

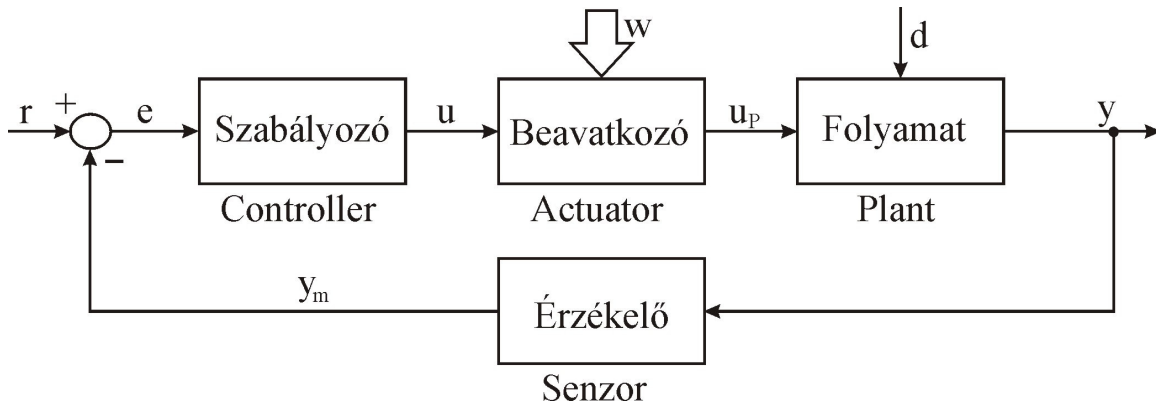
## 2. Folyamatszabályozási alapok

A szabályozás célja, hogy az irányított folyamat kimenete (szabályozott jellemző) megfeleljen az előírt értéknek elfogadható hibahatáron belül. Ha az előírt érték állandó, értéktartó szabályozásról, ha az előírt érték időben változik, követő szabályozásról beszélünk.

Szabályozást a folyamat bemenetének a megfelelő megválasztásával érjük el. A leghatékonyabb módszer a folyamat bemenetének meghatározására a visszacsatolás alkalmazása: mérjük az irányított folyamat kimenetét és a kimenet meg az alapjel függvényében határozzuk meg a folyamat bemenetét.

### 2.1. A szabályozási hurok

A szabályozási hurok a folyamatból és az annak az irányítását biztosító technikai eszközökből álló zárt rendszer (lásd 2.1. Ábra).



2.1 Ábra: A szabályozási hurok felépítése

A hurokban megjelenő jelek elnevezései<sup>1</sup>:

$u_p$  – a folyamat bemenete – plant input (módosított jellemző)

$y$  – a folyamat kimenete – plant output (szabályozott jellemző)

$y_m$  – mért kimenet – measured output (ellenőrző jel)

$r$  – előírt érték – reference signal (alapjel)

$e$  – szabályozási hiba – control error (rendelkező jel)

$u$  – beavatkozó jel – control signal

$d$  – zavaró jel – disturbance

<sup>1</sup> Az elnevezések az angolszász szakirodalomban használtak fordítása. Amennyiben létezik más, magyar elnevezés az adott jelre, az zárójelben fel van tüntetve.

$w$  a rendszerbe beáramló energiát jelöli. Ez nem egy jel a szabályozási hurokban csak azt szemlélteti, hogy a szabályozási körbe a beavatkozón keresztül áramlik be az energia.

A szabályozási hurok elemeinek főbb jellemzői az irányítási algoritmus szempontjából:

*Az irányított folyamat:* Bemenete tipikusan nagyenergiájú jel, amelyet a szabályozási hurkon belül úgy kell meghatározni, hogy a folyamat kimenete kövesse az előírt értéket.

Ahhoz, hogy meg tudjuk határozni a szabályozási feladatnak megfelelő bemenetet, ismernünk kell az irányított folyamatot leíró modellt. A modellt általában átviteli függvény formájában adjuk meg ( $H_{IF}(s)$ ). Ezt megkaphatjuk a folyamat viselkedését meghatározó törvények alapján. A modell paraméterei lehetnek adottak (például egy motor katalógusából kiolvashatjuk a motor modelljének paramétereit). Ha a paraméterek ismeretlenek, identifikációs eljárásokat alkalmazunk ezek meghatározására: a folyamat bemenetére ismert tesztjeleket adunk és mérjük a kimenetét. A bemenet és kimenet függvényében következtetünk a folyamat paramétereire.

Sok esetben a folyamatot leíró modell a működési tartománytól függően változó lehet (például egy kemence  $25^\circ\text{C}$  körüli hőmérsékleten más dinamikus viselkedést mutathat, mint  $1000^\circ\text{C}$  körüli hőmérsékleten). Ez abból adódik, hogy a kemenceét leíró dinamikus modell paraméterei időben változhatnak, illetve a kemence hővesztesége különböző hőmérséklettartományokban más és más lehet. A folyamat modelljét mindig abban a működési tartományban kell pontosan meghatározni, amelyben irányítani akarjuk.

*Érzékelő:* a folyamat kimenetét alakítja az irányítási algoritmus által feldolgozható értékre. A bemeneteük lehet nem-elektromos jel (például hőmérséklet, fordulatszám), a kimenete tipikusan elektromos jel. Célszerű, ha az érzékelő kimenete egyezményesített, szabványos tartományú jel. Ilyen tartományok például a 0-5 V, 0-10 V, 0-20 mA, 4-20 mA. Természetesen nem minden jelet lehet vagy érdemes egyezményesített tartományba átalakítani. Például az érzékelő digitális jelsorozatot is szolgáltatathat, amelynek a frekvenciája változik a folyamat kimenetének függvényében. Fontos azonban, hogy az érzékelő a folyamat teljes működési tartományában képes legyen mérni a kimenetet.

Az érzékelő dinamikája gyorsabb kell legyen, mint a folyamat, kisebb időállandókkal kell rendelkezzen. Általában az érzékelőket kis időállandójú elsőfokú rendszerként modellezhetjük:

$$H_E(s) = \frac{K_E}{T_E s + 1} \quad (2.1)$$

Ha az érzékelő időállandója ( $T_E$ ) elhanyagolható a folyamat domináns időállandói mellett, az érzékelőt leíró modell jó megközelítéssel ideális erősítő:

$$H_E(s) = K_E \quad (2.2)$$

*Beavatkozó:* A beavatkozó szerepe, hogy bemenetén megjelenő kisenergiájú jelet (amit a szabályozó határoz meg) a folyamat bemenetének megfelelő szintre átalakítsa, tehát tipikusan kisenergiájú jeltől nagyenergiájú jelet kell, hogy létrehozzon. Jellemző a

beavatkozóra, hogy ezen az elemen keresztül áramlik be az energia a szabályozási körbe (w a 2.1. Ábrán). Hasznos, ha a bemenete egyezményesített jel. Elvárható, hogy kisebb időállandókkal rendelkezzen, mint az irányított folyamat. Általában első, vagy másodfokú rendszerként modellezhető:

$$H_B(s) = \frac{K_B}{T_B s + 1}; \quad H_B(s) = \frac{K_B}{(T_{B1}s + 1)(T_{B2}s + 1)} \quad (2.3)$$

Az *irányítási algoritmus* feladata, hogy az érzékelő által mért kimenet és az előírt érték függvényében meghatározza a beavatkozó bemenetét úgy, hogy a folyamat kimenete kövesse az alapjelet. Az esetek többségében először képezzük az előírt érték és a mért kimenet különbségét, vagyis meghatározzuk a szabályozási hibát:

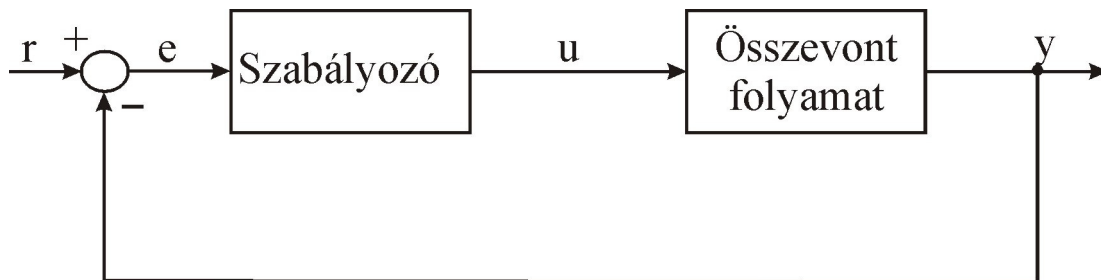
$$e = r - y_m \quad (2.4)$$

A szabályozó bemenetét így a szabályozási hiba képezi. A szabályozónak mind a bemenete mind a kimenete kisenergiájú jel.

A szabályozó megvalósításánál, illetve tervezésénél hasznos, ha a hurok többi részét összevont folyamatként kezeljük. Mivel az érzékelő – irányított folyamat – beavatkozó sorosan vannak csatolva, az összevont folyamat modellje:

$$H_F(s) = H_B(s) \cdot H_{IF}(s) \cdot H_S(s) \quad (2.5)$$

Az összevont folyamattal a szabályozási hurok a 2.2. Ábrán látható. Az összevont folyamat kimenetét  $y$  jelöli. Habár az összevont folyamat utolsó eleme az érzékelő, aminek a kimenetét  $y_m$ -el jelöltük, a szabályozó tervezési algoritmusoknál és a szabályozók megvalósításánál áttérünk az egyszerűbb,  $y$  jelölésre.



**2.2 Ábra: Szabályozási hurok összevont folyamattal**

**2.1 Példa:** A 2.3. Ábrán egy ventilátor fordulatszámát szabályozó hurok látható. A ventilátort villamos gép hajtja, amelyet egy teljesítményerősítőn keresztül irányíthatunk. A teljesítményerősítőt az jellemzi, hogy a bemeneti feszültség függvényében lineárisan változik a motorra adott teljesítmény. A ventilátor fordulatszámát tahogenerátorral mérjük. A fordulatszámot egy betáplált potenciométerrel írjuk elő. A rendszert egy analóg be- és kimenetekkel felszerelt adatbegyűjtő kártyával rendelkező számítógéppel

irányítjuk. Határozzuk meg a szabályozási hurokban megjelenő jeleket és a hurok elemeit.

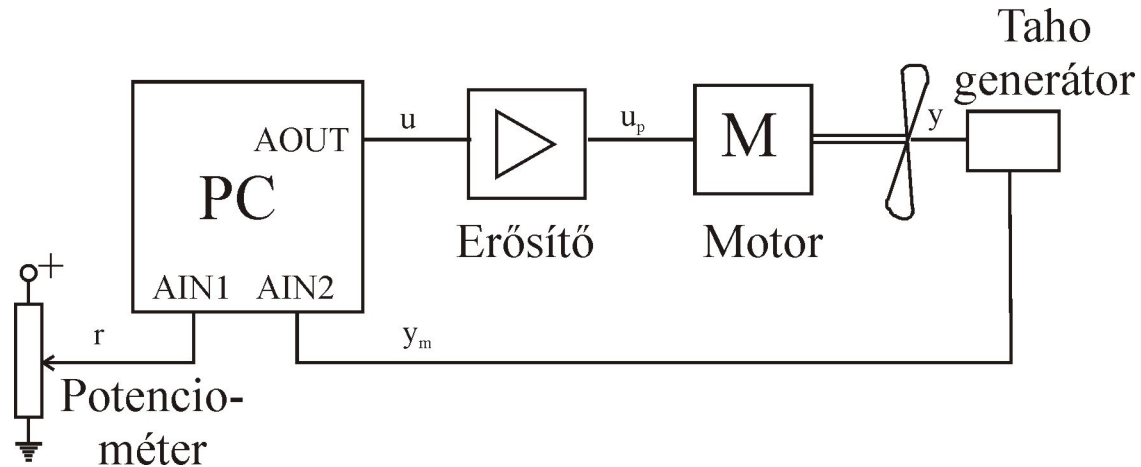
A szabályozókör elemei:

Folyamat: a ventilátor: bemenet a ventilátort forgató nyomaték és a kimenet a ventilátor sebessége

Érzékelő: tahogenerátor

Beavatkozó: a teljesítményerősítő+motor

Szabályozó: a számítógépen futó irányítószoftver (NEM a számítógép maga!)



2.3 Ábra: Fordulatszám szabályozási hurok

A szabályozókör jelei:

$u$  – a folyamat bemenete – a motor által kifejtett nyomaték

$y$  – a folyamat kimenete – ventilátor fordulatszáma

$y_m$  – mért kimenet – a tahogenerátor által létrehozott, számítógépbe beolvasott analóg feszültség

$r$  – a betáplált potenciométerről leolvasott, számítógépbe beolvasott feszültség

$e$  – szabályozási hiba – a jel a számítógépes irányítószoftverben számítható a (2.4) összefüggés alapján

$u$  – az irányító szoftver által kiszámolt, a számítógép analóg kimenetén kiadott jel, ami a teljesítményerősítő bemenete

$d$  – zavaró jel – mérési zajok, a ventilátor csapágyaiban fellépő nem viszkózus súrlódási komponensek, stb.

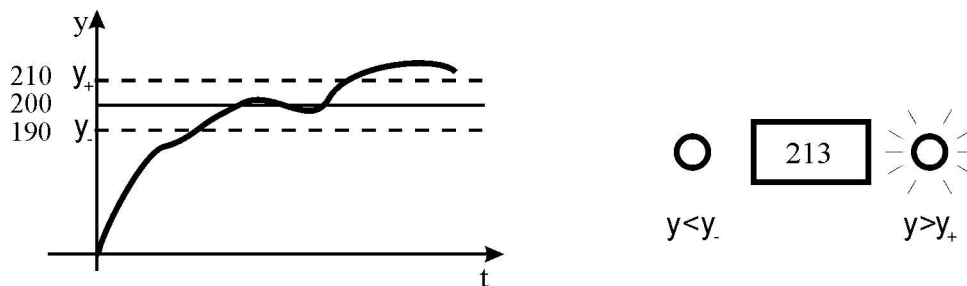
## 2.2. Kézi szabályozás

Kézi szabályozás esetében az ipari folyamatot kezelő, felügyelő személy végezi el az irányítási feladatot. Ebben az esetben a felügyelő személynek egyrészt információja kell, hogy legyen a folyamat kimenetéről, másrészt lehetősége kell legyen arra, hogy a

beavatkozó bemenetét módosítsa. Ebben az esetben is megvalósul a visszacsatolás, csak a kezelőszemélyen keresztül.

Az ipari folyamatok esetén mindig meg kell hagyni a lehetőséget a kézi szabályozásra is, arra az esetre, hogy ha a folyamatot irányító számítógépben szoftver- vagy hardverhiba lép fel.

A folyamat kimenetét általában numerikusan jelezzük ki, a kimenet értékét SI mértékegységekben kifejezve. Lassú folyamatok esetén hasznos a grafikus kijelzés is (a folyamat kimenetének ábrázolása idő függvényében). Célszerű még figyelmeztető kijelzéseket is alkalmazni: például, ha a folyamat kimenete meghalad egy felső határt, vagy kilép egy előre meghatározott toleranciasávból, ami már veszélyes lehet a szabályozó kör helyes működésére, akusztikus jelzést vagy fényjelzést ad az irányítást megvalósító eszköz (lásd 2.4. Ábra).



2.4 Ábra: A szabályozott jellemző (irányított folyamat kimenetének) kijelzése

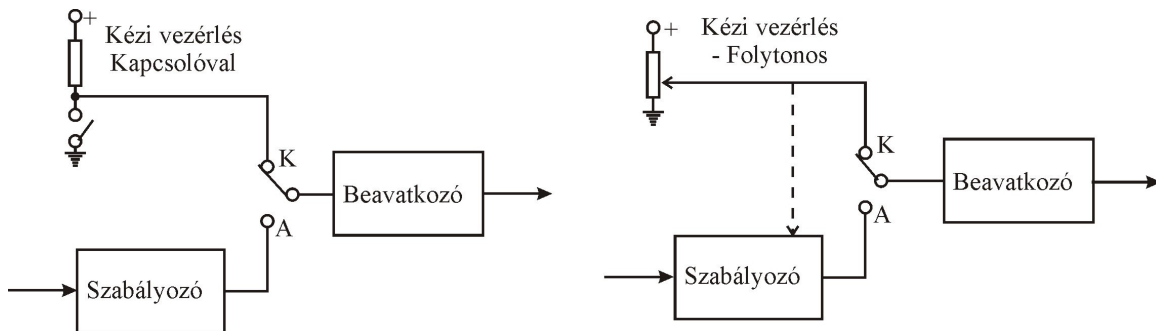
A kezelőszemélyzet beavatkozása:

A folyamatot felügyelő személynek mindig lehetősége kell legyen arra, hogy vészhelyzet esetén leállítsa a szabályozást. Ebben az esetben az irányítási algoritmus inaktívvá válik, a beavatkozó bemenetére nullaértékű jel kerül.

Ha a kezelőszemélyzet irányítani és nem megállítani kívánja a folyamatot, lehetősége kell legyen a beavatkozó bemenetének folytonos módosítására. Ez történhet egy kis energiájú folytonos jellel. (Például a 2.1 példában betáplált potenciométerrel változtatjuk a teljesítményerősítő feszültség bemenetét.) Másik lehetőség, hogy kapcsoló üzemmódu beavatkozást végzünk, ha a beavatkozó bemenetére lehet nemfolytonos jelet adni. (Például egy ellenállással fűtött kemencében az ellenállásra rákapcsoljuk a feszültséget, ha növelni szeretnénk a hőmérsékletet vagy lekapcsoljuk róla a feszültséget, ha azt szeretnénk, hogy csökkenjen a kemence hőmérséklete.) A két megvalósítás elvi rajza a 2.5 Ábrán látható.

A kezelő személy által végzett szabályozás (Kézi (*K*) szabályozás) és a szabályozó elem által végzett szabályozás (Automata (*A*) szabályozás) közötti átkapcsolást kétállású kapcsolóval oldhatjuk meg. Problémát jelenthet, hogy az átkapcsoláskor ugrások jelennek meg a vezérlőjelben. A kézi üzemmódban a kezelő személy a beavatkozó bemenetére adott feszültség szint és a szabályozó által meghatározott feszültség szint különbözhet. Ez az átkapcsolás pillanatában a beavatkozó bemenetén a jel ugrásszerű megváltozásában nyilvánul meg, ami akár a beavatkozó károsodásához is vezethet. Ennek elkerülésére a Kézi - Automata átkapcsoláskor azt a stratégiát alkalmazhatjuk, hogy a szabályozó visszaolvassa a kézi üzemmódban kiadott beavatkozó jelet és átkapcsoláskor a

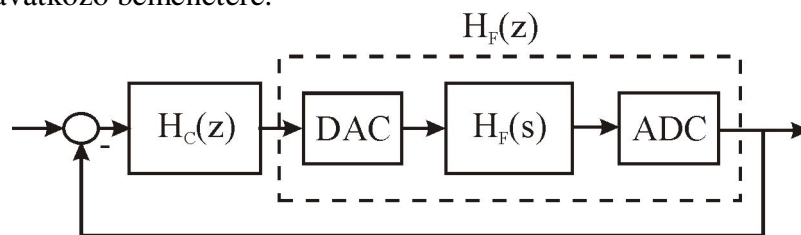
szabályozó ettől a kiinduló értéktől módosítja tovább folyamatosan lesimítva az ugrásokat az irányítási feladatnak megfelelően.



2.5 Ábra: Lehetőségek kézi szabályozás megvalósítására

### 2.3. Mintavételes szabályozások megvalósítása

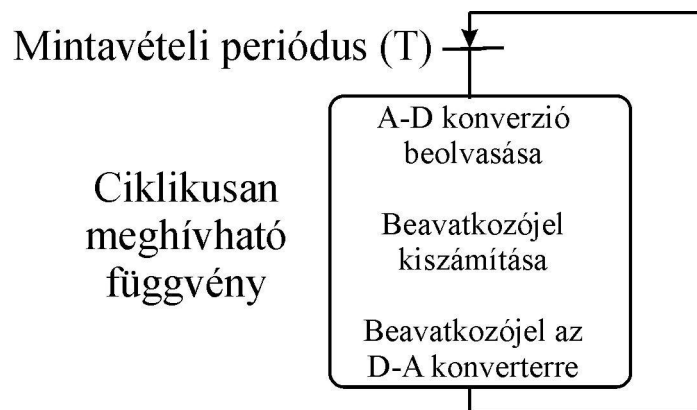
Az irányítási algoritmusok megvalósításának legelterjedtebb módja a szoftveres implementálás. Az irányítási algoritmust megvalósító függvények, eljárások általában egy nagyobb, ipari folyamatok felügyeletére, irányítására, monitorizálására kialakított szoftver része. Az irányítási algoritmus a folyamat mért kimenete és az előírt kimenet (alapjel) alapján számítja ki a beavatkozó jelet. A folyamat kimenetét analóg-digitális átalakítón keresztül olvassuk be. Az alapjelet ugyancsak analóg-digitális átalakítón keresztül olvashatjuk be vagy megadhatjuk számítógépes periférián keresztül is (például billentyűzet segítségével). A második esetben nincs szükség analóg-digitális átalakításra az alapjel esetében. A kiszámolt beavatkozó jelet digitális-analóg átalakítón keresztül küldjük a beavatkozó bemenetére.



2.6 Ábra: Mintavételes szabályozási rendszer kialakítása

A mintavételes megvalósításnál szükségünk van egy ciklikusan (konstans periódusonként) meghívható függvényre. A konstans periódust *mintavételi periódus*nak nevezzük. Az ipari szoftver főszála ciklikusan megszakítódik és végrehajtódik az irányítási algoritmust megvalósító eljárás. Az eljárás főbb részei a következők:

- a folyamat bemenetének beolvasása analóg-digitális átalakítóról.
- a beavatkozó jel kiszámítása az alapjel és a folyamat bemenetének függvényében.
- a beavatkozó jel kiküldése a digitális-analóg átalakítón keresztül.



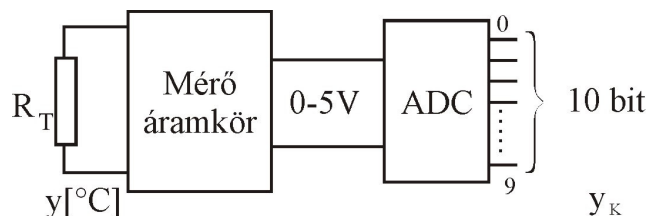
2.7 Ábra: Mintavételes szabályozás megvalósítása

A mintavételes kialakítás felvet néhány speciális problémát mint: az analóg-digitális, digitális-analóg átalakítás miatti kerekítési hibák, a mért jelek digitális szűrése, kalibrációja, Shannon tétel, a mintavételi periódus meghatározása.

### 2.3.1. Analóg-digitális, digitális analóg átalakítás

Az analóg digitális átalakítás miatt kvantálási hibára számíthatunk a mért jel beolvasásánál, a mérés pontossága az átalakító felbontásától függ. Ha az átalakító felbontása  $N$ , a mért jel  $y_{MIN}$ ,  $y_{MAX}$  tartományban van, akkor az elérhető mérési pontosság  $(y_{MIN} - y_{MAX})/2^N$ , ha a mérés lineáris. Ezért az analóg-digitális átalakítót úgy kell megválasztani, hogy nagyobb pontossággal tudjunk mérni, mint az elvárt szabályozási pontosság. A digitális-analóg átalakító (a beavatkozó jel kiküldése) felbontásának meghatározásánál ugyancsak szabályozási pontosságot kell figyelembe venni. Ha nagy felbontással tudjuk csak módosítani a beavatkozó jelet, a folyamat kimenete is nagy felbontással fog változni. A napjainkban ipari alkalmazásokra elterjedt analóg-digitális, digitális-analóg konverterek felbontása tipikusan 10-16 bit között van, de ennél nagyobb felbontású átalakítók is kaphatóak elérhető áron.

**2.2 Példa:** A 2.8. Ábrán látható hőmérsékletmérő 0 - 1000 °C tartományban méri a hőmérsékletet. Az alkalmazott mérőáramkör a 0 °C esetében 0 V feszültséget, 1000 °C esetében 5 V feszültséget szolgáltat a kimenetén. Az alkalmazott analóg-digitális átalakító bemeneti tartománya 0 - 5V, a felbontása 10 bit. Mennyi a mérés pontossága?



2.8 Ábra: Hőmérsékletmérés kialakítása mintavételes szabályozáshoz

A 10 bites pontosság az jelenti, hogy az átalakító kimenete  $0-2^{10}$ , azaz  $0-1024$  között változhat. Így a mérés pontossága  $(1000-0)/1024 \cong 1 \text{ }^\circ\text{C}$ . Ez azt jelenti, hogy az adott mérőrendszerrel  $1 \text{ }^\circ\text{C}$  -nál pontosabban nem tudunk hőmérsékletet szabályozni. Vegyük észre, hogy a jelkondicionáló áramkör kimeneti feszültségtartománya (az átalakító bemeneti feszültségtartománya) nem befolyásolja a mérés pontosságát.

### 2.3.2. Mérések kalibrálása

Az analóg-digitális átalakítóról beolvasott érték egy egész szám  $0 \dots 2^N$  tartományban, ahol  $N$  az átalakító pontossága. A beolvasott jelet mindig át kell alakítani az irányítási algoritmus által feldolgozható, valamint a kezelőszemélyzet által értelmezhető értékre. Kalibrálás alatt az analóg-digitális átalakítóról beolvasott érték transzformálását értjük szabványos (pl.  $SI$ ) mértékegységekkel jellemezhető tartományba.

Amennyiben a mérési tartományban az érték lineárisan változik, a kalibráció  $y_M = a_M y_{AD} + b_M$  lineáris transzformációt jelent. Ha a mérés tartománya  $y_{MIN} \dots y_{MAX}$  és az átalakító pontossága  $N$  ( $y_{MIN}$  bemenetre  $0$ -t,  $y_{MAX}$  bemenetre  $2^N$ -et olvasunk be), akkor:

$$\begin{aligned} y_M &= a_M y_{AD} + b_M \\ a_M &= \frac{y_{MAX} - y_{MIN}}{2^N} \\ b_M &= y_{MIN} \end{aligned} \quad (2.6)$$

Amennyiben a mérés nem változik lineárisan a mérési tartományban, a kalibrációhoz nem elsőfokú, hanem magasabb fokú polinomot alkalmazhatunk. A kalibráció előtt a beolvasott érték még egész előjel nélküli szám, a kalibráció után már célszerű lebegőpontos ábrázolást alkalmazni.

### 2.3.3. A jelek szűrése

Az ipari mérések esetén mindig kell számolni mérési zajokkal. A mérési zajok miatt az irányítási algoritmus bemenetén 'torzítva' jelenik meg a folyamat kimenete, ezért az irányítás minősége romolhat.

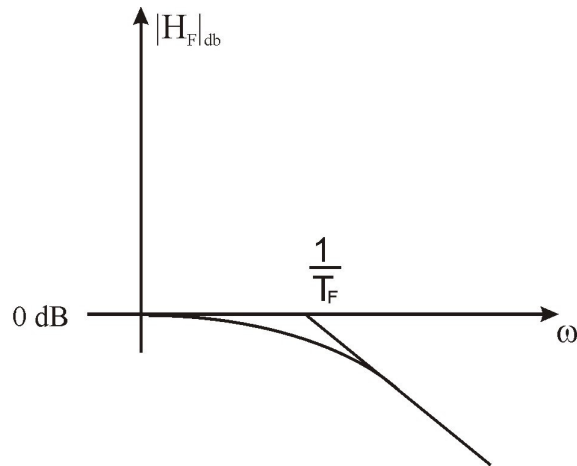
A mérési zajok általában nagy frekvencián jelentkeznek, ezért az alkalmazott szűrő aluláteresztő szűrő kell, hogy legyen. A szűrő megválasztásnál becslésünk kell legyen arról, hogy a mérési zajok milyen frekvenciatartományban jelentkeznek. Irányítástechnikai feladatokban általában elégséges egyszerű struktúrájú, kis fókuszszámú szűrőket alkalmazni.

Lassú irányítások (például hőmérsékletszabályozás) esetében szóba jöhetnek véges impulzusválaszú (FIR – Finite Impulse Response) *átlagoló szűrők*. Ebben az esetben a szűrő kimenete az utolsó  $N$  mérés átlaga:



$$y_{Fk} = \sum_{i=0}^{N-1} \frac{y_{k-i}}{N} \quad (2.7)$$

$y_{Fk}$  a szűrő kimenete,  $y_k$  a mért érték a  $k$ -ik mintavételben. Az  $N$  értékével állíthatjuk be. Minél alacsonyabb frekvencián jelentkeznek a zajok, annál nagyobb  $N$ -t szükséges választani. (lásd 2.9 Ábra)



**2.9 Ábra: Az elsőfokú stabil rendszer amplitúdó menete**

Gyors irányításoknál (például pozíciószabályozásnál az inkrementális adóról beolvasott frekvenciaérték szűrése) alkalmazhatunk *rekurzív szűrőket*. Sok esetben elégséges egy elsőfokú rendszer mintavételes megvalósítása. Közismert, hogy a stabil elsőfokú rendszer aluláteresztő szűrő.

$$H_F(s) = \frac{Y_F(s)}{Y(s)} = \frac{1}{T_F s + 1} \quad (2.8)$$

$T_F$  a rendszer időállandója. A szűrő erősítése egységnyinek választható. Minél alacsonyabb frekvencián jelentkeznek a mérési zajok, annál nagyobbra kell választani  $T_F$  értékét. A szűrő mintavételes megvalósításánál kiindulhatunk az őt leíró differenciálegyenlet alakjából, amire a hátrtartó differenciák közelítést alkalmazunk:

$$\begin{aligned} T_F \frac{dy_F(t)}{dt} + y_F(t) &= y(t) \\ T_F \frac{y_{Fk} - y_{Fk-1}}{T} + y_{Fk} &= y_k \\ y_{Fk} &= \frac{T_F}{T_F + T} \cdot y_{Fk-1} + \frac{T}{T_F + T} \cdot y_k \end{aligned} \quad (2.9)$$

$T$  a mintavételi periódust jelöli. A szűrő kimenete az előző mintavételbeli szűrt értéktől is függ, amely biztosítja a meredek változások (ugrások elkerülését). Ha a  $T_F$  időállandó

értéke zéróhoz közeli, akkor a szűrt kimenet pontosan követi az  $y$  értékét, a zajokat is átengedve. Nagy  $T_F$  értékek esetén jó zajszűrésre számíthatunk, de a szűrő kimenete jelentősen lemarad az  $y$ -hoz képest.

### 2.3.4. Mintavételi periódus

A mintavételi periódus megválasztása kritikus a szabályozás helyes működésének biztosításához. Az irányítási feladatokban *fix, konstans mintavételi periódust* alkalmazunk. Mivel a mintavételi periódus gyakoriságával olvassuk be a folyamat kimenetét, ami alapján az irányítási algoritmus kiszámítja a vezérlőjelet, a periódust úgy kell megválasztani, hogy a beolvasott mintavételezett jel tartalmazza a folytonos jel jellegzetességeit. Így a periódus megválasztása minden esetben alkalmazásfüggő. Gyors válaszú rendszerek esetében kis mintavételi periódust választunk, lassú válaszú rendszerek esetében nagyobb mintavételi periódus alkalmazható.

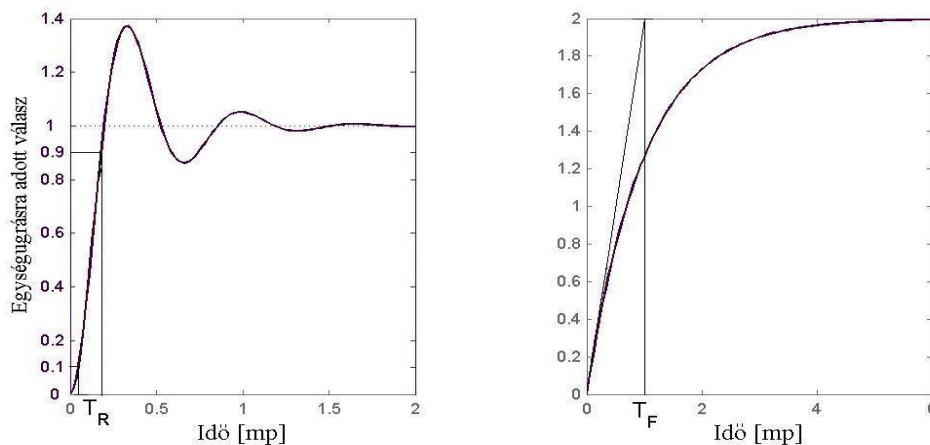
A mintavételi periódus megválasztásánál empirikus, tapasztalati szabályok alkalmazhatóak: amennyiben az irányított rendszer elsőfokú, vagy jól approximálható elsőfokú modellel a periódus megválasztása a rendszer időállandója függvényében történhet:

$$T = (4 \div 10) T_F \quad (2.10)$$

Ha az irányított rendszer viselkedése másodfokú lengőrendszer modelljével közelíthető, akkor a mintavételi periódus az alábbi tartományban történhet:

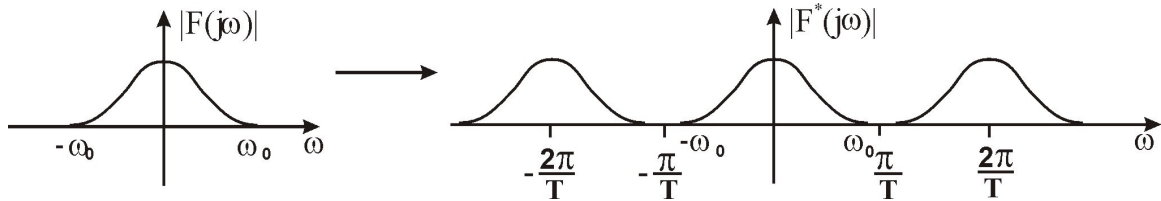
$$T = (4 \div 10) T_R \quad (2.11)$$

$T_R$  (*Rise Time*) azt az időintervallumot jelöli, amely alatt a rendszer egységugrásra adott válasza 10% -ról 90% -ra emelkedik (100% a kimenet az állandósult állapotban).

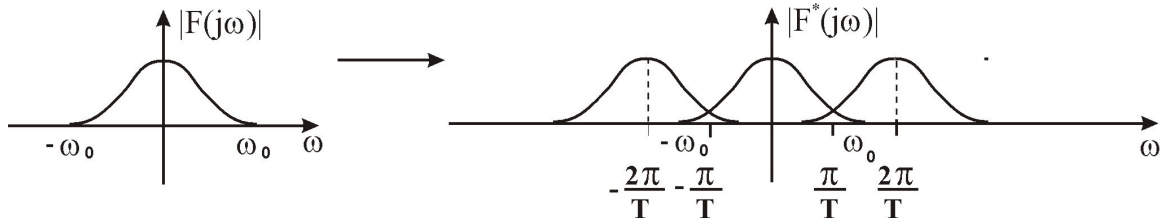


2.10 Ábra: A mintavétel megválasztása különböző rendszerek időállandóinak függvényében

A mintavétel megválasztásánál ugyancsak nagy szerepe van a *Shannon tétel* következményeinek. Közismert, hogy a mintavételezett jel spektruma periodikussá válik, a mintavételezett spektrum  $\pi/T$  periodicitással ismétlődik. Legyen  $F^*(j\omega)$  a mintavételezett  $f(t)$  jel Fourier transzformáltja. Ha a jel spektrumának a határ körfrekvenciáját  $\omega_0$  jelöli, akkor a mintavételezés után a spektrum lehetséges formáit a 2.11 és 2.12 Ábrán láthatjuk.



2.11 Ábra: A mintavételezett jel spektruma ( $\omega_0 < \pi/T$ )



2.12 Ábra: A mintavételezett jel spektruma ( $\omega_0 > \pi/T$ )

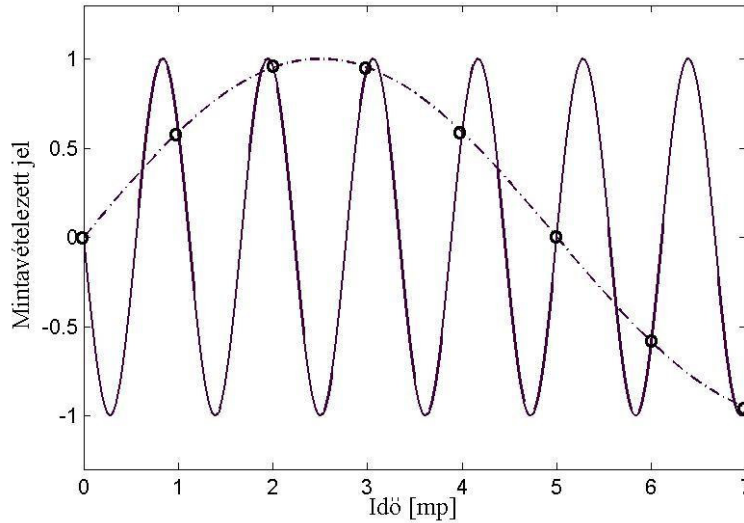
Az első esetben a ( $\omega_0 < \pi/T$ ) az ismétlődő mintavételezett spektrumban az ismétlődő folytonos spektrumelemek jól elkülönülnek egymástól. A második esetben a folytonos spektrumelemek lapolódnak. Ennek az a következménye, hogy a *jel  $\pi/T$ -nél nagyobb frekvenciájú komponensei nem elkülöníthetők* egymástól, tehát a jel e nagyfrekvenciás komponenseiről nincs információnk. Tehát a mintavételi periódus megválasztásának felső határt szab a jel határ körfrekvenciájának inverze. Ahhoz, hogy mintavételezés után a jel rekonstruálható legyen (minden frekvenciakomponensét tökéletesen megkapjuk) az alábbi feltételnek kell teljesülnie:

$$T < \frac{\pi}{\omega_0} \quad (2.12)$$

A valós, mért jeleknek léteznek olyan nagyfrekvenciájú komponensei, amelyre a (2.12) összefüggés nem teljesül, tehát a valós mintavételezés esetén mindig van információvesztés a jel nagy frekvenciás komponenseiről.

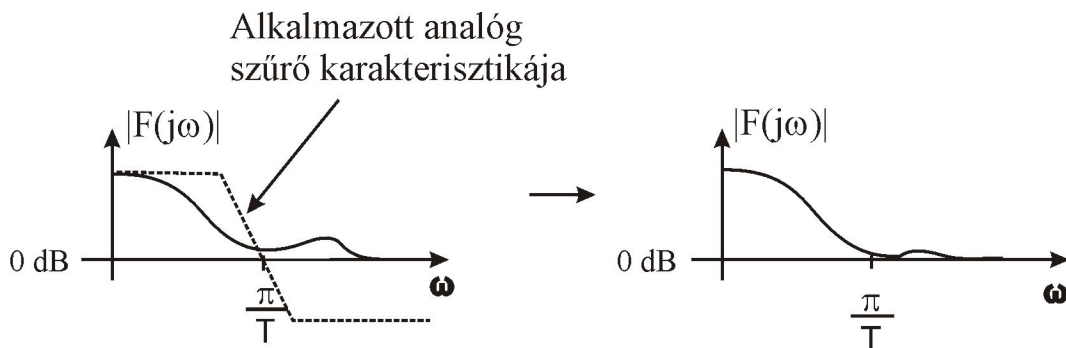
Ugyancsak a Shannon tétellel hozható összefüggésbe az *alulmintavételezés* (downsampling) jelensége. Ha egy  $\omega_0$  körfrekvenciájú jelet mintavételezünk úgy, hogy nem teljesül a (2.12) összefüggés, a jel nem lesz rekonstruálható, tehát mintavételezés után nem 'felismerhető'. Ugyanakkor a nagyfrekvenciás mintavételezett jelnek egy

alacsonyfrekvenciás áljele (*alias*-a) megjelenhet a mintavételezés után. Ez egy alacsony frekvenciájú jel, amely nem tartalmazza a folytonos jel jellegzetességeit, tehát torzított, nem valós képet ad a beolvasott jelről. A 2.13 Ábrán egy  $0.9 [Hz]$  frekvenciájú jel amelyet  $T=1 [sec]$  periódussal mintavételezünk, tehát a 2.12 összefüggés nem teljesül. A jelnek megjelenik  $0.1 [Hz]$  frekvenciájú 'alias'-a.



2.13 Ábra: Alulmintavételezés

A jelenség elkerüléséhez nem szabad megengedni, hogy  $\pi/T$ -nél nagyobb frekvenciájú komponenseket mintavételezzünk. Ezért szükséges a jel analóg szűrése. A folytonos jelből *mintavételezés előtt analóg aluláteresztő szűrő alkalmazásával* a  $\pi/T$ -nél nagyobb frekvenciájú komponenseket el kell távolítani. Az analóg szűrőt a mérőáramkörben kialakíthatjuk passzív áramköri elemekből (ellenállás, kondenzátor) vagy igényesebb alkalmazásoknál műveleti erősítőkkel megvalósított aktív szűrőket alkalmazhatunk.



2.14 Ábra: A mintavételezendő jel szűrése

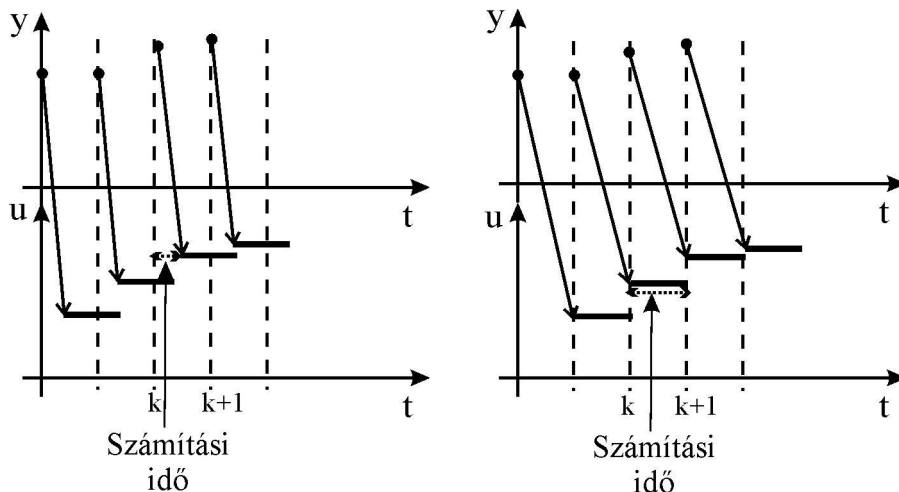
### 2.3.5. Számítási pontosság, számítási idő

A beolvasott jel, ami alapján a beavatkozó jelet számítjuk, mindig egy analóg – digitális átalakítás eredménye. Ezért a beolvasott érték számábrázolása egész számként történik. A beavatkozó jel minél pontosabb meghatározása érdekében az irányítási algoritmusban a számításokat lebegőpontos formában célszerű elvégezni. Így a kalibráció során a beolvasott jelet (a folyamat kimenetének aktuális értékét) már lebegőpontos formában célszerű előállítani. A beavatkozó jel digitális – analóg átalakítón keresztül éri el a folyamatot. A digitális – analóg átalakító bemenete tipikusan egész típusú számot vár. Ezért a számítás lebegőpontos eredményét vissza kell alakítani egész típusú értéké.

Ha a digitális-analóg átalakító felbontása  $N$ , a számítás eredménye (a beavatkozó jel értéke)  $0 \dots 2^N$  közé kell, hogy essen. Ellenkező esetben a beavatkozó jel értékét korlátozni kell ebbe a tartományba.

A mintavétel alatt elvégzendő műveletek (lásd 2.7 Ábra) időkölsége (az AD átalakítás ideje + a számítási idő + a DA átalakítás ideje) kisebb kell legyen, mint a mintavételi periódus. A beavatkozó jel kiküldésénél két stratégiát alkalmazhatunk (lásd 2.15. Ábra):

1. Ha a számítási időkölség számottevően kisebb, mint a mintavételi periódus, a  $k$ -ik mintavételben kiszámított beavatkozó jelet a  $k$ -ik mintavételi periódus végén kiküldjük.
2. Ha a számítási időkölség összemérhető a mintavételi periódussal, a  $k$ -ik mintavételben kiszámított beavatkozó jelet a  $k+1$ -ik mintavételi periódus elején küldjük ki. Ebben az esetben a számítási időkölségek miatt az irányítási rendszerben megjelenik egy mintavételnyi holtidő, amelyet módosíthatja az alkalmazott szabályozó beállításait, mivel figyelembe kell venni az egy mintavételnyi késleltetést.



2.15 Ábra: A kiszámított beavatkozó jel kiküldésének lehetőségei

### 2.3.6. Az irányítás mellett fellépő feladatok

Az irányítást megvalósító számítógépes programnak a beavatkozó jel kiszámítása mellett több feladatot is el kell végeznie. A feladatokat két csoportba sorolhatjuk:

1. Nagy prioritású feladatok: Tipikusan biztonsági problémák, amelyek prioritása nagyobb a beavatkozó jel számításánál, megszakíthatják a vezérlőjel számolását. Például fordulatszám szabályozás esetén, ha a beavatkozó szervként használt motor károsodik, a motor védelme ezt egy digitális bemeneten keresztül jelzi az irányítást megvalósító számítógépes programnak. Ebben az esetben az irányítási algoritmus a beavatkozó jel számítását nem kell elvégezze, a futása felfüggesztődik.

2. Kis prioritású feladatok: Ezek a feladatok ugyancsak párhuzamosan futnak az irányítást megvalósító ciklikusan meghívható függvénnyel, de nem szakíthatják meg az a beavatkozó jel számítását. Ilyen feladatok:

- Megjelenítés (monitorizálás) - a beolvasott, kiküldött, számított jelek numerikus, grafikus kijelzése.

- Naplózás – a rendszer viselkedésének (beolvasott, kiküldött, számított jelek) archiválása adattárolókra az utólagos feldolgozás érdekében (irányítás kiértékelése, rendszerhibák megtalálása). A hasznos adatok elmentése mellett általában tároljuk a mentés idejét is (óra, dátum) ezért az eljárást naplózásnak is nevezzük.