mind gépészeti, mind elektrotechnikai, mind irányítástechnikai megközelítésből.
- képes értelmezni és jellemezni a mechatronikai rendszerek szerkezeti egységeinek, elemeinek felépítését, működését, az alkalmazott rendszerelemek kialakítását és kapcsolatát mind

- gépészeti, mind elektrotechnikai, mind irányítástechnikai megközelítésből.
- képes meghibásodások diagnosztizálására, a megfelelő hibaelhárítási eljárás kiválasztására mind gépészeti, mind elektrotechnikai, mind irányítástechnikai megközelítésből.
 - megérti és használja szakterületének jellemző online és nyomtatott szakirodalmát magyar és idegen nyelven, e tudás birtokában folyamatosan megújul.

c) attitűdje várhatóan kedvezően változik,

- törekszik a gépészeti, az informatikai, a villamosmérnöki és az élettudományi szakterületek közötti összekötő, integráló szerep betöltésére.
- törekszik a szakterületén alkalmazott legjobb gyakorlatok, új szakmai ismeretek, módszerek megismerésére.

d) autonómiája és felelőssége fejlődik,

- tervezési, üzemeltetési, ellenőrzési feladatai megoldása során önállóan választja ki és alkalmazza a releváns problémamegoldási módszereket.
- bekapcsolódik a munkájához kapcsolódó kutatási és fejlesztési projektekbe. A projektcsoportban a cél elérése érdekében autonóm módon, a csoport többi tagjával együttműködve mozgósítja elméleti és gyakorlati tudását, képességeit.

a Szerző

1 Jelek és rendszerek alapjai

1.1 Jelek

A rendszerek vizsgálatához, leírásához és működtetéséhez szükséges információt szerezni a bennük keringő és áthaladó hatásokról. Ezeket a hatásokat jeleknek nevezzük. A jelek információs kapcsolatot valósítanak meg, és legfontosabb jellemvonásuk az információtartalom.

1.1.1 Jelek csoportosítása

A jeleket többféle szempont szerint is feloszthatjuk:

Értékkészlet szerint

- Folytonos értékű – ha tetszés szerinti értéket vehet fel és értékkészlete folytonos.
- Diszkrét értékű vagy szakaszos – csak meghatározott, diszkrét értékeket vehet fel, két szomszédos diszkrét értéke közötti értékkészlete hiányzik.

Értelmezési tartomány (időbeli lefolyás) szerint

- Folyamatos – a független változó minden valós értékre értelmezett, azaz megszakítás nélkül fennáll. Amennyiben a független változó az idő, akkor az ilyen jeleket folytonos idejű jeleknek nevezzük.
- Szaggatott – a jel a független változó szerint csak meghatározott, diszkrét pontokban változhat. Ha az idő a független változó, akkor ez azt jelenti, hogy a jel időközökben megszakad. Ilyen esetben diszkrét idejű jelekről beszélünk.

Az információ megjelenési formája szerint

- Analóg – a jelhordozó értéke vagy értékváltozása az információt közvetlenül képviseli. Ezek a jelek mind értékkészlet, mind időbeli lefolyás szerint folytonosak.
- Digitális – időben és értékkészletben is diszkrét, az információ pedig a jelhordozó számjegyeket kifejező, diszkrét, jelképi értékeiben (kódjaiban) van jelen.

Meghatározottsága szerint

- Determinisztikus – értékük meghatározott függvénnyel megadható.

A determinisztikus jelek tovább feloszthatók:

- Periodikus – egy adott periódus után a jel ismétlődik. A legismertebb és leggyakrabban használt periodikus jel a szinusz jel, de ide tartoznak még a négyszögjel, fűrészjel, háromszögjel, stb.
- Nem periodikus vagy aperiodikus.
- Sztochasztikus – véletlen lefolyású, és csak valószínűségszámítási módszerekkel írható le.

Jelhordozó szerint

Jelhordozó lehet bármilyen fizikai vagy kémiai mennyiség. Segítségükkel az információ anyagi jellegűvé válik, illetve továbbítható és/vagy tárolható. Manapság a villamos jelek alkalmazása a legelterjedtebb, de jelhordozó lehet sűrített levegő pneumatikus rendszereknél, folyadék hidraulikusnál, fény optikainál, vagy rádióhullám elektromágneses rendszereknél. A villamos jelek elterjedésének okai, hogy nagy távolságra jól átvihetők, gyors fizikai változásokat is jól képesek követni, és persze könnyen csatlakoztathatóak elektronikus eszközökhöz.

Villamos jeleknél az információt a feszültség vagy az áramerősség változása hordozza. Az információt közölheti a jel amplitúdója, frekvenciája vagy fázisa, de használhatunk impulzusokat is, ahol az információ megjelenhet az impulzus amplitúdójában, szélességében, az impulzusok közötti szünet hosszában, vagy akár az impulzusok számában. Analóg villamos jelek alkalmazása esetén létezik néhány szabványos tartomány: 0-1V, 0-10 V, 0-5 mA, 0-20 mA, vagy 4-20 mA. A 4-20 mA-es tartomány előnye a 0-20 mA-essel szemben, hogy könnyen detektálható ha szakadás történik a vonalon.

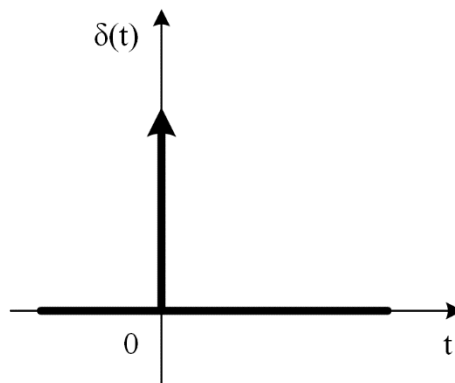
1.1.2 Fontosabb jelek

A jegyzet további részeiben a folytonos idejű jelek $x(t)$ formában, azaz kerek zárójellel lesznek jelölve, míg a diszkrét idejű jelek esetén szögletes zárójel lesz használva, tehát $x[n]$ alakban lesznek megadva. A diszkrét idejű jelölés az $x[nT]$ -ből ered, ahol T a mintavételezési periódusidőt jelképezi, amit el szoktunk hagyni, hiszen minden tagban jelen van.

Dirac-impulzus

A $\delta(t)$ Dirac-impulzus vagy egységimpulzus értéke $-\infty$ -től $+\infty$ -ig nulla, kivéve nullában, ahol végtelen. A Dirac-impulzus egy olyan jel, amely végtelen keskeny, amplitúdója végtelen nagy, területe pedig egységnyi.

$$\int_{-\infty}^{\infty} \delta(t) dt = 1 \quad (1.1)$$



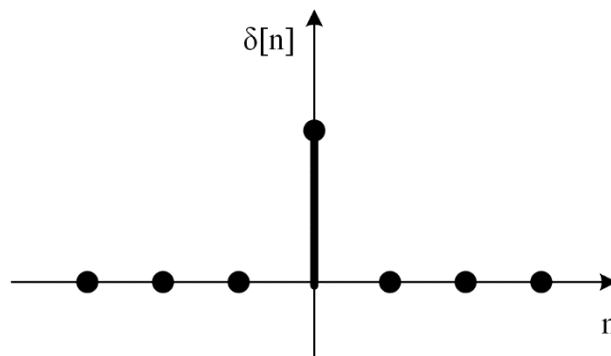
1.1 ábra: Dirac-impulzus ábrázolása folytonos időben.

Az egységimpulzus egyik legfontosabb tulajdonsága, hogy egy függvénnyel való szorzás az adott függvény nullában levő értékét adja vissza.

$$\int_{-\infty}^{\infty} \delta(t) f(t) dt = f(0) \quad (1.2)$$

Diszkrét esetben:

$$\delta[n] = \begin{cases} 1, & n = 0 \\ 0, & n \neq 0 \end{cases} \quad (1.3)$$

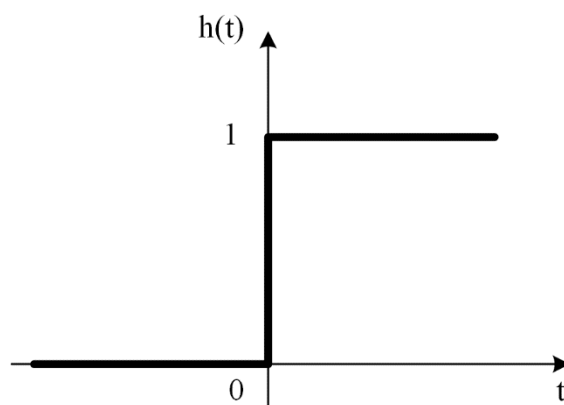


1.2 ábra: Dirac-impulzus ábrázolása diszkrét esetben.

Egységugrás

A $h(t)$ egységugrás jel értéke nulla a nulla pont előtt, ezután pedig egy. Jelölésnél szokás az $1(t)$ -t is használni.

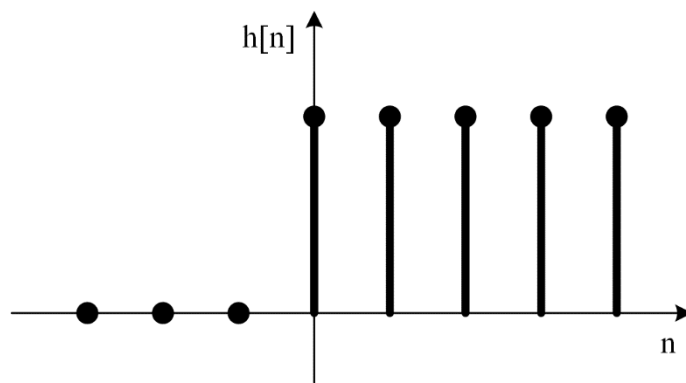
$$h(t) = \begin{cases} 1, & t \geq 0 \\ 0, & t < 0 \end{cases} \quad (1.4)$$



1.3 ábra: Egységugrás jel folytonos esetben.

Diszkrét esetben:

$$h[n] = \begin{cases} 1, & n \geq 0 \\ 0, & n < 0 \end{cases} \quad (1.5)$$

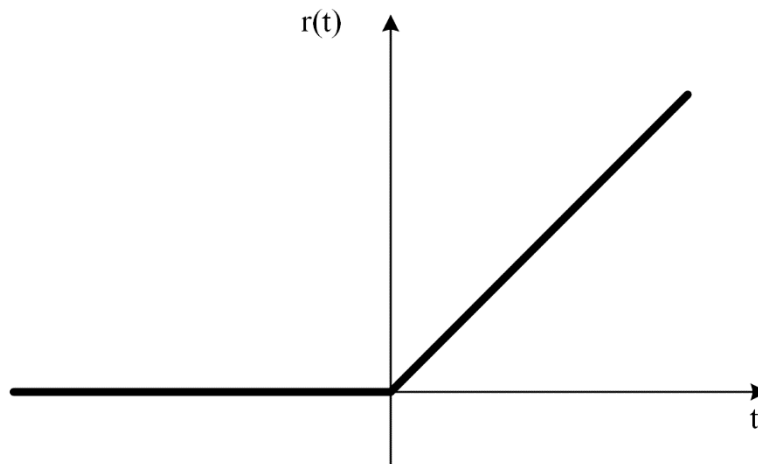


1.4 ábra: Egységugrás jel diszkrét esetben.

Egységsebesség függvény

Az $r(t)$ egységsebesség jel, másnéven sorompó függvény (angolul ramp function) értéke nulláig nulla, majd lineárisan növekszik:

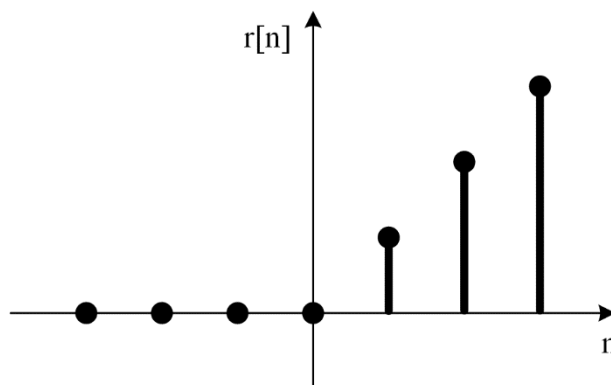
$$r(t) = \begin{cases} t, & t \geq 0 \\ 0, & t < 0 \end{cases} \quad (1.6)$$



1.5 ábra: Folytonos idejű egységsebesség jel.

Diszkrét esetben:

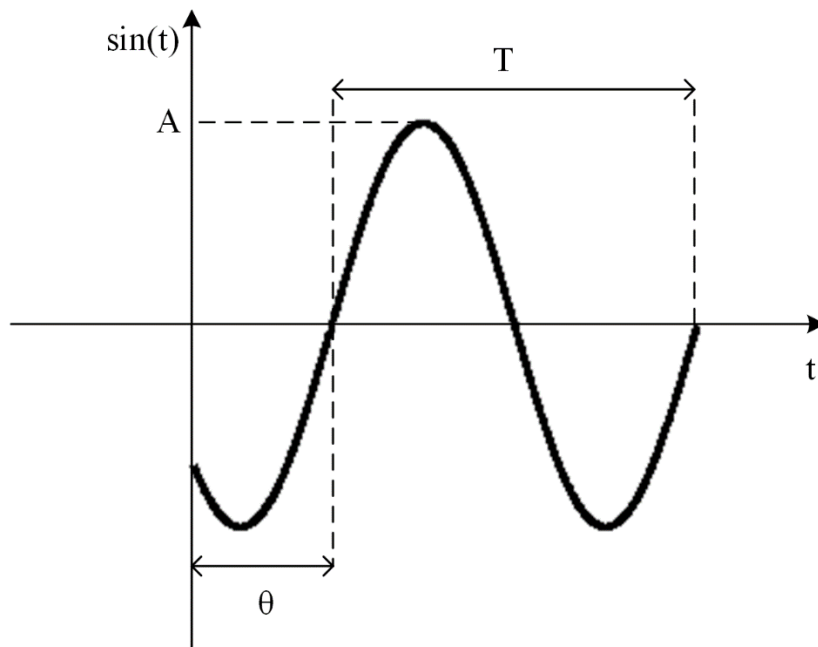
$$r[n] = \begin{cases} n, & n \geq 0 \\ 0, & n < 0 \end{cases} \quad (1.7)$$



1.6 ábra: Diszkrét egységsebesség jel.

Szinusz jel

A szinusz jel periodikus függvény amelyet három paraméter határoz meg: a jel csúcserő (amplitúdója), körfrekvenciája és kezdőfázisa.



1.7 ábra: Szinusz jel és paraméterei.

$$y(t) = A \cos(\omega t + \theta), \quad (1.8)$$

ahol A az amplitúdó, ω a körfrekvencia, és θ a kezdőfázis.

A körfrekvencia és a periódus (T) közötti összefüggés a következő módon adható meg:

$$\omega = \frac{2\pi}{T} \left[\frac{\text{rad}}{\text{s}} \right] \quad (1.9)$$

A frekvencia (f) periódusonkénti rezgésszám, melynek mértékegysége Herz (Hz) és a periódusidővel a következő módon fejezhető ki:

$$f = \frac{1}{T} \quad (1.10)$$

Így a körfrekvencia megadható a frekvencia alapján is:

$$\omega = 2\pi f \quad (1.11)$$

1.1.3 Jelek tulajdonságai

Eltolás

$$x(t) \rightarrow x(t - t_0) \quad (1.12)$$

$$x[n] \rightarrow x[n - n_0] \quad (1.13)$$

Ha $t_0 > 0$, akkor pozitív irányú eltolást kapunk. Például $x(t-2)$ esetén az $x(t)$ 2-vel jobbra lesz eltolva.

Amplitúdó skálázása

$$x(t) \rightarrow Ax(t) \quad (1.14)$$

$$x[n] \rightarrow Ax[n] \quad (1.15)$$

Az amplitúdó skálázása esetén az amplitúdó értékét minden pontban be kell szorozni az A együttható értékével.

Független változó skálázása

$$x(t) \rightarrow x(at) \quad (1.16)$$

$$x[n] \rightarrow x[an] \quad (1.17)$$

Ha a független változót szorzó együttható értéke nagyobb 1-nél ($a > 1$), akkor a jelet összenyomjuk, míg ha $0 < a < 1$, akkor a jelet széthúzzuk.

Reflexió

Reflexió során a jelet az ordinátára (függőleges tengely) tükrözzük.

$$x(t) \rightarrow x(-t) \quad (1.18)$$

$$x[n] \rightarrow x[-n] \quad (1.19)$$

Pároság

Egy függvény akkor páros, ha szimmetrikus az ordinátára.

$$x(-t) = x(t) \quad (1.20)$$

$$x[-n] = y[n] \quad (1.21)$$

Páratlanság

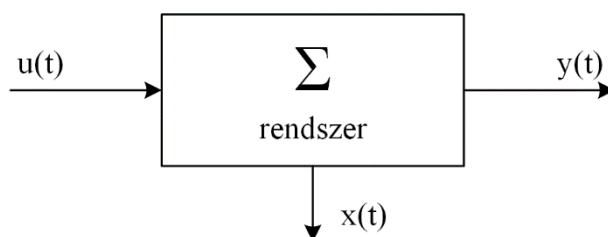
A páratlan függvény szimmetrikus az origóra nézve.

$$x(-t) = -x(t) \quad (1.22)$$

$$x[-n] = -y[n] \quad (1.23)$$

1.2 Rendszerek

Rendszernek nevezzük a valóság térben elhatárolt részét, melynek elemeit különböző kölcsönhatások és kölcsönös összefüggések kötik össze, illetve a rendszer és környezete is kölcsönhatásokon keresztül van kapcsolatban. Mindkét esetben a kölcsönhatások anyag-, energia- vagy információ-átadással járó folyamatok.



1.8 ábra: A rendszer és jellemzői.

A rendszer külső gerjesztésekre válaszokkal reagál. A gerjesztést más néven szokás a rendszer bemenetének, a választ pedig a rendszer kimenetének nevezni. A bemenetek azoknak a hatásoknak az összessége, amelyekkel a környezet hat a rendszerre, míg a kimenetek olyan hatások, amelyekkel a rendszer hat a környezetre. A bemenetek vektorát $u(t)$ -vel, a kimeneteket pedig $y(t)$ -vel szokás jelölni.

A rendszerekre ható hatások két csoportba oszthatóak, irányító és zavaró hatásokra. Ezek között a különbség, hogy az irányító hatások módosíthatóak, míg a zavaróak nem.

A rendszer fontos jellemzője még az állapot, amely valamely belső jellemzők vagy kölcsönhatás értékét határozza meg egy adott időpontban. Az állapotváltozók értékeit $x_1(t)$, $x_2(t)$, ..., $x_n(t)$ időfüggvényekkel szokás jelölni, ezeket vektorba rendezve pedig megkapjuk az $x(t)$ állapotvektort:

$$x(t) = \begin{bmatrix} x_1(t) \\ \vdots \\ x_n(t) \end{bmatrix} \quad (1.24)$$

Az érzékelők (szenzorok) szolgáltatják az információkat a rendszer állapotáról, míg a beavatkozók (aktuátorok) teszik lehetővé a rendszer irányítását.

Fontos még megemlíteni a hírközlő csatorna fogalmát, amely az a rendszer vagy közeg amelyen keresztül a rendszer a jelet kapja.

1.2.1 Rendszerek osztályozása

A rendszerek osztályozása általában viselkedésük és matematikai modelljük alapján történik.

Folytonos vagy diszkrét idejű

- Folytonos idejű – ha bemenetei vagy kimenetei közt található időben folytonos jel.
- Diszkrét idejű – ha csak diszkrét időpontokban van értéke.

Statikus vagy dinamikus

- Statikus – a rendszer kimenete csak az adott pillanatban jelentkező bemenettől függ.
- Dinamikus – a kimenet függ a múltbeli gerjesztésektől is. A diszkrét rendszerek ilyen esetben memóriával kell hogy rendelkezzenek.

Kauzális vagy nem kauzális

- Kauzális rendszer – a kimenet egy pillanatban csak az adott időpontban és az előtte megjelenő gerjesztésektől függ. Például:

$$y(t) = u(t - t_0), t_0 > 0 \quad (1.25)$$

$$y[n] = u[n] + u[n - 1] \quad (1.26)$$

- Nem kauzális rendszer – jövőbeli értékeket is tartalmaz. Az ilyen rendszerek fizikailag nem reálisak, viszont a mérnöki gyakorlat alkalmaz ilyen rendszereket. Például a diszkrét idejű jelek tárolhatók és később feldolgozhatók, ebben az esetben pedig már ismerjük a jövőbeli értékeket is, így fel is tudjuk őket használni feldolgozás közben.

Homogén vagy nem homogén

Homogén rendszer esetén érvényes, hogy ha A -szorosára növeljük a bemenetet, akkor a kimenet is A -szorosára növekszik.

$$u(t) \xrightarrow{\Sigma} y(t) \Rightarrow Au(t) \xrightarrow{\Sigma} Ay(t) \quad (1.27)$$

$$u[n] \xrightarrow{\Sigma} y[n] \Rightarrow Au[n] \xrightarrow{\Sigma} Ay[n] \quad (1.28)$$

Additív vagy nem additív

Ha egy gerjesztésre a rendszer egy adott kimenettel válaszol, egy másikra pedig egy másik kimenettel, akkor additív rendszer esetén a rendszer ha a bemenetére a két bemenet összegét adjuk, akkor kimenetén a két válasz összegét fogja adni.

$$u_1(t) \xrightarrow{\Sigma} y_1(t), u_2(t) \xrightarrow{\Sigma} y_2(t) \Rightarrow (u_1(t) + u_2(t)) \xrightarrow{\Sigma} (y_1(t) + y_2(t)) \quad (1.29)$$

$$u_1[n] \xrightarrow{\Sigma} y_1[n], u_2[n] \xrightarrow{\Sigma} y_2[n] \Rightarrow (u_1[n] + u_2[n]) \xrightarrow{\Sigma} (y_1[n] + y_2[n]) \quad (1.30)$$

Lineáris vagy nemlineáris

A lineáris rendszerek egyben homogének és additívak is.

$$u_1(t) \xrightarrow{\Sigma} y_1(t), u_2(t) \xrightarrow{\Sigma} y_2(t) \Rightarrow (Au_1(t) + Bu_2(t)) \xrightarrow{\Sigma} (Ay_1(t) + By_2(t)) \quad (1.31)$$

$$u_1[n] \xrightarrow{\Sigma} y_1[n], u_2[n] \xrightarrow{\Sigma} y_2[n] \Rightarrow (Au_1[n] + Bu_2[n]) \xrightarrow{\Sigma} (Ay_1[n] + By_2[n]) \quad (1.32)$$

Időinvariáns vagy idővariáns

- Időinvariáns – minden időpillanatban azonos jelre azonos kimenetet ad, tehát kapcsolatai és paraméterei függetlenek az időtől.

$$u(t) \xrightarrow{\Sigma} y(t) \Rightarrow u(t - t_0) \xrightarrow{\Sigma} y(t - t_0) \quad (1.33)$$

$$u[n] \xrightarrow{\Sigma} y[n] \Rightarrow u[n - n_0] \xrightarrow{\Sigma} y[n - n_0] \quad (1.34)$$

- Idővariáns – a rendszer kimenete függ attól, hogy milyen időben adjuk rá az adott bemenetet.

Invertálható vagy nem invertálható

Az invertálható rendszerek esetén a kimenetből egyértelműen következtetni tudunk a bemenetre, és a bemenetből is a kimenetre, tehát különböző bemenetekhez különböző kimeneteket rendel hozzá. A rendszernek így létezik inverze, amely a kimenetből vissza tudja állítani a bemeneteket.

Koncentrált vagy elosztott paraméterű

- Koncentrált paraméterű – az elemek paramétereit idealizáltnak, kiterjedés nélkülinek tekintjük (például tömegpont).
- Elosztott paraméterű – a paraméterek térben folytonos eloszlásban hatnak. Matematikai modelljüket parciális differenciálegyenletekkel adjuk meg.

Determinisztikus vagy sztochasztikus

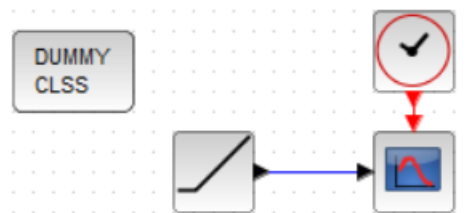
- Determinisztikus rendszerek esetén a kölcsönhatások leírhatóak determinisztikus függvényekkel.
- Sztochasztikus rendszerek esetén jelen vannak zavaró hatások, amelyek csak valószínűségi változókkal írhatóak le.

1.3 Példák

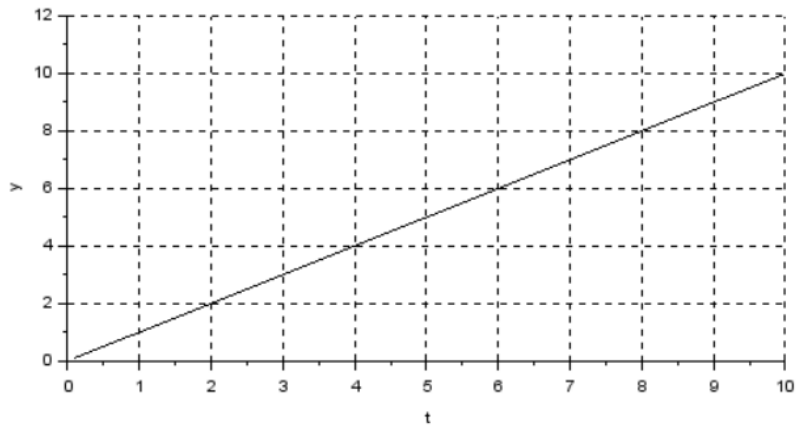
1.1 Példa

Xcos segítségével ábrázoljunk egy egységsebesség jelet amelynek meredeksége 1.

Megoldás:



Eredmény:

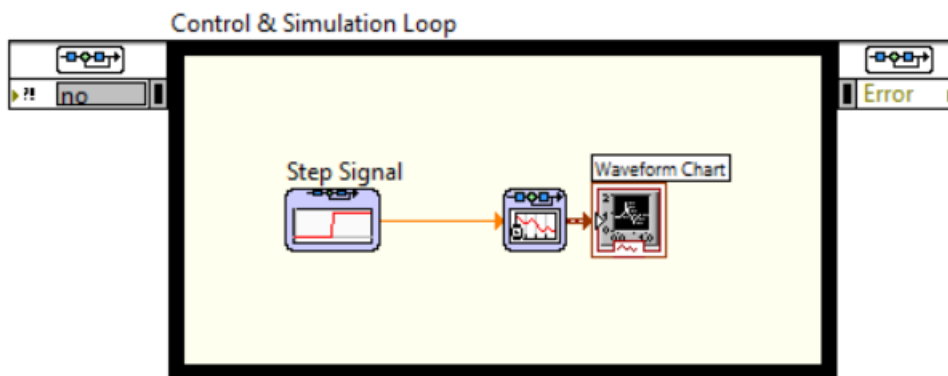


1.2 Példa

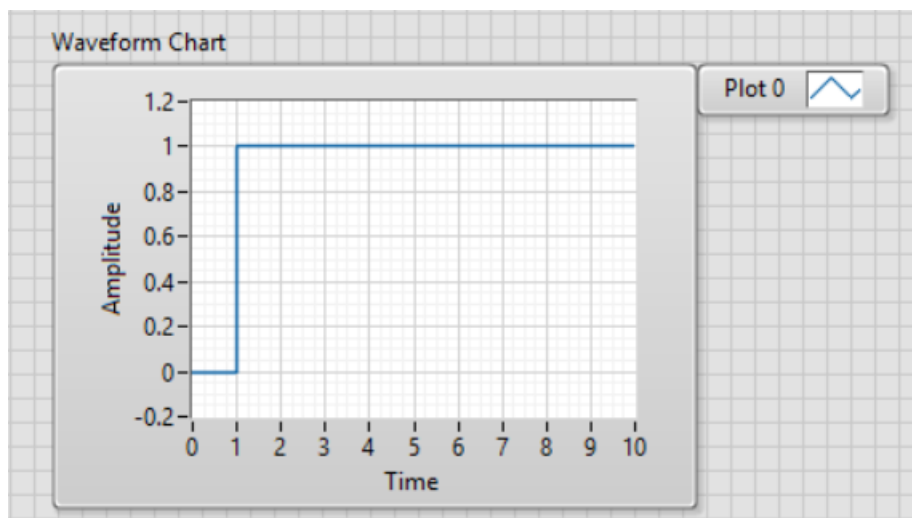
A LabVIEW Control Design and Simulation moduljával valósítsunk meg egy egységugrás jelet.

Megoldás:

A modulban található megfelelő elemeket Control & Simulation Loop-ba kell helyezni.



Eredmény:



1.3 Példa

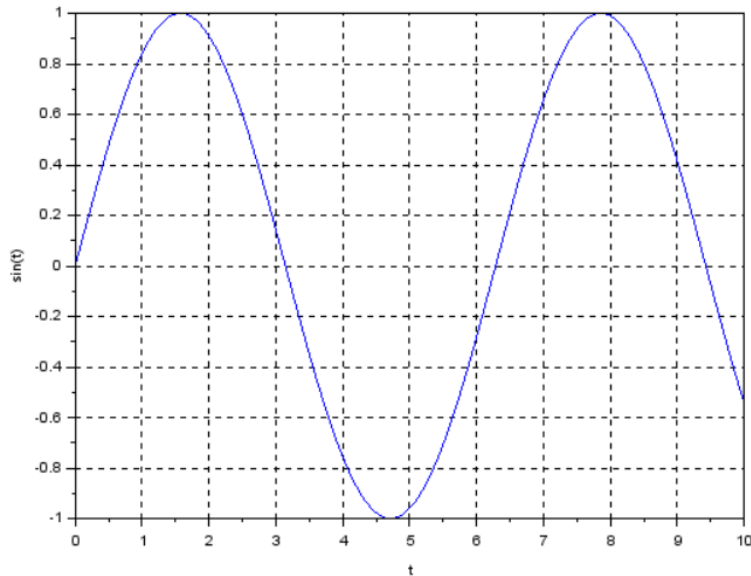
Hozzunk létre majd ábrázoljunk Scilab segítségével egy folytonos szinusz jelet.

Megoldás:

Ilyen környezetben nem hozhatunk létre folytonos jelet, viszont elegendően kis lépésközzel és folytonos ábrázolásmóddal a jelünk folytonos jelnek fog kinézni.

```
t=0:0.01:10;  
x=sin(t);  
plot(t,x)  
xgrid  
xlabel("t");  
ylabel("sin(t)");
```

Eredmény:



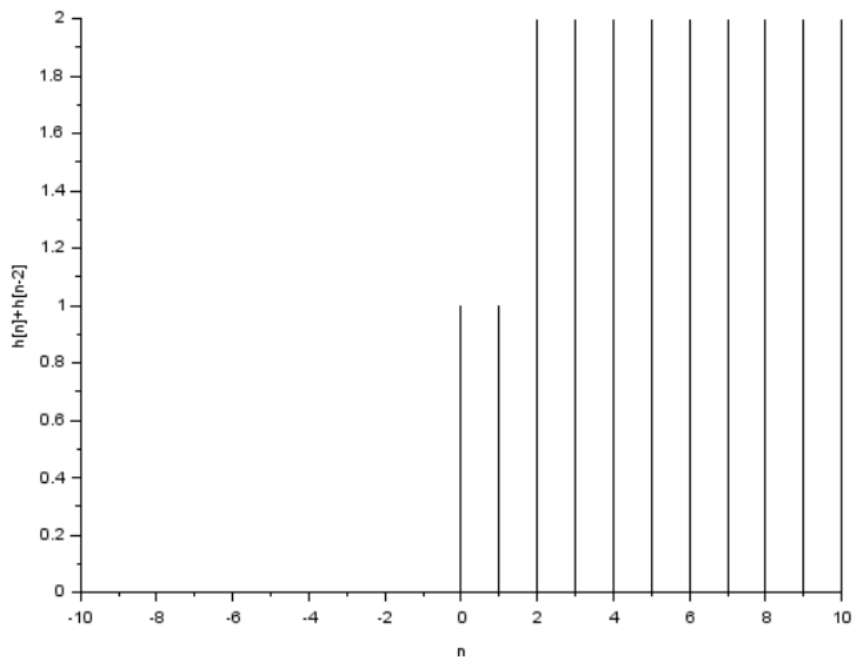
1.4 Példa

Scilab segítségével határozzuk meg a $h[n]$ és $h[n-2]$ diszkrét jelek összegét, majd ábrázoljuk az eredményt.

Megoldás:

```
n=-10:10;  
h_0=ones(1, size(n, 2))  
h_0=h_0.*(n>=0);  
h_2=ones(1, size(n, 2));  
h_2=h_2.*(n>=2);  
h_sum=h_0+h_2;  
plot2d3(n, h_sum)  
xlabel("n");  
ylabel("h[n]+h[n-2]");
```

Eredmény:



1.4 Ellenőrző kérdések

1. [Értékkészlet és időbeli lefolyás szerint mi a különbség az analóg és digitális jelek között?](#)
2. [Milyen tulajdonságokkal rendelkezik a Dirac-impulzus?](#)
3. [Mely paraméterek határozzák meg a szinusz jelet?](#)
4. [Milyenek a kauzális rendszerek?](#)
5. [Mi a különbség az idővariáns és időinvariáns rendszerek között?](#)

2 A mérési és jelfeldolgozási eljárások alapjai

A mérésadatgyűjtő rendszerek alapvető feladata fizikai jelek mérése és létrehozása. Mérés során információt gyűjtünk és dolgozunk fel, hogy egy objektumot jobban megismerjünk.

2.1 Modellezés

A modellezés célja a rendszerről szerzett ismeretek reprezentálása. A rendszerek leírása azok struktúrájával, paramétereivel és pillanatnyi állapotaival lehetséges, vagyis mérés során ezek megismerése a célunk. A struktúra a rendszer elemei közötti összefüggéseket határozza meg, míg a paraméterek ezen összefüggések mennyiségi viszonyai. A mérések megtervezése során is már szükséges, hogy rendelkezünk egy előzetes modellel a rendszerről.

A modellek a következő csoportokba sorolhatók:

- **Funkcionális modell** – a funkcionális modellek esetén a rendszer elemei idealizált funkciójuk alapján jelennek meg. Leginkább a mérések tervezésekor használhatóak. Ilyen modellek például az áramköri sémák, ahol az elektronikai elemek szimbólumait használjuk.
- **Fizikai modell** – a fizikai modell a bonyolult valóságos rendszer egyszerűsített és megvalósított mása. Alkalmazásuk célja lehet mérések elvégzése vagy akár előzetes információk szerzése is.
- **Matematikai modell** – a fizikai modell idealizált, matematikailag leírható változata. Segítségével a rendszer viselkedését írjuk le, a rendszer bemenő és kimenő jellemzőinek kapcsolatát adjuk meg valamilyen matematikai összefüggésekkel. Általában számos egyszerűsítést tartalmaz, mivel nem tudunk minden tényezőt figyelembe venni. Nagy jelentősége van olyan rendszerek esetén amelyeket fizikailag csak nagy költségek mellett, vagy egyáltalán nem tudunk megépíteni. Ha a valós rendszeren és a matematikai modellen azonos kísérleteket végzünk, akkor verifikálni tudjuk a modellünk alkalmazhatóságát. Matematikai modell esetén a jeleket változókkal jelöljük.

Egyéb fontos fogalmak:

- **Rendszeranalízis** – a rendszer struktúrája és paramétereinek alapján hogyan állapítható meg a rendszer viselkedése és tulajdonságai.
- **Rendszerszintézis** – adott tulajdonságok alapján hogyan kell kialakítani a rendszer struktúráját és paramétereinek értékeit meghatározni.
- **Rendszeridentifikáció** – a rendszer tulajdonságainak meghatározása valódi kísérletek végzése során nyert adatok feldolgozásával.
- **Szimuláció** – a matematikai vagy fizikai modellen végzett kísérlet.

Számítógép segítségével a rendszer működését gyakran le is tudjuk játszani.

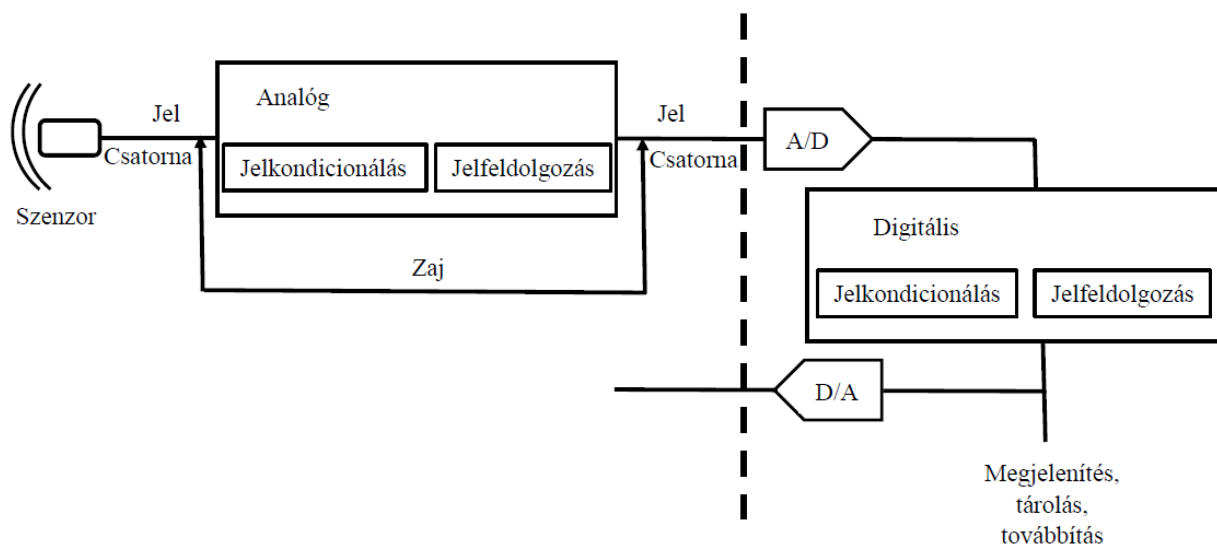
2.2 A jelfeldolgozás modellje

A jelfeldolgozás nagyon fontos részét képezi a mérési folyamatoknak.

A jelfeldolgozást széles körben használják, és számos alkalmazásnak rendkívül fontos eleme, mint pl. folyamatirányítási rendszerek, orvosi berendezések, kamerák, robotikai eszközök, mérőműszerek, szórakoztatóipari eszközök stb.

A jelek feldolgozása történhet analóg vagy digitális rendszerekben. Az analóg jelfeldolgozó eszközöknek nagy volt a szerepe addig, amíg a technológia fejlődésével a digitális eszközök nem jutottak arra a szintre, hogy valós idejű feldolgozásra legyenek képesek. Kezdetben csak analóg folyamatok szimulálására alkalmazták a digitális számítógépeket, majd rájöttek, hogy digitális tartományban megvalósíthatók teljes rendszerek is. Manapság leginkább digitális tartományban dolgozzák fel a jeleket. A technikát amikor analóg jeleket analóg eszközökkel dolgozunk fel, analóg jelfeldolgozásnak, míg digitális esetben pedig digitális jelfeldolgozásnak (angolul Digital Signal Processing – DSP) nevezzük.

A jelfeldolgozás blokksémája az 2.1 ábrán látható.



2.1 ábra: A jelfeldolgozás blokksémája.

Szenzor

A jelfeldolgozás folyamatának első eleme a szenzor, amely valamilyen fizikai vagy kémiai mennyiséget érzékel, és azt általában feszültségjellé alakítja. A szenzor analóg tartományba képezi le a fizikai vagy kémiai paraméter értékét.

Jelkondicionálás és jelfeldolgozás analóg tartományban

A jelek digitalizálása előtt analóg tartományban szükség van a jelek kondicionálására és feldolgozására, mely több folyamatot foglal magába. Ennek a fázisnak a legfőbb feladata a további lépésekben szereplő eszközök védelme, és a mérési értékek megfelelő tartományba illesztése.

A további eszközök védelme érdekében szükség lehet leválasztás alkalmazására, mely pl. optocsatolókkal valósítható meg.

A megfelelő átalakítás érdekében fontos az analóg jelen erősítést, csillapítást, és szűrést végezni. Például gyakran a szenzorok nagyon alacsony feszültségértékeket szolgáltatnak, amelyeket az A/D átalakító által meghatározott tartomány szerint erősíteni kell.

Az átalakítás előtt szükség van egy aluláteresztő szűrőre, amelyet szokás anti-aliasing szűrőnek is nevezni. Ennek feladata, hogy a spektrális átfedések elkerülése érdekében eltávolítsa a magasabb frekvenciájú komponenseket. Erről a problémáról a későbbiekben lesz szó.

A/D átalakítás

Az analóg jelek digitális eszközzel való feldolgozásához először a jelet át kell alakítani digitális jellé. Ezt analóg-digitális (A/D) átalakítóval végezhetjük el. Az A/D átalakítók legfontosabb paraméterei a felbontás és a sebesség. A sebesség meghatározza, hogy milyen frekvenciával frissíti az átalakító az értéket, míg a felbontás a bitek számát, azaz, hogy mennyire lesz pontos az átalakított érték.

Digitális jelfeldolgozás

A digitális jelfeldolgozást valamilyen digitális eszköz végzi. Ebben a folyamatban a cél lehet a jelek szűrése, transzformálása, tömörítése, kódolása vagy egyéb algoritmusok megvalósítása.

Például a digitális jelfeldolgozás során történhet különböző tulajdonságok kinyerése a jelből, melyek alapján egy algoritmus döntéseket hoz. Az algoritmus feladata lehet mondjuk valamilyen esemény detektálása. A tulajdonságok kinyerése történhet idő- vagy frekvenciatartományban is. Ilyen tulajdonságok lehetnek például időtartományban a zérusátmenetek száma, legmagasabb ugrások értéke, frekvenciatartományban adott tartomány energiaértéke, legmagasabb amplitúdóhoz tartozó frekvencia, stb.

Megjelenítés, tárolás és továbbítás

A digitális rendszerek fontos tulajdonsága, hogy könnyen hozzáilleszthetők más digitális eszközökhöz, így a jelek megjelenítése egyszerűen megvalósítható. A jelek továbbítása más távoli digitális rendszereknek is egyszerűen megoldható. Az adattovábbítás biztonságosabb is az

analóg módszernél, mivel biztonsági kódok alkalmazásával akár a közben létrejött hibák is észlelhetőek. Digitális eszközöknél az adatok tárolása is van lehetőség, ami lehetővé teszi az adatok jövőbeli analizálását is. A tárolás és a továbbítás is történhet tömörített adatokkal, amely tárolásnál csökkenti a szükséges tárhelyet, továbbításnál pedig a szükséges időt.

D/A átalakítás

Ha szükség van a feldolgozott jel visszaalakítására analóg tartományba, az digitális-analóg (D/A) átalakítóval valósítható meg.

A D/A átalakítók kimeneti jeleit simítani kell aluláteresztő szűrőkkel. Ennek oka, hogy ezek az analóg jelek szögletesek, ami zavart okozhat a jeltovábbítás vagy felhasználás során.

Szintén gyakran szükséges a végső rész felé a kimenő jelek koncinióálása, például erősítése.

Zaj

A zaj olyan zavaró hatás amely mindig befolyásolja a jelünket, így az eltér a valódi értéktől. Az ábrán csak analóg tartományban lett megjelölve ez a hatás, viszont digitális tartományban is jelen van, csak nagyobb amplitúdójú zaj szükséges, hogy módosítsa az értéket.

2.2.1 A digitális technika előnyei

A digitális technikának az analóg technikához képest számos előnye van:

- Adattárolás – az egyik legfontosabb előnye a digitális rendszereknek. A mérési adatainkat tárolhatjuk, majd utána bármikor elővehetjük, és például újabb módszereket próbálhatunk ki a rögzített jeleken.
- Kisebb zajérzékenység – különösen az adatok átvitelekor fontos tényező. Analóg jel esetében a zaj hatására különbséget észlelünk a valós és mért értékek között. Mivel a digitális jelek esetén a két alkalmazott értékhez („0” és „1”) egy-egy meghatározott feszültség szint tartozik, nagy amplitúdójú zaj szükséges ahhoz, hogy megváltozzon az érték.
- Tömörítés – digitális jeleken végezhető adatsűrítés. Ez a tömörítés lehet veszteségmentes és veszteséges. A veszteséges tömörítést csak a hang- és képanyag sűrítésére szokták használni. Az adatsűrítés segítségével nagy teret tudunk spórolni a memóriában, illetve jelátvitelkor nagy mértékben csökkenthető a szükséges átviteli kapacitás. A tömörített adatokat tárolás vagy adatátvitel után ellentétes algoritmussal kicsomagolhatjuk az eredeti nagyságra és szerkezetre.
- Programozhatóság – a digitális eszközök viselkedése programozással megváltoztatható. Emiatt ugyanazt az eszközt széles körben alkalmazhatjuk különböző problémák megoldására. Ezen kívül fontos lehet például egy szűrő esetében, ahol ha kiderül,

hogy meg kell változtatni a szűrő karakterisztikáját, akkor analóg esetben alkatrészeket kell cserélni, vagy rosszabb esetben akár újra kell tervezni az áramkört, addig digitális eszköz esetén elég lehet egy átprogramozás.

- Hőmérsékleti hatás – az analóg alkatrészek (ellenállások, operációs erősítők, stb.) esetén az egyik legkritikusabb tényező, mivel a hőmérséklet változásának hatására az átviteli jelleggörbéjük megváltozik.
- Öregedés – az analóg rendszerek (különösen a kondenzátorok) érzékenyek az öregedésre, melynek hatására megváltozik az elektromos viselkedésük az idő során.
- Tolerancia (tűrés) – analóg áramkörök tervezésekor ez egy nagyon fontos paraméter, amelyet figyelembe kell venni. Ennek hatására például két azonos típusú elemekből felépített rendszer között nagy különbségek jelenhetnek meg. Digitális rendszerek tervezésével ezek a problémák elkerülhetők.
- Tulajdonságok – vannak olyan igények amelyeket csak digitális technikával lehet megvalósítani. Például lineáris fázisú szűrők, lyuk szűrők, stb.

2.2.2 Digitális jelek

A digitális jelek bináris értékek, azaz nullások és egyesek, sorozatából állnak. A digitális rendszerek ezekkel az értékekkel dolgoznak, és ezeken végeznek megfelelő műveleteket.

Hexadecimális ábrázolás

A hexadecimális (tizenhatos) számrendszerben ábrázolt értékekre sok esetben van szükségünk:

- Programozás során gyakran szükséges vagy egyszerűbb hexadecimális értékeket alkalmaznunk (regiszterek címe, változók, maszkolás, stb.).
- Általában különböző eszközök adatlapjain a gyártók inkább hexadecimális formában tüntetik fel a regiszterek lehetséges értékeit és címeit mint binárisban vagy decimálisban.

Mivel a digitális rendszerek alapja a byte, ami 8 bitet tartalmaz, tehát egy byteot két hexadecimális szám segítségével lehet megadni. Például a 255 decimális értéket 0xFF alakban szokás ábrázolni.

2.1 táblázat: Értékek különböző számrendszerekben

bináris	decimális	hexadecimális	bináris	decimális	hexadecimális
0000	0	0	1000	8	8
0001	1	1	1001	9	9
0010	2	2	1010	10	A
0011	3	3	1100	11	B
0100	4	4	1100	12	C
0101	5	5	1101	13	D
0110	6	6	1110	14	E
0111	7	7	1111	15	F

2.2.3 Jelfeldolgozó eszközök

Valós idejű jelfeldolgozásra többféle eszköz létezik, amelyek közül a feladat határozza meg, hogy melyiket érdemes választani.

Mikrovezérlő

A mikrovezérlők vagy mikrokontrollerek (angolul microcontroller) egyszerű, valós idejű vezérlési feladatok ellátására lettek kifejlesztve, és egyetlen integrált áramkörben (angolul Integrated Circuit - IC) tartalmaznak mikroprocesszort, memóriát és megfelelő perifériákat.

Nem tartalmaznak túl sok belső memóriát (típusosan kB nagyságú mind RAM mind flash memória esetén), és más eszközökhöz képest alacsony frekvencián működnek. Napjainkban, a megbízható működés érdekében az alkalmazott órajel maximálisan 100MHz körüli.

Programozásuk C/C++ vagy Assembly nyelven történik, a program futása pedig szekvenciális, azaz egyszerre mindig csak egy parancsot hajt végre az eszköz. A szekvenciális futás viszont megfelelő programmal rendkívül megbízható futást eredményez. A valós idejű működést belső időzítők (angolul timer) segítik, amely hatására megáll a futás, végigfut egy megfelelő programrész, majd a program folytatja a futást azon a ponton ahol megállt. A kevés memória miatt valódi operációs rendszer nem telepíthető rájuk, viszont léteznek már egyszerű valós idejű operációs rendszerek (például FreeRTOS) amelyek alkalmazhatók egyes mikrovezérlőkön is.

A mikrovezérlők rendelkeznek A/D és D/A átalakítókkal és különböző kommunikációs interfészekkel, így könnyen illeszthetőek más eszközökhöz.

A eszközök belső architektúrája alapján több típus terjedt el: PIC, 8051, AVR, és ARM (ARM Cortex-M a mikrovezérlő család). A belső adatméret alapján pedig léteznek 8-bites, 16-bites, és 32-bites mikrovezérlők.

Mivel gyakran szükséges jelfeldolgozási művelek végzése ilyen eszközökön, így manapság már tartalmaznak különleges számítási műveleteket gyorsító számológységeket, például lebegő-pontos egységet (angolul floating-point unit – FPU).

Processzor

A processzorok a mikrovezérlőkhöz képest sokkal több memóriát tartalmaznak, nagyobb frekvenciájú órajelet alkalmaznak (körülbelül 2 GHz nagyságig), és általában már több mikroprocesszor magot használnak.

A nagyobb memória lehetővé teszi az operációs rendszerek futtatását ezeken az eszközökön. Az operációs rendszerekkel nehezen biztosítható a valós idejű futás, még annak ellenére is, hogy a több processzormag több szál párhuzamos futtatását teszi lehetővé. Ilyen célokra valós idejű operációs rendszereket lehet alkalmazni, amelyek csak adott feladat elvégzésére optimalizáltak.

Ilyen eszközöket használnak például a mai okostelefonok, amelyek általában 32-bites és 64-bites ARM Cortex-A magokat tartalmaznak.

FPGA

Az FPGA-k (angolul Field-Programmable Gate Array) programozható logikai eszközök, amelyek logikai cellák, makró cellák, illetve programozható kapcsolók és összeköttetések kettő dimenziós tömbjéből épülnek fel. Ezek az eszközök párhuzamos működést tesznek lehetővé, így általánosan olyan algoritmusok esetén célszerű őket használni, amelyeknél folyamatosan ugyanazokat a műveleteket kell elvégezni, és azok nagy mértékben párhuzamosíthatóak.

Az FPGA-k a programozható blokkok mellett tartalmaznak memóriablokkokat is, illetve külön a jelfeldolgozási felhasználások miatt DSP blokkokat is, amelyek a bonyolultabb DSP műveleteket képesek gyorsan elvégezni.

Programozásuk VHDL (angolul Verilog Hardware Description Language) nyelven történik.

DSP processzor

Léteznek konkrétan jelfeldolgozási feladatokra konstruált speciális processzorok, melyeket digitális jelfeldolgozó processzoroknak (angolul Digital Signal Processor) vagy röviden DSP-knek nevezünk. Ezeket az eszközöket adott alkalmazásokra optimalizálják, és annak megfelelően rendelkeznek például memóriával.

Ezek az eszközök tipikus DSP műveleteket képesek elvégezni, gyakran egy taktus alatt. A gyors működés érdekében számos gyorsítási módszert alkalmaznak, például pipelining-ot, párhuzamosítást, stb.

Eleinte csak integer értékekkel dolgoztak a DSP processzorok, programozásuk pedig Assembly nyelven történt. Jelenleg többnyire fixpontos aritmetikát használnak, egyes eszközök pedig lebegőpontos értékeket is tudnak kezelni, programozásuk viszont már magasabb szintű programnyelveken is lehetséges, mint például a C nyelv. A leeffektívebb

programokat azonban Assembly nyelven lehet írni.

2.3 Zajok

A jelfeldolgozásban fontos befolyásoló tényező a zaj, amely olyan jel, amely a hasznos információt hordozó jelhez adódik hozzá. A zaj sohasem szüntethető meg teljesen, hatása csak csökkenthető.

A zajok a sztochasztikus jelek csoportjába tartoznak, így csak valószínűségi adatokkal jellemezhetőek.

A zajok teljesítménysűrűség-spektrumuk és eloszlásuk szerint osztályozhatók.

Ha a zaj teljesítménysűrűség-függvénye konstans, akkor fehér zajról beszélünk. A nem állandó értékű teljesítménysűrűség-spektrummal rendelkező zajokat színes zajnak szokás nevezni. Ezeknek több típusa van: Lorentz-féle, $1/f^2$, $1/f$, stb.

Az eloszlásfüggvény alakja szerint a zajok általában normális eloszlásúak, azaz a zaj pillanatnyi amplitúdó értékének eloszlása a Gauss-görbével adható meg. A valóságban előforduló zajok többsége Gauss-eloszlású fehér zaj.

Szokás még egyenletes eloszlású vagy random zajokat is említeni, amelyek esetén minden amplitúdóérték azonos valószínűséggel jelentkezik.

Eljárások tesztelésénél is gyakran szükséges zajjal terhelt jeleket alkalmazni, ilyen esetben mesterségesen állítjuk elő a zajt és adjuk hozzá a jelhez.

2.4 Példák

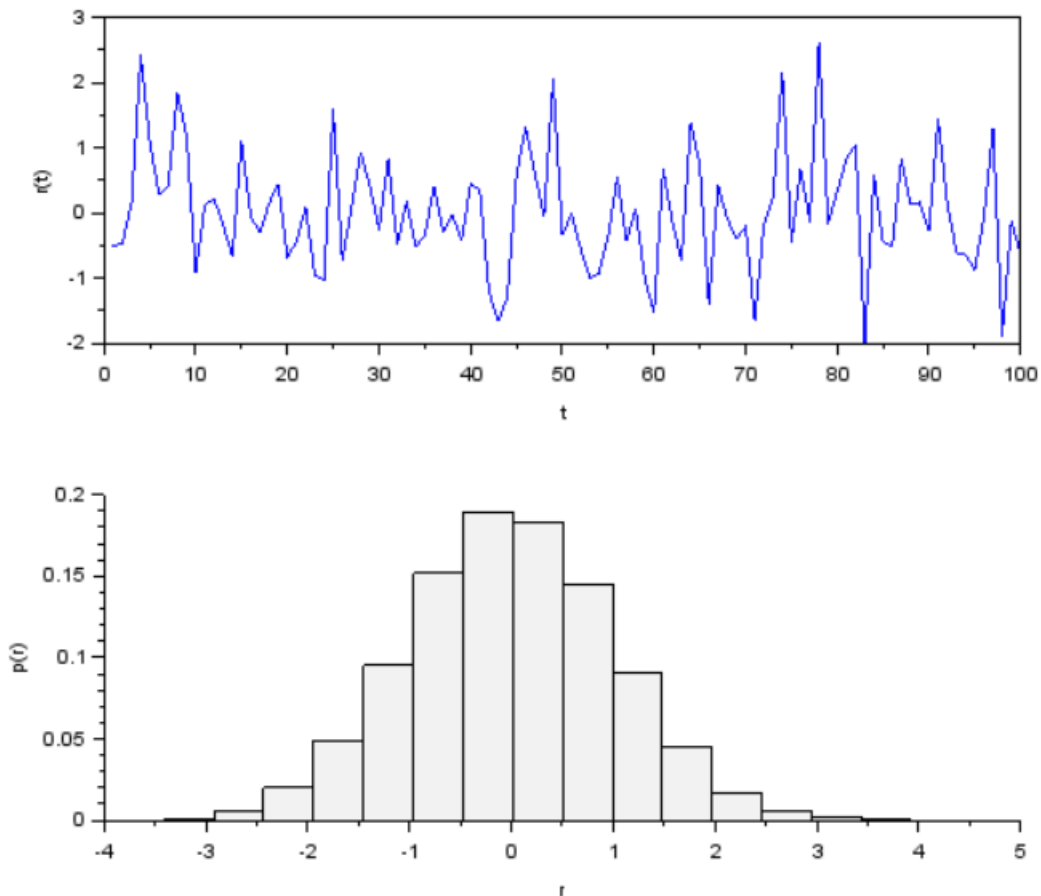
2.1 Példa

Scilab segítségével hozzunk létre normális eloszlású zajt, majd ábrázoljuk azt és annak eloszlásfüggvényét.

Megoldás:

```
r=rand(10000,1,"normal");
subplot(2,1,1)
plot(r(1:100))
xlabel('t','r(t)')
subplot(2,1,2)
histplot(16,r);
xlabel('r','p(r)')
```

Eredmény:



2.2 Példa

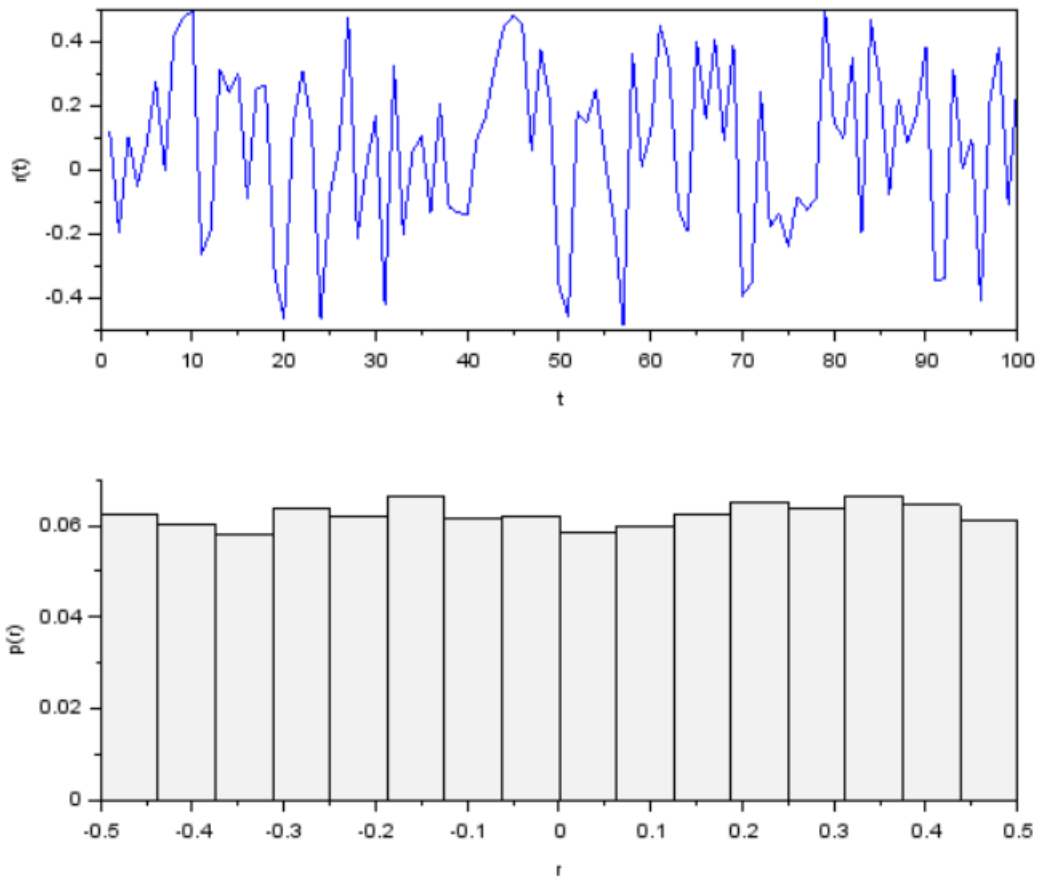
Scilab segítségével hozzunk létre egyenletes eloszlású zajt, majd ábrázoljuk azt és annak eloszlásfüggvényét.

Megoldás:

Egyenletes eloszlás esetén a véletlenszám generátor 0 és 1 közötti értékeket fog létrehozni, így 0,5-öt ki kell vonni minden értékből, hogy nulla körüli értékeket kapjunk.

```
r=rand(10000,1,"uniform")-0.5;  
subplot(2,1,1)  
plot(r(1:100))  
xlabel('t','r(t)')  
subplot(2,1,2)  
histplot(16,r);  
xlabel('r','p(r)')
```

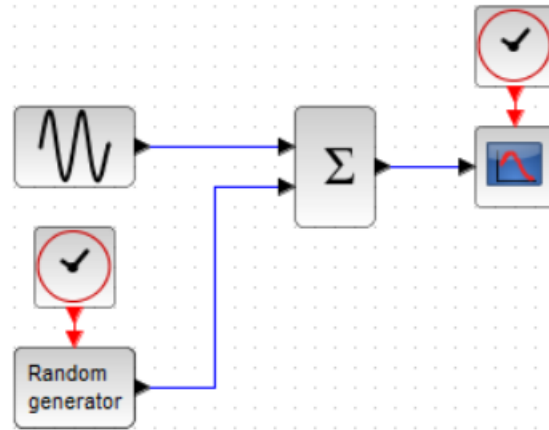

Eredmény:



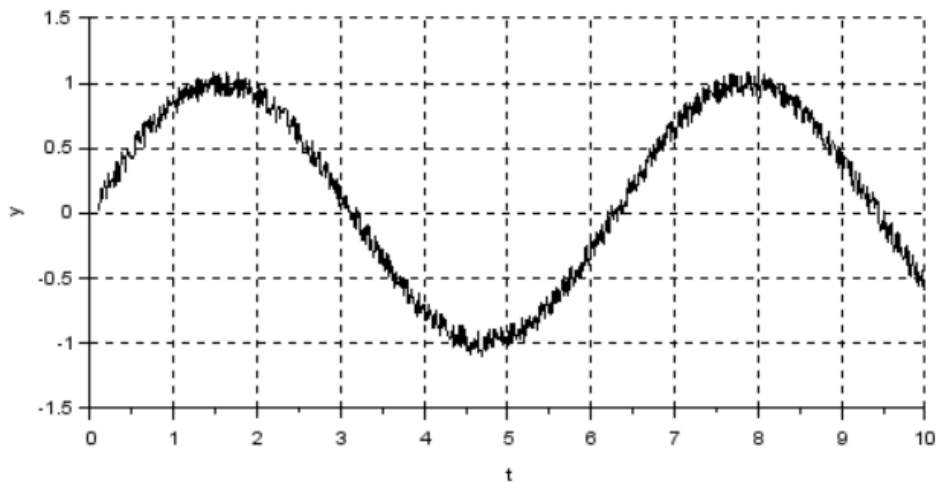
2.3 Példa

Xcos segítségével adjunk zajt egy szinuszjelhez. A jel amplitúdója legyen 1, a zaj pedig legyen egyenletes eloszlású -0.1 és 0.1 értékek között.

Megoldás:



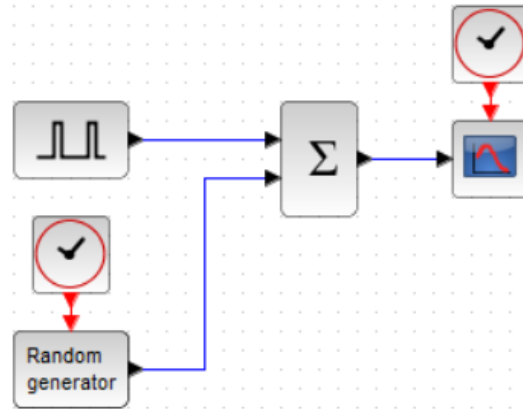
Eredmény:



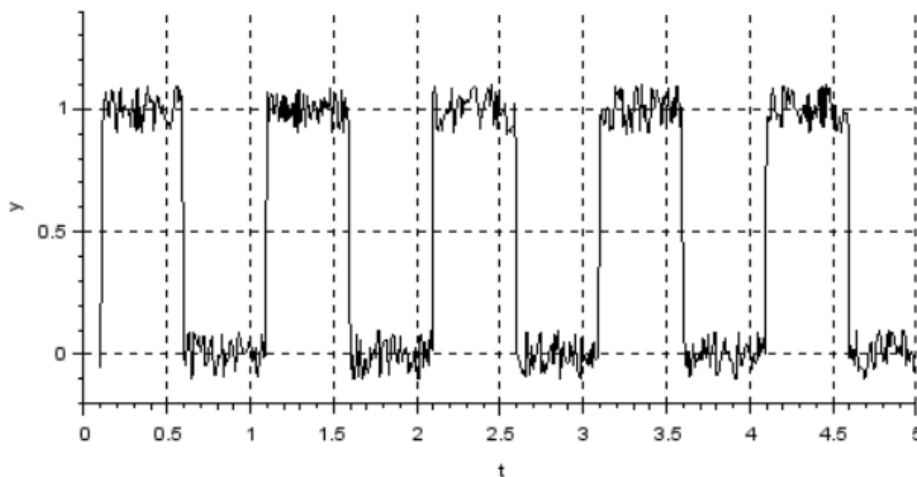
2.4 Példa

Xcos segítségével adjunk az előző példában alkalmazott paraméterekkel rendelkező zajt egy négyszögjelhez, melynek amplitúdója 1.

Megoldás:



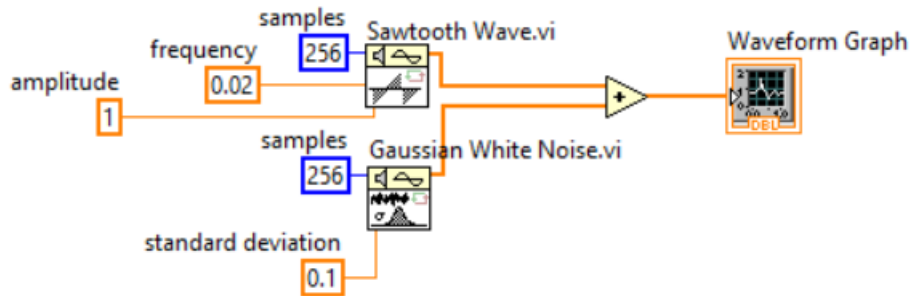
Eredmény:



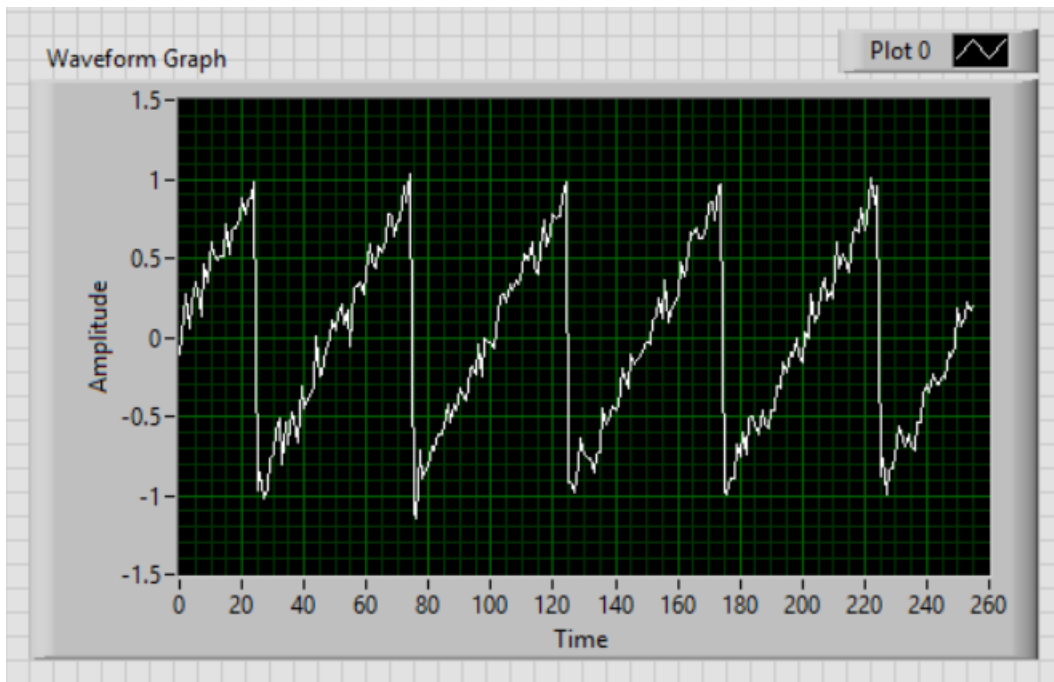
2.5 Példa

LabVIEW segítségével adjunk normális eloszlású zajt egy fűrészelhez

Megoldás:



Eredmény:



2.5 Ellenőrző kérdések

1. [Miért jelentős a matematikai modell?](#)
2. [Mi a rendszeridentifikáció?](#)
3. [Melyek a digitális technika legfontosabb előnyei?](#)
4. [Miben különböznek a FPGA-k a többi említett jelfeldolgozásra alkalmas eszköztől?](#)
5. [Teljesítménysűrűség-spektrumuk és eloszlásuk szerint milyenek a fehér zajok?](#)



3 A/D és D/A átalakítók

Ha az analóg jeleket digitális eszközzel szeretnénk feldolgozni, akkor a jelet először digitalizálnunk kell. A digitalizáció folyamatát analóg-digitális (A/D) átalakító végzi. A digitális tartományban végzett feldolgozás után sok esetben szükség van a feldolgozott számsorok analóg jellé való átalakítására, amit digitális-analóg (D/A) átalakítóval végezhetünk.

3.1 A/D átalakítás

A digitalizálás diszkrét formára hozást jelent mind időben mind amplitúdóban. Az időtartománybeli diszkrétizálást mintavételezésnek, az amplitúdótartománybelit pedig kvantálásnak nevezzük.

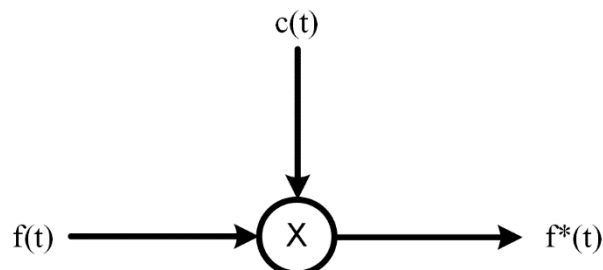
3.1.1 Mintavételezés

A folytonos idejű jelek mintavételezésével állítjuk elő az időben diszkrét jeleket. A mintavételezett jel általában az eredeti jel periodikusan felvett értékeit tartalmazza egy mintasorozat formájában.

Egy ideális mintavételező tulajdonképpen egy szorzó áramkör, melynek működése a következő módon írható le:

$$f^*(t) = c(t)f(t) \quad (3.1)$$

ahol $f^*(t)$ a mintavételezett jel, $c(t)$ a mintavevő jel, $f(t)$ pedig a mintavételezett folytonos jel.



3.1 ábra: Mintavételező szorzó áramkör.

A $c(t)$ mintavevő jel lényegében egy periodikus impulzussorozat amely leírható mint:

$$c(t) = \sum_{n=-\infty}^{+\infty} \delta(t - nT_s) \quad (3.2)$$

ahol $\delta(t)$ a Dirac-impulzus, T_s pedig a mintavételezési idő. Tehát a mintavételező jel egységimpulzusok sorozata, amelyek T_s időközönként vesznek mintát a jelből, így a mintavételezési időpontok $0, T_s, 2T_s, 3T_s, \dots$

A mintavételező jel képletét behelyettesítve:

$$f^*(t) = f(t) \sum_{n=-\infty}^{+\infty} \delta(t - nT_s) \quad (3.3)$$

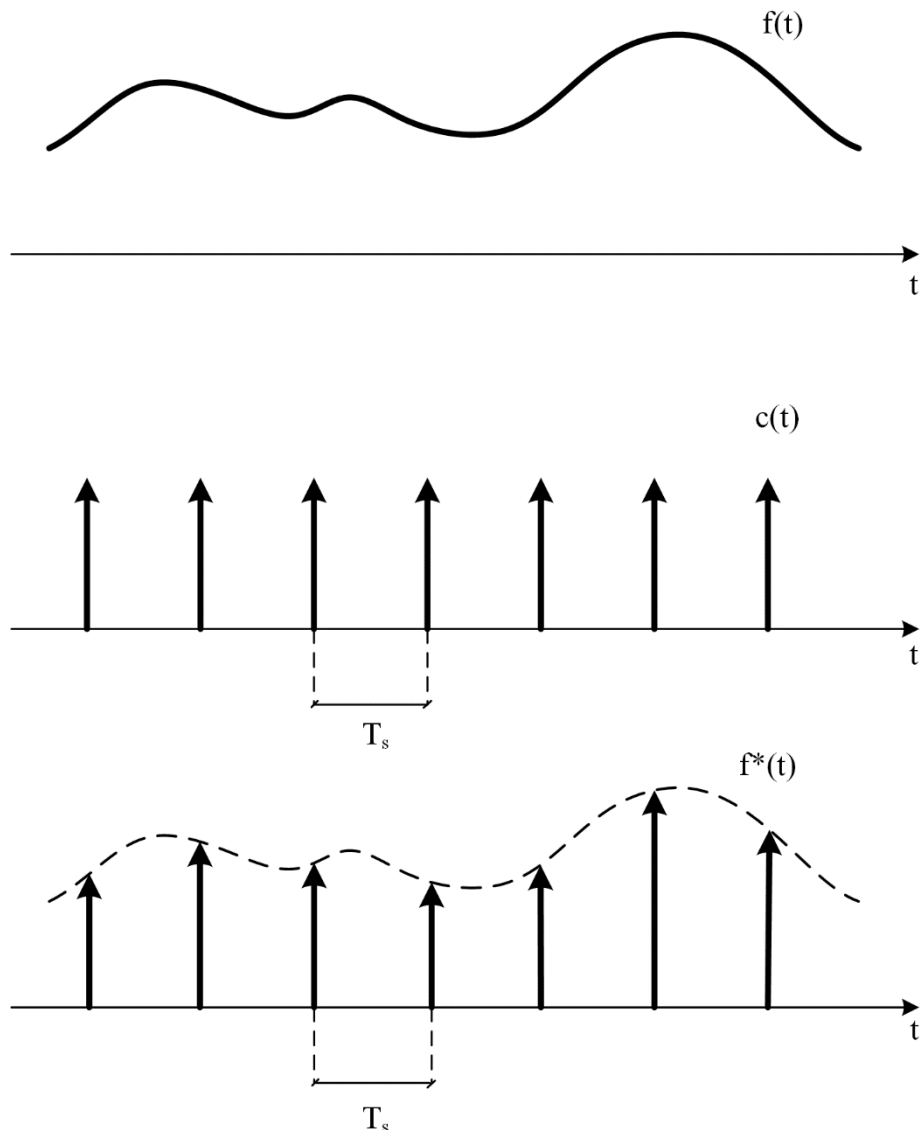
A mintavételezett jelre így megkapjuk:

$$f^*(t) = \sum_{n=-\infty}^{+\infty} f(nT_s) \delta(t - nT_s) = f(0)\delta(0) + f(T_s)\delta(t - T_s) + f(2T_s)\delta(t - 2T_s) + \dots \quad (3.4)$$

A mintavételezési frekvencia, f_s , a mintavételi idő alapján megadható a következő képlettel:

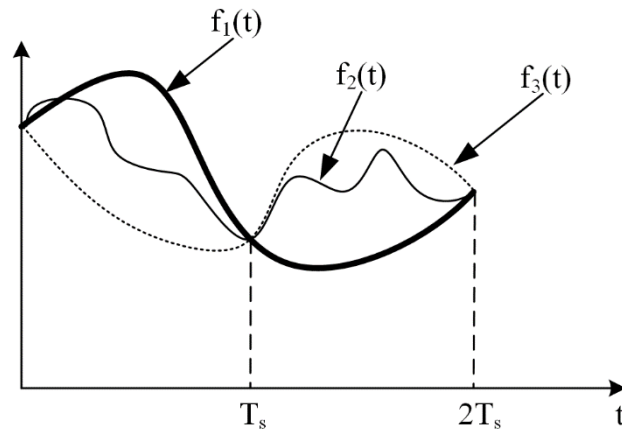
$$f_s = \frac{1}{T_s} \quad (3.5)$$

Az ideális mintavételező működése felfogható egy ideális kapcsolóként is, amely T_s időközönként zár, és nulla ideig ereszt át.



3.2 ábra: Az ideális mintavételező működése

A mintavételezés információvesztéssel is jár, mivel csak az adott mintavételezési pontokban kapunk információt, de a jel alakulásáról két mintavételi időpont között nem. A 3.3 ábrán látható, hogy több függvény létezik amelyek mintavételezésével ugyanazokat az értékeket kapjuk.



3.3 ábra: A mintavételezés információvesztése.

Mintavételezési tétel

Fourier transzformáció segítségével, amelyről a későbbi fejezetekben lesz szó, a jelet átranzformálhatjuk időtartományból frekvenciatartományba, és megkaphatjuk a jel spektrumát. A mintavételezett jelet Fourier transzformálva a következőt kapjuk:

$$F^*(j\omega) = \frac{1}{T_s} \sum_{n=-\infty}^{+\infty} F(j\omega + jn\omega_s) \quad (3.6)$$

ahol ω_s a mintavételezési frekvenciához tartozó körfrekvencia:

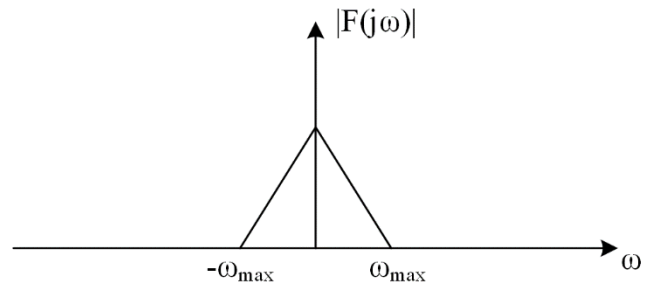
$$\omega_s = \frac{2\pi}{T_s} \quad (3.7)$$

Amint látható a (3.6) képletből, a mintavételezés hatására a spektrum periodikusan ismétlődik ω_s periódussal, vagyis a mintavételezett jel spektruma egy periodikus függvény, amely az eredeti $F(j\omega)$ spektrum eltoltságainak az összege. Az eredeti jel spektrumának kiemeléséhez szűrést kell végeznünk egy aluláteresztő szűrő segítségével.

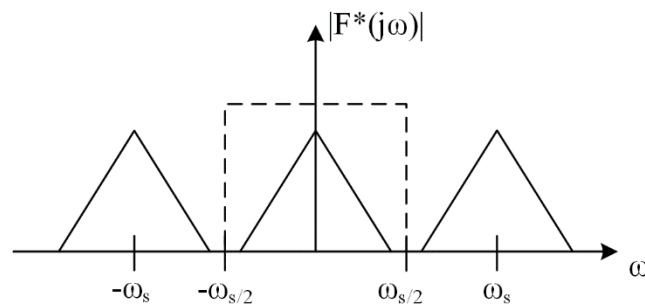
Feltételezzük, hogy az $f(t)$ jel spektrálisan korlátolt, tehát:

$$F(j\omega) = 0, \text{ ha } |\omega| \geq \frac{\omega_s}{2} \quad (3.8)$$

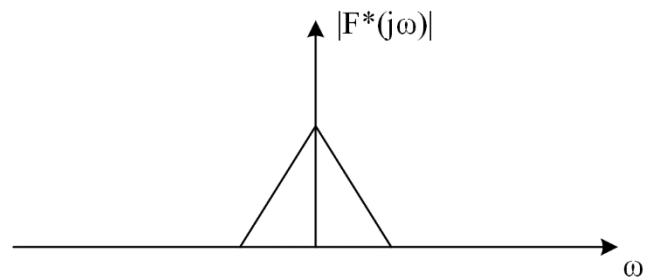
A 3.4 és 3.5 ábrák két különböző esetet mutatnak be az eredeti jel spektrumának kinyerésére.



a.)

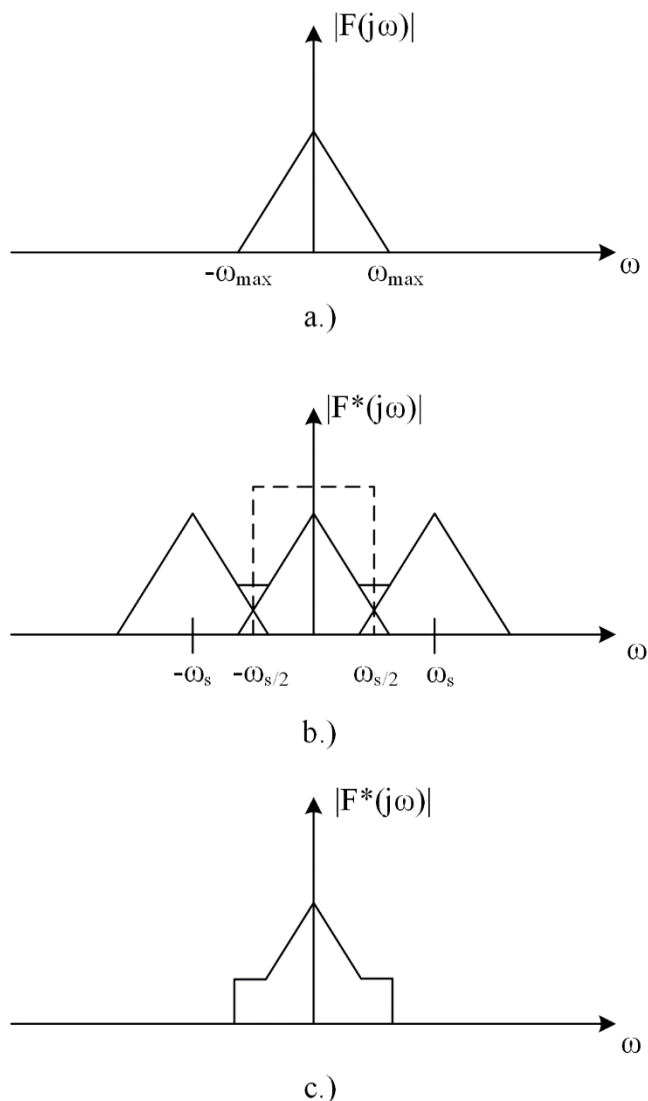


b.)



c.)

3.4 ábra: A spektrum kinyerése megfelelő frekvenciával történő mintavételezés esetén: a) az eredeti jel spektruma, b) a spektrum ismétlődése a mintavételezés hatására, c) a szűréssel kinyert spektrum.



3.5 ábra: A spektrum kinyerése túl alacsony frekvenciával történő mintavételezés esetén: a) az eredeti jel spektruma, b) a spektrum ismétlődése a mintavételezés hatására, c) a szűréssel kinyert spektrum.

Látható, hogy az első esetben a szűrés segítségével visszakapható az eredeti spektrum, viszont a másodikban nem. Ennek oka, hogy a második esetben a túl alacsony mintavételezési frekvencia hatására spektrális átfedések, vagyis aliasing effektus alakul ki.

Ez alapján a mintavételezési tétel, amelyet Nyquist-Shannon mintavételezési törvénynek is szokás nevezni, kimondja, hogy a mintavételezési frekvenciát úgy kell megválasztani, hogy az nagyobb legyen mint mintavételezett analóg jelben található legnagyobb frekvenciájú komponens (ω_{max}) frekvenciájának kétszerese. Vagyis:

$$\omega_s > 2\omega_{max} \quad (3.9)$$

Ez frekvenciákra is felírható:

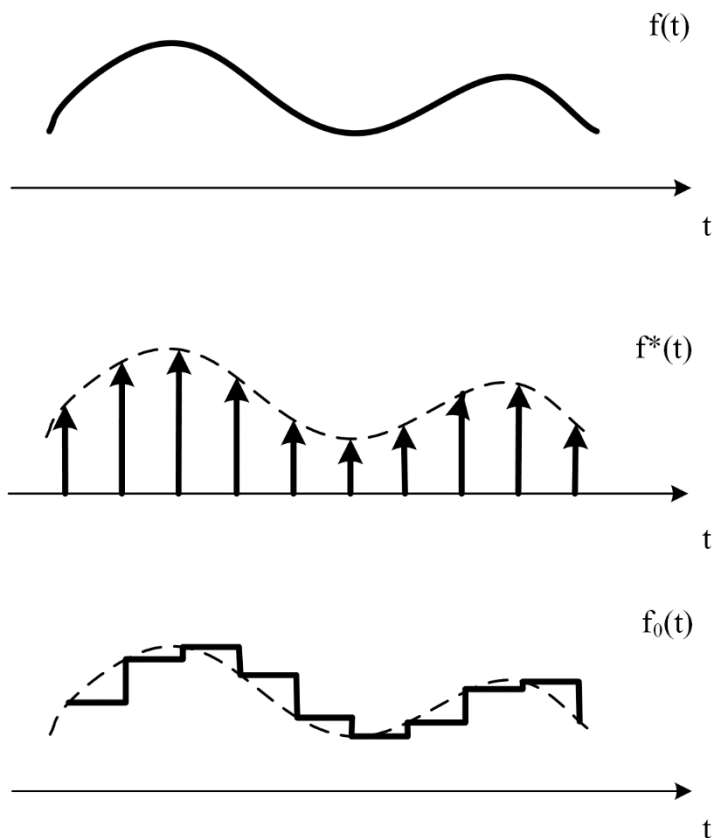
$$f_s > 2f_{max} \quad (3.10)$$

A mintavételezési tétel teljesülése esetén a jel teljes mértékben helyreállítható.

Példaként említhető a hang mintavételezése. Az emberi hallástartomány körülbelül 20Hz és 20kHz között található, így a mintavételezési törvény szerint több mint 40kHz-es frekvenciával szükséges a mintavételezést végezni, hogy a teljes tartomány spektrumát átvigyük. Általában 44100Hz és 48000Hz frekvenciákat szoktak alkalmazni.

A mintavételezés folyamata

Az előzőekben említett rövid és nagy impulzusok megvalósítása a gyakorlatban nehéz és költséges. Emiatt a mintavételezés leggyakrabban egy mintavevő-tartó (sample-and-hold) egységgel van megvalósítva. Lényegében ez egy nulladrendű tartószerv segítségével történik, amely a jel értékét a következő mintavételi pontig tartja.



3.6 ábra: A mintavételezés folyamata mintavevő-tartó egységgel.

Alul- és túlmintavételezés

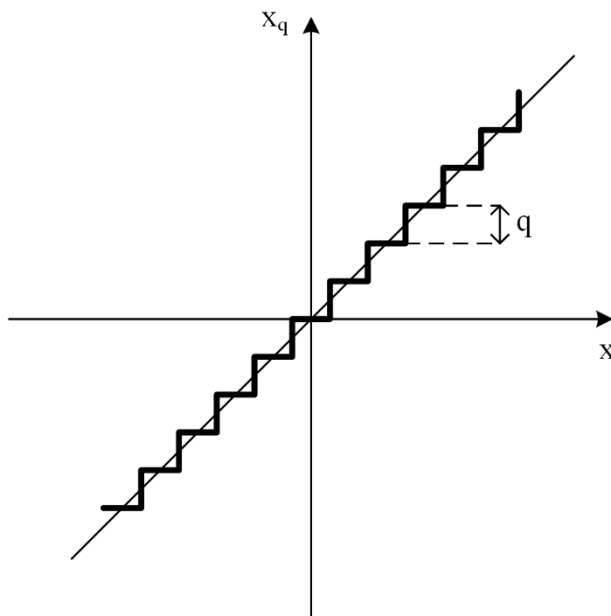
Léteznek olyan mérési feladatok amikor nem szükséges minden lehetséges információt kinyerni a jelből és elegendő alacsonyabb mintavételezési frekvenciát alkalmazni. Például a kinti hőmérséklet mérésénél nagy valószínűséggel nem érdekel minket minden apró változás, elég lehet akár óránként mintavételezni.

Vannak viszont olyan alkalmazások is, amikor célszerű sokkal magasabb mintavételi frekvenciát használni. A mintavételezési törvény alapján egy szinusz hullámból elegendő periódusonként valamivel több mint két mintát venni, viszont a mintákat ábrázolva valószínűleg szabad szemmel nehezen tudnánk felismerni az ábrázolt jelet. Tehát, ilyen esetben célszerű sokkal nagyobb frekvenciával mintavételezni. Tipikus példa ilyen esetre az oszcilloszkóp.

3.1.2 Kvantálás

A kvantáló egység végzi a diszkrét jelek kódolását véges számú bitek sorozatába, azaz 0-ás és 1-es értékekbe.

Kvantálás során a jel pillanatnyi értékéhez a hozzá legközelebb található kvantálási szintet kell hozzárendelni, tehát, leginkább kerekítésen alapszik.



3.7 ábra: A kvantálási görbe.

A kvantálási görbéből látható, hogy a kvantálás nemlineáris folyamat, mivel több bemeneti értékhez ugyanazt a kimeneti értéket rendeljük hozzá.

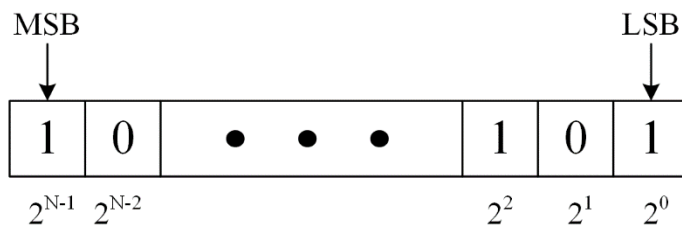
Egy tartomány szélessége a kvantum (q). Lényegében a kvantumok száma, mely 2^N alakban adható meg, határozza meg a kvantálási pontosságot. Például 8-bites átalakító esetén 2^8 , azaz 256.

A bemeneti feszültségtartomány lehet:

- Unipoláris ($0 - U_{max}$)
- Bipoláris ($-U_{max}/2 - U_{max}/2$)

A teljes tartomány (angolul Full Scale – FS) a következő:

$$FS = 2^N \cdot q \quad (3.11)$$



3.8 ábra: Bitek értéke.

A legalacsonyabb helyi értékű bit (angolul Least Significant Bit – LSB) a következő értéket képviseli:

$$LSB = \frac{FS}{2^N} = 2^0 \cdot q = q \tag{3.12}$$

A legmagasabb helyi értékű bité (angolul Most Significant Bit – MSB) pedig:

$$MSB = \frac{FS}{2} = 2^{N-1} \cdot q \tag{3.13}$$

3.1 táblázat: Felbontás nagysága 10V-os feszültségtartomány esetén különböző rezolúciókra.

Rezolúció (bitek száma)	Kvantumok száma	Felbontás [mV]
8	256	39,06
10	1024	9,77
12	4096	2,4
16	65536	0,15

A véges számú bitbe való kódolás hatására a kvantált jelben információvesztés alakul ki, annak ellenére, hogy magas felbontás esetén az eredményezett jel nagyon hasonlít az eredeti jelre. Ezt a hibát kvantálási hibának vagy kvantálási zajnak szokás nevezni. Ha a kvantálási szintek elég keskenyek, azaz nagy a felbontás, akkor a kvantálási zaj tekinthető egyenletes eloszlású, fehér spektrumú zajnak.

3.2 D/A átalakítás

A D/A átalakítók működése is két alapszámra vezethető vissza, a dekódolásra és a tartásra.

3.2.1 Dekódolás

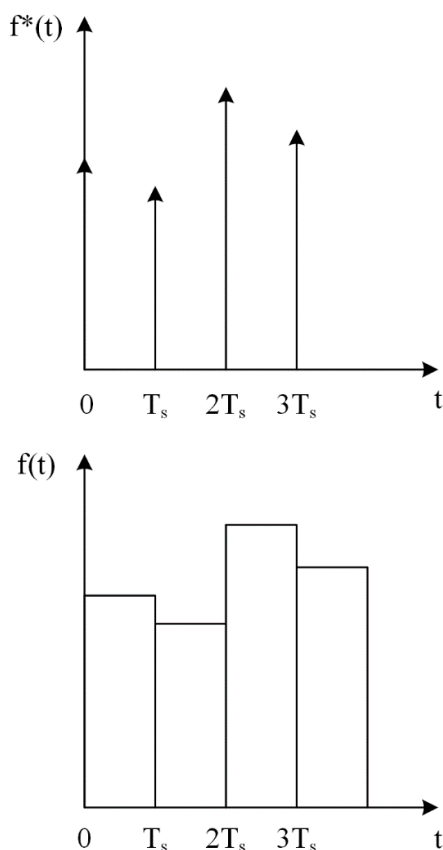
A dekódolás az adott digitális értékekhez megfelelő amplitúdót rendel. Ez a szerv lényegében analóg impulzusok sorozatát hozza létre.

3.2.2 Tartás

Tartás során a dekódolás folyamatából eredményezett impulzussorozatot a tartószerv analóg jellé alakítja át.

Ez a művelet nem egyértelmű, mivel két mintavételi pont között a jel alakja választható. A tartószervek N -edrendű extrapolációs polinom segítségével határozzák meg a mintavételi intervallumon belül a jel értékét, mely N darab korábbi mintavételi értéket alkalmaz.

A legegyszerűbb megoldást a zérusrendű tartószerv szolgáltatja. Ez az egész intervallumon keresztül tartja az impulzus értékét, tehát lépcsős görbét hoz létre.

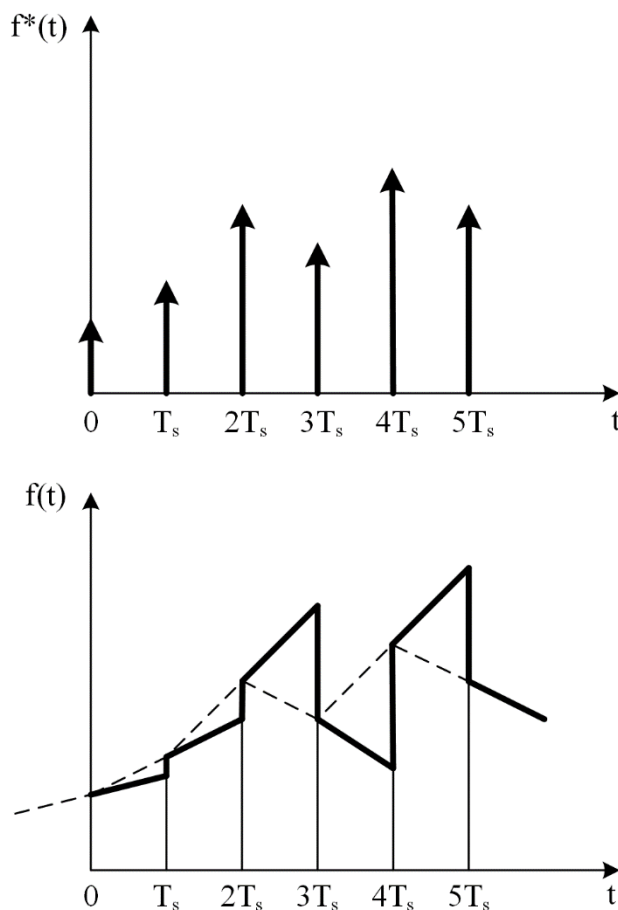


3.9 ábra: Zérusrendű tartószerv működése.

Az ilyen tartószerv működését egységugrás függvények segítségével a következő módon adhatjuk meg:

$$f(t) = \sum_{n=0}^{\infty} f(nT_s)[h(t - nT_s) - h(t - (n + 1)T_s)] \quad (3.14)$$

Az elsőrendű tartószerv olyan analóg jelet képez a kimeneten, amely az impulzus értékéből indul ki az intervallum kezdetén, majd lineárisan változik az intervallum végéig. A meredekséget az intervallum elején levő érték és az előző mintavételi érték határozzák meg.



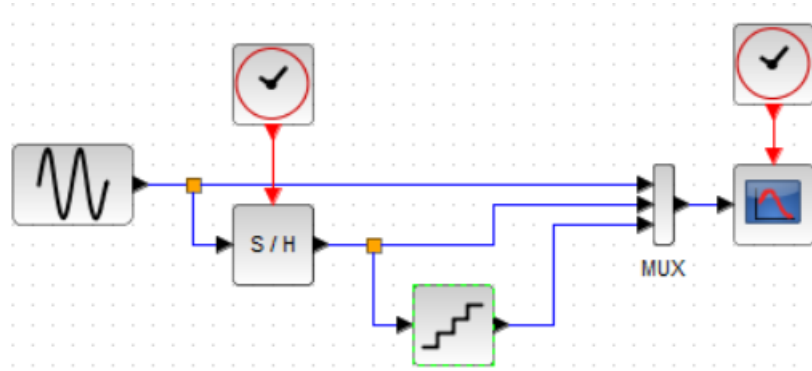
3.10 ábra: Elsőrendű tartószerv működése.

3.3 Példák

3.1 Példa

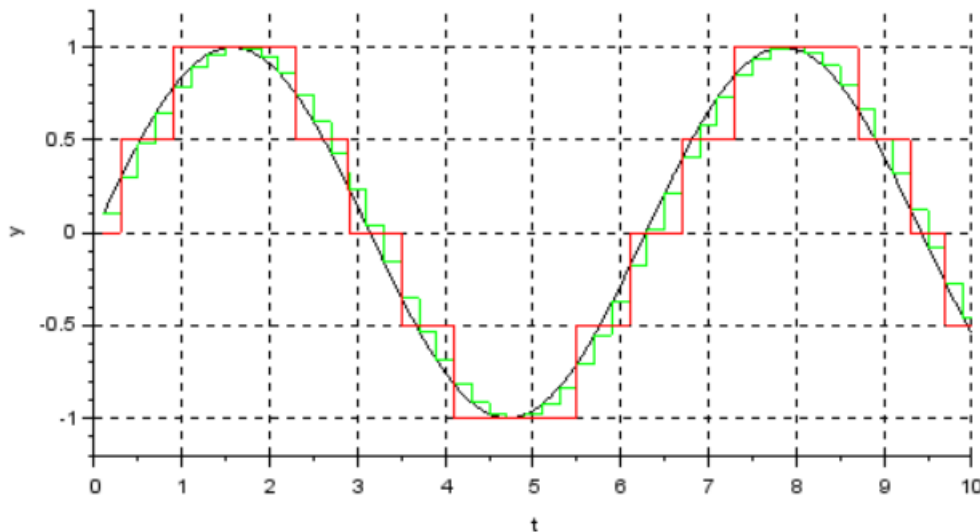
Xcos segítségével vizsgáljuk meg a mintavevő-tartó és a kvantáló elemek hatását egy szinusz jelre.

Megoldás:



Eredmény:

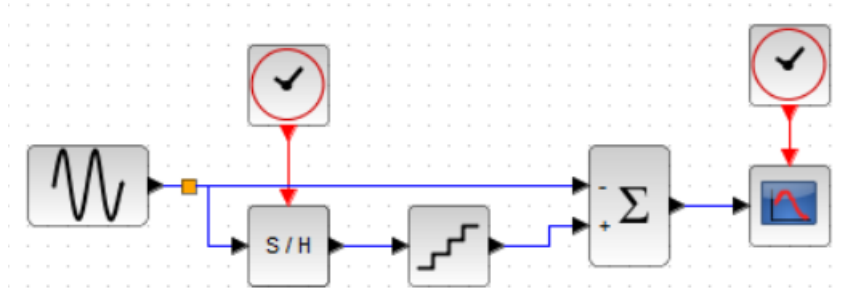
A mintavevő-tartó frissítési idejét 0,2s-ra, a kvantumok szélességét pedig 0,5-re állítva a következőt kapjuk (fekete – eredeti jel, zöld – mintavevő-tartó kimenete, piros – kvantáló kimenete):



3.2. Példa

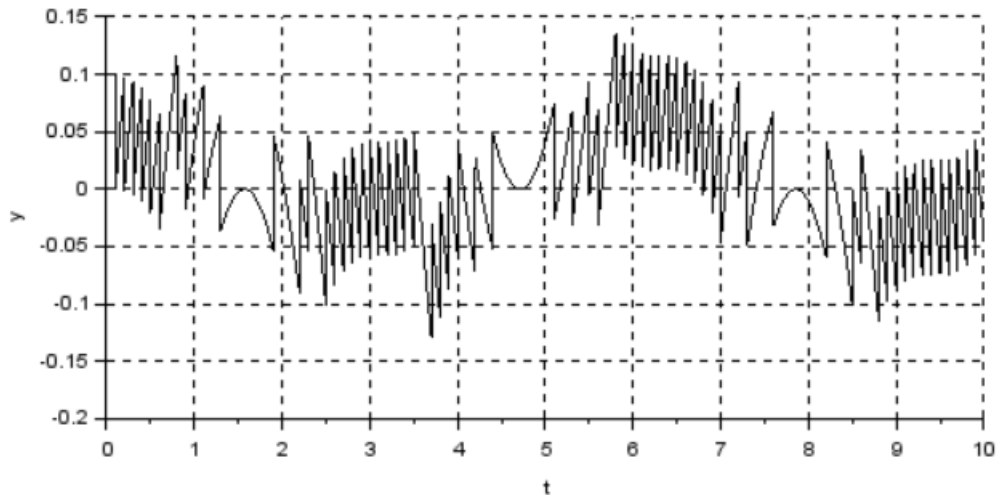
Xcos segítségével határozzuk meg a kvantálási hiba nagyságát szinusz jel esetén.

Megoldás:



Eredmény:

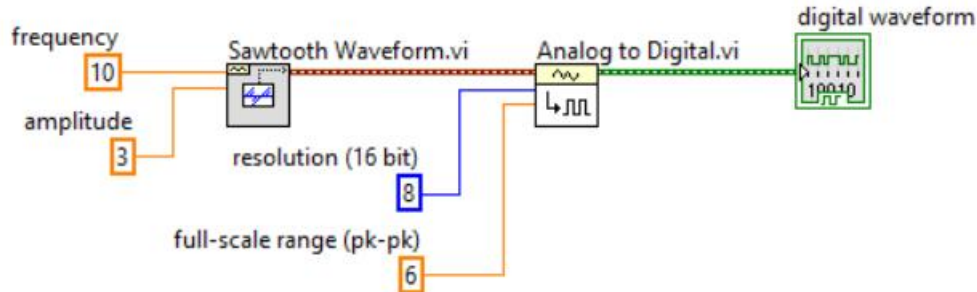
A mintavevő-tartó frissítési idejét 0,1s-ra, a kvantumok szélességét pedig 0,1-re állítva a következő eredményt kapjuk:



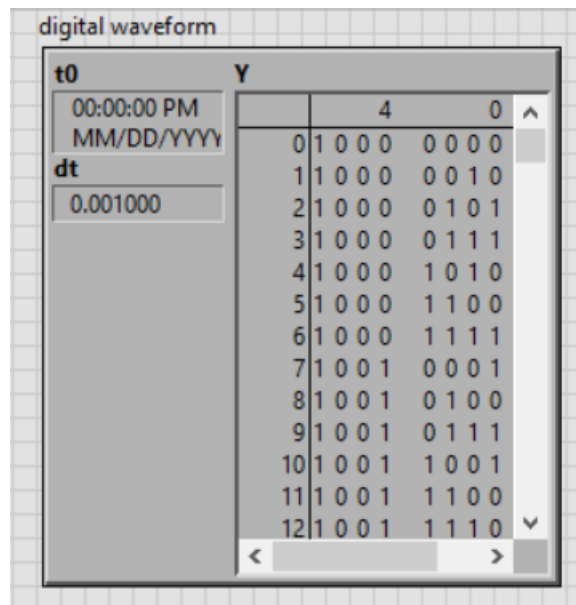
3.3. Példa

Végezzünk A/D átalakítást egy fűrészjelen LabVIEW segítségével.

Megoldás:



Eredmény:



3.4 Ellenőrző kérdések

1. [Miért alakul ki az aliasing effektus?](#)
2. [Milyen szerepe van a mintavevő-tartó egységnek?](#)
3. [Mikor célszerű túlmintavételezést alkalmazni?](#)
4. [Miből ered a kvantálási hiba?](#)
5. [A D/A átalakítás során miért van szükség tartásra?](#)

4 Konvolúciós és korrelációs függvények

4.1 Konvolúció

A konvolúció nagyon fontos művelete a jelfeldolgozásnak és a rendszertechnikának. A konvolúció (*) két jel felett értelmezett művelet, mely egy harmadikat eredményez.

4.1.1 Konvolúció folytonos időben

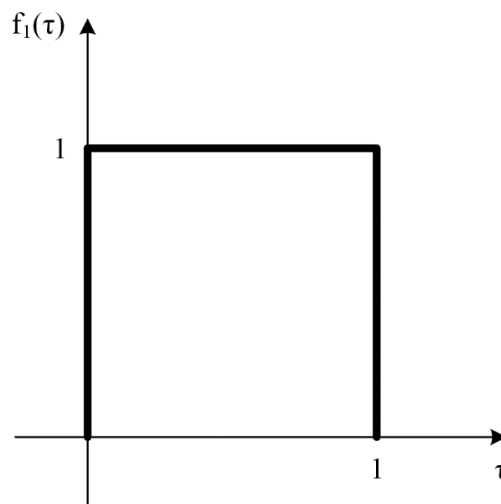
Folytonos időben a következő módon határozható meg a konvolúció, amelyet konvolúciós integrálnak is szokás nevezni.

$$f(t) = f_1(t) * f_2(t) = \int_{-\infty}^{+\infty} f_1(\tau) f_2(t - \tau) d\tau \quad (4.1)$$

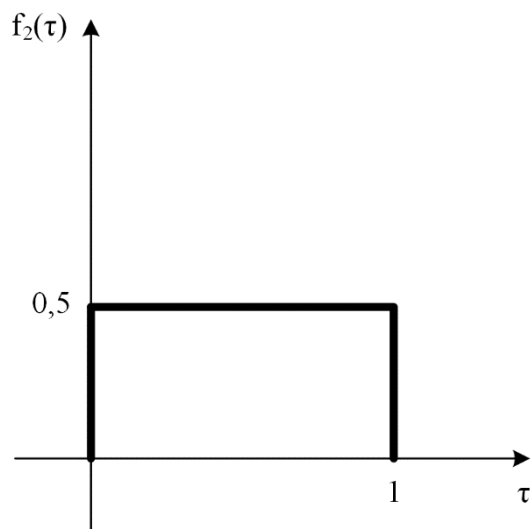
A művelet négy lépésre bontható:

- Reflexió – az $f_2(t)$ jelet tükrözzük, így létrejön $f_2(-\tau)$.
- Eltolás – a $f_2(-\tau)$ eltoljuk t -vel.
- Szorzás – az eltoló $f_2(-\tau)$ jelet szorozzuk $f_1(\tau)$ -val, így megkapjuk a két jel metszetét az adott időben.
- Integrálás – az adott t időben a konvolúció értéke a szorzatfüggvény területének nagysága, amelyet integrálással határozunk meg.

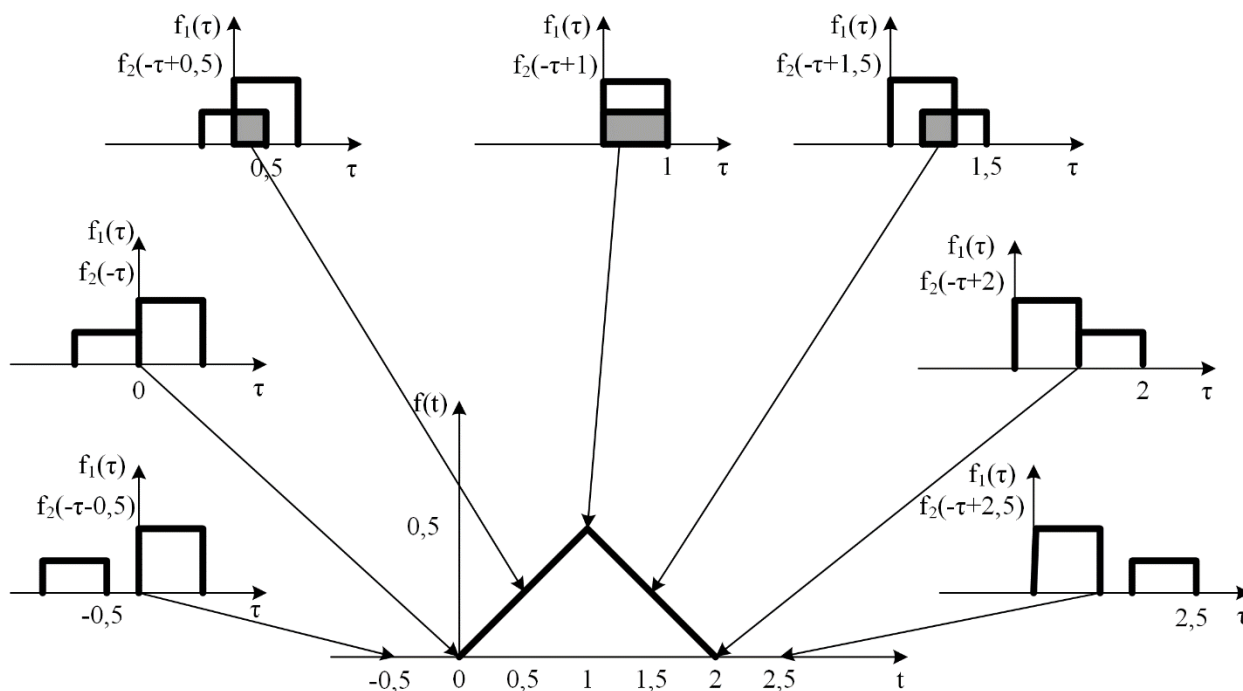
A konvolúció folyamatát a 4.1 és 4.2 ábrákon látható négyszög jelek esetén a 4.3 ábra szemlélteti.



4.1 ábra: Első alkalmazott négyszögjel.



4.2 ábra: Második alkalmazott négyzögjel.

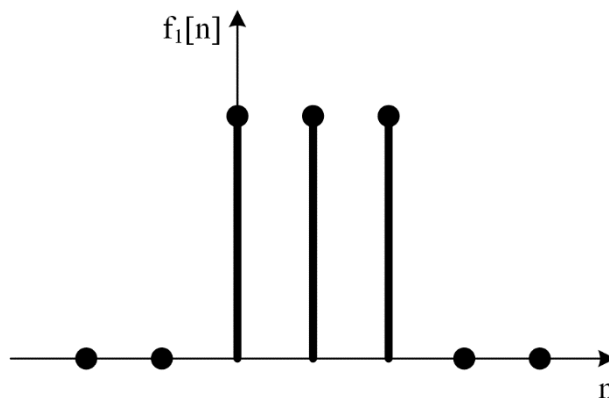


4.3 ábra: A folytonos idejű konvolúció folyamata és eredménye.

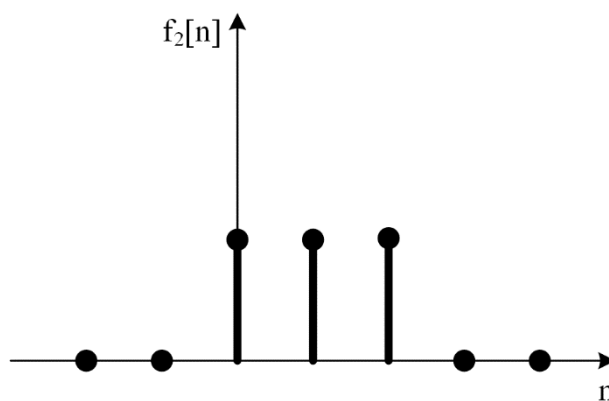
4.1.2 Konvolúció diszkrét időben

Diszkrét idejű esetben a művelet lépései ugyanazok, viszont az utolsó lépésben integrálás helyett szummázást végzünk, így ezt konvolúciós összegnek is nevezzük.

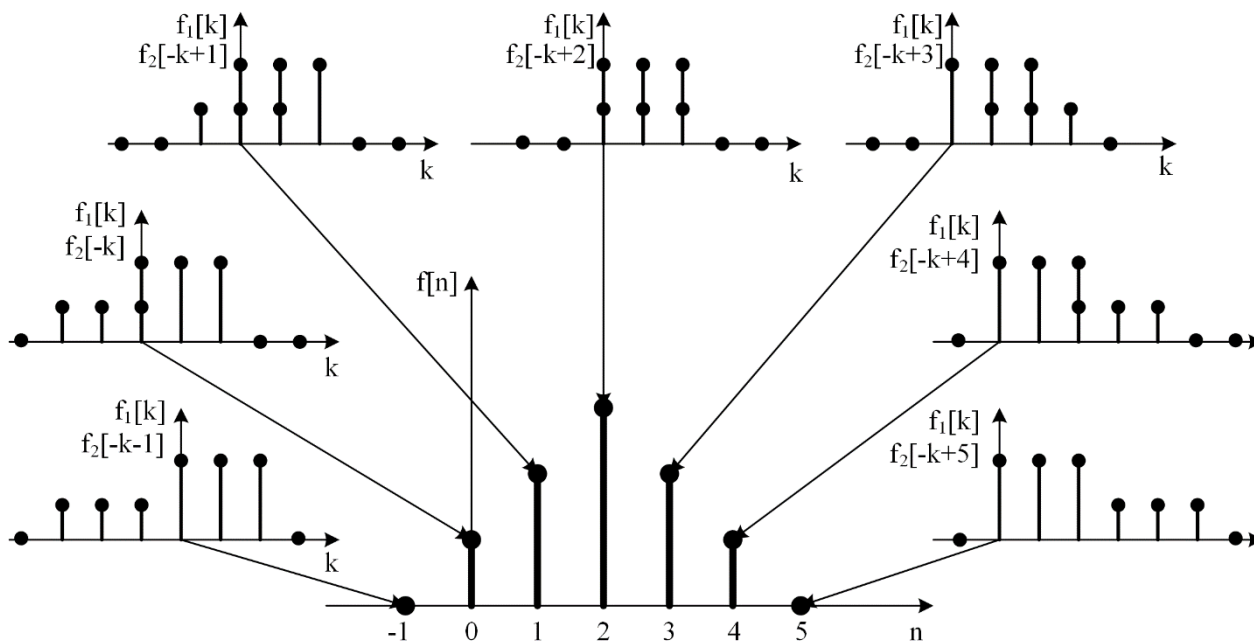
$$f[n] = f_1[n] * f_2[n] = \sum_{k=-\infty}^{\infty} f_1[k]f_2[n-k] \quad (4.2)$$



4.4 ábra: A példában alkalmazott első négyszögjel.



4.5 ábra: A második alkalmazott négyszögjel.



4.6 ábra: A diszkrét idejű konvolúció lépései és eredménye.

A konvolúció eredményeként kapott jel hossza (N_f) diszkrét esetben meghatározható a két jel hossza (N_{f1} és N_{f2}) alapján a következő képlettel:

$$N_f = N_{f1} + N_{f2} - 1 \tag{4.3}$$

4.1.3 A konvolúció tulajdonságai

Kommutativitás

$$f_1(t) * f_2(t) = f_2(t) * f_1(t) \tag{4.4}$$

$$f_1[n] * f_2[n] = f_2[n] * f_1[n] \tag{4.5}$$

Asszociativitás

$$(f_1(t) * f_2(t)) * f_3(t) = f_1(t) * (f_2(t) * f_3(t)) \tag{4.6}$$

$$(f_1[n] * f_2[n]) * f_3[n] = f_1[n] * (f_2[n] * f_3[n]) \tag{4.7}$$

Disztributivitás

$$(f_1(t) + f_2(t)) * f_3(t) = f_1(t) * f_3(t) + f_2(t) * f_3(t) \quad (4.8)$$

$$(f_1[n] + f_2[n]) * f_3[n] = f_1[n] * f_3[n] + f_2[n] * f_3[n] \quad (4.9)$$

Reflexió

$$f(-t) = f_1(-t) * f_2(-t) \quad (4.10)$$

$$f[-n] = f_1[-n] * f_2[-n] \quad (4.11)$$

Időbeni eltolás

$$f(t - t_1 - t_2) = f_1(t - t_1) * f_2(t - t_2) \quad (4.12)$$

$$f[n - n_1 - t_1] = f_1[n - n_1] * f_2[n - n_2] \quad (4.13)$$

Konvolúció Dirac-impulzussal

$$f(t) = f_1(t) * \delta(t - t_0) = f_1(t - t_0) \quad (4.14)$$

$$f[n] = f_1[n] * \delta[n - n_0] = f_1[n - n_0] \quad (4.15)$$

4.2 Keresztkorreláció

A korreláció vagy keresztkorreláció szintén két jelből egy harmadikat eredményez. A konvolúció és a keresztkorreláció műveletek lépései között az a különbség, hogy korreláció esetében nem végzünk tükrözést.

A művelet két jel közötti hasonlóságáról ad információt, megmutatja, hogy az egyik jelben milyen mértékben található meg a másik. Például ha egy hosszabb jelből kiveszünk egy darabot, majd az eredeti jellel korrelációt végzünk, akkor az eredményezett függvényben egy maximumot találunk a jel eredeti helyén, mivel ott teljes fedést kapunk.

$$R_{xy}(t) = \int_{-\infty}^{\infty} x(\tau)y(\tau + t) d\tau \quad (4.16)$$

$$R_{xy}[m] = \sum_{n=-\infty}^{\infty} x[n]y[n + m] \quad (4.17)$$

A korreláció visszavezethető konvolúcióra:

$$R_{xy}(t) = x(-t) * y(t) \quad (4.18)$$

$$R_{xy}[m] = x[-m] * y[m] \quad (4.19)$$

4.3 Autokorreláció

Az autokorreláció a keresztkorreláció speciális esete, amikor egy jelen önmagával végzünk korrelációt.

$$R_{xx}(t) = \int_{-\infty}^{\infty} x(\tau)x(\tau + t) d\tau \quad (4.20)$$

$$R_{xx}[m] = \sum_{n=-\infty}^{\infty} x[n]x[n + m] \quad (4.21)$$

Az autokorreláció felhasználható például periódusok kimutatására az adatsorokban. Legfontosabb felhasználása azonban, hogy segítségével meghatározható a jel $S_{xx}(f)$ teljesítménysűrűség-spektruma (angolul Power Spectral Density – PSD), mely már említésre került a zajok kapcsán. A teljesítménysűrűség-spektrum az autokorrelációs függvény Fourier transzformálásával kapható meg. A Fourier transzformáció a következő fejezetben lesz tárgyalva.

$$S_{xx}(f) = \int_{-\infty}^{\infty} R_{xx}(t)e^{-2\pi f t} dt \quad (4.22)$$

A teljesítménysűrűség-spektrumból a következő inverz kifejezéssel kapható vissza az autokorrelációs függvény:

$$R_{xx}(t) = \int_{-\infty}^{\infty} S_{xx}(f)e^{2\pi f t} df \quad (4.23)$$

Az autokorreláció tulajdonságai

- Az $R_{xx}(0)$ mindig a jel teljes energiáját adja.

$$R_{xx}(0) = \int_{-\infty}^{\infty} x(\tau)x(\tau + 0) d\tau = \int_{-\infty}^{\infty} x^2(\tau) d\tau \quad (4.24)$$

$$R_{xx}[0] = \sum_{n=-\infty}^{\infty} x[n]x[n + 0] = \sum_{n=-\infty}^{\infty} x^2[n] \quad (4.25)$$

- Az autokorreláció eredményeként kapott függvény amplitúdó értéke egy pontban sem lehet nagyobb a nulla eltolásnál kapott értéknél.

- Az autokorrelációs függvény mindig páros.

$$R_{xx}(t) = R_{xx}(-t) \quad (4.26)$$

$$R_{xx}[m] = R_{xx}[-m] \quad (4.27)$$

- Az autokorreláció nem változik ha a független változóban eltoljuk a jelet.

4.4 Példák

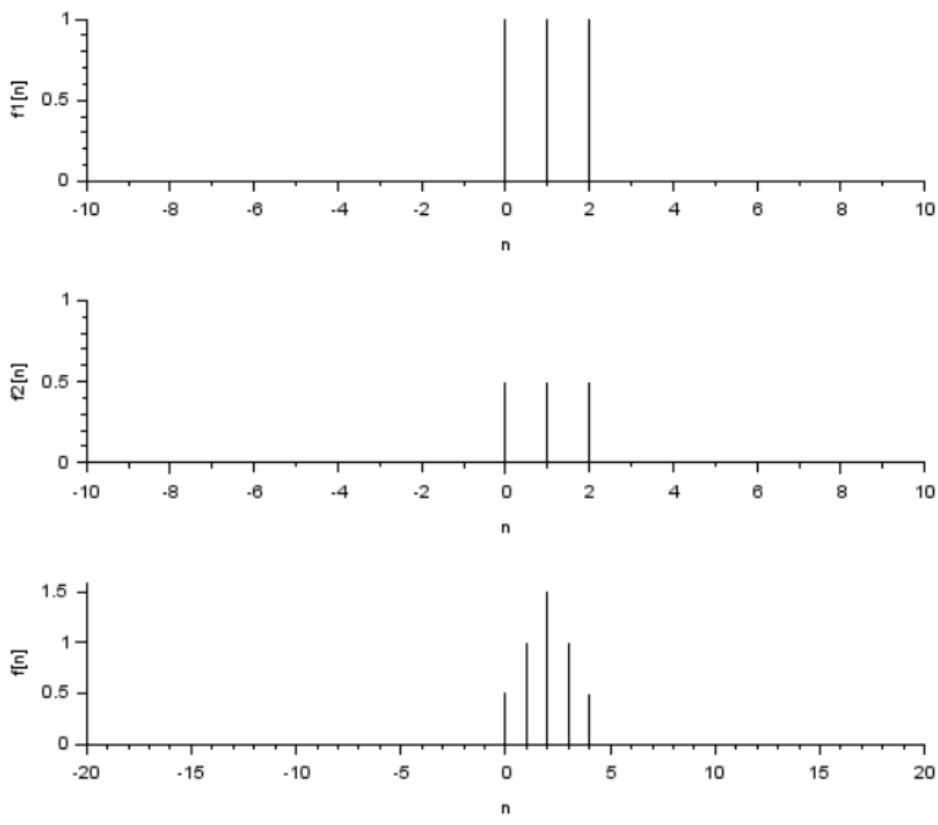
4.1 Példa

Valósítsuk meg a fejezetben található példát diszkrét idejű konvolúcióra Scilab segítségével.

Megoldás:

```
n=-10:10;
f_1=ones(1,size(n,2))
f_1=f_1.*((n>=0)&(n<=2));
f_2=0.5*ones(1,size(n,2));
f_2=f_2.*((n>=0)&(n<=2));
f=conv(f_1,f_2);
n_conv=(-((size(f,2)-1)/2):((size(f,2)-1)/2));
figure
subplot(3,1,1)
plot2d3(n,f_1)
xlabel("n");
ylabel("f1[n]");
subplot(3,1,2)
plot2d3(n,f_2)
xlabel("n");
ylabel("f2[n]");
subplot(3,1,3)
plot2d3(n_conv,f)
xlabel("n");
ylabel("f[n]");
```

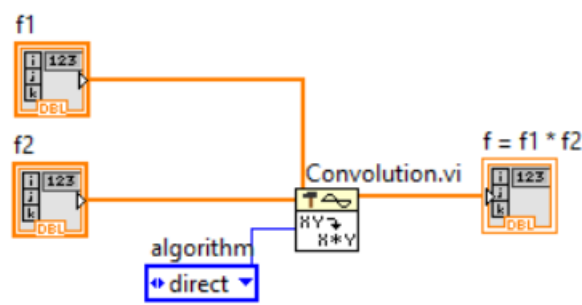
Eredmény:



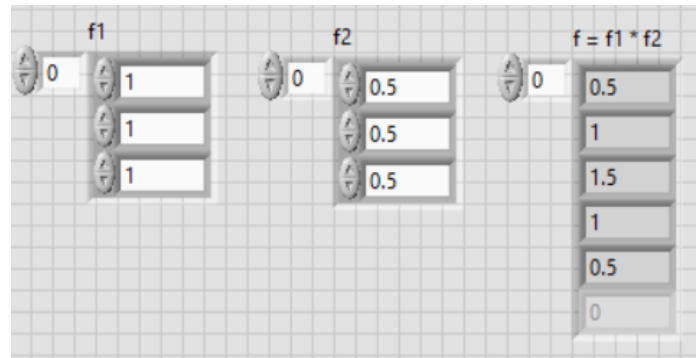
4.2 Példa

Az előző példát valósítsuk meg LabVIEW segítségével.

Megoldás:



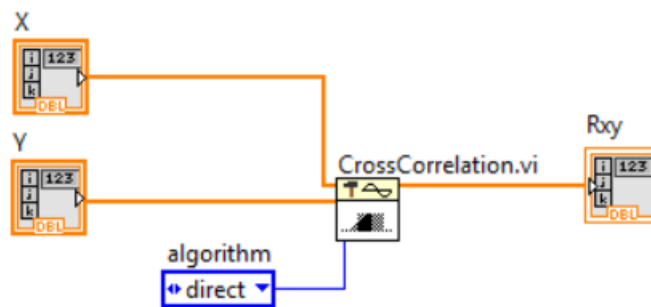
Eredmény:



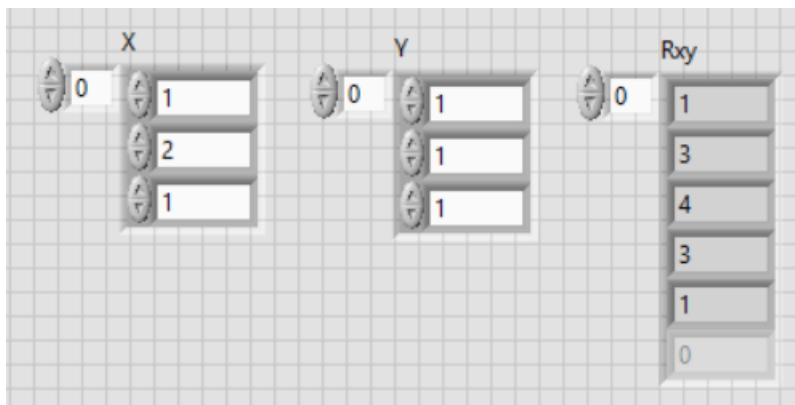
4.3 Példa

LabVIEW segítségével határozzuk meg két jel keresztkorrelációját.

Megoldás:



Eredmény:



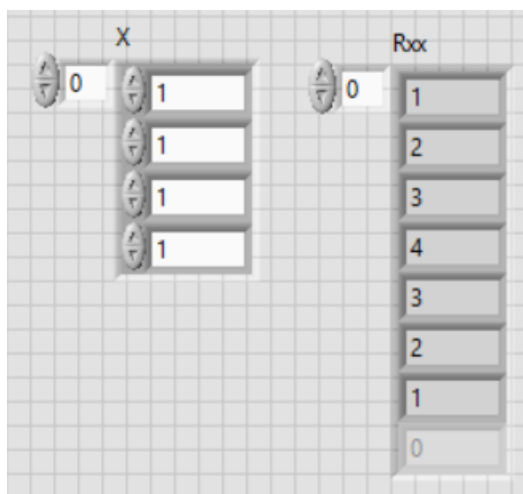
4.4 Példa

Végezzünk autokorrelációt LabVIEW segítségével.

Megoldás:



Eredmény:



4.5 Példa

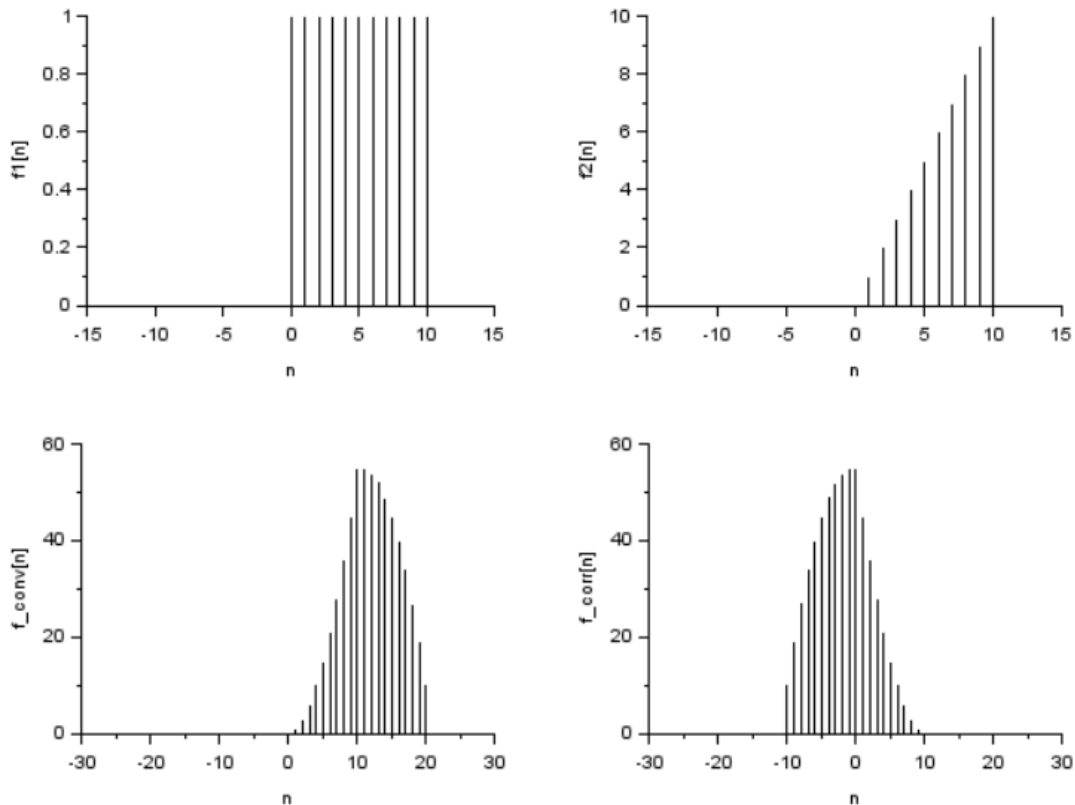
Scilab segítségével végezzünk konvolúciót és keresztkorrelációt egy fűrészel és egy négyszögjel segítségével, majd ábrázoljuk az eredmények közti különbséget.

Megoldás:

```
n=-15:15;
f_1=ones(1, size(n,2))
f_1=f_1.*((n>=0)&(n<=10));
f_2=n.*ones(1, size(n,2));
f_2=f_2.*((n>=0)&(n<=10));
f_conv=conv(f_1, f_2);
f_corr=xcorr(f_1, f_2)
n_conv=-((size(f_conv,2)-1)/2):((size(f_conv,2)-1)/2));
figure
subplot(2,2,1)
plot2d3(n, f_1)
xlabel("n");
ylabel("f1[n]");
subplot(2,2,2)
plot2d3(n, f_2)
```

```
xlabel("n");
ylabel("f2[n]");
subplot(2,2,3)
plot2d3(n_conv,f_conv)
xlabel("n");
ylabel("f_conv[n]");
subplot(2,2,4)
plot2d3(n_conv,f_corr)
xlabel("n");
ylabel("f_corr[n]");
```

Eredmény:



Az eredményből látható, hogy a konvolúció és a keresztkorreláció által kapott eredmények egymás tükörképei.

4.6 Példa

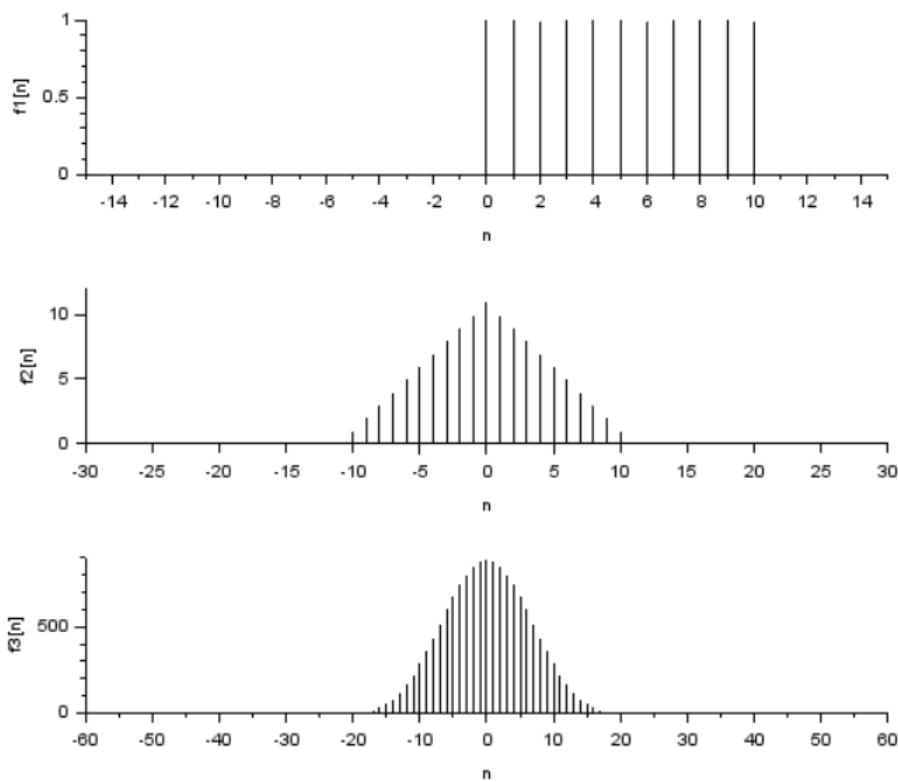
Scilab segítségével végezzünk autokorrelációt egy négyszögjelen, majd az eredményen végezzünk újra autokorrelációt.

Megoldás:

```
n1=-15:15;
f1=ones(1,size(n1,2))
f1=f1.*((n1>=0)&(n1<=10));
f2=xcorr(f1)
n2=-((size(f2,2)-1)/2):(size(f2,2)-1)/2);
f3=xcorr(f2)
```

```
n3= -((size(f3,2)-1)/2) : ((size(f3,2)-1)/2);  
figure  
subplot(3,1,1)  
plot2d3(n1,f1)  
xlabel("n");  
ylabel("f1[n]");  
subplot(3,1,2)  
plot2d3(n2,f2)  
xlabel("n");  
ylabel("f2[n]");  
subplot(3,1,3)  
plot2d3(n3,f3)  
xlabel("n");  
ylabel("f3[n]");
```

Eredmény:



Az eredményből látható, hogy a négyszögjel autókorrelációja háromszöget ad, míg a háromszögé Gauss-görbét.

4.5 Ellenőrző kérdések

1. [Melyek a konvolúció lépései?](#)
2. [Diszkrét konvolúció esetén milyen hosszú lesz az eredményezett jel?](#)
3. [Mi a különbség a konvolúció és a keresztkorreláció között?](#)
4. [Mire használható a keresztkorreláció?](#)
5. [Mely speciális esete a keresztkorrelációnak az autokorreláció?](#)

5 Fourier-sor és Fourier transzformáció

A Fourier analízis nagyon fontos eszköze a jelfeldolgozásnak. Analóg jelek esetén a Fourier-sor és a Fourier transzformáció segítségével frekvenciatartományba transzformálhatjuk a jelet, melyek közül a Fourier-sorba fejtés periodikus a Fourier transzformáció pedig aperiodikus jeleknél alkalmazható.

5.1 Fourier-sor

Minden szakaszonként folytonos és differenciálható korlátolt periodikus függvény felbontható harmonikus szinuszos és koszinuszos rezgések összegére, vagyis előállítható Fourier-sor alakjában.

Egy $f(t, T)$ periodikus függvény Fourier-sorát a következő alakban adhatjuk meg:

$$f(t, T) = \sum_{i=0}^{\infty} (A_i \cos(i\omega t) + B_i \sin(i\omega t)), i = 1, 2, 3, \dots \quad (5.1)$$

Az együtthatók a következő módon számíthatók ki:

$$A_0 = \frac{1}{T} \int_{-\frac{T}{2}}^{\frac{T}{2}} f(t, T) dt \quad (5.2)$$

$$A_i = \frac{2}{T} \int_{-\frac{T}{2}}^{\frac{T}{2}} f(t, T) \cos(i\omega t) dt \quad (5.3)$$

$$B_i = \frac{2}{T} \int_{-\frac{T}{2}}^{\frac{T}{2}} f(t, T) \sin(i\omega t) dt \quad (5.4)$$

Az $i=1$ indexű tagot alapharmonikusnak, míg a többi összetevőt felharmonikusoknak nevezzük. Az A_0 az ún. egyenáramú komponens, a B_0 tag mindenképp 0-át eredményez. Ennek oka, hogy nullás i érték esetén a trigonometrikus függvények argumentuma mindkét esetben 0, bármilyen ω és t értékekre. A koszinusz értéke 0-ban 1, a szinusz függvényé pedig 0, így a B_0 együttható mindenképp nulla lesz.

Speciális esetekben egyszerűsíthetjük az együtthatók számítását:

- Ha a függvény páros, akkor $B_i=0$
- Ha a függvény páratlan, akkor $A_0=0$ és $A_i=0$

Az ugyanahhoz a harmonikushoz tartozó tagok összevonásával egyetlen koszinuszos taggal is leírhatók a komponensek.

$$A_i \cos(i\omega t) + B_i \sin(i\omega t) = F_i \cos(i\omega t + \theta_i) = F_i \cos(i\omega t) \cos \theta_i - F_i \sin(i\omega t) \sin \theta_i \quad (5.5)$$

Az együtthatók segítségével megkaphatjuk az $f(t, T)$ periodikus jel amplitúdó- és fázisspektrumát. Fourier-sor esetén a spektrum diszkrét, a vonalak közti távolság pedig megegyezik az ω értékével.

A spektrális komponenseket az együtthatókból a következő módon tudjuk meghatározni:

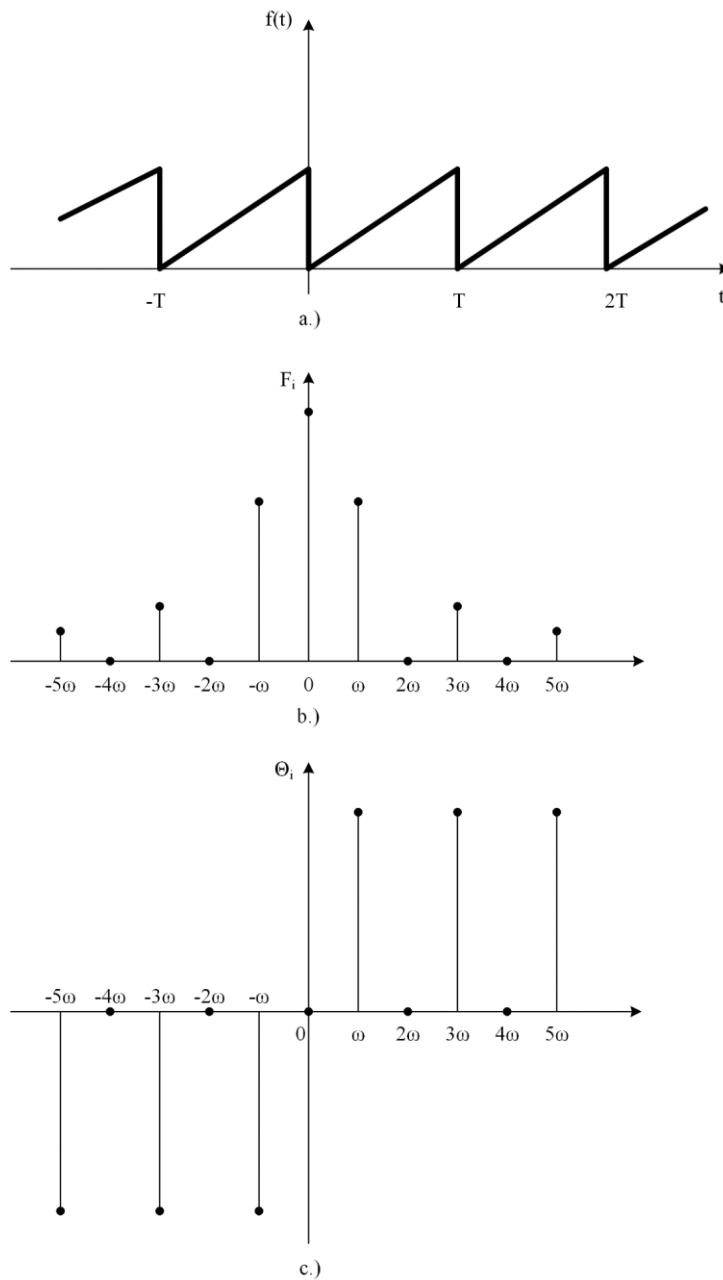
$$A_i = F_i \cos \theta_i, B_i = F_i \sin \theta_i \quad (5.6)$$

$$F_i = \sqrt{A_i^2 + B_i^2} \quad (5.7)$$

$$\operatorname{tg} \theta_i = -\frac{B_i}{A_i} \quad (5.8)$$

A sorba fejtett periodikus jel az amplitúdó és fázis komponensek segítségével a következő módon írható fel:

$$f(t, T) = F_0 + \sum_{i=1}^{\infty} F_i \cos(i\omega t + \theta_i) \quad (5.9)$$



5.1 ábra: Példa periodikus jel spektrumára: a) fűrészel, b) a jel amplitúdóspektruma, c) a jel fázisspektruma.

Az amplitúdóspektrum páros, a fázisspektrum pedig páratlan függvény.

Komplex alak

A következő helyettesítéseket alkalmazva megkaphatjuk a Fourier-sor komplex alakját:

$$\cos(i\omega t) = \frac{e^{ji\omega t} + e^{-ji\omega t}}{2} \quad (5.10)$$

$$\sin(i\omega t) = \frac{e^{ji\omega t} - e^{-ji\omega t}}{2j} \quad (5.11)$$

Behelyettesítés és átalakítás után:

$$f(t, T) = \sum_{i=-\infty}^{+\infty} \frac{A_i - jB_i}{2} e^{ji\omega t} \quad (5.12)$$

A következő alak bevezetésével a komplex együtthatókra:

$$C_i = \frac{A_i - jB_i}{2} \quad (5.13)$$

a Fourier-sor komplex alakja a következő lesz:

$$f(t, T) = \sum_{i=-\infty}^{+\infty} C_i e^{ji\omega t}, \quad (5.14)$$

ahol:

$$C_i = \frac{1}{T} \int_{-\frac{T}{2}}^{\frac{T}{2}} f(t, T) e^{-ji\omega t} dt. \quad (5.15)$$

A C_i együtthatók segítségével is megkaphatjuk a jel amplitúdó- és fázisspektrumát.

5.2 Fourier transzformáció

A Fourier-sor kifejezését csak periodikus jelek esetén tudjuk alkalmazni. Aperiodikus jelek esetén feltételezzük, hogy az adott jel egy olyan periodikus jel amelynek periódusideje végtelen felé tart ($T \rightarrow \infty$).

Ebben az esetben a Fourier-integrált alkalmazzuk, hogy megkapjuk a jel komplex spektrumát ($F(j\omega)$), vagyis a Fourier transzformáltját:

$$F(j\omega) = \mathcal{F}\{f(t)\} = \int_{-\infty}^{+\infty} f(t) e^{-j\omega t} dt \quad (5.16)$$

Frekvenciatartományból időtartományba az inverz Fourier transzformáció segítségével tudunk visszatérni:

$$f(t) = \mathcal{F}^{-1}\{F(j\omega)\} = \frac{1}{2\pi} \int_{-\infty}^{+\infty} F(j\omega) e^{j\omega t} d\omega \quad (5.17)$$

Mivel a periódus végtelen felé tart, a körfrekvencia nulla felé fog tartani ($\omega \rightarrow 0$), így pedig a Fourier transzformációval kapott spektrum folytonos lesz.

Az $F(j\omega)$ általános esetben komplex szám, amelyet az $A(\omega)$ amplitúdóspektrum és $\Phi(\omega)$ fázisspektrum segítségével a következő módon írhatunk fel:

$$F(j\omega) = A(\omega)e^{j\Phi(\omega)} \quad (5.18)$$

$$A(\omega) = |F(j\omega)| \quad (5.19)$$

$$\Phi(\omega) = \arg F(j\omega) \quad (5.20)$$

Spektrum szélessége

A jel megfelelő átviteléhez fontos a szükséges spektrális tartomány meghatározása. A jel sáv szélessége az a frekvenciatartomány amelyben a spektrális komponensek energiája nagyobb egy adott értéknél. A sáv szélesség meghatározása több módszer alapján történhet, például:

Az energiaspektrumban adott energiaszint meghatározásával, amely alapján a határ azon a frekvencián helyezkedik el, ahol a nagyobb frekvenciájú komponensek már csak a meghatározott értéknél kisebb energiával rendelkeznek.

Energiahányad megadásával, amely szerint a jel spektrumának a határát úgy határozzuk meg, hogy a jel teljes energiájának megadott része (például 90%) a határfrekvencián belül helyezkedjen el.

5.2.1 A Fourier transzformáció tulajdonságai

Linearitás

$$\mathcal{F}\{Ax(t) + By(t)\} = AX(j\omega) + BY(j\omega) \quad (5.21)$$

Szimmetria

$$\mathcal{F}\{x^*(t)\} = X^*(-j\omega), \quad (5.22)$$

ahol a csillag komplex konjugáltat jelöl.

Időbeni eltolás

Az időtartományban végzett eltolás fáziseltolást eredményez frekvenciatartományban.

$$\mathcal{F}\{x(t - t_0)\} = e^{-j\omega t_0} X(j\omega) \quad (5.23)$$

Spektrális eltolás

$$\mathcal{F}\{e^{j\omega_0 t} x(t)\} = X(j(\omega - \omega_0)) \quad (5.24)$$

Skálázás időben

$$\mathcal{F}\{x(at)\} = \frac{1}{|a|} X\left(\frac{j\omega}{a}\right) \quad (5.25)$$

Konvolúció

Az időtartománybeli konvolúció frekvenciatartományban szorzást eredményez.

$$\mathcal{F}\{x(t) * y(t)\} = X(j\omega)Y(j\omega) \quad (5.26)$$

Szorzás

Az időtartománybeli szorzás művelet frekvenciatartománybeli megfelelője a konvolúció.

$$\mathcal{F}\{x(t)y(t)\} = \frac{1}{2\pi} X(j\omega) * Y(j\omega) \quad (5.27)$$

Parseval tétel

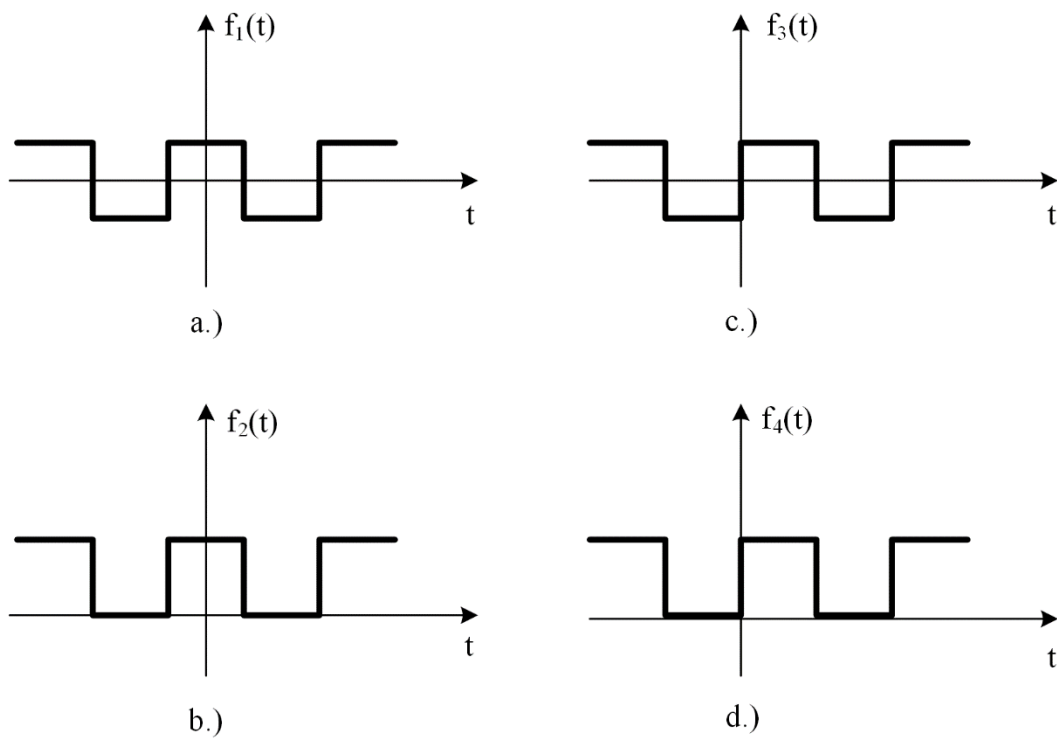
A Parseval tétel alapján a jel energiája nem csak energia per egységidő alapján határozható meg, hanem frekvenciatartományban energia per egységfrekvencia alapján is.

$$\int_{-\infty}^{\infty} |x(t)|^2 dt = \frac{1}{2\pi} \int_{-\infty}^{\infty} |X(j\omega)|^2 d\omega \quad (5.28)$$

5.3 Példák

5.1 Példa

Határozzuk meg az ábrán látható $f_1(t)$, $f_2(t)$, $f_3(t)$ és $f_4(t)$ periodikus jelek esetében, hogy milyen együtthatókból áll a Fourier-soruk.



Megoldás:

- a) Látható, hogy a függvény páros, így nem tartalmazhat B_i együtthatókat, csak A_i -ket. Egyenáramú komponense nincs, mivel egy periódusban az átlagértéke 0.
- b) A függvény páros, így A_i tagokat tartalmaz, és rendelkezik egyenáramú komponenssel is, mivel átlaga nem 0.
- c) A jel páratlan, így csak B_i koefficienseket tartalmazhat.
- d) Az ábrázolt négyszögjel se nem páratlan se nem páros, viszont látható, hogy egyenáramú komponens tartalmaz. Az egyenáramú komponens kivonásával visszakapjuk a c) ábrán látható függvényt, tehát, a Fourier-sor A_0 és B_i együtthatókat tartalmaz.

5.2 Példa

Határozzuk meg az előző példa d) ábráján látható $f_4(t)$ jel Fourier-sorát, ha a periódus hossza 2π , az amplitúdó pedig 1.

Megoldás:

$$A_0 = \frac{1}{2\pi} \int_{-\pi}^{\pi} f_4(t) dt = \frac{1}{2\pi} \left(\int_{-\pi}^0 (0) dt + \int_0^{\pi} (1) dt \right) = \frac{1}{2\pi} t \Big|_0^{\pi} = \frac{1}{2}$$

Felhasználva, hogy:

$$\omega = \frac{2\pi}{T} = 1$$

Az A_i és B_i együtthatókra a következőt kapjuk:

$$A_i = \frac{2}{2\pi} \int_{-\pi}^{\pi} f_4(t) \cos(it) dt = \frac{1}{\pi} \left(\int_{-\pi}^0 (0) \cos(it) dt + \int_0^{\pi} (1) \cos(it) dt \right) = \frac{1}{i\pi} \sin(it) \Big|_0^{\pi} = 0$$

$$B_i = \frac{2}{T} \int_{-\frac{T}{2}}^{\frac{T}{2}} f(t, T) \sin(it) dt = \frac{1}{\pi} \left(\int_{-\pi}^0 (0) \sin(it) dt + \int_0^{\pi} (1) \sin(it) dt \right) = \frac{1}{i\pi} (-\cos(it) \Big|_0^{\pi})$$

$$= \begin{cases} \frac{2}{i\pi}, & i \text{ páratlan} \\ 0, & i \text{ páros} \end{cases}$$

Tehát a Fourier-sor:

$$f_4(t) = \frac{1}{2} + \frac{2}{\pi} \left(\sin(\omega t) + \frac{1}{3} \sin(3\omega t) + \frac{1}{5} \sin(5\omega t) + \dots \right)$$

5.3 Példa

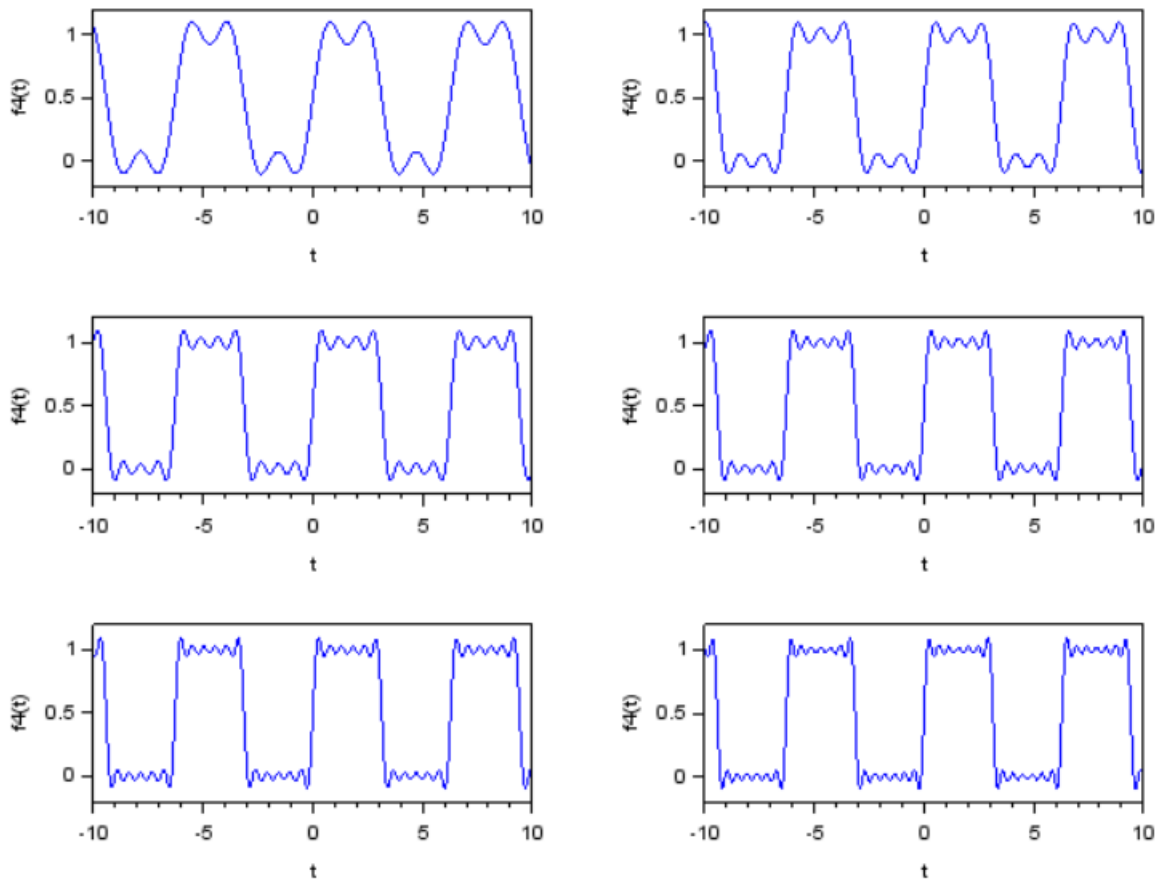
Scilab segítségével ábrázoljuk a jelet az előző példában kapott képlet segítségével különböző i értékek (3, 5, 7, 9, 11, 13) esetén.

Megoldás:

```
t=-10:0.001:10;
om=1;
figure
for j=1:6
    subplot(3,2,j)
    f=[];
    i=j*2+1;
    sin_tmp=zeros(size(t,2));
    for k=1:2:i
        sin_tmp=sin_tmp+sin(k*om*t)/k;
    end
    f=1/2+(2*sin_tmp)/%pi;
    plot(t,f)
```

```
xlabel("t");
ylabel("f4(t)");
end
```

Eredmény:



5.4 Példa

Scilab segítségével ábrázoljuk az előző példákban használt négyszögjel amplitúdó és fázisspektrumát.

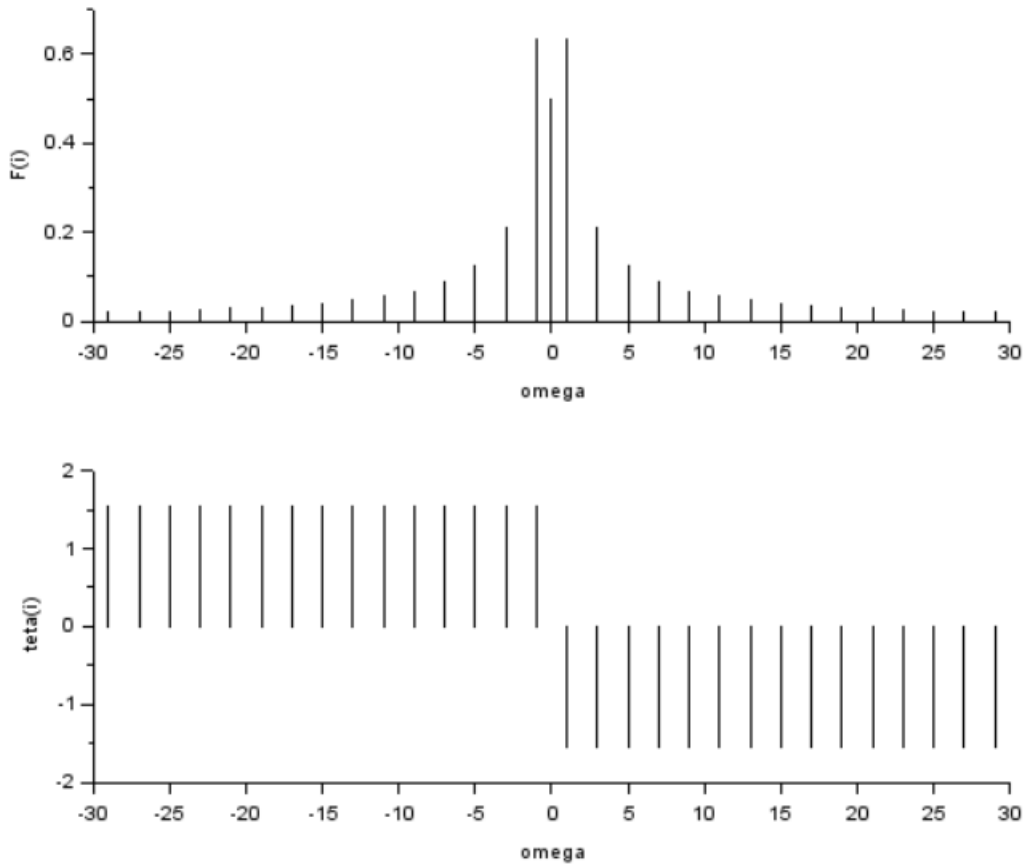
Megoldás:

```
n=-30:30;
Ai=zeros(size(n,2));
Bi=zeros(size(n,2));
Fi=zeros(size(n,2));
tetai=zeros(size(n,2));
for i=1:size(n,2)
    if n(i)==0
        Ai(i)=1/2;
        Bi(i)=0;
    elseif pmodulo(n(i),2)==1
        Ai(i)=0;
        Bi(i)=2/(n(i)*%pi);
    else
        Ai(i)=0;
    end
end
```



```
Bi(i)=0;
end
Fi(i)=sqrt(Ai(i)^2+Bi(i)^2);
tetai(i)=atan(-Bi(i)/Ai(i));
end
figure
subplot(2,1,1)
plot2d3(n,Fi)
xlabel("omega");
ylabel("F(i)");
subplot(2,1,2)
plot2d3(n,tetai)
xlabel("omega");
ylabel("teta(i)");
```

Eredmény:



5.5 Példa

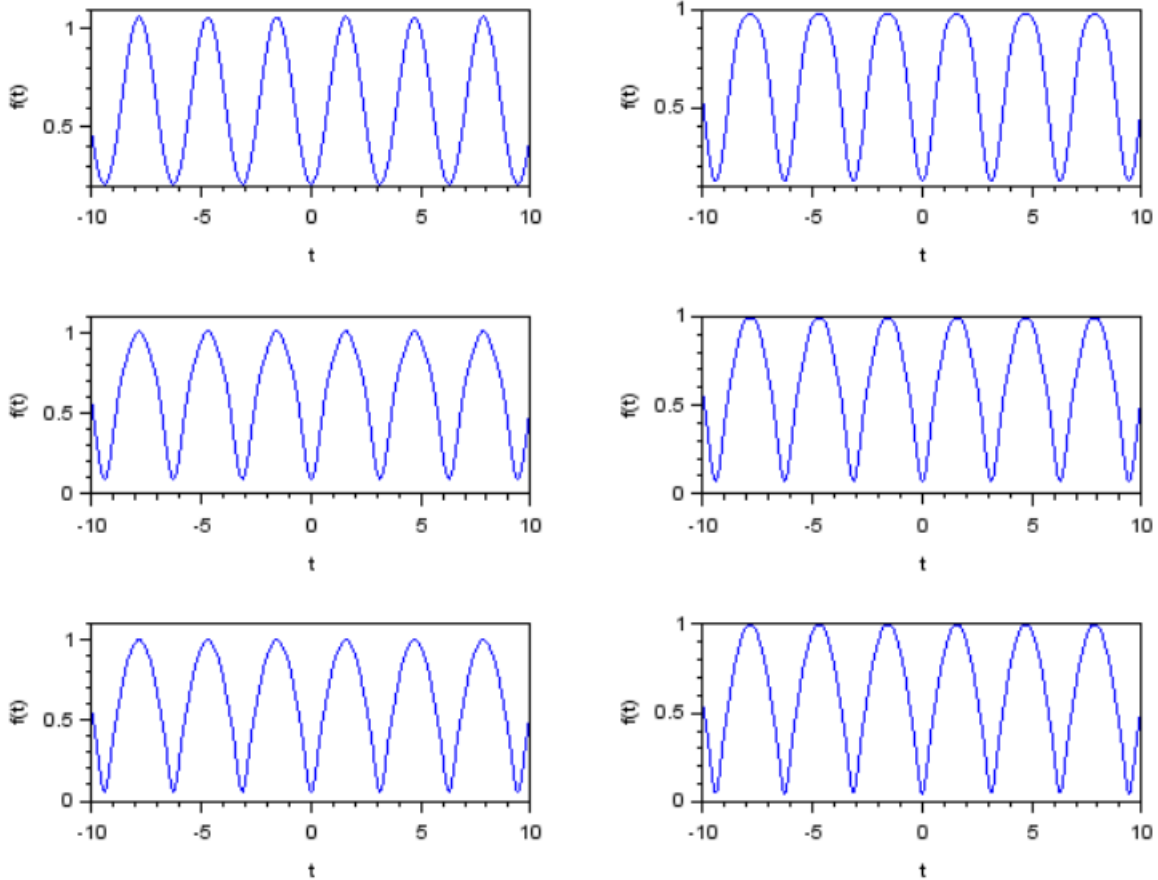
Scilab segítségével ábrázoljuk azt a periodikus jelet amelynek Fourier-sora a következő alakban adható meg:

$$f(t) = \frac{2}{\pi} - \frac{4}{\pi} \left(\frac{1}{1 \cdot 3} \cos(2\omega t) + \frac{1}{3 \cdot 5} \cos(4\omega t) + \frac{1}{5 \cdot 7} \cos(6\omega t) + \dots \right)$$

Megoldás:

```
t=-10:0.001:10;
om=1;
figure
for j=1:6
    subplot(3,2,j)
    f=[];
    i=j*2;
    cos_tmp=zeros(size(t,2));
    for k=2:2:i
        cos_tmp=cos_tmp+cos(k*om*t)/((k-1)*(k+1));
    end
    f=2/%pi-(4*cos_tmp)/%pi;
    plot(t,f)
    xlabel("t");
    ylabel("f(t)");
end
```

Eredmény:



Az eredményből látható, hogy a Fourier-sor a következő jelet határozza meg:

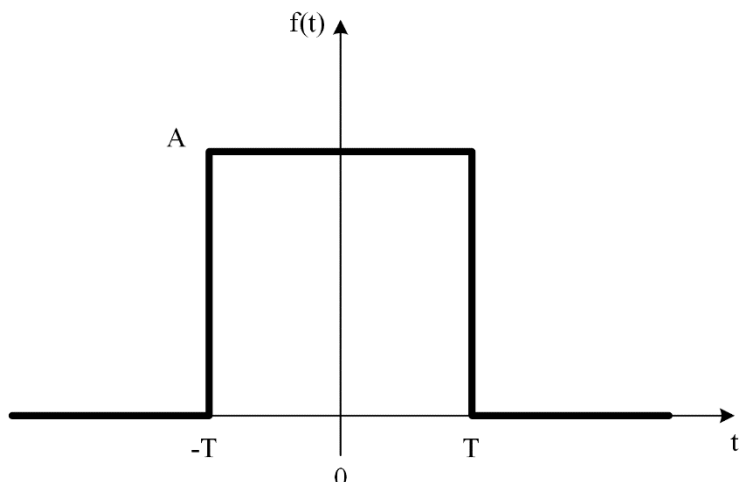
$$f(t) = |\sin(\omega t)|$$

Annak ellenére, hogy a jel a szinusz függvény abszolút értékét adja, mégis koszinusz jelekből állítható össze, mivel az abszolút érték hatására a függvény páros lesz.

Az ábrából szintén észrevehető, hogy az előző négyszögjelhez képest sokkal kevesebb tag elegendő, hogy jó közelítést kapjunk a jelre.

5.6 Példa

Határozzuk meg a következő jel Fourier transzformáltját.



Megoldás:

$$F(j\omega) = \int_{-\infty}^{+\infty} f(t)e^{-j\omega t} dt = \int_{-T}^T Ae^{-j\omega t} dt = -\frac{A}{j\omega} e^{-j\omega t} \Big|_{-T}^T = \frac{2A}{\omega} \frac{e^{j\omega T} - e^{-j\omega T}}{2j}$$

$$= 2AT \frac{\sin(\omega T)}{\omega T}$$

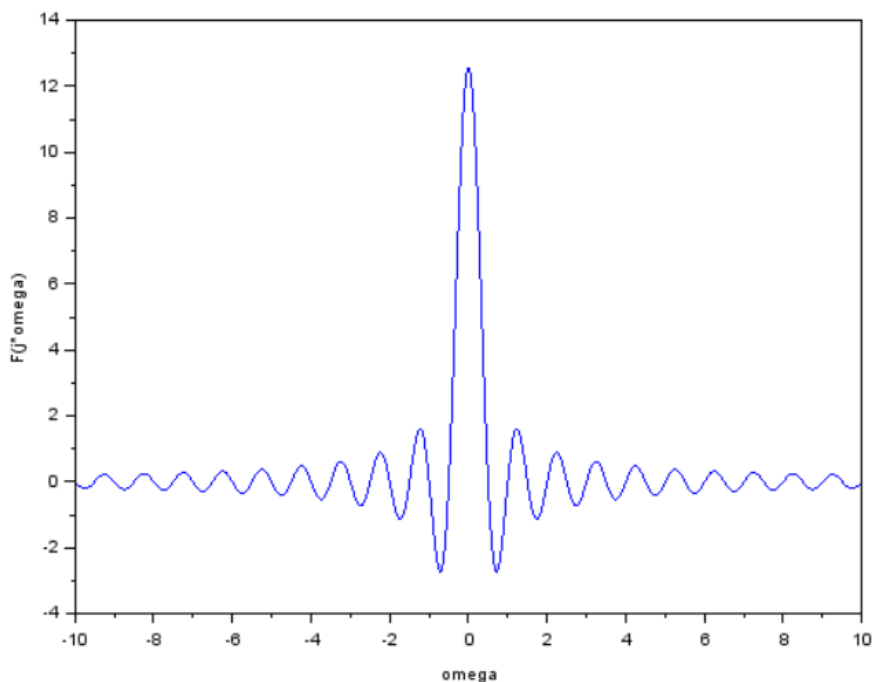
5.7 Példa

Ábrázoljuk Scilab segítségével az előző példában kapott Fourier transzformált amplitúdó spektrumát ha $A=1$ és $T=2\pi$.

Megoldás:

```
omega=-10:0.01:10
A=1;
T=2*pi;
F_jomega=[];
for i=1:size(omega,2)
    F_jomega(1,i)=2*A*T*sin(omega(i)*T)/(omega(i)*T);
end
plot(omega,F_jomega)
xlabel("omega");
ylabel("F(j*omega)");
```

Eredmény:



5.4 Ellenőrző kérdések

1. [Milyen jeleknek határozható meg a Fourier-sora?](#)
2. [Mely speciális esetekben egyszerűsíthető a Fourier-sor együtthatóinak számítása, és hogyan?](#)
3. [Milyen lesz a Fourier-sorba fejtett jel spektruma?](#)
4. [Mit feltételezünk a Fourier transzformáció esetén, hogy az aperiodikus jeleket transzformálni tudjuk?](#)
5. [Milyen lesz a Fourier transzformáció segítségével előállított spektrum?](#)

6 Diszkrét Fourier transzformáció (DFT) és gyors Fourier transzformáció (FFT)

6.1 Diszkrét Fourier transzformáció (DFT)

Az eddigi megoldások a spektrum meghatározására folytonos idejű jelek esetén voltak alkalmazhatók. Mintavételezett jelek esetén a diszkrét Fourier transzformációt (Discrete Fourier Transform – DFT) használjuk fel.

Diszkrét idejű jelek Fourier transzformáltja

A diszkrét idejű rendszerek frekvenciafüggvényei periodikusak 2π periódusidővel, így elegendő őket $0 \leq \omega \leq 2\pi$ vagy $-\pi \leq \omega \leq \pi$ tartományban vizsgálni.

Az $x[n]$ diszkrét idejű jelek Fourier transzformálva a következőt kapjuk:

$$X(e^{j\omega}) = \sum_{n=-\infty}^{\infty} x[n]e^{-j\omega n} \quad (6.1)$$

A kifejezésből látható, hogy $x[n]$ diszkrét idejű, transzformáltja pedig folytonos és periodikus 2π -vel. A folytonos és periodikus függvények Fourier-sorba fejthetők, az $X(e^{j\omega})$ sorbafejtésével pedig épp az $x[n]$ elemeket kapjuk. Tehát, az eredményezett kifejezés egyben a diszkrét idejű jelek inverz Fourier transzformációját adja:

$$x[n] = \frac{1}{2\pi} \int_{-\pi}^{\pi} X(e^{j\omega})e^{j\omega n} d\omega \quad (6.2)$$

Ez a kifejezés szerint az $x[n]$ végtelenül sok komplex amplitúdójú komplex szinusz jelből áll.

Mintavételezés frekvenciatartományban

A spektrum 2π hosszúságú alapperiódusát diszkrétizáljuk N egyenlő távolságú mintával. A minták távolsága így:

$$\Delta\omega = \frac{2\pi}{N} \quad (6.3)$$

Ezt behelyettesítve a következőt kapjuk:

$$X\left(\frac{2\pi k}{N}\right) = \sum_{n=-\infty}^{\infty} x[n]e^{-j\frac{2\pi}{N}kn}, k = 0, 1, 2, \dots, N-1 \quad (6.4)$$

Ezt felosztva végtelen számú N tagú parciális összegre, majd megfelelő helyettesítéssel a következőt kapjuk:

$$X\left(\frac{2\pi k}{N}\right) = \sum_{n=0}^{N-1} \left[\sum_{l=-\infty}^{\infty} x[n-lN] \right] e^{-j\frac{2\pi}{N}kn}, k = 0, 1, 2, \dots, N-1 \quad (6.5)$$

Az $x_p[n]=x[n-lN]$ az $x[n]$ idősor periodikus ismétlését adja, ahol N a periódus nagysága. Ennek Fourier-sorba fejtésével a következőt kapjuk:

$$x_p[n] = \sum_{l=-\infty}^{\infty} x[n-lN] = \sum_{n=0}^{N-1} c_k e^{j\frac{2\pi}{N}kn}, n = 0, 1, 2, \dots, N-1 \quad (6.6)$$

A c_k együtthatók a következő módon adhatók meg:

$$c_k = \frac{1}{N} \sum_{n=0}^{N-1} x_p[n] e^{-j\frac{2\pi}{N}kn} = \frac{1}{N} X\left(\frac{2\pi k}{N}\right), k = 0, 1, 2, \dots, N-1 \quad (6.7)$$

Ebből megkapjuk:

$$x_p[n] = \frac{1}{N} \sum_{k=0}^{N-1} X\left(\frac{2\pi k}{N}\right) e^{j\frac{2\pi}{N}kn}, n = 0, 1, 2, \dots, N-1 \quad (6.8)$$

Tehát, a mintavételezett spektrumból visszaállítható az $x_p[n]$ periodikus idősor, amely az $x[n]$ periodikus kiterjesztése.

Az időtartománybeli mintavételezésnél ügyelni kellett a spektrális átfedések elkerülésére, így a periodikus idősor létrehozásakor a spektrális tartományban történő mintavételezés esetén az időbeli átfedésekre kell ügyelni. Abban az esetben ha L , amely az $x[n]$ hossza, kisebb mint N , akkor átfedés nem fog jelentkezni. Ilyenkor a sort kiegészítjük $N-L$ számú nullával.

A frekvenciatartományban végzett mintavételezés már könnyebben használható alakot eredményez, mivel idő- és frekvenciatartományban is diszkrét és véges számú tagot tartalmaz. Az előzőek alapján a DFT a következő kifejezéssel határozható meg:

$$X[k] = \mathcal{DFT}\{x[n]\} = \sum_{n=0}^{N-1} x[n] e^{-j\frac{2\pi}{N}kn}, k = 0, 1, 2, \dots, N-1 \quad (6.9)$$

Az idősor visszaállítására az inverz diszkrét Fourier transzformációt (angulul Inverse Discrete Fourier Transform – IDFT) alkalmazzuk:

$$x[n] = \mathcal{IDFT}\{X[k]\} = \frac{1}{N} \sum_{k=0}^{N-1} X[k] e^{j\frac{2\pi}{N}kn}, n = 0, 1, 2, \dots, N-1 \quad (6.10)$$

6.1.1 Spektrum

Vegyünk a diszkrét időbeli jelből N pontot, majd ezen végezzük el a DFT műveletet. Az f_s mintavételezési frekvencia alapján tudjuk, hogy a mintavételi periódus:

$$T_s = \frac{1}{f_s} \quad (6.11)$$

Az eredeti jelből kivett idősor hossza ez alapján:

$$T = N \cdot T_s \quad (6.12)$$

A mintavételezés hatására a spektrum ismétlődni fog f_s periódussal. A DFT segítségével meghatározott spektrum határai $-f_s/2$ és $f_s/2$, amelyek között N pontot kapunk, tehát, a frekvenciatengelyen az egyes pontok közötti távolság N -pontos DFT esetén:

$$\Delta f = \frac{f_s}{N} \quad (6.13)$$

Mivel az eredményezett spektrum páros, és a negatív frekvenciaértékek nehezen értelmezhetők, így a spektrumot csak 0 és $f_s/2$ között szokás ábrázolni, és ebben a tartományban az értékeket megduplázzuk, mintha átlapoltuk volna a negatív tartományt a pozitívra, és összeadtuk volna az értékeket.

Az $X[0]$ az egyenáramú komponens jelöli, amely megadja a jel középértékét:

$$X[0] = \sum_{n=0}^{N-1} x[n] \quad (6.14)$$

6.2 Gyors Fourier transzformáció (FFT)

A DFT hatékony módszer a jelek spektrumának meghatározására, viszont nagy számú műveletet igényel, így meghatározása nagyon időigényes. Meghatározásához N -pontos DFT esetén N^2 komplex szorzás és $N(N-1)$ komplex összeadás szükséges.

A gyors Fourier transzformáció (angolul Fast Fourier Transform – FFT) jelentősége a DFT-hez képest a sokkal alacsonyabb műveletigénye. Az algoritmusokban a szorzás művelet jelenti a legjelentősebb lassító tényezőt. FFT esetén a szükséges szorzások száma jóval alacsonyabb, ami jelentős sebességnövekedést eredményez.

FFT abban a speciális esetben végezhető amikor $N=2^p$. Ha ez nem érvényes, akkor alkalmazhatjuk a következő kettő hatványt és a sort feltölthetjük nullákkal, ami nem fogja jelentősen módosítani a spektrumot. Például $N=1000$ esetén a következő kettő hatvány 1024, így 24 nullát adunk a jelhez.

Az algoritmus alapgondolata, hogy DFT esetén egy N hosszúságú jel transzformációja N^2 szorzást igényel, míg egy $N/2$ hosszúságú jel esetében ez $2(N/2)^2$. Ha a két részeredmény összekombinálható, akkor jelentős gyorsulás érhető el.

Ha az eredményezett két ágat további két-két ágra bontjuk, majd ezt addig folytatjuk, amíg egy ágban két elem marad, akkor elegendő kétpontos DFT műveletek végzése.

Az FFT algoritmus lényegében a DFT szimmetrikusságát és periodikusságát használja ki. Használjuk a DFT leírásánál a következő helyettesítést:

$$W_N = e^{-j\frac{2\pi}{N}} \quad (6.15)$$

Így a DFT kifejezés a következő:

$$X[k] = \sum_{n=0}^{N-1} x[n]W_N^{kn} \quad (6.16)$$

Bontsuk szét a szummát egy páros indexű tagokat és egy páratlan indexű tagokat tartalmazó összegre.

$$X[k] = \sum_{n=2m}^{N-1} x[n]W_N^{kn} + \sum_{n=2m+1}^{N-1} x[n]W_N^{kn} \quad (6.17)$$

A következő szimmetriákat használjuk fel:

$$W_N^{2ik} = W_{N/2}^{ik} \text{ és } W_N^{(2i+1)k} = W_N^k W_{N/2}^{ik} \quad (6.18)$$

Végezzük még el a következő helyettesítéseket: $x_{10}(m)=x(2m)$ és $x_{11}(m)=x(2m+1)$. Így a következő alakot kapjuk:

$$X[k] = \sum_{m=0}^{N/2-1} x_{10}[m]W_{N/2}^{km} + W_N^k \sum_{m=0}^{N/2-1} x_{11}[m]W_{N/2}^{km} \quad (6.19)$$

Mivel ez két $N/2$ hosszúságú DFT-nek felel meg, felírható a következő:

$$X[k] = X_{10}[k] + W_N^k X_{11}[k], k = 0, 1, 2, \dots, N/2 - 1 \quad (6.20)$$

Az $N/2$ -nél nagyobb indexű tagok felírásánál az $X_{10}[k]$ és $X_{11}[k]$ periodikusságát:

$$X_{10}[k + N/2] = X_{10}[k] \quad (6.21)$$

$$X_{11}[k + N/2] = X_{11}[k] \quad (6.22)$$

és hogy

$$W_N^{k+N/2} = -W_N^k \tag{6.23}$$

Így felírható:

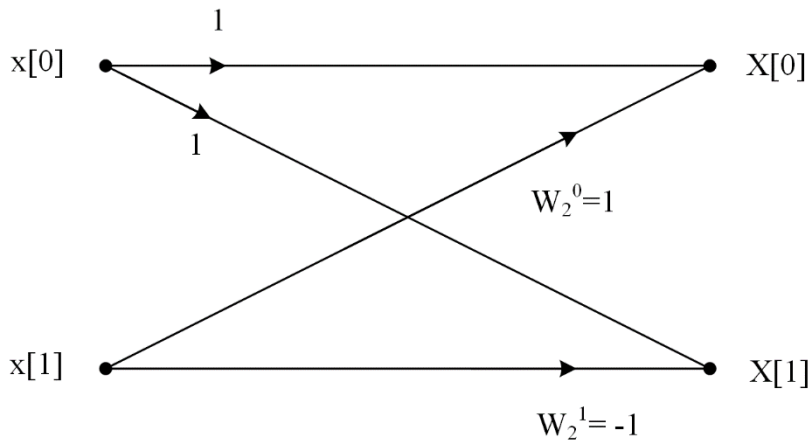
$$X[k + N/2] = X_{10}[k] - W_N^k X_{11}[k], k = 0, 1, 2, \dots, N/2 - 1 \tag{6.24}$$

A k és $k+N/2$ indexű tagokat tartalmazó DFT-eket összehasonlítva megállapítható, hogy csak a második összeadandó előjelében különböznek. Ezt az eljárást pillangóműveletnek vagy lepke-műveletnek (angolul butterfly) nevezzük.

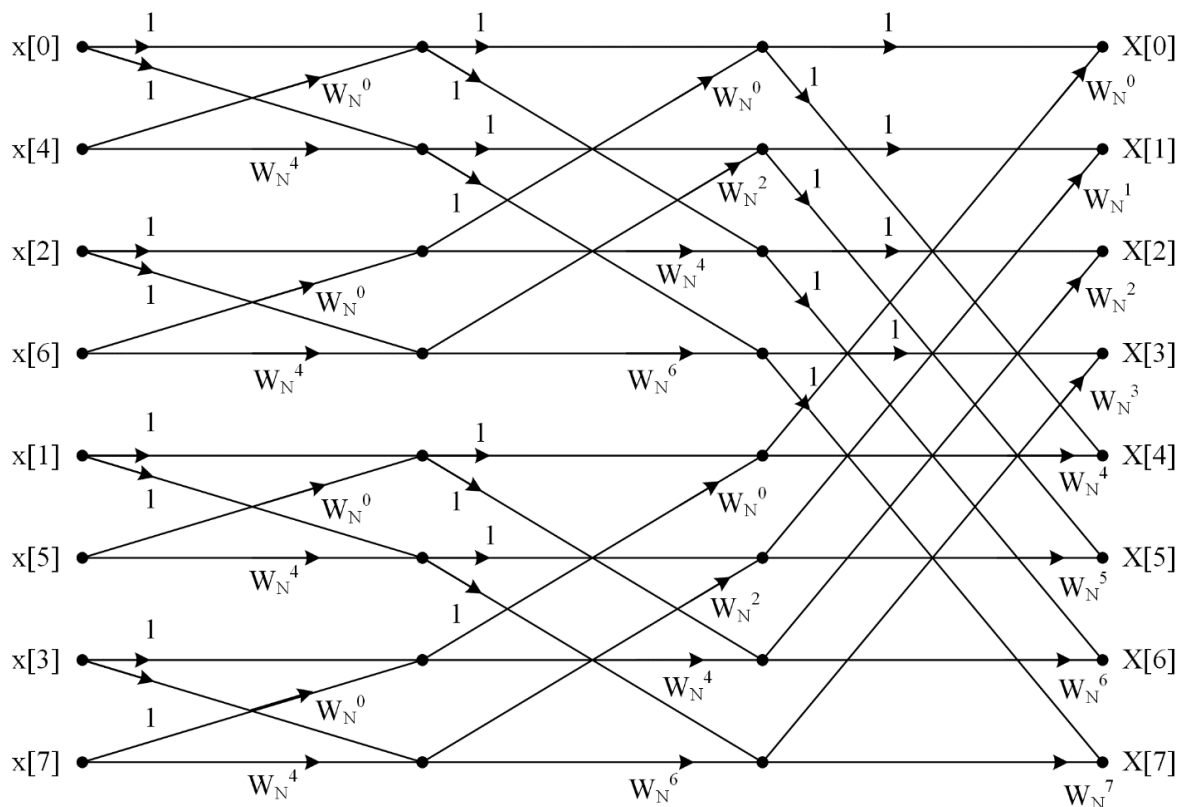
Kétpontos FFT-re, ahol az idősor elemei $x[0]$ és $x[1]$, a következőt kapjuk:

$$X[0] = x[0] + x[1] \tag{6.25}$$

$$X[1] = x[0] - x[1] \tag{6.26}$$



6.1 ábra: Kétpontos lepke-művelet.



6.2 ábra: Nyolcpontos FFT.

N számú pont esetén a teljes transzformáció elvégzéséhez $(N/2)\log_2 N$ lepkeművelet szükséges. Mivel a lepkeművelet egyetlen komplex szorzást alkalmaz, a teljes FFT művelethez $M\log_2 N$ szorzást kell elvégezni.

6.1 táblázat: Szükséges műveletek száma DFT és FFT esetén különböző N értékekre.

N	1000	10^6	10^9
N^2	10^6	10^{12}	10^{18}
$M\log_2 N$	10^4	$20 \cdot 10^6$	$30 \cdot 10^9$

A táblázatból látható, hogy $N=1024$ esetén körülbelül százszoros sebességnövekedés érhető el, míg az N növelésével egyre számottevőbb a különbség. Ha például a középső esetet vesszük, és behelyettesítünk a műveletidőre 1ns-ot, akkor a DFT-hez 1000 másodperc, azaz majdnem 17 perc szükséges, míg az FFT-hez 20ms.

Az FFT algoritmusnál az adatok sorrendbe rendezéséhez is szükséges gépidő, viszont erre léteznek gyors eljárások.

Számos megoldás létezik teljes FFT algoritmus megvalósítására.

6.3 Példák

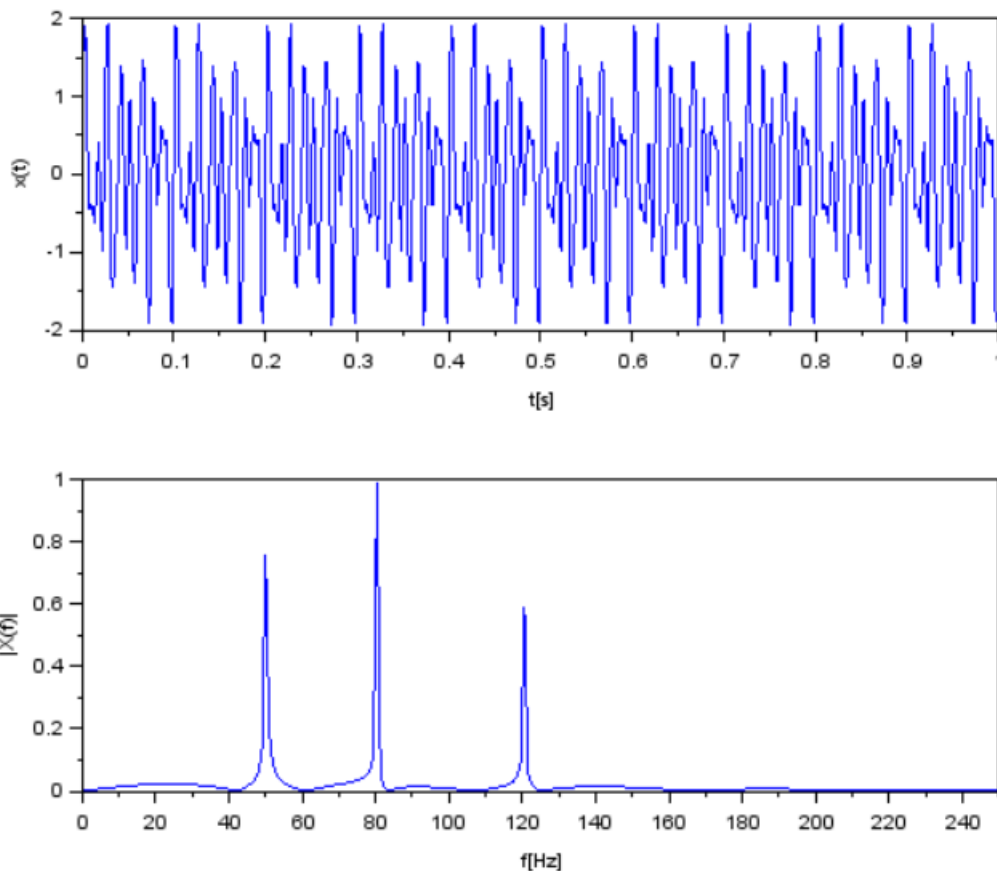
6.1 Példa

Scilab segítségével adjunk össze három különböző frekvenciájú és amplitúdójú szinusz jelet, majd ábrázoljuk az eredő jelet és spektrumát.

Megoldás:

```
Fs = 500;  
T = 1/Fs;  
L = 500;  
t = (0:L-1)*T;  
x = 0.8*sin(2*pi*50*t) + sin(2*pi*80*t) + 0.6*sin(2*pi*120*t);  
NFFT = 2^nextpow2(L);  
X = mtlb_fft(x,NFFT)/L;  
f = Fs/2*linspace(0,1,NFFT/2);  
subplot(2,1,1)  
plot(t,x)  
xlabel('t[s]')  
ylabel('x(t)')  
subplot(2,1,2)  
plot(f,2*abs(X(1:NFFT/2)))  
xlabel('f[Hz]')  
ylabel('|X(f)|')
```

Eredmény:



6.2 Példa

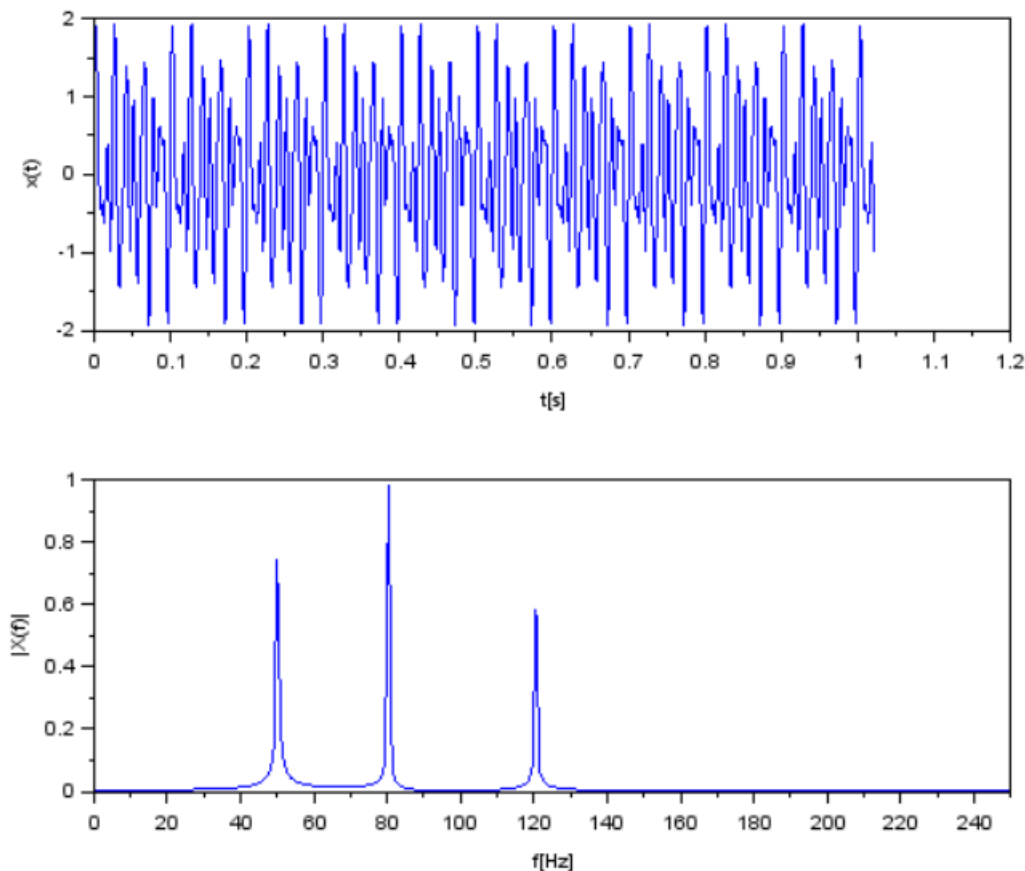
Az előző példához képest növeljük meg a FFT-ben felhasznált minták számát úgy, hogy ne legyen szükség nullákkal való kiegészítésre, majd ábrázoljuk a spektrumot.

Megoldás:

```
Fs = 500;  
T = 1/Fs;  
L = 512;  
t = (0:L-1)*T;  
x = 0.8*sin(2*pi*50*t) + sin(2*pi*80*t) + 0.6*sin(2*pi*120*t);  
NFFT = 2^nextpow2(L);  
X = mtlb_fft(x,NFFT)/L;  
f = Fs/2*linspace(0,1,NFFT/2);  
subplot(2,1,1)  
plot(t,x)  
xlabel('t[s]')  
ylabel('x(t)')  
subplot(2,1,2)  
plot(f,2*abs(X(1:NFFT/2)))  
xlabel('f[Hz]')  
ylabel('|X(f)|')
```



Eredmény:



Az ábrákat összehasonlítva látható, hogy a nullák hozzáadása kis hullámzást eredményez a komponensek között, de nincs számottevő hatása.

6.3 Példa

Adjunk zajt a 6.1 példában szereplő jelhez, majd határozzuk meg a spektrumát.

Megoldás:

```

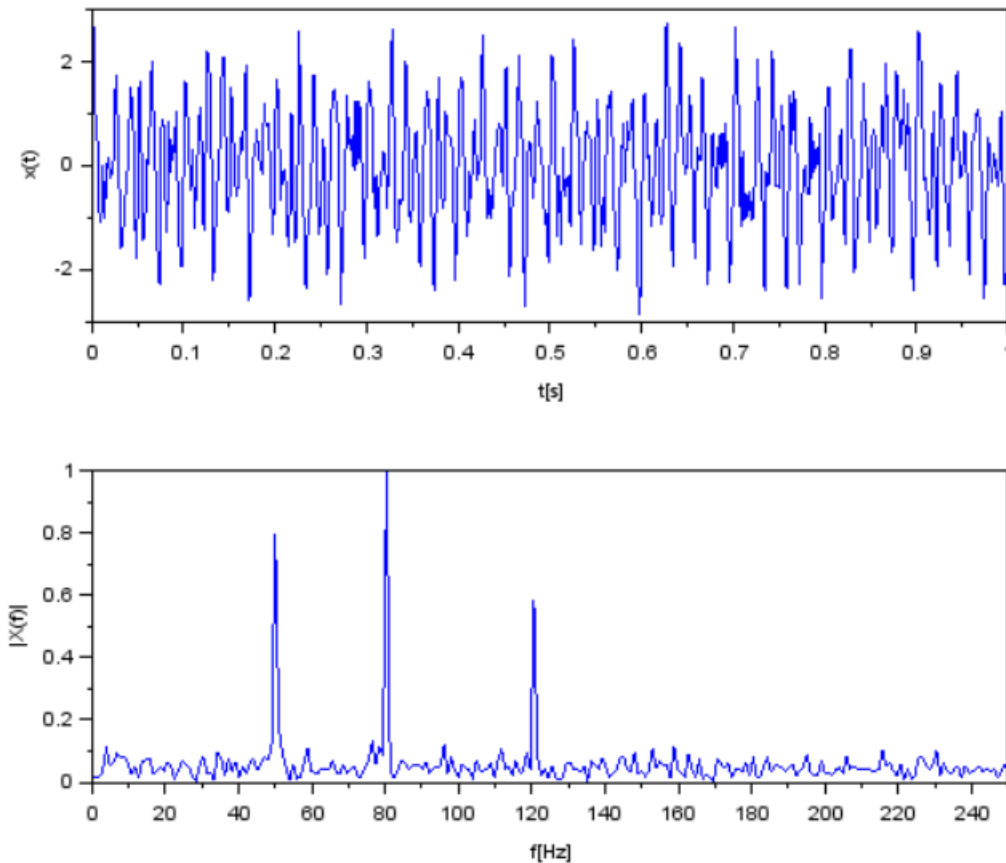
Fs = 500;
T = 1/Fs;
L = 500;
t = (0:L-1)*T;
x = 0.8*sin(2*pi*50*t) +
sin(2*pi*80*t) +
0.6*sin(2*pi*120*t) +
(2*rand(1:size(t,2),"uniform")-
1);
NFFT = 2^nextpow2(L);
X = mtlb_fft(x,NFFT)/L;
f = Fs/2*linspace(0,1,NFFT/2);
subplot(2,1,1)
plot(t,x)
xlabel('t[s]')
ylabel('x(t)')

```



```
subplot(2,1,2)  
plot(f,2*abs(X(1:NFFT/2)))  
xlabel('f[Hz]')  
ylabel('|X(f)|')
```

Eredmény:

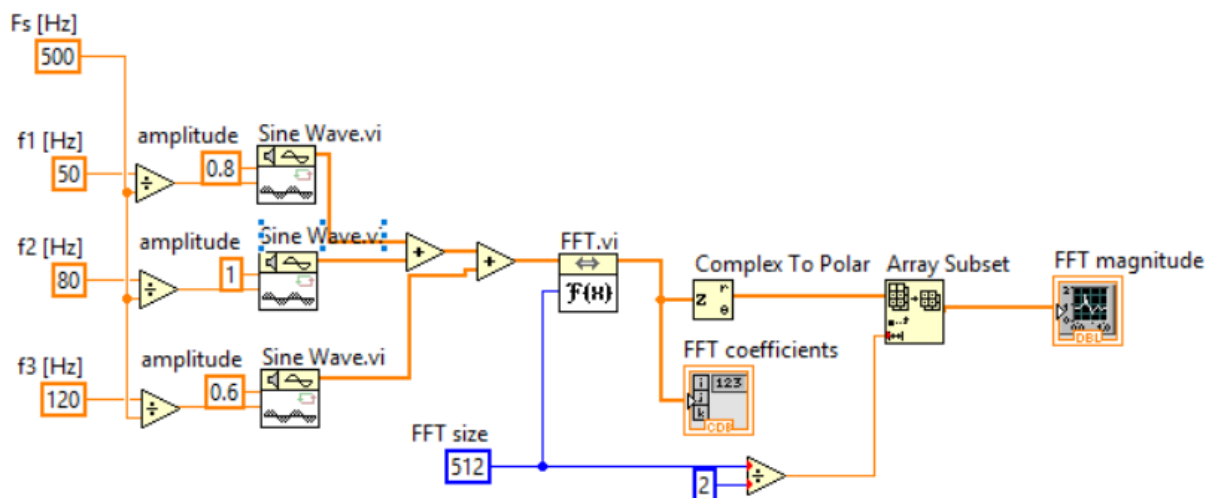


Látható, hogy a nagy amplitúdójú egyenletes zaj hatására felismerhetetlenné vált a jel, viszont a spektrumban könnyen meghatározhatóak a komponensek, mivel a zaj minden frekvencián jóval kisebb amplitúdóval rendelkezik.

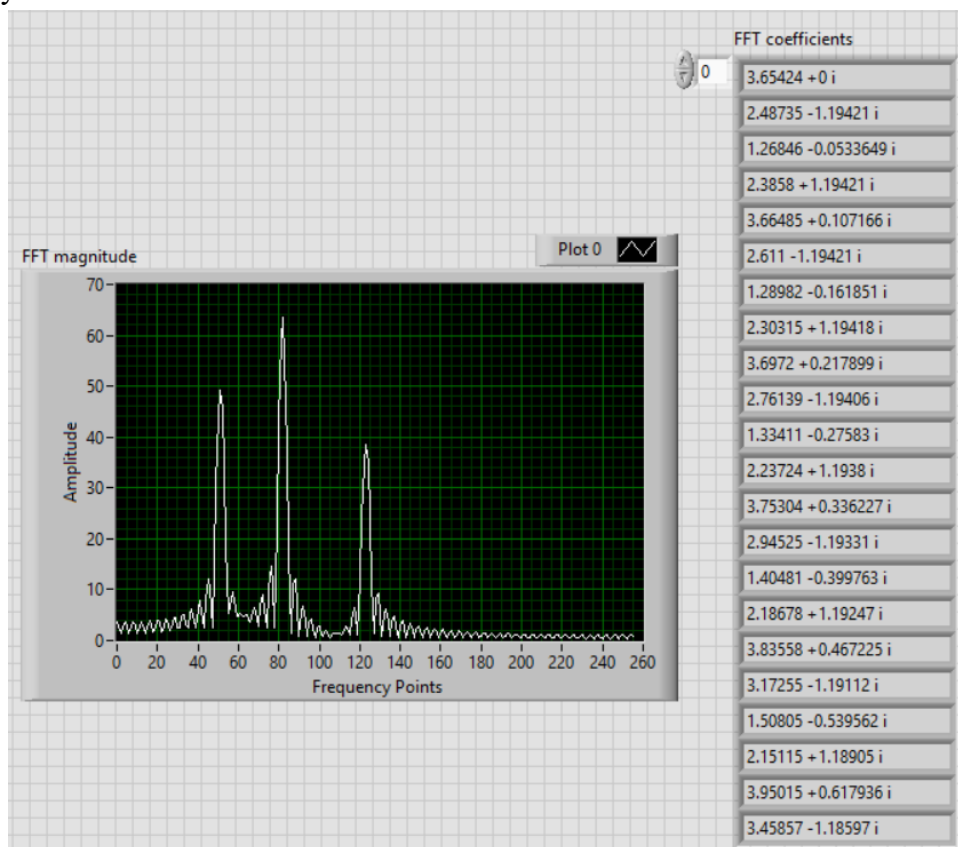
6.4 Példa

Állítsuk elő LabVIEW segítségével a 6.1 példában szereplő jelet, majd ábrázoljuk a spektrumát és az egyes komponensek komplex értékét.

Megoldás:



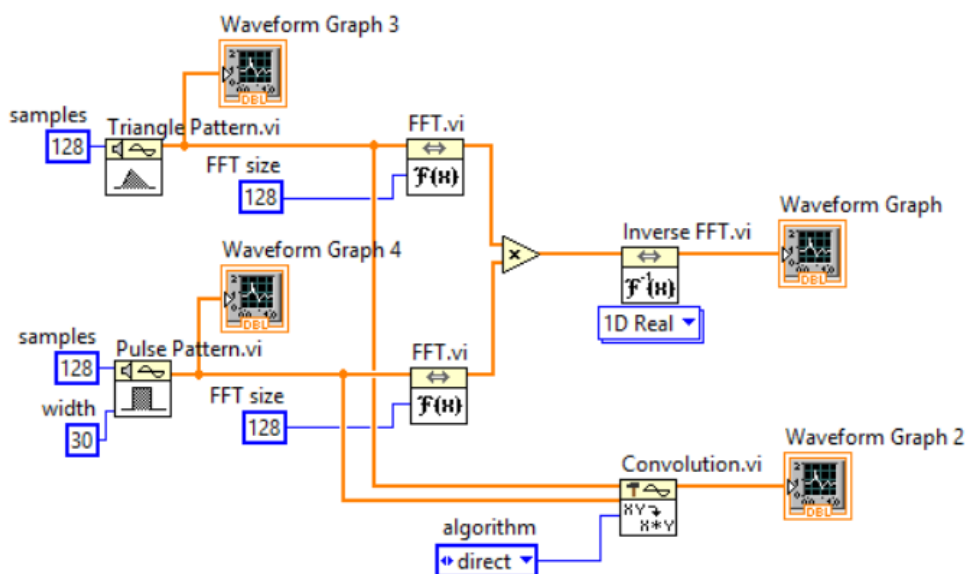
Eredmény:



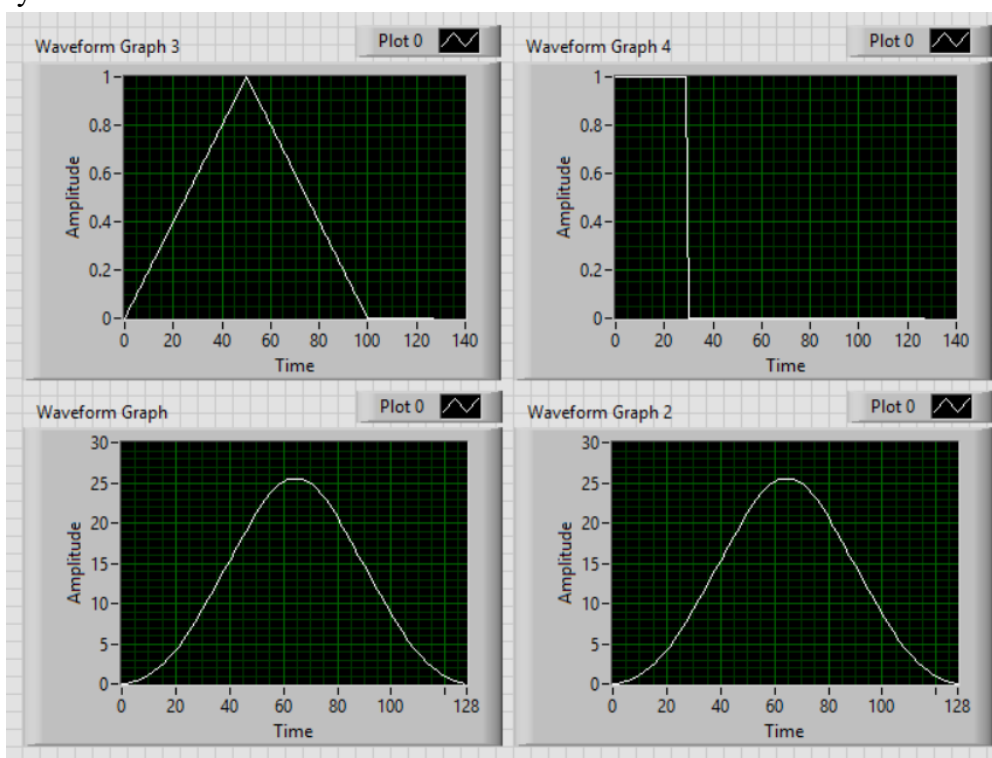
6.5 Példa

Igazoljuk LabVIEW segítségével, hogy két jel időbeli konvolúciója frekvenciatartománybeli szorzással is előállítható.

Megoldás:



Eredmény:



6.4 Ellenőrző kérdések

1. [Mire használható a DFT?](#)
2. [DFT esetén milyen tartományt határoz meg a spektrum és a frekvenciatengelyen mekkora a lépések nagysága?](#)
3. [A DFT milyen tulajdonságait használja ki az FFT algoritmus?](#)
4. [Milyen speciális esetben használható az FFT?](#)
5. [Hogyan határozhatóak meg a kétpontos lepkeművelet kimenetei?](#)



7 LTI rendszerek és vizsgálatuk

Azokat a rendszereket, amelyek kielégítik a lineáritás feltételét és időinvariánsak, LTI rendszereknek (Linear Time-Invariant systems) nevezzük.

Az olyan rendszerek esetén amelyekben minden jel folytonos a megoldás differenciálegyenletekre és differenciálegyenlet rendszerekre vezethető vissza. Egy folytonos idejű LTI rendszer a következő differenciálegyenlettel írható le:

$$a_n \frac{d^n y(t)}{dt^n} + a_{n-1} \frac{d^{n-1} y(t)}{dt^{n-1}} + \dots + a_0 y(t) = b_m \frac{d^m u(t)}{dt^m} + b_{m-1} \frac{d^{m-1} u(t)}{dt^{m-1}} + \dots + b_0 u(t) \quad (7.1)$$

amely megadható a következő formában is:

$$\sum_{i=0}^N a_i \frac{d^i y(t)}{dt^i} = \sum_{j=0}^M b_j \frac{d^j u(t)}{dt^j} \quad (7.2)$$

Ha legalább egy jel diszkrét akkor differenciaegyenleteket és differenciaegyenlet rendszereket alkalmazunk. Az LTI rendszerek időtartományban a következő differenciaegyenlettel írhatóak le:

$$y[n] = \sum_{k=0}^M b_k u[n-k] - \sum_{k=1}^N a_k y[n-k] \quad (7.3)$$

A foksámot a rendszer energiatárolóinak száma határozza meg, és $M \leq N$. Villamos rendszerek esetén az energiatárolók a kondenzátor és az induktivitás.

7.1 Súlyfüggvény és átmeneti függvény

7.1.1 Súlyfüggvény

A rendszer egységimpulzusra adott válaszát súlyfüggvénynek nevezzük, melyet folytonos időben $h(t)$ -vel, diszkrét időben pedig $h[n]$ -nel jelölünk. Egy LTI rendszert teljes mértékben meghatározza impulzusválasza.

Az LTI rendszerek fontos tulajdonsága, hogy a rendszer kimenete megadható az $u(t)$ bemenő jel és a $h(t)$ konvolúciójával. Ez érvényes diszkrét időben is $u[n]$ és $h[n]$ jelek esetén.

$$y(t) = u(t) * h(t) \quad (7.4)$$

$$y[n] = u[n] * h[n] \quad (7.5)$$

7.1.2 Átmeneti függvény

A rendszert jellemezhetjük az egységugrás jelre adott válasszal is, melyet átmeneti függvénynek nevezünk, jelölése pedig $s(t)$ és $s[n]$.

Az átmeneti függvény a (7.4) és (7.5) alapján meghatározható a következő módon:

$$s(t) = 1(t) * h(t) \quad (7.6)$$

$$s[n] = 1[n] * h[n] \quad (7.7)$$

Folytonos esetben érvényes, hogy

$$h(t) = \frac{ds(t)}{dt}, \quad (7.8)$$

diszkrét esetben pedig

$$h[n] = s[n] - s[n - 1], \quad (7.9)$$

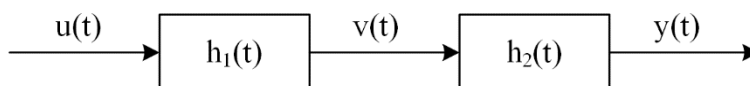
így a rendszer kimenete a súlyfüggvénnyel a következő kifejezésekkel adható meg:

$$y(t) = u(t) * \frac{ds(t)}{dt}, \quad (7.10)$$

$$y[n] = u[n] * (s[n] - s[n - 1]). \quad (7.11)$$

7.2 LTI rendszerek soros és párhuzamos kapcsolása

7.2.1 Soros kapcsolás



7.1 ábra: LTI rendszerek soros kapcsolása.

Az ábrán látható LTI rendszerek kaszkád kapcsolása esetén az $y(t)$ kimenet a következő módon határozható meg:

$$v(t) = u(t) * h_1(t), y(t) = v(t) * h_2(t) \quad (7.12)$$

$$y(t) = [u(t) * h_1(t)] * h_2(t) \quad (7.13)$$

A kommutativitás alapján a rendszerek sorrendjét felcserélve is azonos kimenetet kapunk:

$$y(t) = [u(t) * h_2(t)] * h_1(t) \quad (7.14)$$

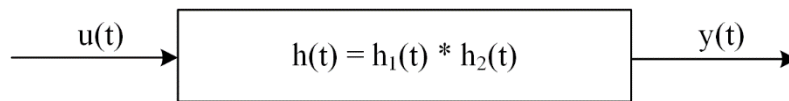
A konvolúció asszociatív tulajdonságát felhasználva:

$$y(t) = u(t) * [h_1(t) * h_2(t)] \quad (7.15)$$

$$h(t) = h_1(t) * h_2(t) \quad (7.16)$$

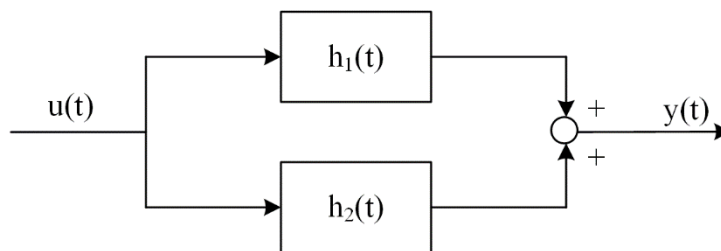
$$y(t) = u(t) * h(t) \quad (7.17)$$

Tehát, LTI rendszerek soros kapcsolása esetén alkalmazható egy eredő súlyfüggvény, amely a két rendszer súlyfüggvényének konvolúciójával határozható meg.



7.2 ábra: Eredő súlyfüggvény kaszkád kapcsolás esetén.

7.2.2 Párhuzamos kapcsolás



7.3 ábra: Párhuzamosan kapcsolt LTI rendszerek.

Párhuzamosan kapcsolt LTI rendszerek esetén a kimenet a következő kifejezéssel adható meg:

$$y(t) = [u(t) * h_1(t)] + [x(t) * h_2(t)] \quad (7.18)$$

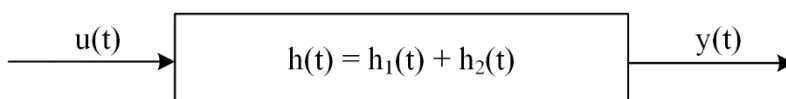
A konvolúció disztributív tulajdonságát felhasználva ez felírható mint:

$$y(t) = u(t) * [h_1(t) + h_2(t)] \quad (7.19)$$

$$h(t) = h_1(t) + h_2(t) \quad (7.20)$$

$$y(t) = u(t) * h(t) \quad (7.21)$$

Ezek alapján megállapítható, hogy párhuzamosan kötött rendszerek esetén is használható ekvivalens súlyfüggvény, viszont ebben az esetben a két súlyfüggvény összegét kell alkalmazni.



7.4 ábra: Eredő súlyfüggvény párhuzamos kapcsolás esetén.

7.3 Időálló és sajátfrekvencia

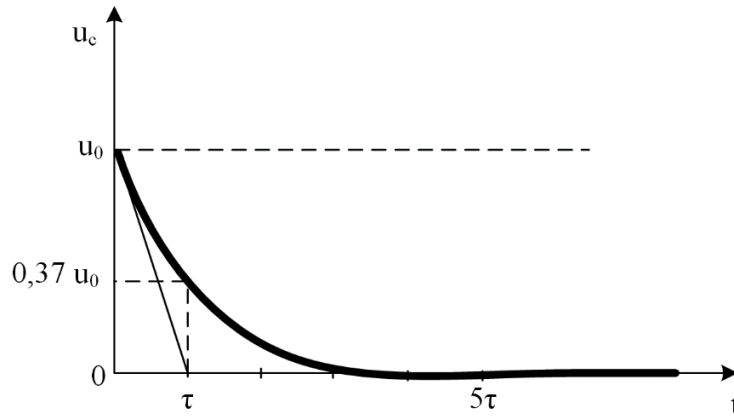
A rendszer homogén egyenletének (gerjesztés nélküli) megoldása sok információt szolgáltat a rendszer viselkedéséről. A karakterisztikus polinom gyökei a rendszer sajátértékeit képezik.

A λ_i sajátértékek vagy valós értékek vagy komplex konjugált párokat alkotnak.

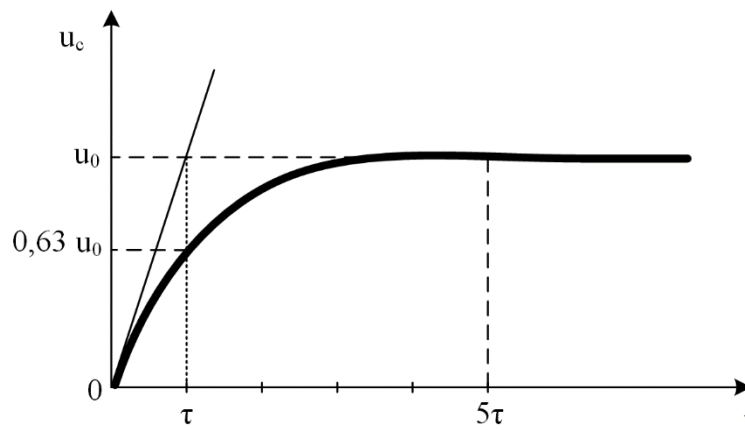
$$\lambda_i = -\sigma \pm j\omega \quad (7.22)$$

A sajátérték valós része (σ) adja a csillapítási tényezőt, melynek reciproka határozza meg az időállandót (τ). Pozitív σ és τ értékek esetén az eredmény nullához tart, és minél kisebb az időállandó értéke (minél nagyobb a csillapítási tényező), annál gyorsabb lesz a lecsengés.

Egytárolós esetben pontosan megállapítható, hogy egy magára hagyott rendszerben egy időállandónyi idő után az érték pontosan 63,71%-al kisebb lesz a kezdeti értékhez képest, öt időállandó után pedig 99,33%-kal. Például egy kondenzátor kisülésekor a feszültség értéke az említett módon változik. Ez a jelenség fordított irányban is érvényes, azaz a kondenzátor töltésekor is az időállandó határozza meg a feszültség értékének növekedését.



7.2 ábra: Kondenzátor kisülésének folyamata az időállandó alapján.



7.3 ábra: Kondenzátor töltődésének folyamata az időállandó alapján.

A sajátérték képzetes része adja a rendszer saját körfrekvenciáját (ω), a sajátfrekvencia pedig meghatározható mint:

$$f = \frac{\omega}{2\pi} \quad (7.23)$$

Ha a sajátérték csak képzetes részt tartalmaz, akkor az csillapítatlan rezgést jelent, míg ha van valós és képzetes rész is, akkor csillapodó rezgést kapunk.

Abban az esetben ha a rendszert a sajátfrekvenciával megegyező frekvenciájú jellel gerjesztjük akkor egy speciális jelenség jelentkezik, amelyet rezonanciának nevezünk. Ilyenkor a gerjesztés által bevitt energia összegződik, és a rezgés amplitúdója egyre nagyobb lesz.

7.4 Stabilitás

A stabilitás a rendszerek nagyon fontos tulajdonsága. Az LTI rendszer általánosan akkor mondható stabilnak, ha korlátos bemenő jelre korlátos kimenettel válaszol.

Folytonos és diszkrét esetben is a bemenő jel akkor lesz korlátos, ha létezik egy olyan B_u valós pozitív érték, amelyre minden időpont esetén teljesül, hogy:

$$|u[n]| \leq B_u < \infty \quad (7.24)$$

$$|u(t)| \leq B_u < \infty \quad (7.25)$$

A rendszer kimente így folytonos és diszkrét esetben akkor lesz korlátos, ha:

$$|y(t)| = \left| \int_{-\infty}^{\infty} h(\tau) u(t - \tau) d\tau \right| \leq \int_{-\infty}^{\infty} |h(\tau)| |u(t - \tau)| d\tau \leq B_u \int_{-\infty}^{\infty} |h(\tau)| d\tau < \infty \quad (7.26)$$

$$|y[n]| = \left| \sum_{k=-\infty}^{\infty} h[k] u[n - k] \right| \leq \sum_{k=-\infty}^{\infty} |h[k]| |u[n - k]| \leq B_u \sum_{k=-\infty}^{\infty} |h[k]| < \infty \quad (7.27)$$

Vagyis, korlátos bemenet esetén akkor lesz a kimenet is korlátos ha:

$$\int_{-\infty}^{\infty} |h(\tau)| d\tau < \infty \quad (7.28)$$

$$\sum_{k=-\infty}^{\infty} |h[k]| < \infty \quad (7.29)$$

7.5 Laplace transzformáció

LTI rendszerek analízisének és szintézisének fontos szerepe van a transzformációs eljárásoknak. A jelek és rendszerek sok tulajdonságát könnyebben lehet analizálni, illetve sok művelet sokkal egyszerűbbé válik a transzformált tartományban.

A Laplace transzformáció folytonos idejű jelek és rendszerek vizsgálatára szolgál. Segítségével lehetséges az átmeneti jelenségek analizálása, és LTI rendszerek analízise ha a gerjesztés nem szinuszos.

Egy folytonos idejű $x(t)$ jel Laplace transzformáltja a következő módon adható meg:

$$X(s) = \mathcal{L}\{x(t)\} = \int_{-\infty}^{\infty} x(t) e^{-st} dt \quad (7.30)$$

Ezt a formát bilaterális Laplace transzformációnak nevezzük.

Laplace transzformáció során ún. operátortartományba transzformálunk, ahol az operátor:

$$s = \sigma \pm j\omega \quad (7.31)$$

Abban az esetben ha $\sigma=0$, megkapjuk Fourier transzformáltat, tehát a Fourier transzformáció a Laplace transzformáció speciális esete.

Az időtartományba való visszatranszformáláshoz az inverz Laplace transzformációt alkalmazzuk, amely a következő kifejezéssel valósítható meg:

$$x(t) = \mathcal{L}^{-1}\{X(s)\} = \frac{1}{2\pi j} \int_{\sigma-j\omega}^{\sigma+j\omega} X(\sigma + j\omega) e^{st} ds \quad (7.32)$$

Az $f(t)$ függvényt $1(t)$ egységugrással szorozva megkapjuk az unilaterális Laplace transzformációt:

$$F(s) = \mathcal{L}\{f(t)\} = \int_{-\infty}^{\infty} 1(t) \cdot f(t) e^{-st} dt = \int_0^{\infty} f(t) e^{-st} dt \quad (7.33)$$

Az egységugrással való szorzás hatására az $F(s)$ nem függ az $f(t)$ értékeitől negatív t értékek esetén.

7.5.1 Laplace transzformáció tulajdonságai

Linearitás

$$\mathcal{L}\{ax_1(t) + bx_2(t)\} = a\mathcal{L}\{x_1(t)\} + b\mathcal{L}\{x_2(t)\} = aX_1(s) + bX_2(s) \quad (7.34)$$

Eltolás

$$\mathcal{L}\{x(t - t_0)\} = e^{-st_0} \mathcal{L}\{x(t)\} = e^{-st_0} X(s) \quad (7.35)$$

A független változó skálázása

$$\mathcal{L}\{ax(t)\} = \frac{1}{|a|} X\left(\frac{s}{a}\right) \quad (7.36)$$

Differenciálás

$$\mathcal{L}\left\{\frac{df(t)}{dt}\right\} = sF(s) \quad (7.37)$$

Ez alapján a deriváló tagokban gyakran találunk s jelölést.

Integrálás

$$\mathcal{L}\left\{\int x(t) dt\right\} = \frac{1}{s}X(s) \quad (7.38)$$

Az integráló tagokat gyakran szokták $1/s$ alakban jelölni.

Konvolúció

$$\mathcal{L}\{x(t) * y(t)\} = X(s) \cdot Y(s) \quad (7.39)$$

Kezdőérték egyoldalas esetben

$$\lim_{t \rightarrow 0} x(t) = x(0) = \lim_{s \rightarrow \infty} sX(s) \quad (7.40)$$

Végérték egyoldalas esetben

$$\lim_{t \rightarrow \infty} x(t) = x(\infty) = \lim_{s \rightarrow 0} sX(s) \quad (7.41)$$

7.6 z-transzformáció

A diszkrét időtartományban is lehetséges a diszkrét jelek és rendszerek leírására transzformációt végezni a Laplace transzformáció általánosításával. Ezt a transzformációt z-transzformációnak nevezzük.

A diszkrét jelet Laplace transzformálva a következőt kapjuk:

$$X^*(s) = \mathcal{L}\{x^*(t)\} = x(0) + x(T_s)e^{-T_s s} + x(2T_s)e^{-2T_s s} + \dots = \sum_{n=0}^{\infty} x(nT_s)e^{-nT_s s} \quad (7.42)$$

Bevezetve a

$$z = e^{T_s s} \quad (7.43)$$

helyettesítést és az n értékét kiterjesztve $-\infty$ -ig, az $x[n]$ diszkrét idősor z-transzformáltját a következő módon definiáljuk:

$$X(z) = \mathcal{Z}\{x[n]\} = \sum_{n=-\infty}^{\infty} x[n]z^{-n} \quad (7.45)$$

Az egyoldalas z-transzformáció pedig a következő módon adható meg:

$$X(z) = \sum_{n=0}^{\infty} x[n]z^{-n} \quad (7.46)$$

Az inverz z-transzformáció segítségével visszakaphatjuk a diszkrét jel értékét adott időpontban a z-transzformált alakból.

$$x[n] = \mathcal{Z}^{-1}\{X(z)\} = \frac{1}{2\pi j} \oint X(z) z^{n-1} dz \quad (7.47)$$

7.6.1 A z-transzformáció tulajdonságai

Linearitás

$$\mathcal{Z}\{ax_1[n] + bx_2[n]\} = a\mathcal{Z}\{x_1[n]\} + b\mathcal{Z}\{x_2[n]\} = aX_1(z) + bX_2(z) \quad (7.48)$$

Késleltetés

$$\mathcal{Z}\{x[n - k]\} = z^{-k}X(z), \text{ ha } k > 0 \quad (7.49)$$

A késleltető tagokat így gyakran szokták z^{-1} jelöléssel megadni.

Idő előrehozás

$$\mathcal{Z}\{x[n + k]\} = z^k X(z) - z^k \sum_{n=0}^{k-1} x[n]z^{-n}, \text{ ha } k > 0 \quad (7.50)$$

Konvolúció

$$\mathcal{Z}\{x_1[n] * x_2[n]\} = X_1(z)X_2(z) \quad (7.51)$$

Kezdőérték egyoldalas esetben

$$\lim_{n \rightarrow 0} x[n] = x[0] = \lim_{z \rightarrow \infty} X(z) \quad (7.52)$$

Végérték egyoldalas esetben

$$\lim_{n \rightarrow \infty} x[n] = x[\infty] = \lim_{z \rightarrow 1} [(1 - z^{-1})X(z)] \quad (7.53)$$

7.7 Átviteli függvény

Az átviteli függvény a kimenő és bemenő jel Laplace transzformáltjának hányadosa zérus kezdeti feltételek mellett. Az átviteli függvény megegyezik a súlyfüggvény Laplace transzformáltjával.

$$H(s) = \frac{Y(s)}{U(s)} \quad (7.54)$$

A megfelelő polinomokat behelyettesítve:

$$H(s) = \frac{b_m s^m + b_{m-1} s^{m-1} + \dots + b_1 s + b_0}{a_n s^n + a_{n-1} s^{n-1} + \dots + a_1 s + a_0} \quad (7.55)$$

Ezt az alakot szokás megadni a következő formában is:

$$H(s) = \frac{N(s)}{D(s)} \quad (7.55)$$

ahol N az angol numerator kifejezésből származik, ami számlálót jelent, a D pedig a denominator szóból, ami nevezőt jelent.

A polinom alak mellett az átviteli függvény felírható zérus/pólus alakban is, ahol a zérusok a számláló gyökei, a pólusok pedig a nevezőé.

$$H(s) = k \frac{(s-z_1)(s-z_2)\dots(s-z_m)}{(s-p_1)(s-p_2)\dots(s-p_m)} = k \frac{\prod_{j=1}^m (s-z_j)}{\prod_{i=1}^m (s-p_i)}, k = \frac{b_m}{a_n} \quad (7.56)$$

A gyökök mindig vagy valós értékek vagy konjugált komplex párok.

7.7.1 Diszkrét idejű átviteli függvény

z -transzformált tartományban a következő átviteli függvényt kapjuk:

$$H(z) = \frac{Y(z)}{U(z)} = \frac{\sum_{k=0}^M b_k z^{-k}}{1 + \sum_{k=1}^N a_k z^{-k}} \quad (7.57)$$

Blokk diagram segítségével ezt ábrázolhatjuk összeadó, szorzó és késleltető elemek felhasználásával.

7.8 Frekvenciafüggvény

A bemenő és kimenő jelek Fourier transzformációjával a következőt kapjuk:

$$a_n(j\omega)^n Y(j\omega) + a_{n-1}(j\omega)^{n-1} Y(j\omega) + \dots + a_0 Y(j\omega) = b_m(j\omega)^m U(j\omega) + b_{m-1}(j\omega)^{m-1} U(j\omega) + \dots + b_0 U(j\omega) \quad (7.58)$$

A rendszer frekvenciafüggvénye meghatározható mint a kimenet és a bemenet Fourier transzformáltjainak hányadosa:

$$G(j\omega) = \frac{Y(j\omega)}{U(j\omega)} = \frac{b_m(j\omega)^m + b_{m-1}(j\omega)^{m-1} + \dots + b_0}{a_n(j\omega)^n + a_{n-1}(j\omega)^{n-1} + \dots + a_0} \quad (7.59)$$

A frekvenciafüggvény segítségével, amelynek meghatározásához a kimenő és bemenő jelek ismeretére van szükségünk, az időtartományba történő visszatérés nélkül következtethetünk a rendszer időbeli viselkedésére. A $G(j\omega)$ komplex kifejezés, amely, mint az egyenletből is látszódik, frekvenciafüggő abszolút értéke és fázisa is.

Egy LTI rendszer a bemenetére adott szinuszos ω körfrekvenciájú jelet csak amplitúdóban és fázisban módosítja. A frekvenciafüggvény megadja különböző frekvenciákon a kimenő és bemenő jel amplitúdóbeli viszonyát, illetve a kimenő jelnek a bemeneti jelhez viszonyított fáziseltolási szögét.

Egy adott ω_0 körfrekvencia esetén az $A(\omega_0)$ amplitúdó és $\varphi(\omega_0)$ fázis a következő egyenletekkel határozható meg:

$$A(\omega_0) = \sqrt{\operatorname{Re}(G(j\omega))^2 + \operatorname{Im}(G(j\omega))^2} \quad (7.60)$$

$$\varphi(\omega_0) = \arctan \frac{\operatorname{Im}(G(j\omega))}{\operatorname{Re}(G(j\omega))} \quad (7.61)$$

7.8.1 A frekvenciafüggvény ábrázolása

A frekvenciafüggvény ábrázolására két módszer terjedt el, a Bode diagram és a Nyquist diagram.

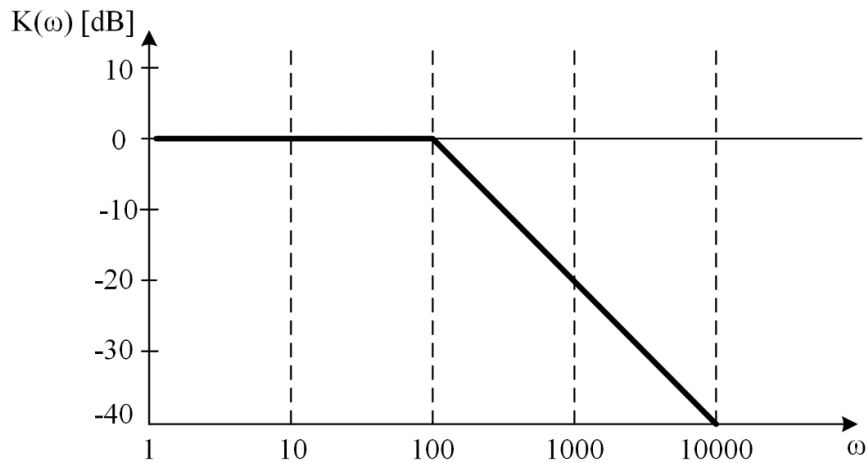
Bode diagram

A Bode diagram esetén külön ábrázoljuk az amplitúdó és fázis jelleggörbéket az ω körfrekvencia függvényében.

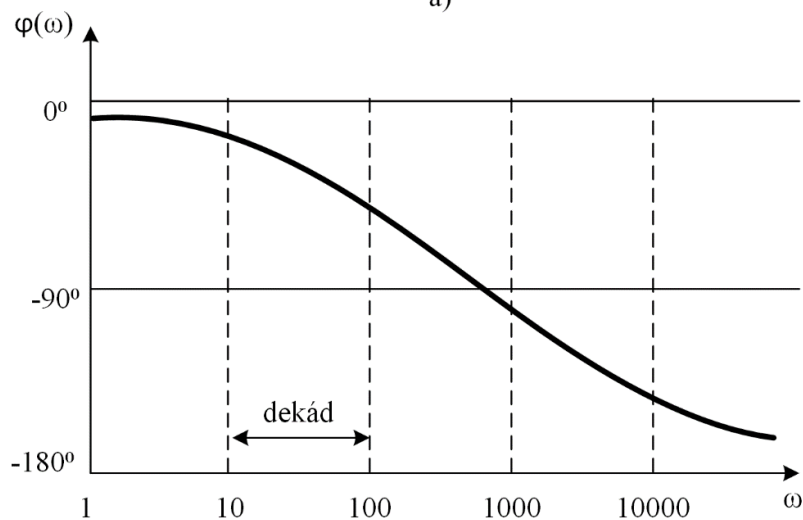
Az amplitúdóviszonyt decibelben (dB) szokás megadni, amely a következő módon határozható meg:

$$K(\omega) = 20 \log_{10} A(\omega)$$

A körfrekvenciákat célszerű logaritmikus léptékben ábrázolni a szélesebb frekvenciaátfogás miatt. Ilyen esetben alkalmazzuk a dekád egységet, amely 10:1 arányú frekvenciaváltozást jelent.



a)

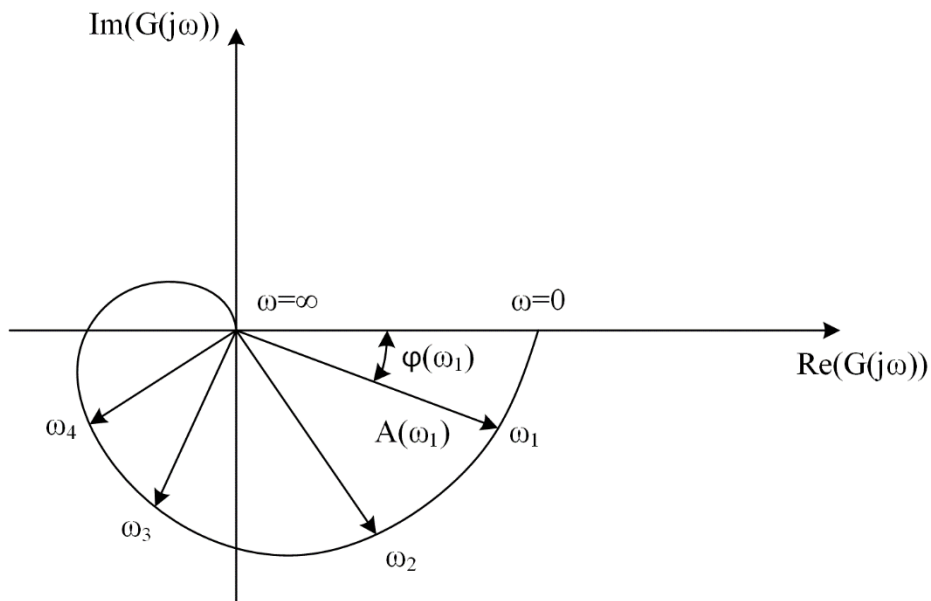


b)

7.4 ábra: Bode diagram: a) amplitúdó jelleggörbe, b) fázis jelleggörbe.

Nyquist diagram

Nyquist diagram esetében a frekvenciafüggvényt komplex síkban ábrázoljuk. Minden körfrekvenciára meghatározzuk a kimeneti jelnek a bemeneti jelhez viszonyított $\varphi(\omega)$ fáziseltolási szögét és $A(\omega)$ amplitúdóját, amelyek az egyes pontokat fogják képezni, ezeket összekötve pedig megkapjuk az amplitúdó-fázis jelleggörbét.



7.5 ábra: Nyquist diagram.

7.9 Példák

7.1 Példa

Határozzuk meg a következő függvény Laplace transzformáltját:

$$f_1(t) = 2 \cdot 1(t)$$

Megoldás:

$$F_1(s) = \int_{-\infty}^{\infty} f_1(t)e^{-st} dt = \int_0^{\infty} 2e^{-st} dt = 2 \frac{e^{-st}}{-s} \Big|_0^{\infty} = 0 - \frac{2}{-s} = \frac{2}{s}$$

7.2 Példa

Határozzuk meg a Dirac-impulzus z-transzformáltját.

$$f_2[n] = \delta[n]$$

Megoldás:

$$F_2(z) = \mathcal{Z}\{f_2[n]\} = f_2[0] + f_2[1]z^{-1} + f_2[2]z^{-2} + \dots = 1 + 0 + 0 + \dots = 1$$

7.3 Példa

Határozzuk meg a diszkrét egységugrás jel z-transzformáltját.

$$f_3[n] = h[n]$$

Megoldás:

$$\begin{aligned} F_3(z) = \mathcal{Z}\{f_3[n]\} &= f_3[0] + f_3[1]z^{-1} + f_3[2]z^{-2} + \dots = 1 + z^{-1} + z^{-2} + \dots = \frac{1}{1 - z^{-1}} \\ &= \frac{z}{z - 1} \end{aligned}$$

7.10 Ellenőrző kérdések

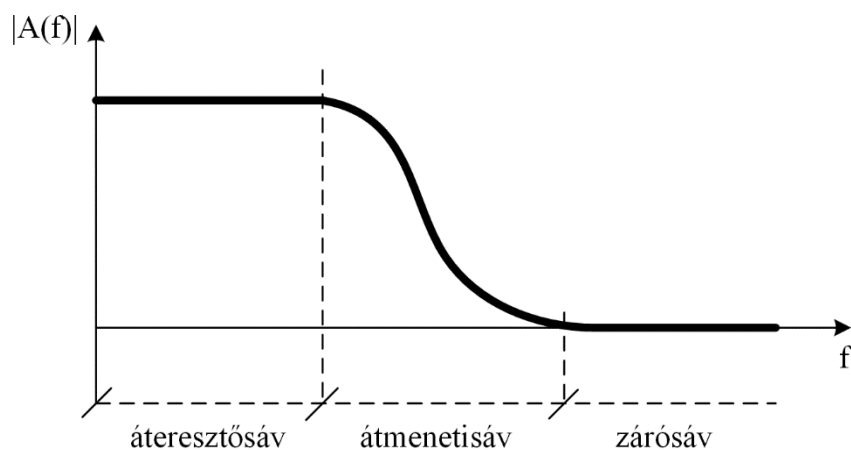
1. [Mi a súlyfüggvény és az átmeneti függvény között a különbség?](#)
2. [Milyen műveletekkel határozható meg az ekvivalens súlyfüggvény LTI rendszerek soros és párhuzamos kapcsolása esetén.](#)
3. [Mi állapítható meg az időállandó segítségével?](#)
4. [Hogyan határozható meg az átviteli függvény?](#)
5. [Hogyan határozható meg a frekvenciafüggvény?](#)

8 Analóg szűrők

A szűrők szerepe nagyon jelentős a jelfeldolgozásban, mivel számtalan helyen használják őket. A szűrők olyan eszközök, amelyek lineárisak, időinvariánsak és idő- vagy frekvenciatartományban megadott módon módosítják az adott jelet.

8.1 Szűrők sávjai

A szűrőknél két sáv különböztethető meg ideális esetben: áteresztősáv (angolul passband) és zárósáv (angolul stopband). Ideális esetben a szűrő az áteresztősávban az adott tartományban elhelyezkedő spektrális komponenseket torzítás nélkül átviszi, míg a zárósávban nem visz át, vagyis teljes mértékben csillapítja. Tehát, ebben az esetben az erősítés értéke áteresztősávban egy, zárósávban pedig nulla. Az átmenet a két sáv között ideális esetben végtelen keskeny. Mivel ilyen szűrő nem valósítható meg, szükséges egy átmenetisáv (angolul transition band) az áteresztősáv és zárósáv között.



8.1 ábra: A szűrők sávjai.

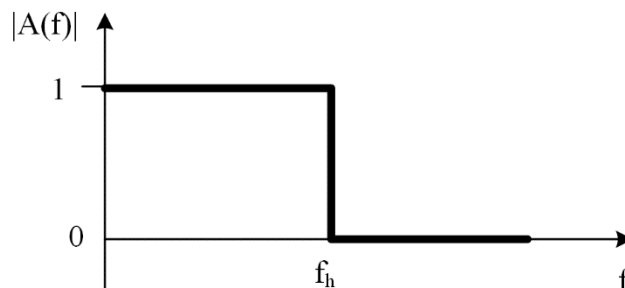
Az ideális szűrőknek hátrányai is vannak a nem ideális szűrőkhöz képest. Az ideális aluláteresztő szűrő átmeneti függvénye, azaz egységugrásra adott válasza, túllövést és fluktuációt tartalmaz, míg nem ideális esetben a beállítás exponenciális jellegű.

8.2 Szűrők típusai

A szűrők az áteresztősáv és a zárósáv elhelyezkedése alapján több típusra oszthatók.

8.2.1 Aluláteresztő szűrő

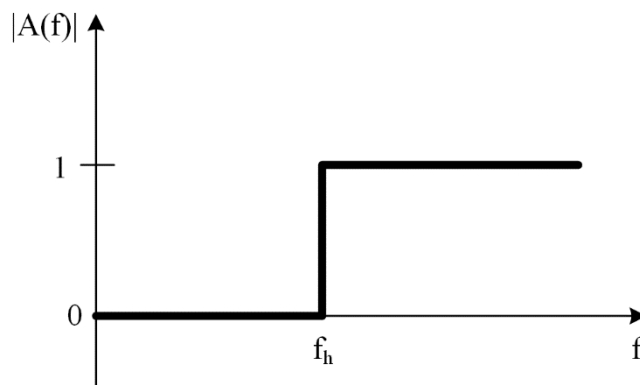
Egy aluláteresztő szűrő (angolul lowpass filter) egy adott f_h határfrekvenciánál alacsonyabb komponenseket átereszt, míg a határfrekvenciától a végtelenig zárósáv található. Ezt a határfrekvenciát szokás vágási frekvenciának (angolul cutoff frequency) is nevezni, és f_c -vel jelölni.



8.2 ábra: Ideális aluláteresztő szűrő.

8.2.2 Feluláteresztő szűrő

A feluláteresztő szűrő (angolul highpass filter) esetén egy adott f_h határfrekvenciától kisebb frekvenciáknál zárósáv, ez a frekvencia felett pedig áteresztősáv található. A feluláteresztő szűrő lényegében egy aluláteresztő inverzeként is felfogható.

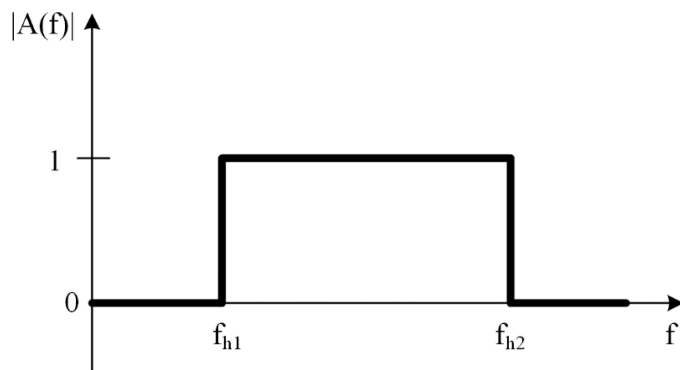


8.3 ábra: Ideális feluláteresztő szűrő.

8.2.3 Sáváteresztő szűrő

A sáváteresztő szűrő (angolul bandpass filter) két határfrekvencia között átenged, a kisebb f_{h1} határfrekvencia alatt és a nagyobb f_{h2} határfrekvencia felett pedig csillapít.

Sáváteresztő szűrő megvalósítható egy aluláteresztő és egy feluláteresztő szűrő soros kapcsolásával, amelyeknél az aluláteresztő határfrekvenciája f_{h2} , a feluláteresztőé pedig f_{h1} .

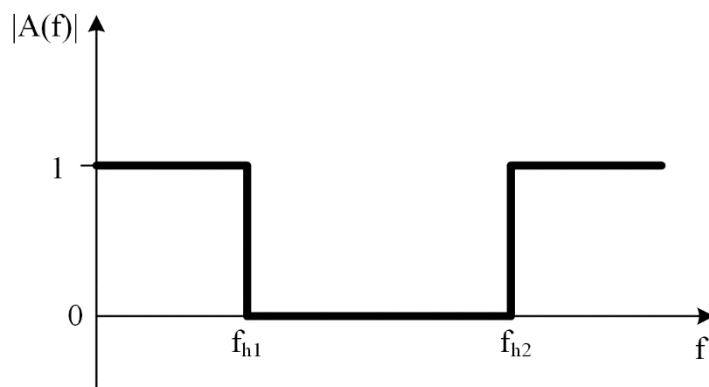


8.4 ábra: Ideális sáváteresztő szűrő.

8.2.4 Sávzáró szűrő

Sávzáró szűrő (angolul bandstop filter) esetén két határfrekvencia között zárósávunk van, az alacsonyabb f_{h1} határfrekvencia alatt és a nagyobb f_{h2} határfrekvencia felett pedig áteresztősávok találhatók.

Egy sávzáró szűrő elképzelhető egy aluláteresztő és egy feluláteresztő szűrő párhuzamos kapcsolásként, ahol a két szűrő kimenete összeadódik. Az aluláteresztő szűrő határfrekvenciája f_{h1} , míg a feluláteresztőé f_{h2} .



8.5 ábra: Ideális sávzáró szűrő.

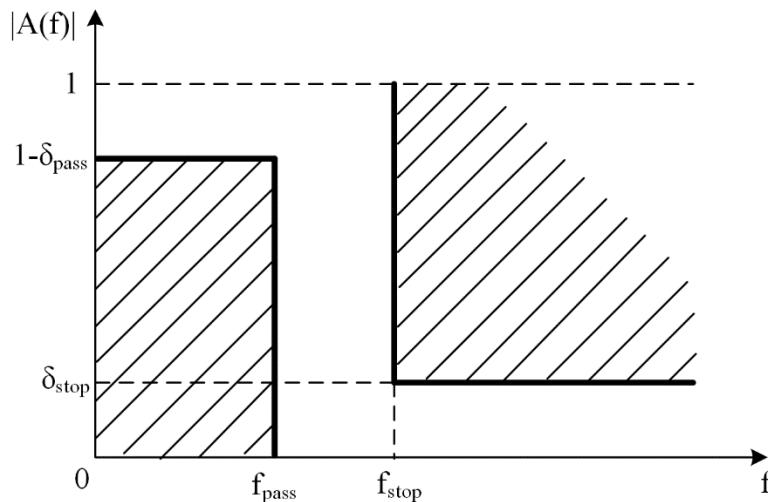
8.2.5 Mindentáteresztő szűrő

Ezek a szűrők nem változtatják meg az amplitúdó spektrumát a jelnek, viszont módosítják a fázisspektrumát.

8.3 Szűrők tervezése

Szűrők tervezése során egy ún. toleranciasémát alkalmazunk, amely meghatározza, hogy az átviteli függvény milyen alakot vehet fel. A toleranciasémát frekvenciatartományban adjuk meg.

Aluláteresztő szűrő esetén ez a toleranciaséma az 8.6 ábra szerint ábrázolható. Az amplitúdó esetében szokás dB-ben is az ábrázolás és a szintek megadása.



8.6 ábra: Aluláteresztő szűrő toleranciasémája.

A frekvenciatengelyen két frekvenciát szükséges megadni, f_{pass} és f_{stop} , amelyek az átmeneti sáv szélességét határozzák meg. Az f_{pass} (angolul passband edge) egyben az áteresztősáv végét és az átmeneti sáv kezdetét adja meg, míg az f_{stop} (angolul stopband edge) lesz az átmeneti sáv vége és a zárósáv kezdete. Fontos, hogy $f_{stop} > f_{pass}$, az átmenetisáv szélessége pedig így:

$$\Delta f_t = f_{stop} - f_{pass} \tag{8.1}$$

Szükséges még az amplitúdókaraktisztikában is két adat megadása. Az első paraméter δ_{pass} megadja az áteresztősávban megengedett maximális csillapítási értéket, amely egyben a fodrozódás legnagyobb értéke is (angolul passband ripple), a másik δ_{stop} pedig a zárósávban elvárt legalacsonyabb csillapítási szintet határozza meg (angolul stopband attenuation), de zárósávi fodrozódásnak is szokás nevezni (angolul stopband ripple).

8.4 Közelítési eljárások

Szűrőtervezés során a szűrő zérusait és pólusait szükséges meghatározni az adott követelmények szerint. A gyökök meghatározására több fajta approximációs módszer létezik, amelyek különböző tulajdonságokkal rendelkeznek. Ezek a módszerek különböző módon próbálják közelíteni az ideális átviteli karakterisztikát.

A függvénynek a lehető legkisebb foksámúnak kell lennie, és a toleranciaséma által meghatározott területen belül kell elhelyezkednie.

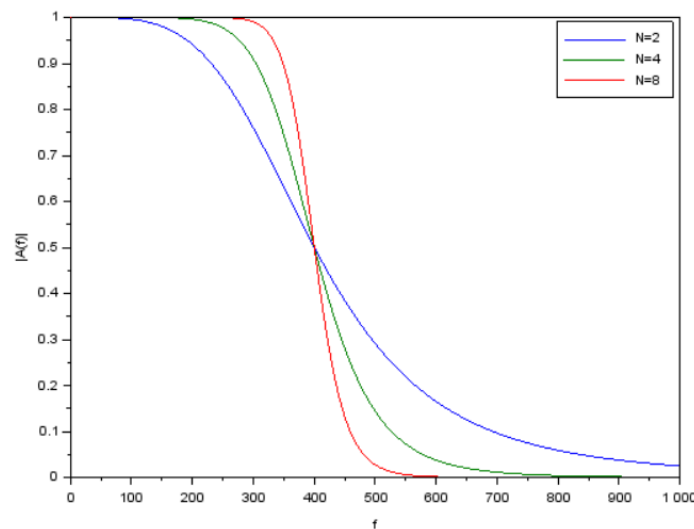
8.4.1 Legismertebb közelítési eljárások

Butterworth

A Butterworth-féle approximációs eljárásnak a legjellegzetesebb tulajdonsága, hogy maximálisan lapos függvényt eredményez az áteresztősávban és a zárósávban is.

Ez az approximáció Taylor-sorral közelíti az ideális aluláteresztő szűrő karakterisztikát az $\omega=0$ és $\omega=\infty$ körfrekvenciák esetén. Az átviteli karakterisztikája N -ed fokú szűrő esetén a következő:

$$|H(j\omega)|^2 = \frac{1}{1+\omega^{2N}} \quad (8.2)$$



8.7 ábra: Butterworth eljárással tervezett aluláteresztő szűrő különböző foksámok esetén.

Chebyshev

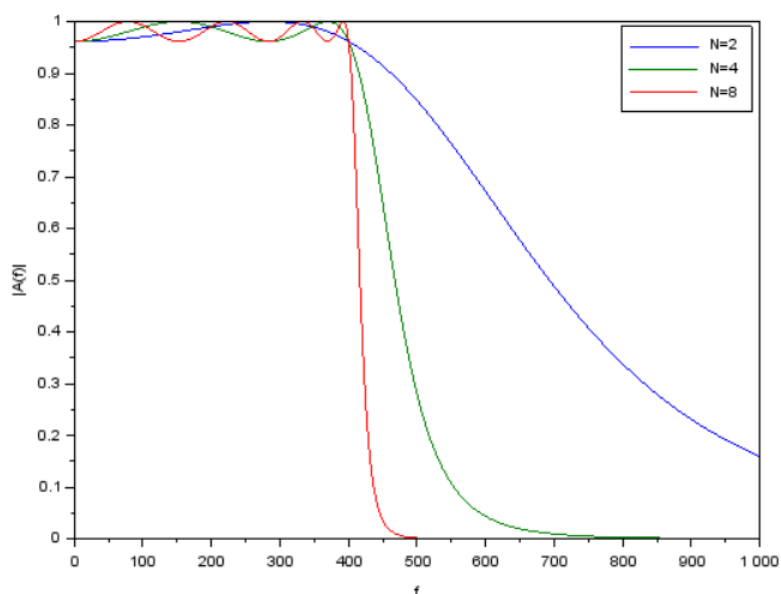
A Chebyshev-féle közelítés gyorsabb átmenetet biztosít az áteresztősáv és a zárósáv között, viszont az áteresztősávban hullámos karakterisztikát eredményez.

A Chebyshev approximáció $\omega=\infty$ -ben közelíti az ideális aluláteresztő karakterisztikát Taylor-sorral, viszont az áteresztősávban minimalizálja a hiba maximumát.

$$|H(j\omega)|^2 = \frac{1}{1+\varepsilon^2 C_N^2(\omega)} \quad (8.3)$$

ahol ε meghatározza az áteresztősávban az ingadozást, $C_N(\omega)$ pedig az N -ed rendű Chebyshev polinom, amely:

$$C_N(\omega) = \cos(N \arccos(\omega)) \quad (8.4)$$



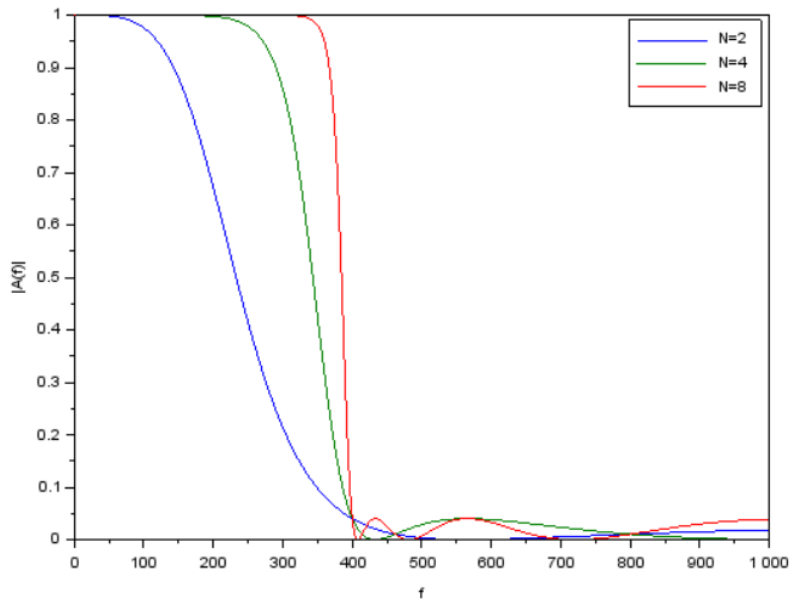
8.8 ábra: Chebyshev approximációval tervezett aluláteresztő szűrő különböző fokszámok esetén.

Inverz Chebyshev

Az inverz Chebyshev (Chebyshev II) módszer ugyanolyan átmenetet biztosít mint a Chebyshev-féle közelítés, de a hullámosság az áteresztősáv helyett a zárósávban jelentkezik.

Ennél az approximációnál felcseréljük a tartományokat, így felüláteresztő szűrőt kapunk. Átviteli karakterisztikája a következő:

$$|H(j\omega)|^2 = \frac{\varepsilon^2 C_N^2(1/\omega)}{1 + \varepsilon^2 C_N^2(1/\omega)} \quad (8.5)$$



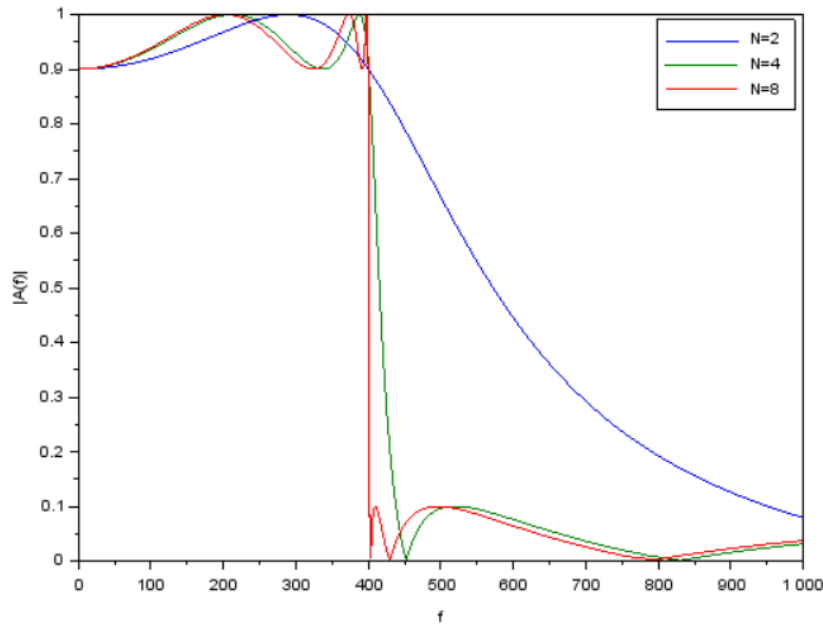
8.9 ábra: Inverz Chebyshev approximációval tervezett aluláteresztő szűrő különböző foksámok esetén.

Elliptikus

A Cauer féle elliptikus eljárás biztosítja a leggyorsabb átmenetet a zárósáv és az átteresztősáv között, viszont mindkét sávban hullámos közelítést ad.

$$|H(j\omega)|^2 = \frac{1}{1 + \varepsilon^2 G_N^2(\omega)}, \tag{8.6}$$

ahol $G(\omega)$ a Chebyshev polinom általánosítása, melynek meghatározásához elliptikus integrálokra van szükség.

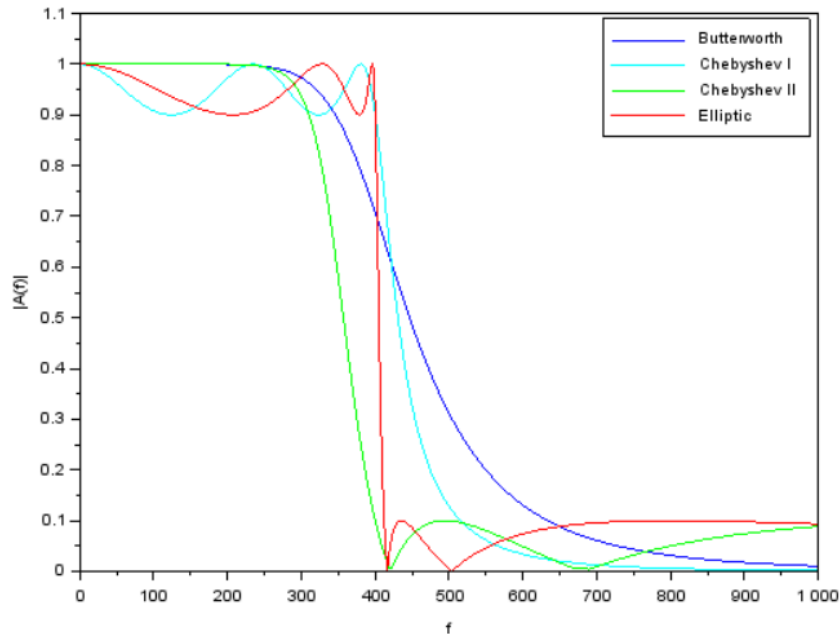


8.9 ábra: Elliptikus eljárással tervezett aluláteresztő szűrő különböző fokszámok esetén.

Az eljárások összehasonlítása

Ideális közelítés nem létezik, a megfelelő szűrő kiválasztását a feladatban elvárt karakterisztikák határozzák meg.

A közelítési eljárások amplitúdókarakterisztikáinak összehasonlítására az 8.10 ábrán azonos feltételekkel tervezett aluláteresztő szűrők láthatóak. Az alkalmazott paraméterek: $f_{hi}=400\text{Hz}$, $N=5$, $\delta_p=0.1$, $\delta_s=0.1$.



8.10 ábra: Aluláteresztő szűrő tervezése különböző módszerekkel, azonos feltételek mellett.

Az ábrából látható, de az előzőek alapján is kikövetkeztethető, hogy a legsimább átviteli függvényt a Butterworth-féle eljárás szolgáltatja, míg a Chebyshev-féle approximációk vagy az áteresztő- vagy a zárósávban eredményeznek hullámos karakterisztikát, az elliptikus módszer pedig mindkét sávban hullámos. Viszont az is észrevehető, hogy a leggyorsabb átmenetet az elliptikus eljárás adja, a leglassabbat pedig a Butterworth. Ez alapján a toleranciasémában egy adott szélességű átmenetisáv esetén az elliptikus módszerrel szükséges a legkisebb fokszám, a Butterworth approximációval pedig a legnagyobb.

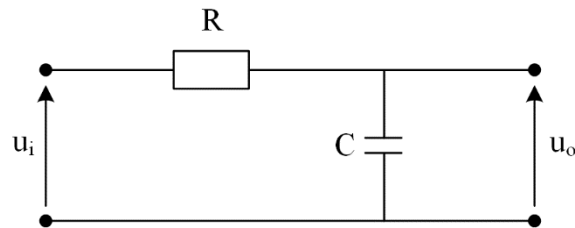
A fáziskarakterisztika alapján is a Butterworth módszer a legjobb, ugyanis ez biztosítja a leglineárisabb karakterisztikát. A legrosszabb a négy approximáció közül, az amplitúdó-karakterisztikához hasonlóan, az elliptikus eljárás.

8.5 Passzív elemekkel megvalósított szűrők

Szűrők megvalósíthatók kizárólag passzív elemek felhasználásával. Az ilyen megoldások előnyei az aktív elemeket is felhasználó megoldásokkal szemben, hogy nem igényelnek tápfeszültséget, megbízhatóak, stb.

RC aluláteresztő szűrő

Az egyik legegyszerűbb elsőrendű szűrő az ellenállás és a kondenzátor soros kapcsolásával kapható.



8.11 ábra: RC aluláteresztő szűrő.

Ha a kondenzátor feszültségét használjuk fel a rendszer kimeneteként, akkor aluláteresztő karakterisztikát kapunk. Ezt az esetet a következő differenciálegyenletet írhatjuk le:

$$RC \frac{du_C(t)}{dt} + u_C(t) = u_i(t) \quad (8.7)$$

Ebben az esetben a frekvenciafüggvény:

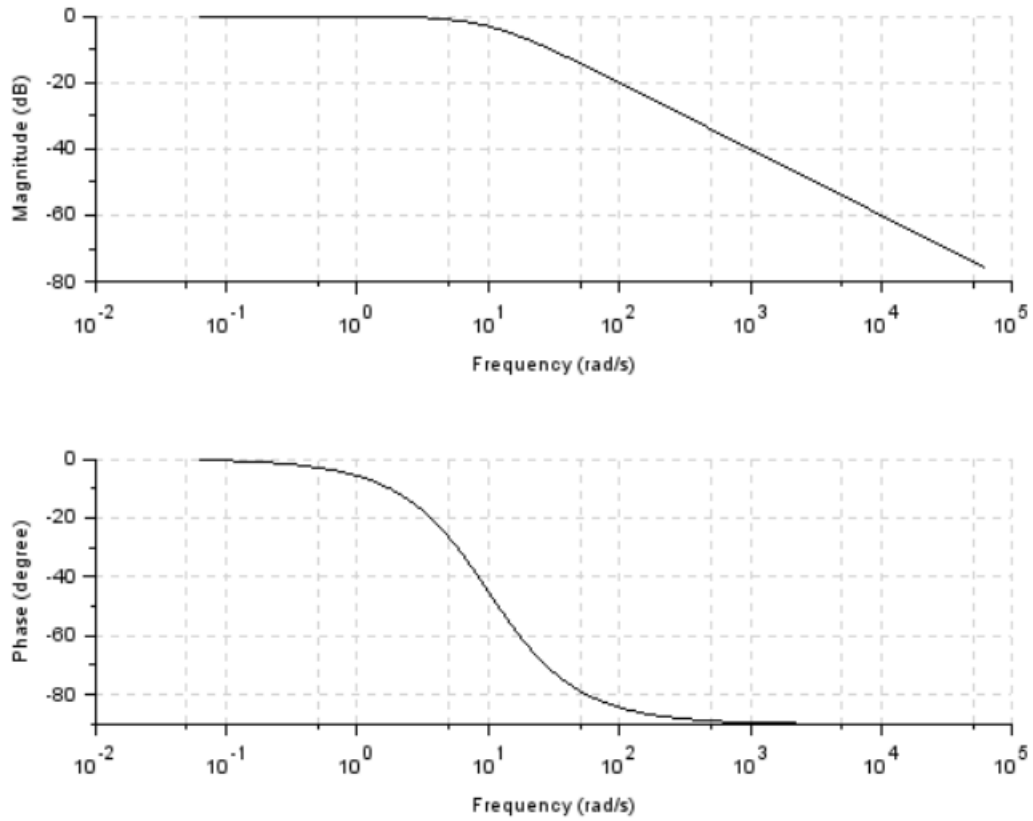
$$G(j\omega) = \frac{u_o(j\omega)}{u_i(j\omega)} = \frac{1}{j\omega RC + 1} = \frac{1}{j\omega\tau + 1} \quad (8.8)$$

Tehát az időállandó:

$$\tau = RC \quad (8.9)$$

A szűrő határfrekvenciája ez alapján megadható mint:

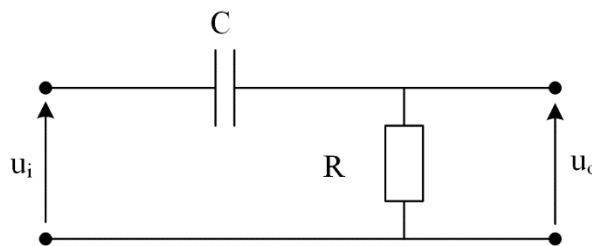
$$\omega_h = \frac{1}{\tau} = \frac{1}{RC} \quad (8.10)$$



8.12 ábra: RC aluláteresztő szűrő Bode diagramja.

RC felüláteresztő szűrő

Ha a kondenzátor feszültsége helyett az ellenálláson levő feszültséget használjuk fel kimenetként, akkor ez felüláteresztő karakterisztikát ad.



8.13 ábra: RC felüláteresztő szűrő.

A differenciálegyenlet ebben az esetben a következőképpen adható meg:

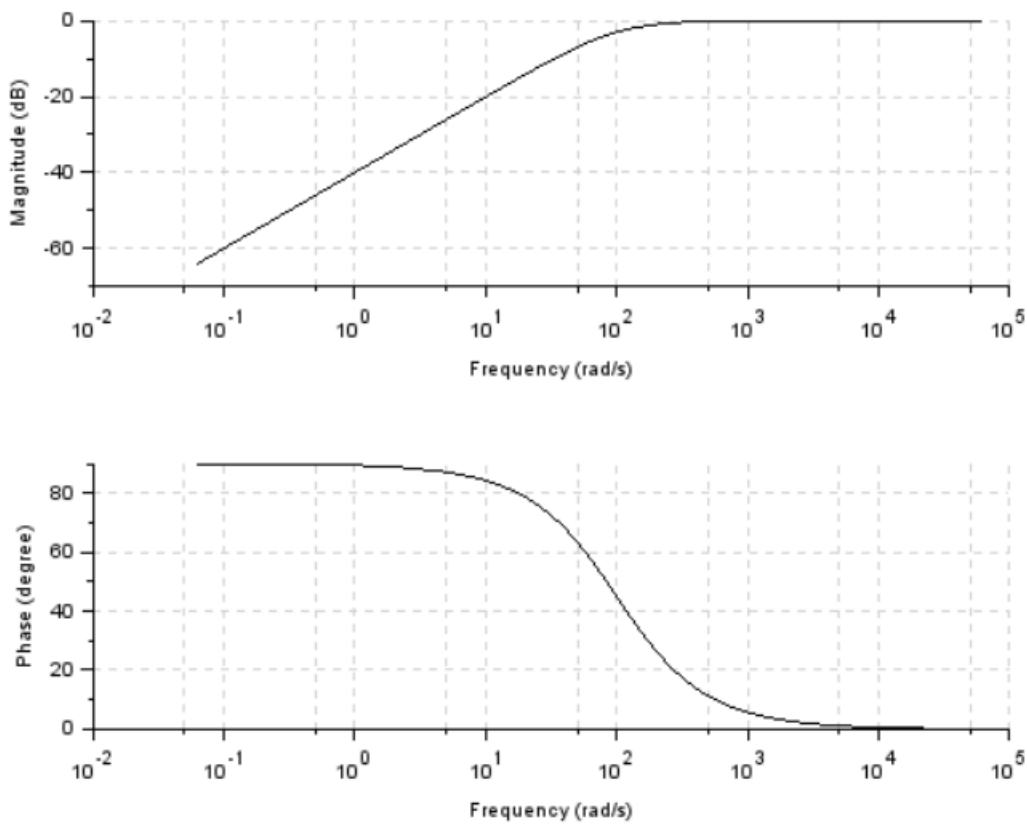
$$RC \frac{du_R(t)}{dt} + u_R(t) = RC \frac{du_i(t)}{dt} \quad (8.11)$$

Frekvenciafüggvénye:

$$G(j\omega) = \frac{u_o(j\omega)}{u_i(j\omega)} = \frac{j\omega RC}{j\omega RC + 1} \quad (8.12)$$

A szűrő határfrekvenciája ebben az esetben is:

$$\omega_h = \frac{1}{\tau} = \frac{1}{RC} \quad (8.13)$$

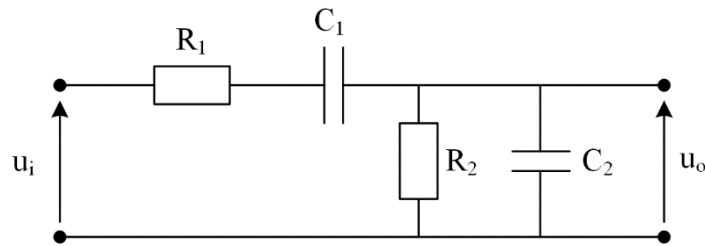


8.14 ábra: RC aluláteresztő szűrő Bode diagramja.

Ha az átmenet meredekségénél élesebb átmenetre van szükségünk, akkor összetettebb kapcsolásra van szükségünk, amely magasabb rendű szűrőt valósít meg.

RC sáváteresztő szűrő

Sáváteresztő jelleg elérhető például az 8.15 ábrán látható egyszerű RC kapcsolással.



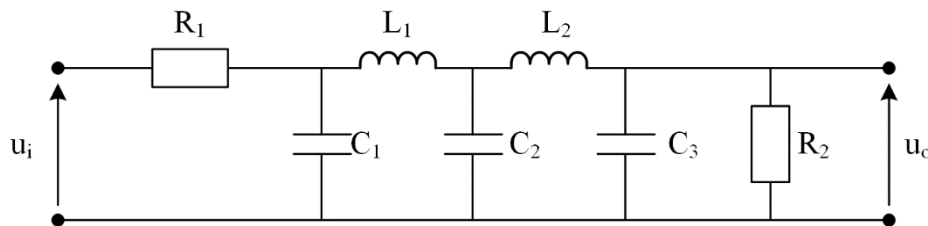
8.15 ábra: RC sáváteresztő szűrő.

LC szűrők

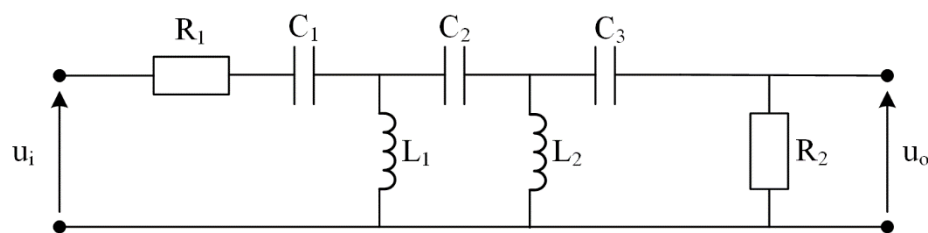
Tekercsek és kondenzátorok megfelelő kapcsolásával a legösszetettebb szűrési feladatok is megvalósíthatók.

Sokféle kapcsolat összeállítható ezen passzív elemek felhasználásával, de leginkább a létrahálózatok terjedtek el. Ezeknél az elrendezéseknél a szűrő fokszámát az LC tagok száma határozza meg.

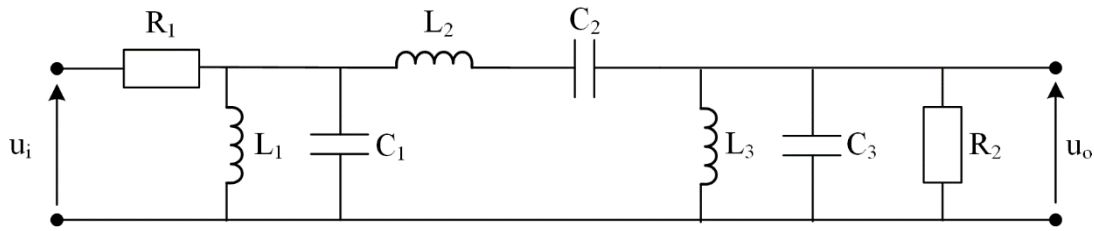
A 8.16, 8.17 és 8.18 ábrákon különböző típusú szűrőket megvalósító LC létrahálózatok láthatóak.



8.16 ábra: LC aluláteresztő szűrő.



8.17 ábra: LC felüláteresztő szűrő.



8.18 ábra: LC sáváteresztő szűrő.

8.6 Ellenőrző kérdések

1. [Milyen sávokat különböztetünk meg a nem ideális szűrők esetén?](#)
2. [Melyek a szűrők alaptípusai?](#)
3. [Mely paraméterek szükségesek a toleranciaséma meghatározásához?](#)
4. [Szükséges fokszám alapján melyik említett approximációs eljárás a legelőnyösebb?](#)
5. [Hogyan határozható meg egy egyszerű RC szűrő esetén a határfrekvencia?](#)

9 Digitális szűrők

A digitális szűrők olyan rendszerek, amelyek diszkrét idejű jeleken végeznek amplitúdó- és fázismódosítást. Az ilyen szűrők digitális eszközök segítségével vannak megvalósítva.

A digitális szűrők tervezésének lépései:

- Specifikáció – a probléma alapján meghatározzuk a kívánt rendszer amplitúdó- és fázisjellegét, illetve a megengedett toleranciákat.
- Szerkezet és típus – megválasztjuk a szűrő szerkezetét és típusát.
- Szintézis – meghatározzuk a zérusok és pólusok értékét, amelyekkel meg lehet valósítani a szűrőt.
- Megvalósítás.

A digitális szűrőknek két típusa van, a véges impulzusválaszú (Finite Impulse Response – FIR) és a végtelen impulzusválaszú (Infinite Impulse Response – IIR) szűrők.

9.1 FIR szűrők

FIR szűrők esetében a kimenet nem függ a korábbi kimenetektől, csak a jelenlegi és a korábbi bemenetek határozzák meg.

Átviteli függvényük a következő módon adható meg:

$$H(z) = \sum_{k=0}^M b_k z^{-k} \quad (9.1)$$

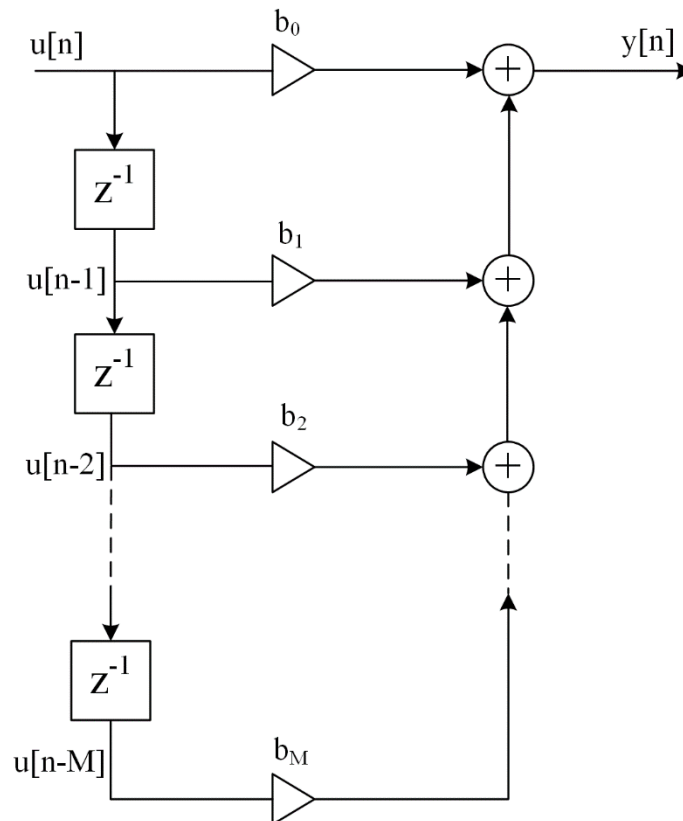
Ez azt jelenti, hogy ezeknek a szűrőknek nincs visszacsatolásuk, azaz csak zérusaik vannak, pólusaik nincsenek. A FIR szűrők mindig stabilak, mivel a pólusok hatnak ki a rendszerek stabilitására.

Az analóg rendszerek esetében nem lehet megvalósítani olyan átviteli függvényt amelynek csak zérusai vannak, így a FIR szűrők átviteli függvényét nem lehet megvalósítani analóg szűrők alapján, csak direkt szintézissel.

A FIR szűrő kimenete meghatározható a következő módon:

$$y[n] = \sum_{i=0}^M b_i u[n-i] \quad (9.2)$$

Az egyenletből látható, hogy a kimenet a koefficiensek és a bemenetek diszkrét konvolúciójával határozható meg. A szerkezetből és az egyenletből megállapítható az is, hogy a FIR szűrő impulzusra adott válasza visszaadja a szűrő koefficienseit.



9.1 ábra: FIR szűrő blokk diagramja.

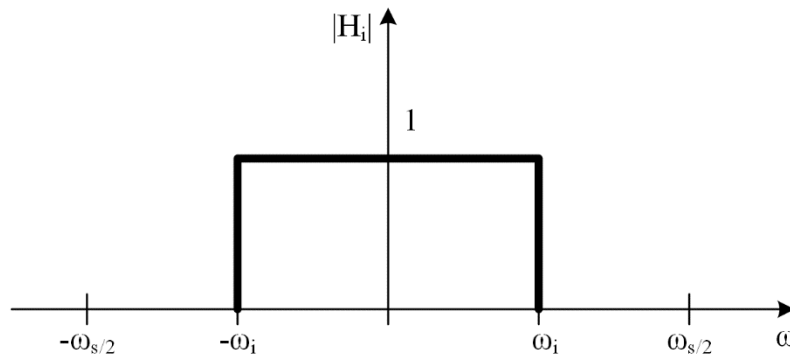
FIR szűrők tervezése során meg kell állapítani a fokszámot amellyel kielégíthető a kívánt átviteli karakterisztika és a koefficienseket. Tervezésükre több eljárás létezik, melyek közül a legismertebb az ablakozó módszer.

9.1.1 FIR szűrők tervezése ablakozással

Az ablakozó módszer egy gyors és egyszerű eljárás. Azt használja ki, hogy a FIR szűrők súlyfüggvénye megegyezik az együtthatókészlettel, és a koefficienseket a specifikációban megadott ideális átviteli karakterisztikának diszkrét idejű inverz Fourier transzformációjával állítja elő.

A tervezés a következő lépésekből áll:

- Ideális frekvenciafüggvény meghatározása.
- A frekvenciafüggvény impulzusválaszának meghatározása diszkrét idejű inverz Fourier transzformáció segítségével.
- Az impulzusválaszt szimmetrikusan rövidíteni kell, hogy véges számú együtthatóból álljon, és el kell tolni, azaz késleltetni kell, hogy kauzális legyen.



9.2 ábra: Ideális frekvenciafüggvény aluláteresztő szűrő esetén.

Az ábrán látható aluláteresztő szűrő esetén a következőt kapjuk:

$$h_i[n] = \frac{T}{2\pi} \int_{-\frac{\omega_s}{2}}^{\frac{\omega_s}{2}} H_i(e^{j\omega t}) e^{j\omega t} d\omega = \frac{T}{2\pi} \int_{-\omega_i}^{\omega_i} 1 e^{j\omega t} d\omega = \frac{\sin(\omega_i nT)}{n\pi} \quad (9.3)$$

Az együtthatók számát végesre csökkentve:

$$-\frac{N-1}{2} \leq n \leq \frac{N-1}{2} \quad (9.4)$$

A kauzalitás érdekében ezt jobbra eltolva és behelyettesítve a következőt kapjuk:

$$h_i[n] = \frac{\sin(\omega_i T [n - \frac{N-1}{2}])}{[n - \frac{N-1}{2}]\pi} \quad (9.5)$$

A csonkítás hatására azonban Gibbs-jelenség fog jelentkezni, ami oszcillációt okoz a sáváteresztő tartomány végén és a sávzáró tartomány elején, függetlenül az N értékétől. Ezt a jelenséget csökkenteni tudjuk szimmetrikus simító ablak használatával. Az impulzusválasz függvényt meg kell szoroznunk egy megfelelő $w(n)$ ablakozó függvénnyel:

$$h_w[n] = h_i[n]w[n] \tag{9.6}$$

Többféle ablakozó függvényt alakítottak ki, melyek közül a legismertebbek: Hanning, Hamming, Kaiser, Blackman, stb.

9.2 IIR szűrők

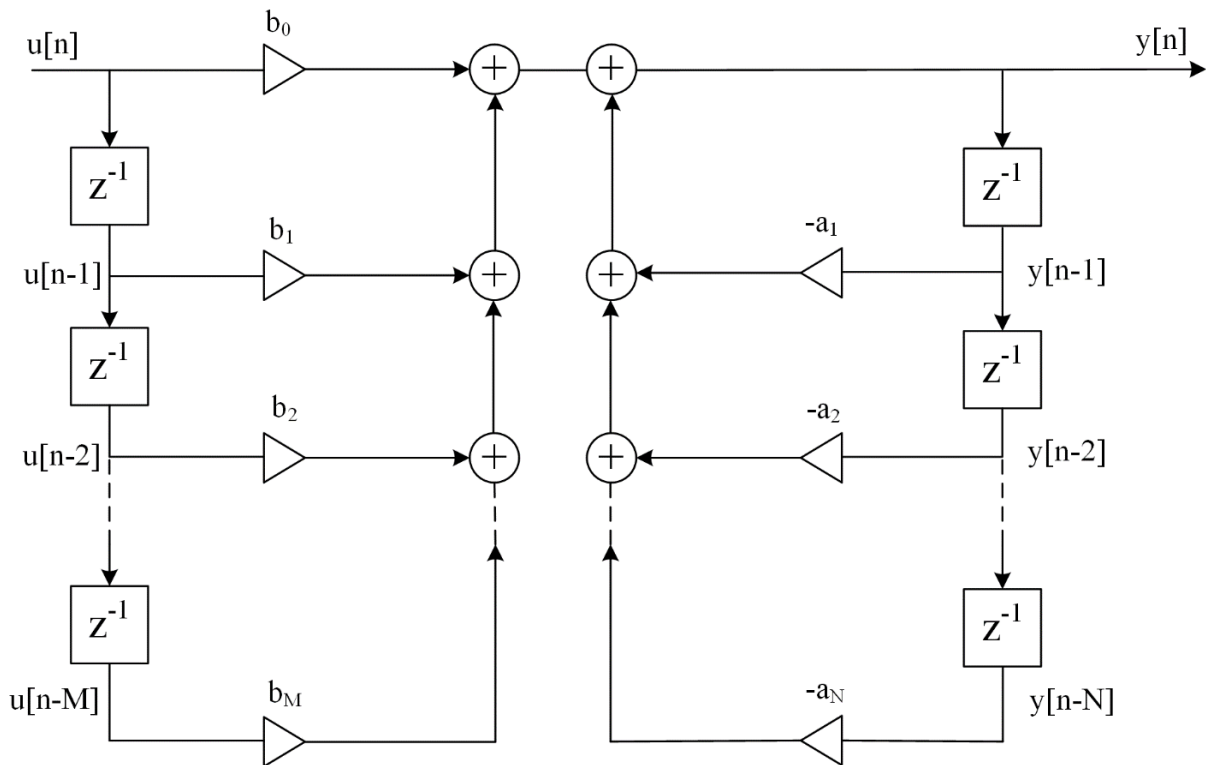
IIR szűrők esetében a korábbi bemenetek mellett a korábbi kimeneteket is figyelembe vesszük a kimenet meghatározásánál. Tehát, az IIR struktúrák visszacsatolást is tartalmaznak.

Az IIR szűrők átviteli függvénye a következő formában adható meg:

$$H(z) = \frac{\sum_{k=0}^M b_k z^{-k}}{1 + \sum_{k=1}^N a_k z^{-k}} \tag{9.7}$$

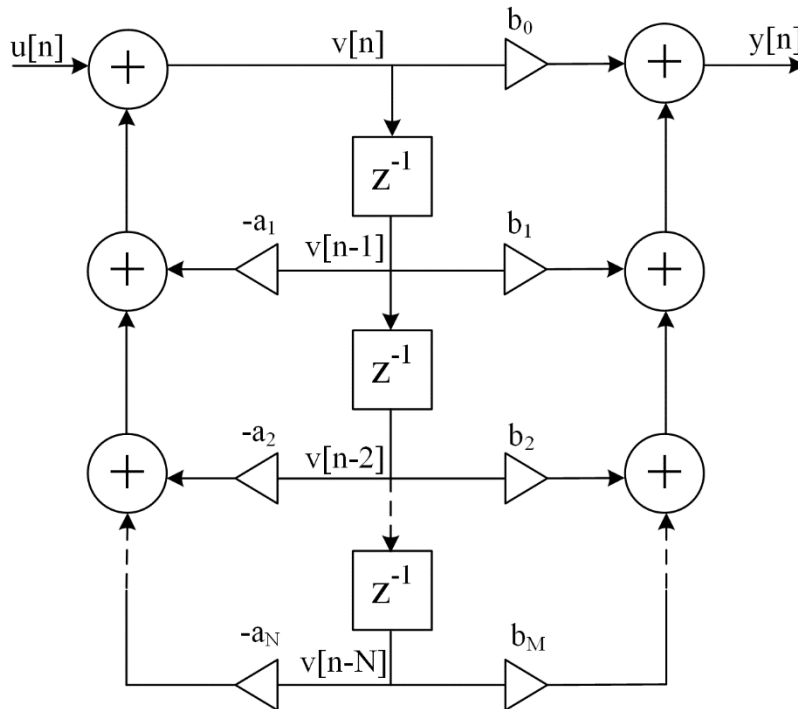
Az IIR szűrő kimenete a következő módon fejezhető ki:

$$y[n] = \sum_{i=0}^{M-1} b_i u[n-i] - \sum_{i=0}^{N-1} a_i y[n-i] \tag{9.8}$$



9.3 ábra: IIR szűrő blokk diagramja I. direkt formában.

Az ábrán látható I. direkt formán kívül szokás a II. direkt formát is használni, amely kevesebb késleltető elemet használ fel.



9.3 ábra: IIR szűrő blokk diagramja II. direkt formában ($N=M$).

Ebben az esetben a kimenet következő módon határozható meg:

$$v[n] = u[n] - \sum_{i=1}^N a_i v[n - i] \tag{9.9}$$

$$y[n] = \sum_{i=0}^M b_i v[n - i] \tag{9.10}$$

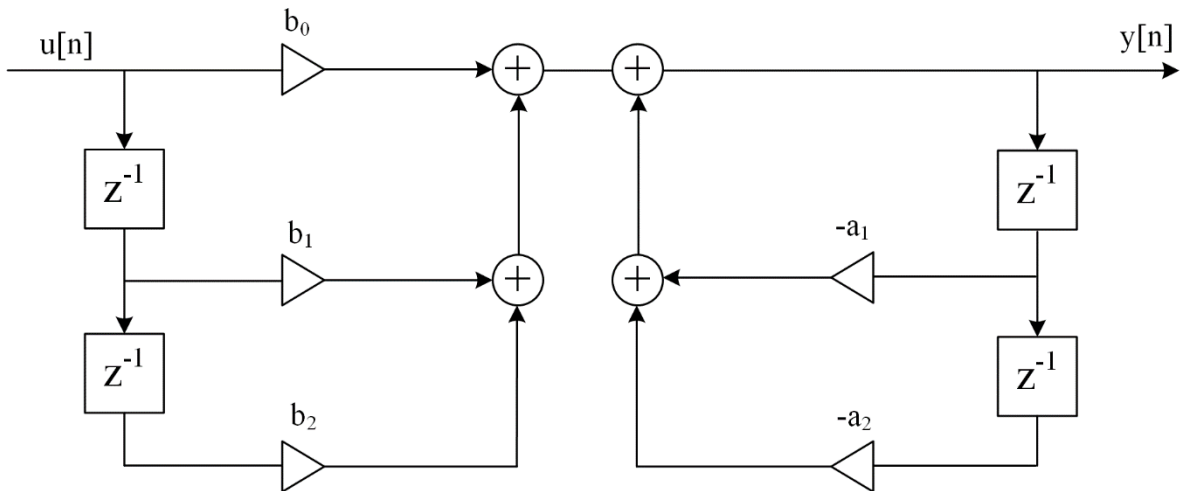
Az IIR szűrőket gyakran szokták kaskád formában megvalósítani, amely másodfokú tagokból (biquad) áll. Az átviteli függvény ebben az esetben a következő módon adható meg:

$$H(z) = \prod_{k=1}^{N_s} H_k(z) = \prod_{k=1}^{N_s} \frac{b_{0k} + b_{1k}z^{-1} + b_{2k}z^{-2}}{1 - a_{1k}z^{-1} - a_{2k}z^{-2}} \tag{9.11}$$

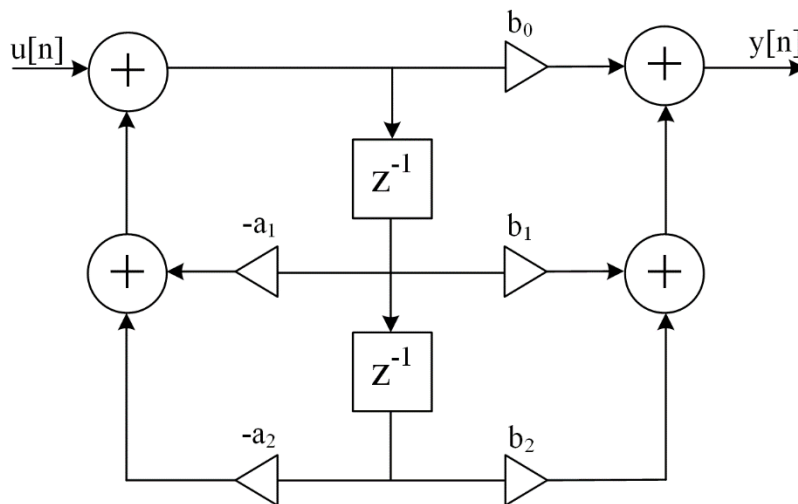
A szükséges biquadok száma a következő kifejezéssel számolható ki:

$$N_s = \frac{\max(N,M)+1}{2} \tag{9.12}$$

Az egyes biquadok realizálásakor általában a II. direkt formát szokták alkalmazni.



9.4 ábra: Biquad szűrő I. direkt formában.

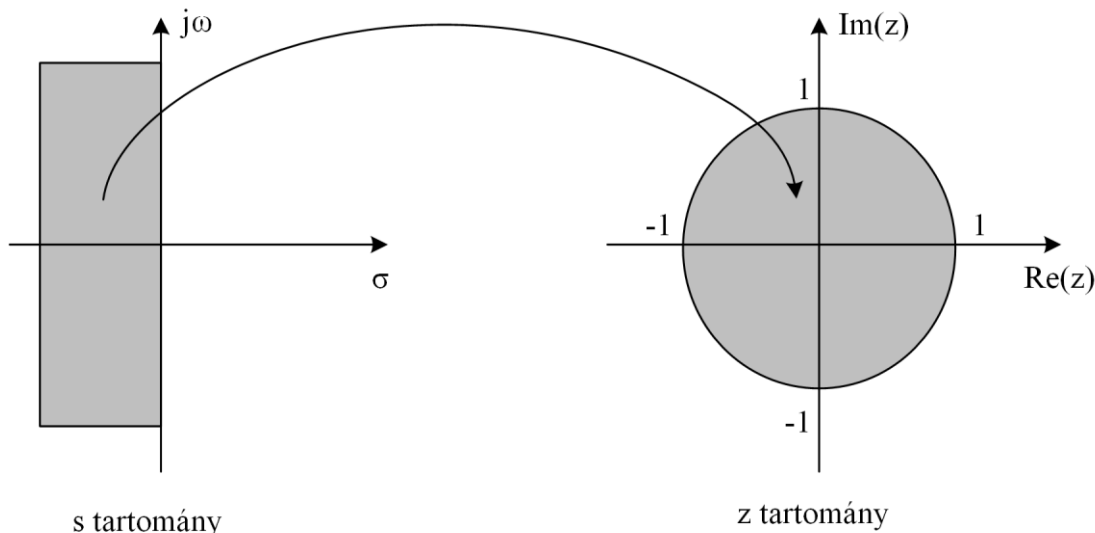


9.5 ábra: Biquad szűrő II. direkt formában.

IIR szűrők tervezésére a legegyszerűbb módszer az analóg szűrők transzformálása, tehát egy $H(s)$ átviteli függvényhez keresünk megfelelő $H(z)$ -t. Erre több módszer létezik, melyek közül a legismertebb a bilineáris transzformáció.

9.2.1 Bilineáris transzformáció

Ez a transzformáció az s -tartomány bal félsíkját a z -tartomány egységkörének belsejébe képezi le, tehát, stabil s -beli pólust stabil z -beli pólusba képez le.



9.6 ábra: Bilineáris transzformáció.

Az analóg átviteli függvény esetén ehhez a következő helyettesítést kell alkalmazni:

$$s = \frac{2}{T_s} \frac{1-z^{-1}}{1+z^{-1}} \tag{9.13}$$

9.3 FIR és IIR szűrők összehasonlítása

Mindkét digitális szűrőtípusnak vannak előnyei és hátrányai.

A FIR szűrők előnyei:

- A FIR struktúrákkal olyan speciális karakterisztikák is megvalósíthatók amelyeket IIR szűrőkkel nem lehet létrehozni. Pl.: lineáris fáziskarakterisztika (a jel spektrális komponenseinek fázisa ugyanazon mértékben változik a szűrőn keresztülhaladva), lineáris amplitúdó-karakterisztika, állandó 90 fokos fáziseltolás, lyuk szűrő (angolul notch filter, ez egy olyan sávzáró szűrő amelynek zárósávja nagyon keskeny), illetve más olyan karakterisztikák amelyet nem lehet analóg tartományból való átképezéssel megvalósítani.

- IIR szűrőkkel lineáris fázisátmenet nem valósítható meg, viszont elérhető, hogy az áteresztősávban közel lineáris legyen a karakterisztika.
- A FIR szűrők mindig stabilak, míg az IIR esetén lehetnek stabilitási problémák.
- A FIR szerkezetek implementációjához kisebb számításigény szükséges.

Az IIR szűrőknek is van több előnyös tulajdonsága a FIR struktúrákhoz képest:

- Az IIR struktúrák alacsonyabb foksámot igényelnek. A FIR szűrők sok esetében akár több száz együttthatóra is szükséges lehet, ilyen esetben pedig a számításigény is sokkal magasabb.
- Az IIR szűrők tervezése egyszerűbb.
- Az IIR szűrők jól modelleznek fizikai rendszereket.

9.4 Analóg és digitális szűrők összehasonlítása

Egy adott feladat megoldása során több szempontot figyelembe kell venni amikor el szeretnénk dönteni, hogy analóg vagy digitális szűrőt alkalmazzunk. Mindkét típusnak vannak előnyei és hátrányai.

Az analóg szűrők előnyei:

- Az analóg szűrők megvalósítása egyszerű szűrők esetén sokkal olcsóbb.
- Az analóg szűrők működését nem korlátozza a mintavételezési frekvencia, így nagy frekvenciákon is alkalmazhatók. A technológia fejlődésével egyre nagyobb mintavételezési frekvencia alkalmazható, így sok helyen váltják fel az analóg rendszereket digitálisak.
- A mintavételezési frekvencia mellett digitális eszközöknél figyelembe kell venni azt is, hogy a szükséges számítások is igényelnek időt. Például egy 48000 Hz-es frekvenciával mintavételezett hangjel esetén másodpercenként ennyi adaton kell a megfelelő műveleteket elvégezni.

A digitális szűrők előnyei:

- Az analóg szűrőkben használt elemek pontatlanabbak, a digitális szűrők pedig pontosak.
- Az analóg szűrők elemeinek tulajdonságai időben változnak, tehát időinvariánsak, míg a digitális szűrőkre ez nem érvényes.
- A digitális szűrők könnyen átprogramozhatóak. Analóg szűrőknél ha változtatásra van szükség akkor egyszerűbb esetben az elemek más értékűekre kell cserélni, de nagyobb változtatás esetén akár új szerkezet megvalósítására is szükség lehet.
- A digitális szűrőknél csak kvantálási zajok jelennek meg, míg az analóg szűrőknél nagy elektronikus zaj jelentkezik.

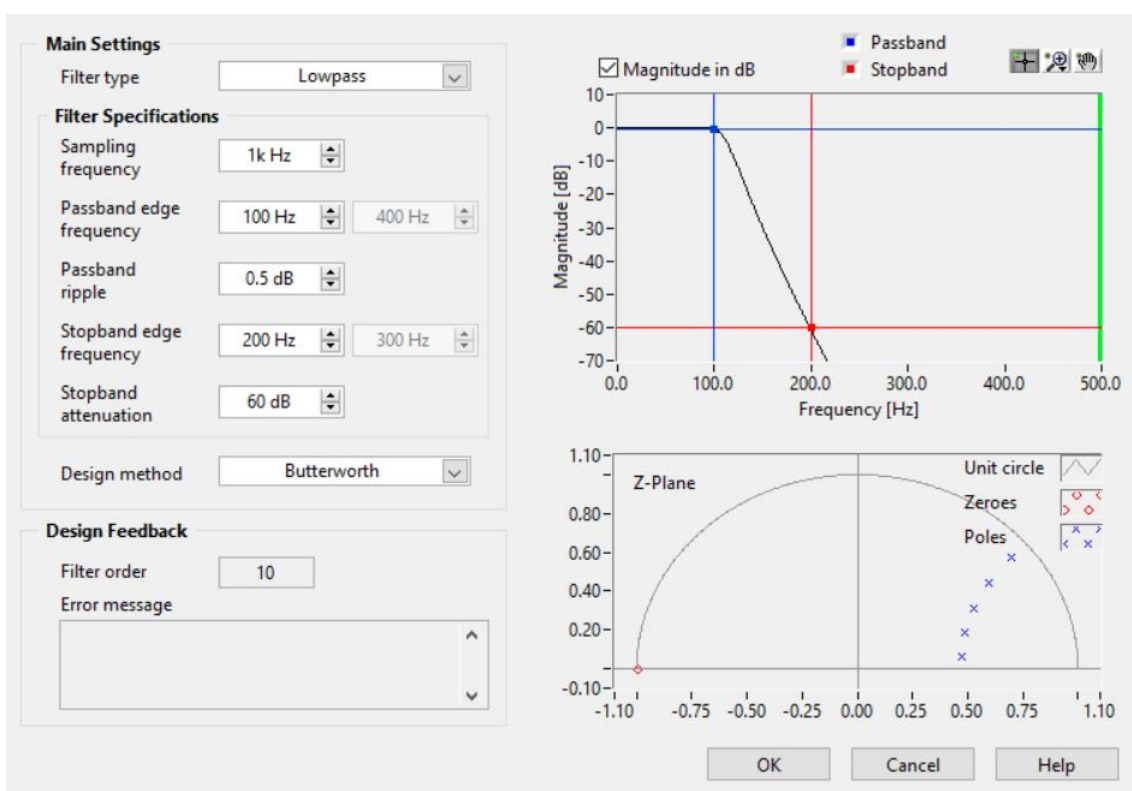
9.5 Példák

9.1 Példa

A LabVIEW Classical Filter Design blokkjának segítségével vizsgáljuk meg a szükséges fokszámot különböző szűrőtervezési eljárások használata esetén. Használjuk a következő paramétereket: aluláteresztő szűrő, $f_s=1000\text{Hz}$, $f_{pass}=100\text{Hz}$, $f_{stop}=200\text{Hz}$, $\delta_{pass}=0.5\text{dB}$, $\delta_{stop}=60\text{dB}$. Ezután az átmenetisávot szűkítsük le, például változtassuk meg az f_{pass} paramétert 190Hz-re, majd figyeljük újra a szükséges fokszámokat.

Megoldás:

Az első paramétersorozat esetén a beállítások a következők:



Eredmény:

Első paramétersorozat esetén:

- Butterworth: 10
- Chebyshev: 6
- Inverse Chebyshev: 6
- Elliptic: 5
- Equi-Ripple FIR: 22

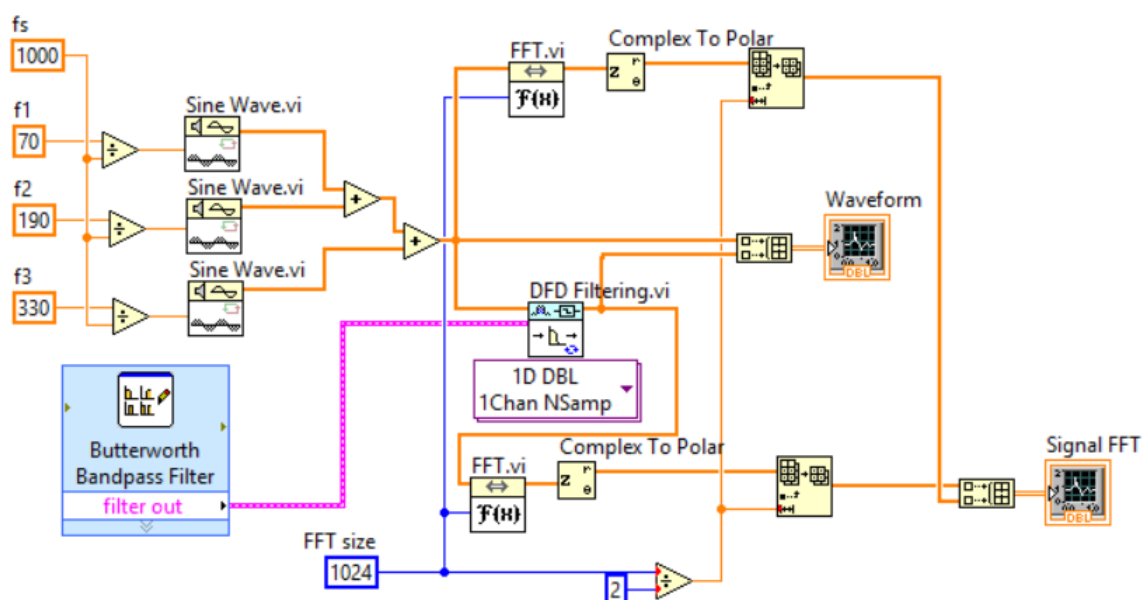
Második paramétersorozat esetén:

- Butterworth: 120
- Chebyshev: 24
- Inverse Chebyshev: 24
- Elliptic: 10
- Equi-Ripple FIR: 202

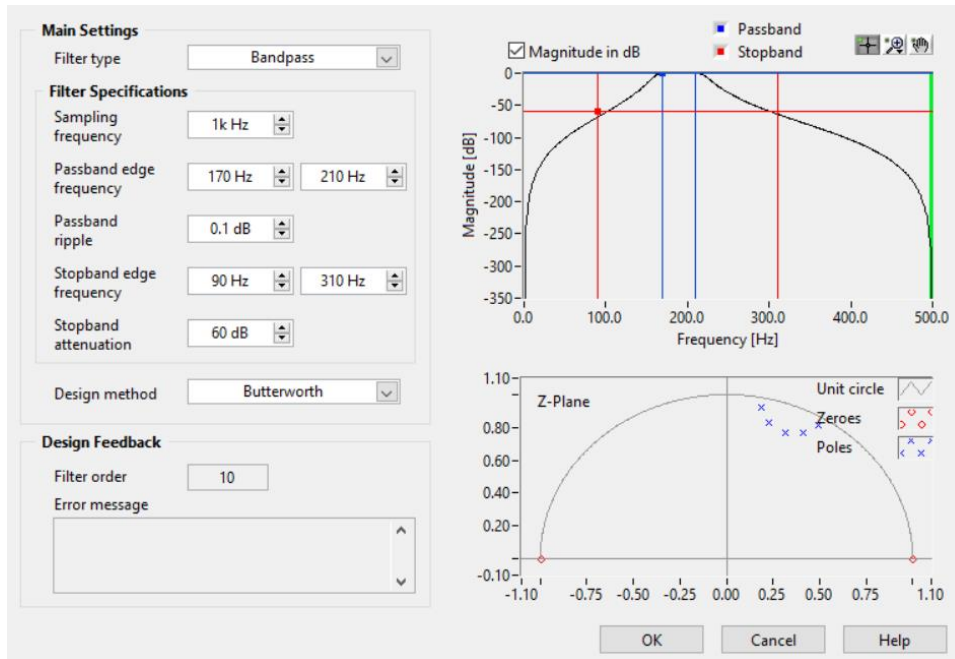
9.2 Példa

LabVIEW segítségével állítsunk össze három különböző frekvenciájú szinusz jelből egy eredő jelet, majd megfelelő szűrő segítségével szűrjük úgy, hogy csak a középső komponenst engedje át. Törekedjünk a síma karakterisztikára és a minél alacsonyabb fokszám érdekében alkalmazzunk minél szélesebb átmenetisávot.

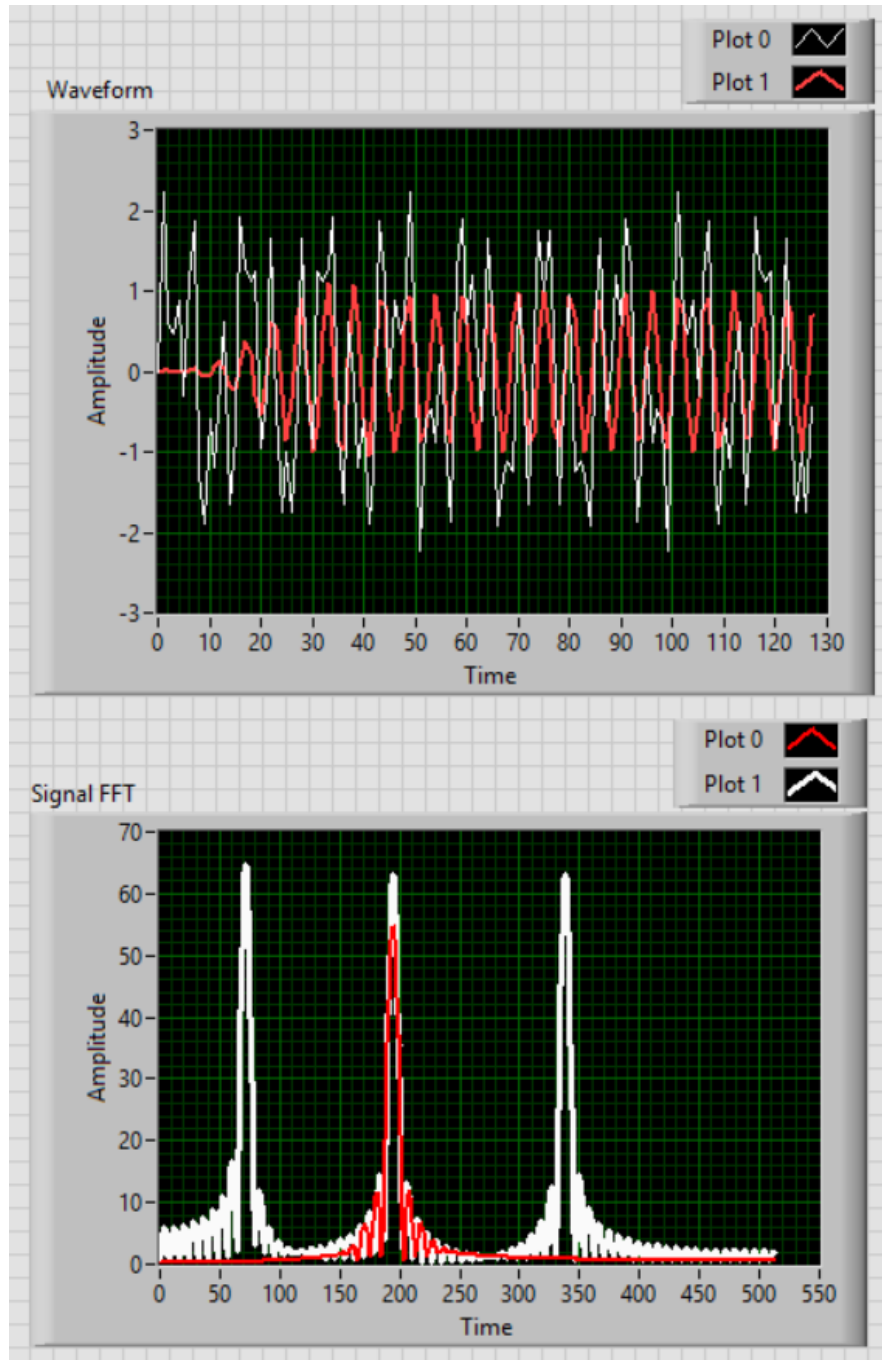
Megoldás:



A szűrő paramétereit:



Eredmény:



9.6 Ellenőrző kérdések

1. [Miért stabilak a FIR szűrők?](#)
2. [Mi az alapvető különbség a FIR és IIR szűrők felépítésében?](#)
3. [Miért előnyösebb a II. direkt forma használata?](#)
4. [Milyen módszer segítségével transzformálhatók az analóg szűrők?](#)
5. [Mik az IIR szűrők előnyei a FIR szűrőkkel szemben és mik az előnyei a FIR szűrőknek az IIR szűrőkhöz képest?](#)

10 Adatgyűjtő rendszerek

Az adatgyűjtő rendszerek (angolul Data Acquisition Systems – DAS, de a DAQ is gyakran használt rövidítés) feladata a méréstechnikában az információgyűjtés fizikai rendszerekről, majd az információ feldolgozása és esetleges tárolása megfelelő szoftver segítségével. Az ilyen rendszerek lehetőséget adnak az adatok azonnali, azaz real-time vagy online feldolgozására, de az adatok tárolásával erre a mérések elvégzése után, vagyis offline, is van lehetőség.

Gyakran az ilyen rendszerek számítógéppel (PC) vezérelt mérőrendszerek, amelyek adatgyűjtés mellett sokszor a műszerek és folyamatok vezérlését is végzik. Ezeket a feladatokat lényegében egy a számítógépen futó szoftver végzi. A szoftver kifejlesztésére számos programnyelv áll rendelkezésre, például LabVIEW vagy akár C, C++, C#, stb. Az adatok mentésére is több lehetőség van, használhatóak MySQL, Oracle, vagy egyéb adatbázisok, de akár szöveges fájlokba is menthetőek az adatok.

A mérésadatgyűjtő rendszerek állhatnak adatgyűjtő célhardverekből (DAQ eszköz) is, amelyekre szenzorok vannak illesztve, az adatgyűjtők pedig egy PC számára szolgáltatják az adatokat. De léteznek már olyan eszközök is, amelyeket nem kell feltétlenül PC-hez kapcsolni, mivel önállóan is képesek működni.

Olyan rendszerek esetén, amelyek beavatkoznak a folyamatba és online feldolgozást végeznek, nagyon fontos szempont a sebesség, mivel meghatározott időn belül a rendszernek reagálnia kell.

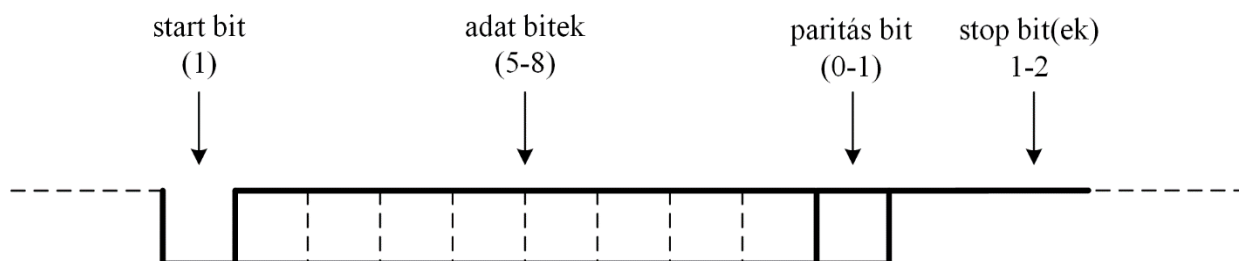
A mérési rendszerek struktúráját és a mérések folyamatát a mérési feladat szabja meg. Gyakran célszerű úgy létrehozni a rendszert, hogy új mérőműszer minél könnyebben hozzáilleszhető legyen. Abban az esetben ha műszerek száma eléri azt a szintet, hogy már egy egység nem képes minden műszerrel egyszerre kommunikálni és az adatokat feldolgozni, akkor több adatgyűjtő, például PC, is beilleszhető. Ilyenkor ezek az eszközök általában egymással is kommunikálhatnak, például LAN (Local Area Network) hálózaton keresztül.

10.1 Protokollok

A rendszerben található eszközök közötti kommunikációra számos szabványos protokoll áll rendelkezésre, amelyek közül itt néhány kerül csak felsorolásra.

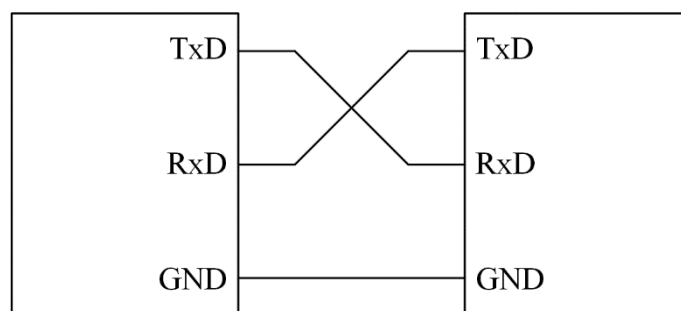
RS-232

Az RS-232 a gyakorlatban széles körben alkalmazott protokoll, amelyet soros kommunikációnak is nevezünk. Általános esetben két eszköz között alakít ki kapcsolatot teljesen-duplex (full-duplex) módon.



10.1 ábra: Csomagok szerkezete az RS-232 kommunikációban.

A legegyszerűbb megoldás kétirányú kommunikáció esetén három vezetéket igényel, kettőt adatátvitelre, a harmadikat pedig a közös földpotenciálhoz. Csak az egyik eszközt figyelve, az TxD-n keresztül küldi (angolul transmit data), míg az RxD-n keresztül fogadja (angolul receive data) az adatokat.



10.2 ábra: Szükséges vezetékek a legegyszerűbb esetben.

Tipikus sebességek (angolul baud rate) [bit/s]: ..., 2400, 4800, 9600, 19200, 38400, 57600, és 115200. Létezik néhány megoldás magasabb sebességekre is, de ezeket ritkán használják.

Ennek a protokollnak a hátrányai az alacsony átviteli sebesség és a kis átviteli távolság (kb. 10-15m).

Manapság már a számítógépek ritkán rendelkeznek soros port csatlakozóval, így leginkább USB-n keresztül van lehetőségünk a használatára, amelyhez átalakítót iktatunk be.

RS-485

Ez a protokoll az RS-232 továbbfejlesztett változata, amellyel nagyobb sebességgel és nagyobb távolságokra is lehetséges az adatátvitel. Ezen kívül több, akár 32 eszköz is felfűzhető a buszra.

A adatvitel itt már differenciális jellel történik csavart érpáron keresztül, és nincs szükség a földpotenciál átvitelére. Egyszerre két irányban viszont nem történhet kommunikáció, tehát fél-duplex (angolul half-duplex).

Az RS-485 protokollal kis távolságoknál akár 35Mb/s sebesség is elérhető, míg 100kb/s esetén akár 1200m távolságra is átvihető a jel.

USB

Az USB (univerzális soros busz, angolul Universal Serial Bus) esetén is soros az adatáramlás. Fontos tulajdonságai, hogy USB hub-ok segítségével osztható, és hogy a vezetékek lehetőséget adnak táplálásra is.

Ennek a protokollnak a nagy sebesség az előnye. Az USB-nek három generációja létezik:

- USB 1.x – 12Mbit/s a maximális sebesség
- USB 2.0 – 480 Mbit/s a maximális sebesség
- USB 3.x – a 3.0 típusnál 5 Gbit/s, a 3.1-nél 10 Gbit/s, a 3.2-nél pedig 20 Gbit/s a maximális sebesség.

MODBUS

A MODBUS protokoll nyílt forráskódú protokoll ipari felhasználásokra. Előre definiált kódokat és master/slave adatátvitelt alkalmaz.

A baud rate és a paritás is állítható, illetve két üzemmód közül lehet választani: ASCII és RTU (Remote Terminal Unit). Az ASCII üzemmód esetén a nyolcbites adatok két darab ASCII kódnak megfelelő karakterként továbbítja az adatokat, míg az RTU módnál két négybites hexadecimális kódként. Mindkét üzemmód alkalmaz hibadetektáló kódokat a csomagokban.

PROFIBUS

A PROFIBUS (Process Field Bus) szintén automatizálási rendszerekben gyakran használt protokoll. A 7 rétegű OSI modell első két rétegét használja. Az első réteg a fizikai réteg (angolul Physical Layer), mely az átvitel fizikai karakterisztikáját határozza meg, és a megalkotók hozzá az RS-485 protokollt választották. Az átviteli sebesség 9,6 kbit/s és 12Mbit/s között választható. A második a buszelérési protokollt definiálja, és adatkapcsolati rétegnek (angolul Data Link Layer) nevezik.

CAN BUSZ

A CAN (angolul Controller Area Network) esetében a szenzorok, aktuátorok és vezérlők egy buszra vannak felfűzve, melyen az adatátvitel soros. Fontos, hogy nincs szükség központi vezérlő egységre. A protokoll az OSI modell fizikai és adatkapcsolati rétegének felel meg.

Legfontosabb tulajdonságai a nagy sebesség, olcsó implementálhatóság, egyszerű konfigurálhatóság, és a központi hibaelLENŐRZÉS. A CAN busz legismertebb felhasználása a járműipar, amely ezt a protokollt alkalmazza a járművekben.

I2C

Az I2C (angolul Inter-Integrated Circuit, egyéb használt rövidítései még az I²C és az IIC) egy kétirányú, kétvezetékes buszrendszer amelyet IC-k közötti kommunikációra fejlesztett ki a Philips. A digitális szenzoroknál manapság ezt és/vagy az SPI protokollt alkalmazzák a gyártók. Digitális szenzorok esetén például egy mikrovezérlő segítségével olvashatjuk a mérési adatokat a szenzorokból, majd továbbíthatjuk őket egy adatgyűjtőnek.

A működéshez szükséges két vezeték az SDA (soros adatvonal) és az SCL (soros órajel). A buszhoz csatlakoztatható több master vagy több slave eszköz. A slave eszközök rendelkeznek egy egyedi címmel, amelynek egy részét a gyártó határozza meg, a többit pedig huzalozással lehet tetszőlegesen állítani.

A kommunikáció soros, 8-bites, a sebesség pedig normál üzemmódban 100kbit/s, míg gyors üzemmódban 400kbit/s. Kis távolságon szokás alkalmazni, leginkább lapon belül.

SPI

Az SPI (angolul Serial Peripheral Interface) protokoll egy master-slave alapú szinkron adatátviteli rendszer, melyet a Motorola fejlesztett ki.

A busz sebessége körülbelül 2MHz-ig növelhető, és egy adatátviteli ciklus nyolc órajel alatt megy végbe.

A protokollhoz négy vezeték szükséges: kimeneti adatvonal (angolul Master Out Slave In – MOSI), bemeneti adatvonal (angolul Master In Slave Out – MISO), soros órajel (angolul Serial Clock – SCL), és az egységet kiválasztó (angolul Chip Select – CS). Kommunikáció során a master állítja elő az órajelet és választja ki a slave eszközt.

Az SPI előnye az I2C-vel szemben a magasabb sebesség, viszont több vezeték igényel.

UART

Az UART (univerzális aszinkron adóvevő, angolul Universal Asynchronous Receiver-Transmitter) szintén IC-k közötti kommunikációra használható, viszont nagyobb távolságra átvihető és nincs szükség órajelre. A kommunikáció folyamata az előzőekben kifejtett legegyszerűbb RS-232-höz hasonlít, viszont ebben az esetben 8 adatbit van.

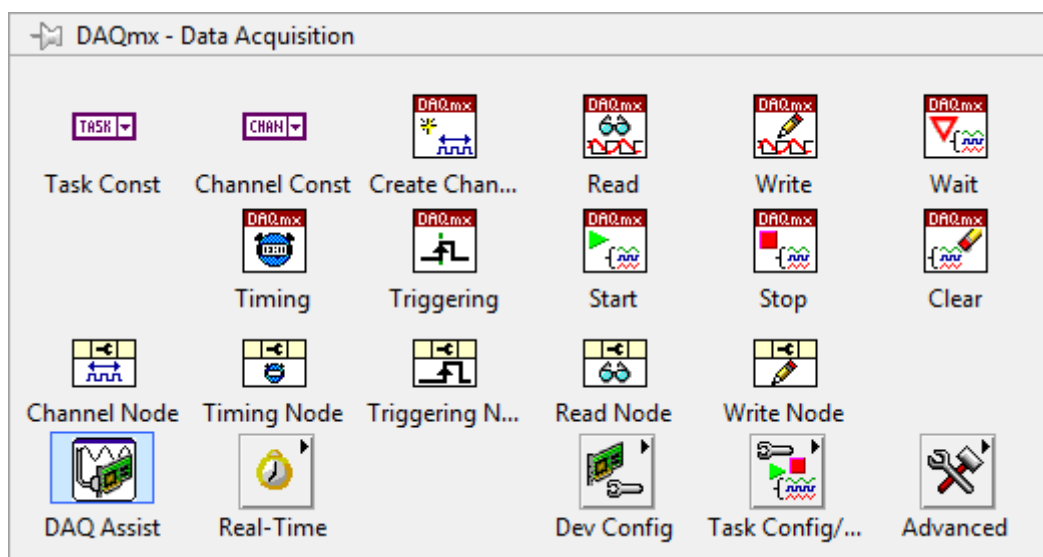
10.2 NI alapú adatgyűjtő rendszerek

A National Instruments (NI) számos adatgyűjtő eszközt gyárt. Ezeknek nagy előnye, hogy segítségével prototípus rendszer nagyon gyorsan összeállítható a könnyű illesztés miatt, és LabVIEW-ban gyorsan fejleszthető hozzájuk megfelelő szoftver.

Az általuk gyártott eszközökön kívül lehetőségünk van más műszereket is összekötni a LabVIEW-val. Ilyen esetben szükség van driver-re, amely segítségével az operációs rendszer felismeri az eszközt, illetve LabVIEW-n belül egy instrument driver-re, amely lehetőséget ad a műszer programon keresztül használatához.

A műszerek könnyű konfigurálására és tesztelésére a Measurement & Automation Explorer (NI MAX) programon keresztül van lehetőség.

A LabVIEW-ból az NI által gyártott műszereket a DAQmx könyvtáron keresztül lehet elérni.



10.3 ábra: A LabVIEW DAQmx palettája.

A paletta tartalmaz blokkokat csatorna létrehozására (Create Channel), a mérés elindítására (Start) és leállítására (Stop), az erőforrás felszabadítására (Clear), adat olvasására (Read) és írására (Write), stb. A mérés történhet Channel vagy Task segítségével. A Channel az eszköz egy fizikai csatornája, míg a Task egy mérési feladat, amelyet az NI MAX segítségével konfigurálhatunk, és akár az eszköz több csatornáját is felhasználhatja.

A mintavételezés időzítését végezheti a szoftver és a hardver is. Szoftveres esetben a szoftver tetszés szerinti időpontban kérhet pillanatnyi mérést, míg hardveres esetben egy állandó mintavételezési frekvenciát alkalmaz, ami konfigurálható.

10.2.1 Egyszerű adatgyűjtő eszközök

Ezek számítógéphez köthető univerzális műszerek, melyek több analóg bemenettel és kimenettel, és digitális I/O vonalakkal rendelkeznek. Ezek az eszközök általában USB interfészen keresztül kommunikálnak a számítógéppel.

Ilyen eszköz például az NI USB-6001, mely a következő paraméterekkel rendelkezik:

- Analóg bemenetek
 - Csatornák – 8 (differenciális módban 4)
 - A/D felbontás – 14-bit
 - Maximális mintavételezési sebesség – 20 kHz
 - Feszültségtartomány – $\pm 10V$
- Analóg kimenetek
 - Csatornák – 2
 - D/A felbontás – 14-bit
 - Maximális frissítési sebesség – 5 kHz
 - Feszültségtartomány – $\pm 10V$
- Digitális I/O
 - Vonalak – 13 (Port 1: 8, Port 2: 4, Port 3: 1)



10.4 ábra: NI USB-6001 adatgyűjtő

10.2.2 Valós idejű rendszerek

Az NI gyárt valós idejű platformokat is, mint például a cRIO. Az ilyen eszközök alkalmaznak valós idejű operációs rendszert, amely a hétköznapiakban használt operációs rendszerekkel ellentétben képes adott időkorlátokon belül választ generálni. A hétköznapi operációs rendszer számos task számára szolgáltatja a processzoridőt, és megtörténhet, hogy a mi programunk hosszú ideig nem kap időt, mivel másik lefoglalta.

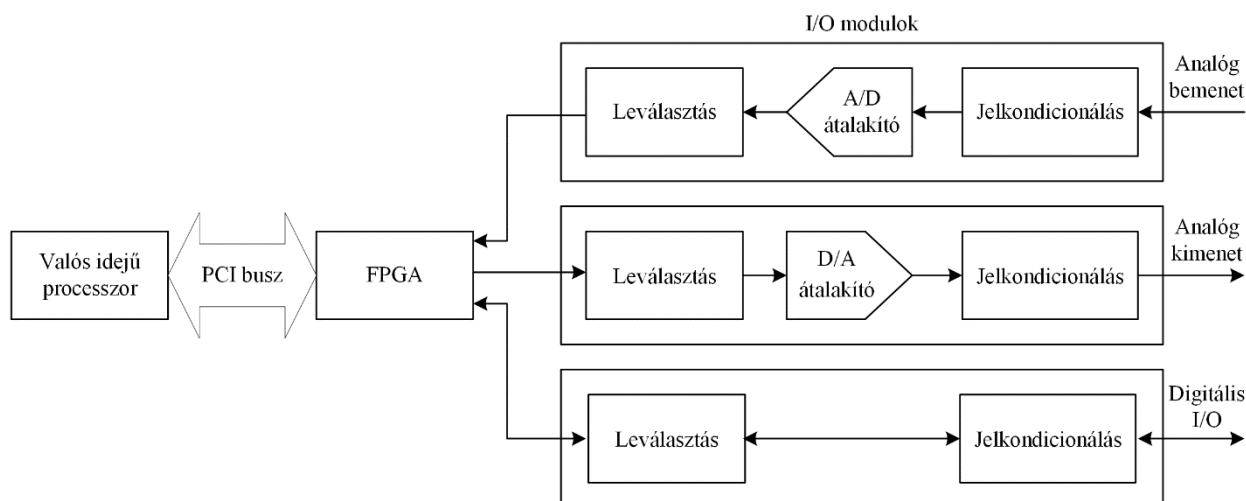
A cRIO egy valós idejű processzorból és egy hátlapból (chassis) áll. A hátlaphoz feladattól függően különböző modulokat csatlakoztathatunk.



10.5 ábra: Különböző modulokkal ellátott cRIO-9068.

A valós idejű processzoron fut a valós idejű operációs rendszer. A hátlap tartalmaz egy FPGA-t (Xilinx gyártmányút), amely kezeli a modulokat, és jelfeldolgozási feladatokat is elláthat. A processzor és az FPGA egy PCI buszon keresztül van kapcsolatban.

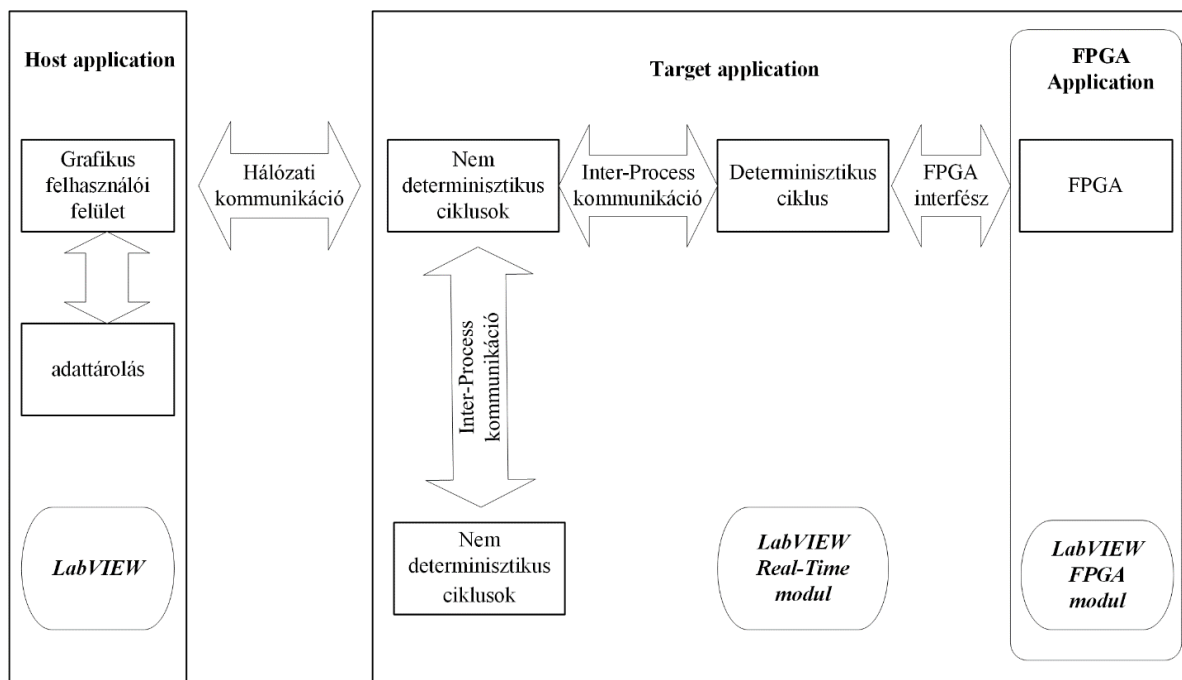
A modulokkal felszerelt cRIO belső felépítése az 10.6 ábrán látható. A modulok galvanikusan le vannak választva. A modulokat működés közben ki lehet húzni és újra lehet csatlakoztatni (angolul hot-swappable).



10.6 ábra: Modulokkal felszerelt cRIO belső felépítése.

Egy cRIO alkalmazás tipikusan a következő három részből áll:

- Host Application – ez a számítógépen futó alkalmazás, amely a valós idejű egységgel hálózaton keresztül kommunikál. Ez az alkalmazás tartalmazza a grafikus felhasználói felületet (angolul Graphical User Interface – GUI), illetve végzi az adatok tárolását és offline feldolgozását. Fejlesztése egyszerű LabVIEW-ban történik, amellyel Windows operációs rendszerre készíthetünk programokat.
- Target Application – ez az alkalmazás a cRIO valós idejű processzorán fut. Párhuzamosan több szálát futtathat, melyek közül egy determinisztikus ciklus (angolul deterministic loop), a többi pedig nem determinisztikus (angolul non-deterministic loop). A determinisztikus ciklus végzi a valós idejű feladatokat, melyek lefutását a magas prioritás biztosítja. Lényegében a többi ciklus akkor fog processzoridőt kapni, amikor a determinisztikus ciklusnak nincs rá szüksége. A kommunikáció, azaz az adatscere, az egyes szálak között Inter-Process kommunikációval történik. A számítógéppel történő kommunikáció is nem determinisztikus cikluson keresztül történik, mivel ez a folyamat nem kell, hogy valós időben történjen. A program létrehozása a LabVIEW Real-Time moduljával történik.
- FPGA Application – ez a cRIO részét képező FPGA-n fut. Ez a rész teremt kapcsolatot a processzor és a modulok között, illetve nagy sebességű feldolgozás is végezhető segítségével. A program fejlesztése a LabVIEW FPGA moduljával történik, mely fordítás során a blokk diagramból köztes fájlokat hoz létre, majd ezekből a Xilinx compiler megfelelő bitfájlt, amely már az FPGA-ra tölthető.



10.7 ábra: cRIO alkalmazás részei.

Hasonló elven működik a szintén NI által gyártott MyRIO is, viszont illeszthető modulok helyett programozható I/O-kat tartalmaz.



10.8 ábra: NI MyRIO.

10.3 Vezetéknélküli szenzorhálózatok

Adatgyűjtés területén fontos még megemlíteni a vezetéknélküli szenzorhálózatok (angolul Wireless Sensor Network – WSN) fogalmát, melyek felhasználása egyre jobban terjed el.

Az ilyen hálózatok általában a térben szétszórt elemekből (angolul node) állnak, melyek különböző környezeti körülményeket (hőmérséklet, páratartalom, légszennyezés, stb.) figyelnek megfelelő szenzorok segítségével, és valamilyen vezeték nélküli technológiával kommunikálnak. A mérési adatokat általában egy központi egység felé továbbítják.

Az alkalmazott eszközök (angolul mote) általában mikrovezérlő alapúak és nem rendelkeznek állandó táplálással, így a fogyasztás minimalizálása érdekében (és egyben az élettartam növelése miatt) csak nélkülözhetetlen egyéb elektronikai elemeket tartalmaznak.

Több cég is gyárt ilyen célra optimalizált eszközöket, illetve speciális operációs rendszerek is lettek már fejlesztve hozzájuk, mint pl. a TinyOS vagy a Contiki.

A környezeti körülmények egyszerű figyelése mellett léteznek olyan alkalmazások is, amelyeknek a célja valamilyen esemény detektálása. Ilyen lehet például az erdőtüz, lavina, stb.

Ipari alkalmazásokban is egyre gyakrabban kerülnek felhasználásra ilyen rendszerek. Például használhatjuk őket gépek állapotának figyelésére, de akár a termékek lokalizációjára és követésére egy gyártósoron.

A WSN-ek több problémával rendelkeznek, melyek közül néhány az alábbiakban röviden kifejtésre kerül. Ezek megoldására folynak a különböző kutatások, és számos lehetséges megoldás kerül kidolgozásra.

Szinkronizáció

Az alacsony fogyasztás elérése érdekében célszerű az eszközöket alvó módban tartani (angolul sleep mode) amikor csak lehetséges. A mikrovezérlőt érdemes csak néha felébreszteni, hogy a szenzorokból olvasson vagy kommunikációt végezzen. A szenzorokat is ajánlott csak ilyen esetekben bekapcsolni, és egyébként kikapcsolva vagy alvó módban tartani. A kommunikációs egység esetében pedig mind a küldés mind a fogadás sok energiát igényel, sőt, a fogadó módhoz általában nagyobb áram szükséges. Így az sem megvalósítható, hogy a fogadó rész állandóan aktív legyen, és érkező csomag hatására ébresszük fel a mikrovezérlőt. Ezen problémák megoldására fontos egy közös időt szinkronizálni a hálózatban, amely alapján minden eszköz tudja, hogy mikor kell küldenie és fogadnia csomagokat.

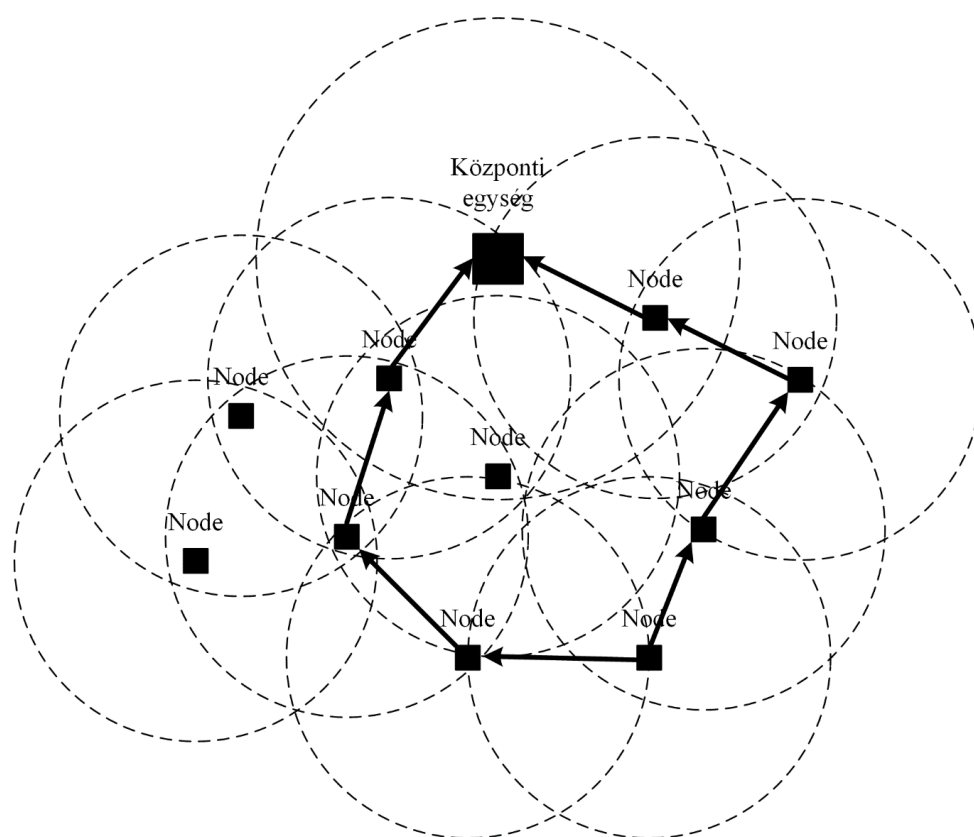
Kommunikáció

Az eszközök gyakran random alapon kerülnek elhelyezésre, illetve néhány esetben a helyük is változhat az idő során. Ez a kommunikáció folyamatára van nagy kihatással.

A moteok általában, az alacsonyabb fogyasztás miatt, csak kis hatótávolságú kommunikációs eszközzel vannak felszerelve. Ha nagy területen vannak szétszórva az eszközök, akkor nagy valószínűséggel van közöttük olyan, amely nem tud kommunikálni a központi egységgel. Sőt, általában csak néhány szomszédos egységgel tudnak kommunikálni. További nehezítő körülmény, hogy az eszközök helyváltoztatása miatt vagy más okokból kifolyólag, a mote szomszédságából eltűnhetnek egyes eszközök.

A vázolt problémák alapján az adatok eljuttatása a központi egységhez nem egyértelmű.

A 10.9 ábrán látható egy példa szituáció véletlenszerűen elszórt nodeokkal, amelyek egy központi egységnek továbbítják az adatokat. A szagatott vonal az eszközök kommunikációs hatótávolságát jelöli. Szintén látható két lehetséges útvonal is, amelyen keresztül az egyik egység eljuttathatja az adatok a központnak.



10.9 ábra: Véletlenszerűen elszórt eszközök és azok kommunikációs hatótávolsága.

Lokalizáció

A WSN-ekben gyakran felmerülő probléma az eszközök lokalizációja, egyes alkalmazásokban lehet akár ez a rendszer feladata is.

Ez a feladat kültérben egyszerűbb folyamat, de különböző okokból kifolyólag itt sem egyértelmű. Egyes esetekben GPS használatával megoldható a probléma, de egyes környezeti körülmények között a GPS sem működik megbízhatóan, vagy az alkalmazásunk nem teszi lehetővé ennek használatát. Léteznek olyan megoldások, amelyek például a rádiókommunikációból kinyert adatokat használják fel. Két eszköz közötti távolság becslésére például felhasználható az RSSI érték (angolul Received Signal Strength Indicator), amelynek nagy hátránya a pontatlanság. Több eszköz között meghatározva a távolságot különböző algoritmusokkal végezhető lokalizáció. Egyes lokalizációs algoritmusok csak azt az információt használják fel, hogy mely eszközök vannak egymás hatótávolságában, azaz mely eszközök képesek kommunikálni egymással a rendszeren belül.

Még nehezebben megoldható feladat a beltéri lokalizáció, ahol a falak és különböző tárgyak mind befolyásolják a rádióhullámok terjedését, a GPS pedig beltérben nehezen ritkán alkalmazható megbízhatóan.

10.4 Ellenőrző kérdések

1. [Melyek a legfontosabb ipari kommunikációs protokollok?](#)
2. [Hogyan épül fel egy csomag RS-232 kommunikáció esetén?](#)
3. [Milyen részekből áll egy modulokkal felszerelt cRIO?](#)
4. [Mely részekből áll egy valós idejű cRIO alkalmazás?](#)
5. [Melyek a legfontosabb problématicák a vezeték nélküli szenzorhálózatokban?](#)

Irodalomjegyzék

- [1] Balogh László, Kollár István, Németh József, Péceli Gábor, Sujbert László: *Digitális jelfeldolgozás*, Műegyetemi Kiadó, 2006.
- [2] Burány Nándor: *Az elektronika alapjai*, Szabadkai Műszaki Főiskola, 2001.
- [3] Fodor Attila, Vörösházi Zsolt: *Beágyazott rendszerek és programozható logikai eszközök*, Typotex, 2011, ISBN 978-963-279-500-3.
- [4] Fodor Dénes: *Digitális jelfeldolgozás*, Pannon Egyetem, 2014.
- [5] Gerzson Miklós: *Méréselmélet*, Typotex, 2011, ISBN 978-963-279-502-7.
- [6] Gerzson Miklós, Pletl Szilveszter: *Irányítástechnika*, Typotex, 2011, ISBN 978-963-279-529-4.
- [7] Jeges Zoltán: *Jelek és rendszerek*, Szabadkai Műszaki Főiskola, 2003.
- [8] Zoher Z. Karu: *Signals and Systems: Made Ridiculously Simple*, ZiZi Press, 2001, ISBN 0-9643752-1-4.
- [9] Odry Péter: *Jelfeldolgozás*, Szabadkai Műszaki Főiskola, 2000.
- [10] Alan V. Oppenheim, Roland W. Schaffer, John R. Buck: *Discrete-Time Signal Processing*, Prentice Hall, 1999, ISBN 0-13-754920-2.
- [11] John Park, Steve Mackay: *Practical Data Acquisition for Instrumentation and Control Systems*, Newnes, 2003, ISBN 07506-57960.
- [12] Sárosi József: *Mérési adatok gyűjtése*, Szegedi Tudományegyetem, Mérnöki Kar, 2014, ISBN 968-963-306-284-5.
- [13] Steven W. Smith: *Digital Signal Processing: A Practical Guide for Engineers and Scientists*, Newnes, 2003, ISBN 0-750674-44-X.
- [14] Váradiné Szarka Angéla, Hegedűs János, Bátorfi Richárd, Unhauzer Attila: *Méréstechnika*, Széchenyi István Egyetem.
- [15] National Instruments hivatalos honlapja (www.ni.com).