



EFOP-3.5.1-16-2017-00017

**„NYE-DUÁL- Új utakon a duális felsőoktatással a Nyíregyházi Egyetemen,  
az Északkelet-Magyarországi térség felemelkedéséért”**

# Automatizálási alapismeretek

## II. rész

### Szabályozástechnika és számítógépes folyamatirányítás

Szerkesztette:  
Ferenczi Ildikó

A tananyag elkészítését a „NYE-DUÁL- Új utakon a duális felsőoktatással a Nyíregyházi Egyetemen, az Északkelet-Magyarországi térség felemelkedéséért” az EFOP-3.5.1-16-2017-00017 számú projekt támogatta. A projekt az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósul meg.

**SZÉCHENYI** 2020



MAGYARORSZÁG  
KORMÁNYA

**Európai Unió**  
Európai Szociális  
Alap



**BEFEKTETÉS A JÖVŐBE**



**EFOP-3.5.1-16-2017-00017**

**„NYE-DUÁL- Új utakon a duális felsőoktatással a Nyíregyházi Egyetemen,  
az Északkelet-Magyarországi térség felemelkedéséért”**

---

**Szerkesztő:  
Ferenczi Ildikó**

**Szerző:  
Ferenczi Ildikó**

**Lektor:  
Dr. Ferenczi István**

Kézirat lezárva: 2019. 09. 15.

**ISBN 978 615 6032 02 7**

**Kiadja a  
NYÍREGYHÁZI EGYETEM**

**2019**



EFOP-3.5.1-16-2017-00017

**„NYE-DUÁL- Új utakon a duális felsőoktatással a Nyíregyházi Egyetemen,  
az Északkelet-Magyarországi térség felemelkedéséért”**

## TARTALOMJEGYZÉK

1. Bevezetés a szabályozástechnikába, alapfogalmak.....	5
1.1. A szabályozási kör részei .....	6
1.2. Új fogalmak ismertetése .....	7
1.3. Egy villamos motor fordulatszám szabályozása .....	8
1.4. A szabályozások osztályozása [1] .....	10
1.4.1. Kétállású hőmérsékletszabályozó működési elve .....	12
1.4.2. Egyhurkos és kaszkádszabályozás .....	13
2. A szabályozások vizsgálata .....	15
2.1. A szabályozási tag fogalma .....	15
2.1.1. Laplace transzformáció [2].....	16
2.1.2. A tagok csoportosítása .....	17
2.2. Hatásvázlat, a tagok közötti kapcsolatok .....	17
2.2.1. Feladatok tagokkal .....	18
2.3. Vizsgálójelek.....	19
2.4. Arányos (proporcionális) tagok.....	20
2.4.1. Energiatároló nélküli arányos tag.....	20
2.4.2. Egytárolós tag.....	23
2.4.3. Kéttárolós tag .....	25
2.4.4. Lengő tag.....	26
2.4.5. Holtidős tag .....	27
2.5. Integráló tag.....	27
2.6. Differenciáló tag.....	30
2.7. Összetett szabályozók .....	32
2.7.1. PI szabályozó.....	32
2.7.2. A P tag kompenzálása [1].....	34
2.7.3. PD szabályozó .....	35
2.7.4. PID szabályozó.....	36
2.7.5. A szabályozók hangolása [1].....	38
2.7.6. Egy PID szabályozó gyakorlati megvalósítása .....	39
3. Számítógépes folyamatirányítás.....	40
3.1. Alapfogalmak .....	40
3.2. Valós idejű (Real-Time) rendszerek .....	41
3.3. Intelligens (smart) távadók.....	43
3.3.1. D-generációs távadók.....	43
3.3.2. Kommunikáció a távadóval. A HART protokoll működése [7] .....	44
3.3.3. E- és F-generációs távadók .....	46
3.4. Digitális PID szabályozók [8] .....	47
3.4. Mérésadatgyűjtő rendszerek.....	48
3.4.1. Korszerű mérésadatgyűjtő rendszerek. NI DAC rendszer .....	49
3.5. SCADA rendszerek .....	51
3.6. OPC szerver (Open Platform Communication) [22] .....	54
3.7. Osztott intelligenciájú (DCS) rendszerek.....	56
4. Valós idejű ipari Ethernet rendszerek .....	59
4.1. Az irodai Ethernet alapjai [11] .....	59
4.1.1. OSI és TCP/IP modellek .....	60



EFOP-3.5.1-16-2017-00017

**„NYE-DUÁL- Új utakon a duális felsőoktatással a Nyíregyházi Egyetemen,  
az Északkelet-Magyarországi térség felemelkedéséért”**

---

4.1.2. A CSMA/CD protokoll működése .....	61
4.1.3. Az IEEE 802.3 Ethernet keret felépítése.....	62
4.1.4. Követelmények egy ipari Ethernettel szemben .....	63
4.2. Fontosabb ipari Ethernet rendszerek [13] .....	64
4.2.1. Ethernet Powerlink .....	65
4.2.2. PROFINET .....	67
4.2.3. EtherCAT .....	70
5. Robottechnika és robotizáció .....	73
5.1. Ipari robotok .....	74
5.1.1. Az ipari robotok szerkezete [1] .....	74
5.1.2. A robotok hajtása [26] .....	78
5.1.3. A robotok irányítórendszere .....	79
5.1.4. A robotok programozása .....	81
6. IoT az iparban. Ipar 4.0 .....	82
6.1. Az Ipar 4.0 fontosabb területei .....	83
6.2. Prediktív diagnosztika (adatközpontú karbantartás) .....	84
Felhasznált szakirodalom .....	85

## 1. BEVEZETÉS A SZABÁLYOZÁSTECHNIKÁBA, ALAPFOGALMAK

A korszerű gépipari technológiák berendezései között ma már szinte kivétel nélkül találkozunk a szabályozástechnika eszközeivel. Az ipari folyamatok megkövetelik, hogy a termelési körülmények mindig a legoptimálisabb körülmények között valósuljanak meg, azaz a különböző termelési eszközök működése a tervezett értékeknek (hőmérséklet, nyomás, anyagáram, stb.) megfelelően alakuljon. A műszaki fejlődés kezdeti szakaszában ezeket a viszonyokat a technológiai berendezések (esztergapad, prés gép, stb.) mellett álló kezelő személy kézi beavatkozásokkal igyekezett kedvező értéken tartani. A múlt század közepén felgyorsult technikai fejlődés nagymértékben hozzájárult az emberi tevékenység elektronikai és informatikai eszközökkel történő helyettesítéséhez. A termelési folyamatokba olyan irányító eszközöket illesztettek, amelyek segítségével a szabályozók vették át az emberi felügyeleti szerepet. A fejlődés ma is rohamos. A kezelőnek most az a feladata, hogy a megfelelő termelési programot töltsse be az irányító rendszerbe és állítsa be a termelési folyamat paramétereit.

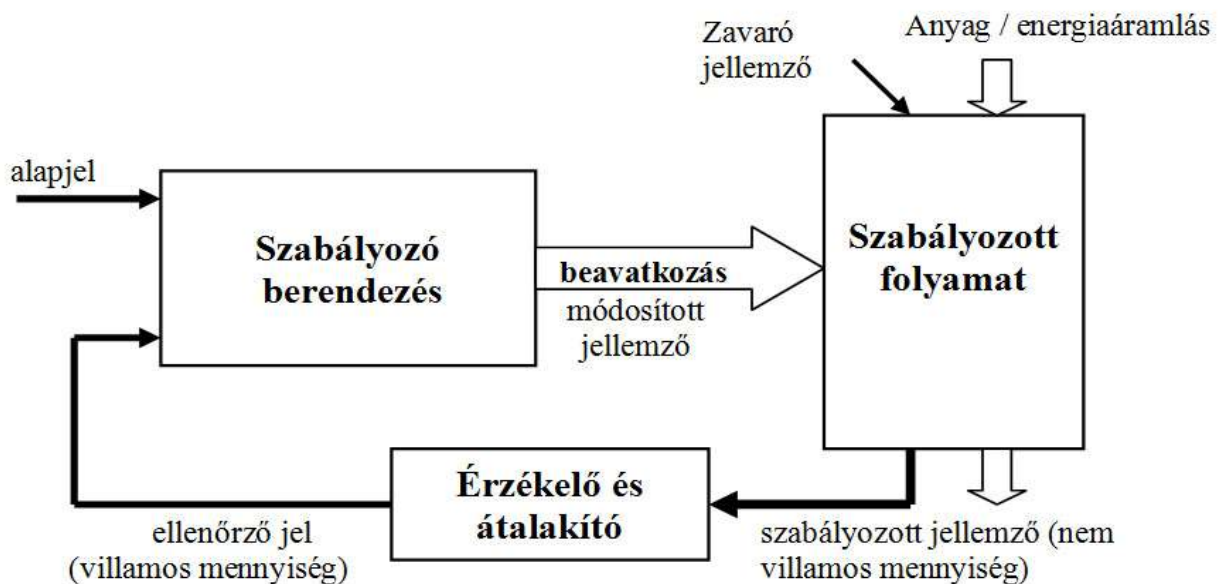
A korszerű gyártástechnológiai folyamatokat tervező és kiszolgáló gépészmérnöktől ma már elvárható, hogy rendelkezzenek a szabályozás alapvető ismereteivel. A szabályozástechnikában megszerzett alapismeretek birtokában legyenek képesek szakszerűen ismertetni a szakmérnökökkel egy adott technológiai folyamat automatizálási lehetőségei. Értsék meg a szakmérnök kérdéseit, találják meg a közös nyelvet az adott tématerületen, és szakszerűen adják át használható tapasztalataikat, műszaki elgondolásaikat.

A szabályozás kialakítása a technológiai berendezéshez vagyis a *szabályozott szakaszhoz* kell igazodjon. Első lépés tehát ez utóbbi alapos megismerése. A szabályozó eszköz kiválasztása, majd a szakaszhoz illetve a követelményekhez való illesztése, hangolása csak a termelő rendszer alapos megismerése után történik. A szabályozó eszköz a termelőeszköz kiszolgálója.

A *szabályozás* az irányításnak az a formája, amikor folyamatosan vagy időközönként, de legalább egyszer visszajelzést kapunk a folyamat állapotáról. *Zárt hatáslánc* jellemzi (1.1. ábra), ami azt jelenti, hogy a hurok bármely jelének megváltozása ugyanarra a jelre visszahat. Működési elve a *negatív visszacsatoláson* alapszik és minél erősebb ez a visszacsatolás, annál stabilabb a szabályozó

működése. Ugyanis egy jól működő szabályozótól alapvető elvárás, hogy ha valamilyen külső beavatkozás hatására kénytelen kimozdulni egyensúlyi helyzetéből, véges időn belül, automatikusan újra stabil állapotba kerüljön. Ennek érdekében a folyamatra történő hatásgyakorlás előjele bármikor megváltozhat azért, hogy az alapjelnek megfelelő szinten tartsa a folyamatot.

Az 1.1. ábrán egy szabályozás általános struktúráját és fontosabb jeleit láthatjuk. A szabályozás elengedhetetlen része az érzékelő és átalakító szerv, amely információt szolgáltat a szabályozó részére a folyamat állapotáról. (Az érzékelő és átalakító szerv elemeit, szenzorait a tananyag I. részében már bemutattam.)



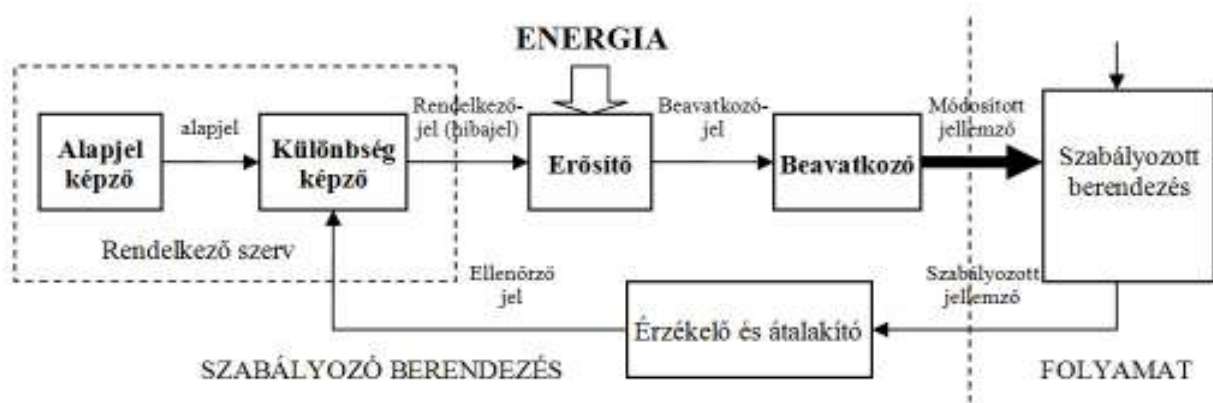
1.1. ábra. A szabályozás elvi megvalósítása zárt hatásláncon keresztül

A szabályozás tárgyát képező folyamat állapota a külső zavaró tényezők hatására bármikor megváltozhat. A szabályozó feladata, hogy ezeket a változásokat megfelelően kompenzálja és a szabályozott folyamatot, mindig az alapjelnek megfelelő értéken tartsa.

### 1.1. A szabályozási kör részei

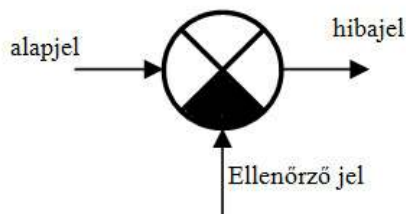
A szabályozó berendezést vagy egyszerűen csak a szabályozót akárcsak a vezérlésnél, itt is szervek alkotják (1.2. ábra). A már az imént említett érzékelő és átalakító szerv mellett, a szabályozásnál is megtalálható a beavatkozó szerv és az erősítő szerv. Ezek működése és funkcióik megegyeznek a vezérlésnél tárgyaltakkal, amelyeket a tananyag első részében már ismerttettem.

A szabályozásnál a vezérléshez képest leginkább a rendelkező szerv módosul. Két fontos részre tagolódik: *alapjel képző szerv* és *különbségképző szerv*. Valójában ez utóbbi látja el a szabályozási feladatokat.



1.2. ábra. A szabályozás szervei és jelei

Szabályozási rendszerek vázlatos ábrázolásánál gyakran a következő (1.3. ábra) szimbólumot használják a különbségképző szerv jelölésére.



1.3. ábra. A különbségképző szerv gyakori jelölési módja

Megjegyzés: A jelölés sötétített része azt jelenti, hogy az ide érkező jel rendszerint ellenkező előjelű az alapjelhez képest.

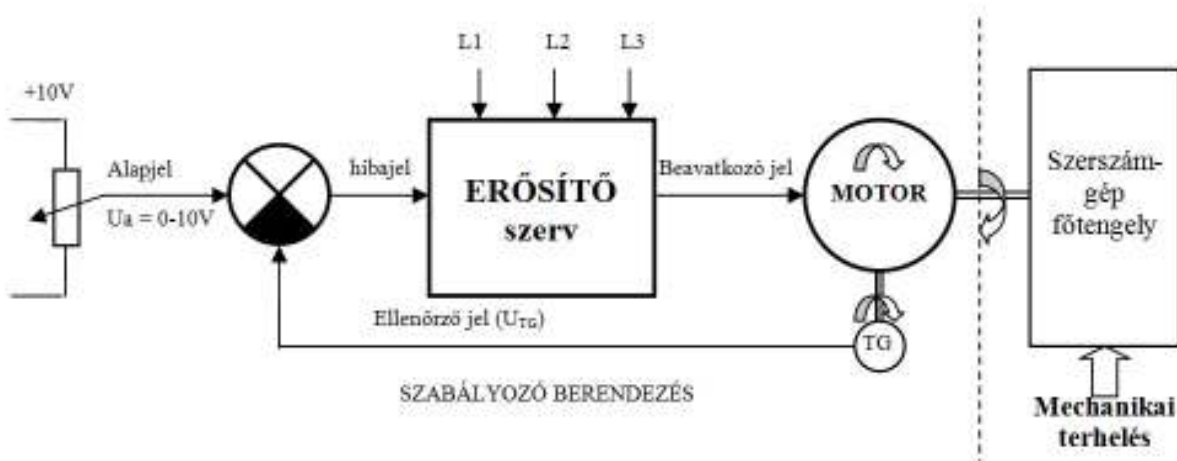
## 1.2. Új fogalmak ismertetése

- Alapjel: - a szabályozott jellemző értékét meghatározó és képviselő jel. Villamos szabályozásnál ez rendszerint egyenfeszültség vagy áram. ( $0 \pm 10 \text{ V}$ ;  $4 - 20 \text{ mA}$ ) Pneumatikus szabályozásnál  $0,2 - 1 \text{ bar}$ . Értékét rendszerint előre megadjuk.
- Két formája van:
  - o Beállítójel: amikor az alapérték üzemszerűen állandó, a szabályozás pedig *értéktartó*.
  - o Vezetőjel: értéke változtatható a kívánt szabályozási hatásnak megfelelően.

- Alapjelképző szerv: feladata az alapjel előállítása. Ez határozza meg a beavatkozás mértékét.
  - o klasszikus szabályozónál ez lehet feszültség stabilizátor, mechanikus közegekben csavar, rugó, stb. Az így létre hozott mennyiség lesz a beállítójel. A termosztáttal, potenciométerrel vagy ikerfémen keresztül biztosított alapjel pedig a vezetőjelet határozza meg.
  - o korszerű szabályozásoknál ezt rendszerint valamilyen intelligens eszköz: mikroprocesszor, mikrokontroller vagy folyamatirányító számítógép szolgáltatja.
- Ellenőrző jel: az érzékelő és átalakító szerv kimenő jele, amely a szabályozott jellemzővel arányos villamos mennyiség.
- Különbségképző szerv: feladata az alapjel és az ellenőrző jel összehasonlítása és a különbséggel arányos és megfelelő előjelű rendelkező jel előállítása. Ezt nevezzük *hibajelnek*. Egyszerű villamos szabályozókban Wheatstone híddal műveleti erősítővel vagy komparátorral megvalósítható. Nem villamos szabályozóknál kétkarú emelő, membrán vagy differenciálmű végzi ezt a feladatot.

### 1.3. Egy villamos motor fordulatszám szabályozása

Ipari irányítórendszerek és technológiai folyamatok egyik gyakori tényezője a villamos motorok fordulatszám szabályozása. A precíz megmunkálási folyamatok megkövetelik egyrészt a szerszámgépek főtengelyének szabályozott fordulatszámát másrészt, pedig a geometriai tengelyek mentén történő elmozdulások pontosságát. Ez csak úgy valósítható meg, ha a tengelyeket hajtó villamos motorok minden pillanatban a megfelelő fordulatszámon járnak. A következő ábra (1.4. ábra) egy klasszikus villanymotor fordulatszám szabályozásának elvi vázlatát mutatja.

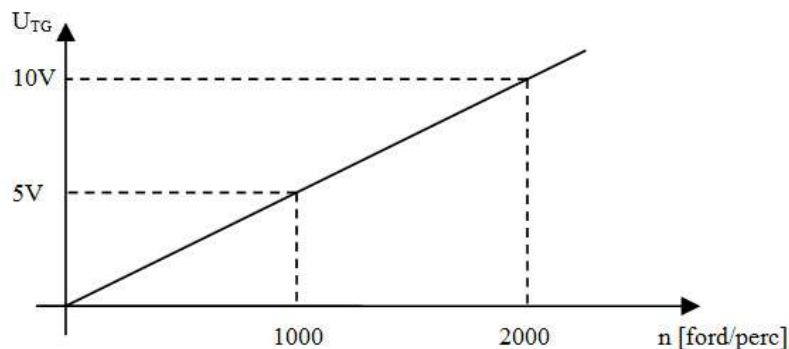


1.4. ábra. Villamos motor fordulatszám szabályozása

A korszerű szerszámgépeknél alapvető követelmény, hogy a főtengely fordulatszáma a megmunkálásnak megfelelően beállítható legyen a kívánt értékre, és ez a fordulatszám ne változzon meg a munkafolyamat közben csak akkor, amikor ez szükségessé válik.

Az 1.4. ábrán a beavatkozó szerv a villamos motor. Ez lehet akár egy egyenáramú szervomotor, de akár háromfázisú aszinkronmotor is. A szabályozás szempontjából ez most lényegtelen. Az erősítő szerv feladata a villamos motor számára szükséges megfelelő nagyságú teljesítmény biztosítása. Az érzékelő és átalakító szerv szerepét a mi esetünkben egy tachogenerátor biztosítja, de ez lehet bármilyen más forgásjeladó (encoder).

A tachogenerátor nem más, mint egy egyszerű, kis teljesítményű egyenáramú generátor, amely közvetlen kapcsolatban van a motor tengelyével, így a motor fordulatszámaival arányos, és ami nagyon fontos lineáris egyenfeszültséget ad (1.5. ábra). A gyártó rendszerint megadja a tachogenerátor paramétereit, vagyis azt, hogy milyen feszültség érték milyen fordulatszámnak felel meg. (Pl.  $n = 1000 \text{ ford/min}$ ;  $U_{TG} = 5 \text{ V}$ )



1.5. ábra. A tachogenerátor feszültségének változása a fordulatszám függvényében

A szabályozó működését hat lépésben vizsgálhatjuk, figyelembe véve mindazokat a körülményeket, amelyek előfordulhatnak a megmunkálás során.

1. Feltételezzük, hogy azt akarjuk, hogy a főtengely  $500 \text{ ford/perc}$  állandó fordulatszámon működjön. Ennek megfelelően választjuk az alapjelet:  $U_a = 2,5 \text{ V}$ . A hibajel kezdetben:  $\varepsilon = U_a - U_{TG} > 0$ , azt eredményezi, hogy az erősítő növeli a beavatkozó jelet, vagyis növekszik a motor fordulatszáma ezáltal az  $U_{TG}$  is növekszik.
2. Egy adott pillanatban:  $U_{TG} > 2,5 \text{ V}$ ; a hibajel  $\varepsilon = U_a - U_{TG} < 0$ ; azaz negatívvá válik és ennek megfelelően az erősítő csökkenti a beavatkozó jelet, amíg az egyensúlyi helyzet be nem áll:  $\varepsilon \approx 0$ . (A hibajel nulla körüli értéke azt jelenti, hogy a szabályozó stabil állapotban van és tartja az alapjel szerint előírt értéket.)

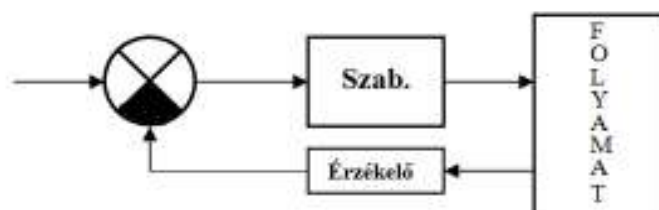
3. Feltételezzük, hogy növekszik a mechanikai terhelés a motor tengelyén. Ez a motor fordulatszám csökkenéséhez vezet, vagyis  $U_{TG}$  csökken. Következik, hogy a hibajel pozitív lesz;  $\varepsilon = U_a - U_{TG} > 0$ ; a beavatkozó pedig növeli a fordulatszámot, míg újra el nem éri a beállított értéket.
4. Ha csökken a terhelés, a motor fordulatszáma megnövekszik, azaz  $U_{TG}$  nő; a hibajel negatív lesz,  $\varepsilon = U_a - U_{TG} < 0$  és a beavatkozó csökkenti a fordulatszámot.
5. Ha növelni akarjuk a fordulatszámot, növeljük az alapjelet;  $U_a$  növekszik; következik, hogy  $\varepsilon = U_a - U_{TG} > 0$ ; az erősítő növeli a beavatkozó jelet, ameddig a kíván fordulatszám be nem áll. ( $\varepsilon \approx 0$ ).
6. Ha csökkenteni akarjuk a fordulatszámot, csökkentjük az alapjelet;  $U_a$  csökken; ennek eredményeként  $\varepsilon = U_a - U_{TG} < 0$ ; az erősítő csökkenti a beavatkozó jelet, ameddig a kíván fordulatszámot el nem éri. ( $\varepsilon \approx 0$ ).

#### 1.4. A szabályozások osztályozása [1]

A szabályozások rendkívül sokfélék és változatosak. Ezért több szempont alapján csoportosítjuk őket:

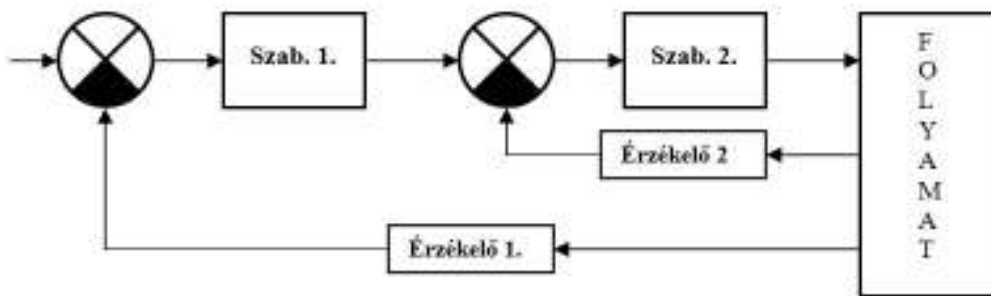
- A rendelkezés létrejötte alapján,
  - Az alapjel időbeli lefolyása szerint,
  - A hatáslánc folytonossága alapján,
  - A szabályozás folyamatossága szerint
  - A rendszer szerkezete alapján.
1. A rendelkezés létrejötte alapján:
    - a. Kézi,
    - b. Önműködő,
    - c. Villamos,
    - d. Pneumatikus,
    - e. Segédenergia nélküli.
  2. Az alapjel időbeli lefolyása szerint:
    - a. Értéktartó: - alapjele állandó. Feladata, hogy a zavaró hatások ellenére az előírt alapjelnek megfelelő szinten tartsa a szabályozott jellemzőt.
    - b. Követő szabályozás: - az alapjel időben változik. Feladata: az alapjelnek megfelelően változtassa a szabályozott jellemzőt is.

- c. Arányszabályozások: - két vagy több jellemző arányát tartja állandó értéken, amikor az egyik véletlenszerűen változik. Pl. a gépjármű üzemanyag és levegő keverékének arányát szabályozó berendezés.
  - d. Menetrendi szabályozás: - az alapjel az idő függvényében előre meghatározott módon változik. Pl. a hőkezelő kemence hőfokszabályozása.
  - e. Szervoszabályozók: - a szabályozott jellemzőt az irányított berendezés valamely elemének térbeli helyzete (elmozdulása, szögelfordulása) határozza meg. Pl. szerszámgépek, robotok helyezetsabályozásai, nagyobb járművek hajók, repülők kormányzása.
3. A hatáslánc jeleinek folytonossága szerint:
- a. Folytonos: - analóg jelű szabályozók. Pl. fordulatszám szabályozás.
  - b. Nem folytonos: - ha a hatáslánc valamely pontján diszkrét (nem folytonos) jel jelenik meg. Nagyon egyszerű de üzembiztos szabályozást valósítanak meg. A szabályozás diszkrét volta azt jelenti, hogy a jelnek csak kétféle értéke van. Pl. termosztátos hőmérsékletszabályozó. Kétállású szabályozó (1.4.1. fejezet).
  - c. Digitális szabályozó: - a korszerű szabályozástechnika alapját képezi. A/D, D/A átalakítók segítségével valósítják meg a kapcsolatot az analóg jellemzőkkel. A mikroprocesszoros, mikrokontrolleres szabályozók tartoznak ebbe a csoportba. Programozott módon működnek.
4. A szabályozás folyamatossága szerint:
- a. Folyamatos működésű: - a hatáslánc tartósan zárt. Pl. fordulatszám szabályozás.
  - b. Időszakos működésű: a hatáslánc csak időnként, rendszerint periodikusan záródik. Ilyen a mintavételes szabályozás, amely a digitális szabályozás alapját képezi. Egy azon intelligens szabályozó, akár több folyamatot is tud irányítani látszólag egyidejűleg párhuzamosan.
5. A rendszer szerkezete alapján:
- a. Egyhurkos: - egyetlen szabályozott jellemző és egyetlen beavatkozó jel van (1.6. ábra).



1.6. ábra. Egyhurkos szabályozás

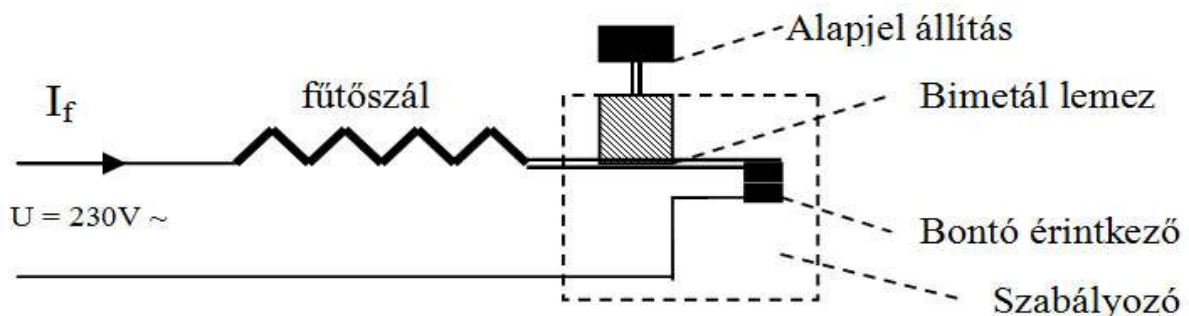
- b. Többhurkos szabályozás: - kaszkádszabályozásnak is hívják, amelynél az egyik szabályozás (belső) a másik szabályozási kör (külső) része (1.7. ábra).



1.7. ábra. Kaszkádszabályozás

#### 1.4.1. Kétállású hőmérsékletszabályozó működési elve

Egyszerű de nagyon elterjedt szabályozó. Ha feltételezzük, hogy a fűtés elektromos táplálású, akkor a fűtőszál soros kapcsolásban van a termosztát bontó érintkezőjével (1.8. ábra).

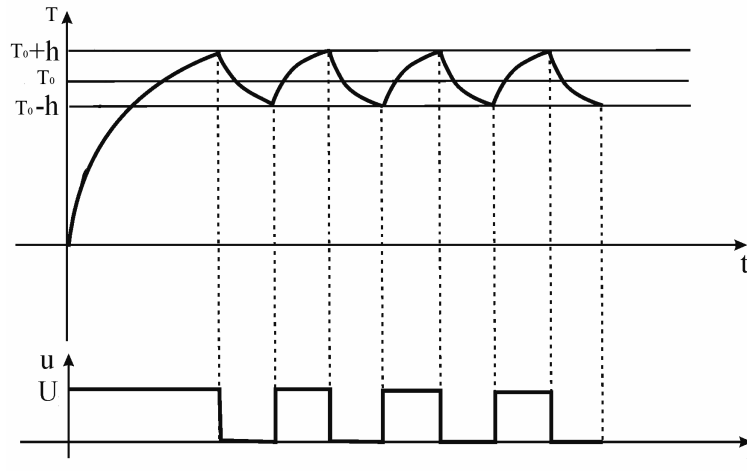


1.8. Kétállású hőmérsékletszabályozó

A kívánt hőmérsékleti érték beállítása (alapjel) a legegyszerűbb esetben mechanikusan történik.  $U$  feszültség hatására a fűtőszálon keresztül  $I_f$  áram halad át. A hőmérséklet addig emelkedik, amíg a beállított értéket el nem éri. Ekkor a bimetal lemez érintkezői bontják a fűtőszál áramkörét. A hőmérséklet egy rövid ideig még növekszik, majd csökkenni kezd. Azért, hogy megakadályozzuk a beállított érték körüli gyakori ki-bekapcsolásokat, a szabályozó  $h$  szélességű histerézissel (holtsávval) rendelkezik. Így a kikapcsolási hőmérséklet értéknél egy alacsonyabb értéken fog újra bekapcsolni.

A kétállású szabályozókat egyszerűségük és üzembiztos működésük miatt kedvelik az iparban, de közvetlen környezetünkben is számos helyen találkozhatunk velük (hőmérsékletszabályozós vasaló,

hűtőszekrény, kapcsolóüzemű tápegységek feszültségszabályozói, stb.). A szabályozó hőmérsékletváltozásának diagramját valamint a fűtőszál feszültségét az 1.9. ábrán láthatjuk.

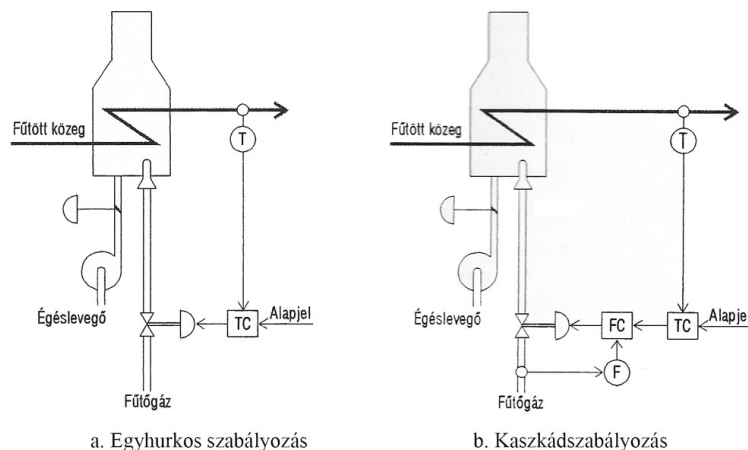


1.9. A szabályozó hőmérsékletváltozása és a fűtőszál feszültség

Alaphelyzetben a termosztát érintkezője zárt állapotú. Bekapcsolás után a fűtőszálon áthaladó áram felmelegíti a közegét. A beállított  $T_0$  hőmérséklet elérése után a holtidőnek köszönhetően a hőmérséklet még emelkedik egy keveset, a termosztát érintkezője bontja a fűtőszál áramát. A hőmérséklet csökkenni kezd, de a holtidőnek köszönhetően nem  $T_0$  értéken fog újra bekapcsolni, hanem kicsivel alacsonyabban. Az érintkező zár, a hőmérséklet újra emelkedik.

#### 1.4.2. Egyhurkos és kaszkádszabályozás

Az 1.10.a. ábrán egy egyhurkos fűtésszabályozás vázlatát látjuk. A gáztüzelésű kemencében a hőcserélőn átáramló folyadékot melegítjük. A folyadék hőmérsékletét a  $T$  távadó érzékeli, és a TC szabályozó a gázvezetékbe szerelt szeleppel módosítja a fűtőgáz mennyiségét a kilépő hőmérséklet állandó értéken tartásához.



1.10. ábra. Egyhurkos és kaszkádszabályozás elvi vázlata (forrás [1])



**EFOP-3.5.1-16-2017-00017**

***„NYE-DUÁL- Új utakon a duális felsőoktatással a Nyíregyházi Egyetemen,  
az Északkelet-Magyarországi térség felemelkedéséért”***

Ha a gáz nyomása valamilyen oknál fogva ingadozik az egyhurkos szabályozási kör erről csak későn, a hőmérséklet változásából szerez tudomást. Ennek a zavaró tényezőnek a hatása nagymértékben csökkenthető, ha érzékeljük a gáz mennyiségét (F jeladó) és egy külső szabályozóval (FC) állandó értéken tartjuk (1.10.b ábra). Figyeljük meg, hogy az FC alapjelét a TC adja, így a két szabályozó sorosan kapcsolódik egymáshoz. FC a belső kör, TC a külső kör.

## 2. A SZABÁLYOZÁSOK VIZSGÁLATA

### 2.1. A szabályozási tag fogalma

Az eddigiek során a szabályozásokat a hatásláncban szereplő szervek alapján vizsgáltuk. Mint ismeretes, a szervek a szabályozás fizikai alkotóelemei, amelyek mindegyikének jól meghatározott szerepe van. Például: érzékelés, erősítés, beavatkozás, stb. A TAG fogalmának bevezetésével a szabályozó egyes részeinek *jelátviteli tulajdonságait* vizsgáljuk. Rendszerint a kimenő és a bemenő jel közötti matematikai kapcsolatot fejezi ki. A kimeneti jel nem hat vissza a bemenetre, vagy ennek visszahatását elhanyagoljuk. Az átvitel lehet lineáris vagy nem lineáris. Egy ilyen kapcsolat kifejezhető egyenlettel, függvénnyel vagy grafikus módon. A tag jelölését egy négyzetbe beírt elnevezéssel vagy szimbólummal jelöljük (2.1. ábra)



2.1. ábra. A tag jelölése

Nagyon gyakran a tag jelátviteli tulajdonságát egy olyan mennyiséggel fejezzük ki, amellyel ha megszorozzuk a bemenő jel változását, a hozzá tartozó kimenő jel változását kapjuk. Ezt a mennyiséget *átviteli tényezőnek* hívjuk. Ez a tag egyik legfontosabb jellemzője. Legyen a tag bemeneti jele  $t = t_i$  időpillanatban  $X_i$ , kimeneti jele ennek megfelelően  $Y_i$ , az átviteli tényezőt a két mennyiség hányadosa adja meg.  $X_i$  és  $Y_i$  állandósult (nyugalmi) állapotú mennyiségek.

$$A = \frac{X_i}{Y_i} \quad (2.1)$$

Amikor a tag időbeli viselkedését is vizsgálni kell, akkor a tag *időállandójával* ( $T, \tau$ ) is számolnunk kell. Összefoglalva tehát a tag legfontosabb jellemzői az *átviteli tényező* és az *időállandó*.

A szabályozástechnikában azonban nemcsak a jelek nyugalmi értékei közötti összefüggéseket kell vizsgálnunk, hanem a jelek időbeli lefolyására is figyelemmel kell legyünk. Lényeges ugyanis,

hogyan az adott szabályozás milyen mértékben reagál a zavaró tényezőkre, vagy a bementi mennyiség változásaira és ezeket mennyi idő alatt, milyen időbeli lefolyással kezeli. A vizsgálatokhoz különböző tulajdonságú *vizsgálójeleket* használunk. A bementeket leggyakrabban ugrásszerűen változtatjuk meg, az erre adott kimentei válaszfüggvényt, pedig *átmeneti függvénynek* nevezzük.

Egy rendszer általános viselkedését, így a tagét is  $n$ -ed rendű differenciálegyenlet írja le.

$$T_n^n \frac{d^n y(t)}{dt^n} + \dots + T_1 \frac{dy(t)}{dt} + y(t) = A \left[ \tau_1 \frac{dx(t)}{dt} + x(t) \right] \quad (2.2)$$

Az egyenlet megoldása idő- és frekvenciatartománybeli viselkedésre ad választ, valamint meghatározhatók az  $A$ ,  $T$ ,  $\tau$  mennyiségek is.

### 2.1.1. Laplace transzformáció [2]

A (2.2.) egyenlet megoldása azonban nem egyszerű. Felhasználva a matematika ide vonatkozó tételeit, egyszerűbb megoldáshoz jutunk, ha az időtartománybeli viselkedést, frekvenciatartománybeli függvényre alakítjuk. Erre alkalmazható a *Laplace transzformáció*.

$$L[y(t)] = \int_0^{+\infty} y(t)e^{-st} dt = Y(s) \quad (2.3)$$

Az így kapott  $Y(s)$  függvényt átviteli függvénynek hívjuk, ahol  $s = j\omega$ , komplex mennyiség,  $\omega$  pedig a körfrekvencia. Az operátornak köszönhetően számos bonyolult, differenciált vagy integrált tartalmazó időtartománybeli függvény a frekvenciatartományban lineáris egyenletté redukálódik. Az alábbiakban néhány példát mutatok be ezzel kapcsolatban.

$$L\left[\frac{d^n y(t)}{dt^n}\right] = s^n Y(s) \quad (2.4)$$

Vagyis az  $n$ -ed rendű differenciál hatványozássá és szorzássá redukálódik. A (2.4) összefüggésben feltételezzük, hogy a kezdeti időpillanat nulla ( $t_0 = 0$ ). Elsőrendű differenciálegyenlet esetében a (2.4) a következőképpen egyszerűsödik:

$$L\left[\frac{dy(t)}{dt}\right] = sY(s) \quad (2.5)$$

A Laplace transzformációt alkalmazva az integrált tartalmazó időtartománybeli egyenletekre, a következő összefüggést kapjuk:

$$L\left[\int_0^t y(t) dt\right] = \frac{1}{s} Y(s) \quad (2.6)$$

Vagyis az integrált tartalmazó időtartománybeli függvény frekvenciatartományban osztássá redukálódik.

A frekvenciatartománybeli összefüggésből bármikor visszatérhetünk időtartománybelibe, alkalmazva az inverz Laplace transzformációt.

$$L^{-1}[Y(s)] = \frac{1}{2\pi} \int_{-j\omega}^{+j\omega} Y(s) e^{-st} ds = y(t) \quad (2.7)$$

### 2.1.2. A tagok csoportosítása

Viselkedésüket tekintve a tagokat három csoportra oszthatjuk:

- arányos tag: - jele  $P$ ; jellemzői:  $A_p$ ;  $T_p$
- integráló tag: - jele:  $I$ ; jellemzői:  $A_I$ ;  $T_I$
- differentiáló tag: jele:  $D$ ; jellemzői:  $A_D$ ;  $T_D$ .

A további fejezetekben ezekkel a tagokkal részletesen is megismerkedhetünk.

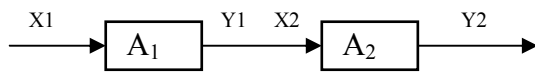
## 2.2. Hatásvázlat, a tagok közötti kapcsolatok

A hatásvázlat tagokból épül fel. A szabályozási kör olyan jelképes rajza, amelyben a tagok közötti kapcsolatokat és jeleket tüntetik fel. Rendszerint belerajzoljuk a tagok kimenő jeleinek időbeli változását a bemenő jel hatására, vagy a használt matematikai összefüggést. A hatásvázlatban alkalmazott szimbólumokat a 2.2. ábrán láthatjuk.



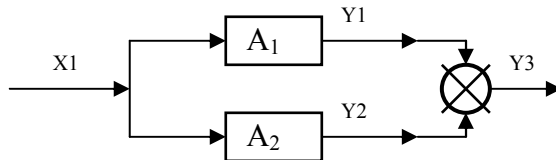
2.2. ábra. Hatásvázlati szimbólumok (forrás [1])

A tagok közötti leggyakoribb alapkapcsolatok a következők: a soros kapcsolat, párhuzamos kapcsolat és negatív visszacsatolás. Ezeket láthatjuk a 2.3. ábrán. Az ábra jobb oldalán a kapott kimeneti összefüggéseket olvashatjuk.



$$Y_2 = A_1 A_2 X_1$$

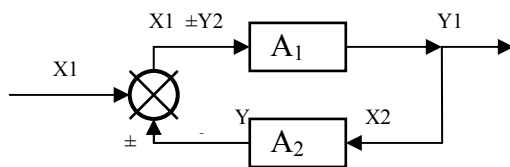
Soros kapcsolat



$$Y_3 = Y_1 + Y_2 = A_1 X_1 + A_2 X_1$$

$$Y_3 = (A_1 + A_2)X_1$$

Párhuzamos kapcsolat



$$Y_1 = \frac{A_1}{1 \pm A_1 A_2} X_1$$

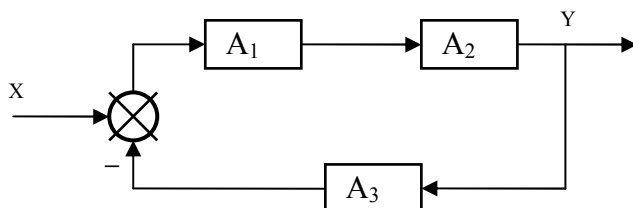
Visszacsatolás (+ negatív, - pozitív visszacsatolás)

2.3. ábra. Műveletek tagokkal

### 2.2.1. Feladatok tagokkal

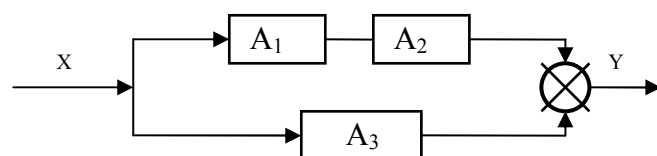
Határozzuk meg az alábbi hatásvázlatok kimeneti mennyiségeit!

#### 1. feladat



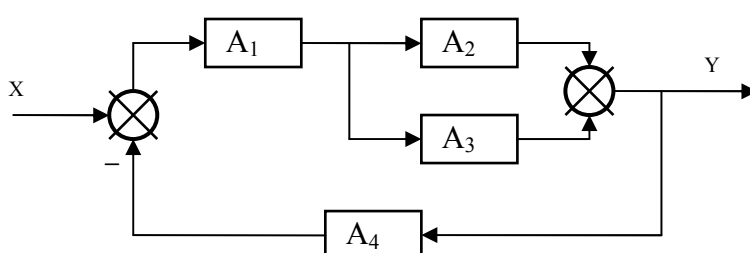
$$Y =$$

#### 2. feladat



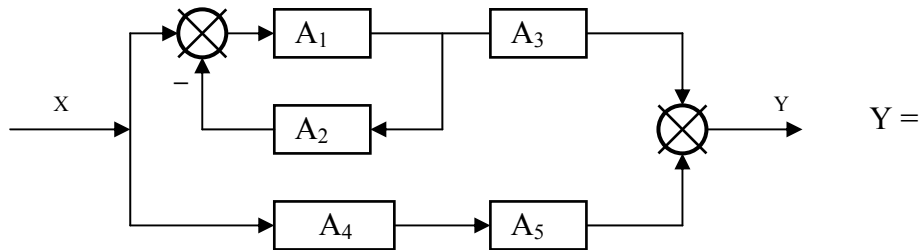
$$Y =$$

#### 3. feladat

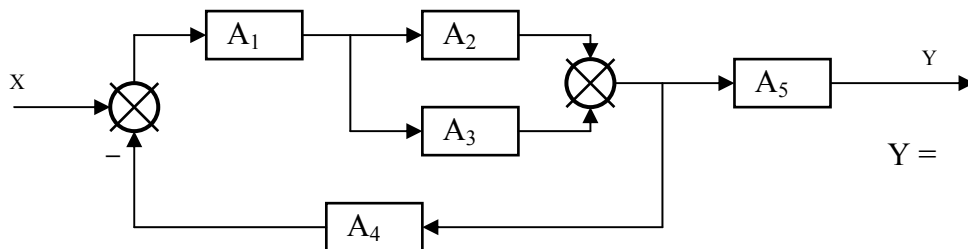


$$Y =$$

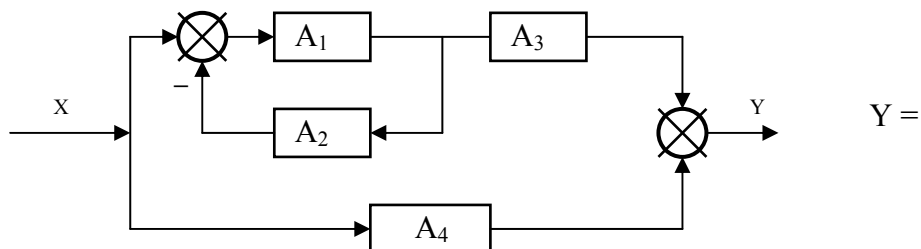
#### 4. feladat



#### 5. feladat



#### 6. feladat



### 2.3. Vizsgálójelek

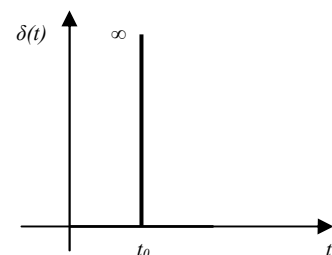
A vizsgálójelek segítségével a tagok viselkedése elméletileg leírható és tanulmányozható mindenféle viszonyok között.

#### 1) DIRAC impulzus, vagy DIRAC delta ( $\delta(t)$ )

- Matematikai formája:  $f(t) = \delta(t)$

- Csak elméletben létezik, mert nulla időtartamú, végtelen amplitúdójú, egységnyi területű impulzus, de fontos a vizsgálatok szempontjából.

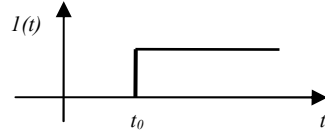
- Ha egy tag bemenetére Dirac impulzust adunk, akkor a kimenő jelet súlyfüggvénynek hívjuk.



## 2) Egységugrás függvény ( $1(t)$ )

- Matematikai formája:

$$f(t) = \begin{cases} 0; & \text{ha } t < t_0 \\ 1; & \text{ha } t > t_0 \end{cases}$$



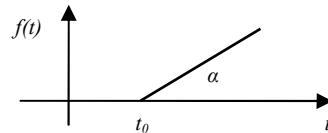
Ez a leggyakrabban használatos vizsgálójel. Gyakorlati alkalmazása igen jelentős, mert nagyon sok esetben a valóságban is a szabályozó bemenetére ehhez hasonló alapjel érkezik. Ha tag bemenetére egységugrás függvényt adunk, akkor átmeneti függvényt kapunk.

## 3) Egységsebesség függvény ( $at$ )

- Matematikai formája:

$$f(t) = t1(t) = at; \text{ ha } t > t_0$$

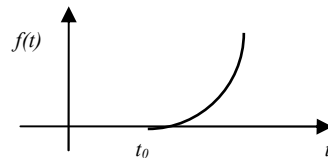
$$f(t) = 0 \text{ ha } t < t_0$$



## 4) Egységgyorsulás függvény

$$f(t) = \frac{1}{2} t^2 1(t); \text{ ha } t > t_0$$

$$f(t) = 0 \text{ ha } t < t_0$$



Megjegyzés: Az egységsebesség és az egységgyorsulás függvényekre adott kimeneti válaszfüggvényeknek nincs külön elnevezésük.

A különböző vizsgálójeltek közötti kapcsolatok az alábbi összefüggésekkel jellemezhetők [2]:

$$\frac{1}{2} \cdot \frac{dt^2}{dt} \cdot 1(t) = t \cdot 1(t); \frac{d}{dt} t \cdot 1(t) = 1(t); \frac{d}{dt} 1(t) = \delta(t) \quad (2.8)$$

Az előbbi összefüggésekből jól látható, hogy a súlyfüggvény az átmeneti függvény deriváltja.

## 2.4. Arányos (proporcionális) tagok

### 2.4.1. Energiatároló nélküli arányos tag

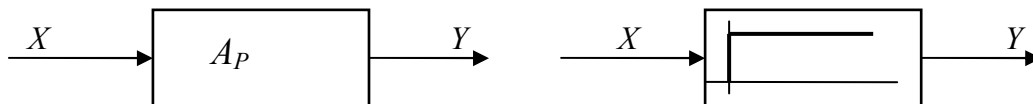
Az energiatároló nélküli tag a legegyszerűbb. Egyenlete nem tartalmaz differenciál elemeket csak szabad tagot. Ennek megfelelően kiindulva a (2.2) összefüggésből, a következő egyszerű kimeneti függvényt kapjuk:

$$y(t) = A \cdot x(t) \quad (2.9)$$

A beavatkozó jel késedelem nélkül, arányosan követi a rendelkező jelet. Jelölését a 2.9. ábrán láthatjuk.

- Átmeneti függvénye:

$$Y = A_P X \quad (2.10)$$



2.9. ábra. Az energiatároló nélküli arányos tag jelölése

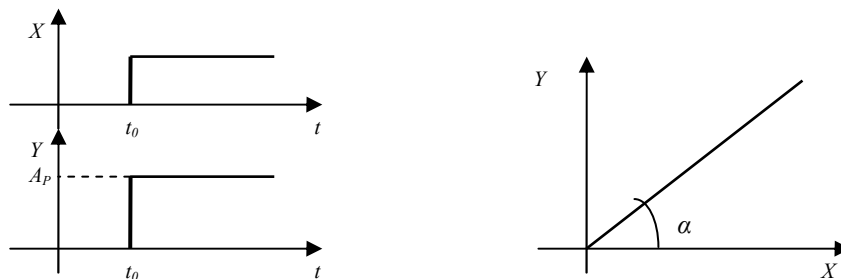
- Átviteli tényezője:

$$A_P = \frac{Y}{X} \quad (2.11)$$

- Időtényezője nulla. ( $T = 0$ )

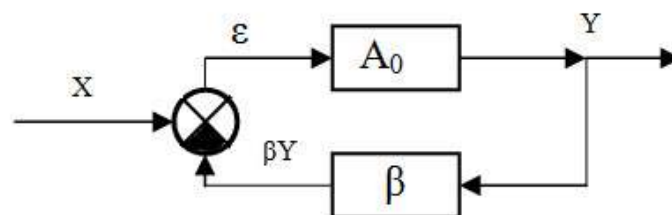
Megjegyzés: Az átviteli tényező a  $P$  indexet a proporcionális elnevezésből kapta.

A következő ábrán (2.10. ábra) az energiatároló nélküli arányos tag egységugrás függvényre adott válaszát, illetve átmeneti függvényének alakulását láthatjuk.



2.10. ábra. Az energiatároló nélküli arányos tag átmeneti függvénye és lineáris változása

Gyakorlati megvalósítása a legtöbb esetben negatív visszacsatolású erősítővel történik (2.11. ábra).



2.11. ábra. Arányos tag negatív visszacsatolással

A negatív visszacsatolás mértéke ( $\beta$ ) nagymértékben meghatározza a szabályozó stabilitását, az átviteli tényező pedig hatással van a szabályozás minőségére. Minél nagyobb az átviteli tényező, annál jobb a szabályozás hatékonysága, de csökken a stabilitása. Éppen ezért nagyon fontos ennek a tényezőnek a helyes megválasztása. Erre a kérdésre a későbbiekben még visszatérünk.

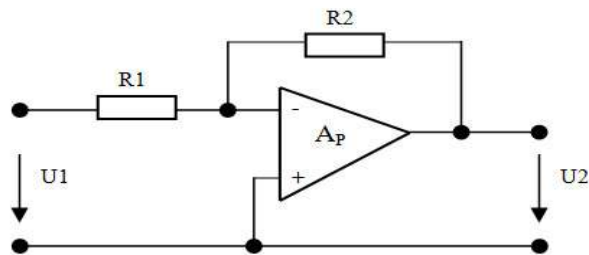
Az átviteli tényező meghatározásához az alábbi számításokat végezzük a 2.11. ábra alapján:

$$\begin{aligned}\varepsilon &= X - \beta Y \\ Y &= A_0 \varepsilon = A_0 X - A_0 \beta Y \\ Y(1 + \beta A_0) &= A_0 X\end{aligned}\tag{2.12}$$

Ahol  $A_0$  a nyílthurkú erősítés,  $\beta$  a visszacsatolási tényező,  $\varepsilon$  pedig a hibajel. Felhasználva az előbbi összefüggéseket, következik, az átviteli tényező kiszámítása:

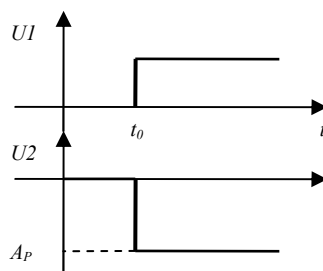
$$A_p = \frac{Y}{X} = \frac{A_0}{1 + \beta A_0} \approx \frac{1}{\beta} = \text{állandó}\tag{2.13}$$

Látható, hogy az átviteli tényezőt kizárólag csak a negatív visszacsatolási tényező határozza meg, abban az esetben, ha a nyílthurkú erősítés nagy ( $A_0 \gg 1$ ). Erre a célra legmegfelelőbb egy műveleti erősítő, ezért az esetek többségében ezt alkalmazzák. A 2.12. ábrán egy ilyen megoldást láthatunk.



2.12. ábra. Energiatároló nélküli arányos tag műveleti erősítővel

A szabályozó egységugrás függvényre adott válaszát (átmeneti függvényét) a 2.13. ábrán láthatjuk.



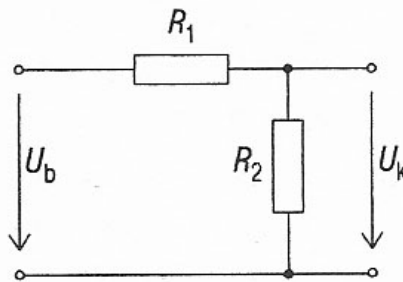
2.13. ábra. Energiatároló nélküli arányos tag átmeneti függvénye

A negatív visszacsatolásnak köszönhetően a zavaró jel hatására a szabályozó nem veszíti el egyensúlyi állapotát, hanem a szabályozott jellemző új, véges értéket vesz fel. Sok esetben azonban ez hátrányt jelent, főleg a hirtelen bementi változások esetében és nem is mindig sikerül megvalósítani.

Megjegyzés: A kimeneti feszültség ( $U_2$ ) azért negatív, mert a műveleti erősítő invertáló bemenetét használja. Ennek megfelelően az átviteli tényező a következő egyszerű összefüggéssel számítható ki (Lásd Elektronika elektrotechnika tárgy idevonatkozó fejezetét!):

$$A_p = -\frac{U_2}{U_1} = -\frac{R_2}{R_1} \quad (2.14)$$

Egy ettől még egyszerűbb modellt is megvizsgálhatunk. A feszültségosztó szintén energiatároló nélküli arányos tagnak tekinthető, viszont itt az átviteli tényező mindig egynél kisebb lesz. ( $A_p < 1$ )



2.14. ábra. A feszültségosztó, mint energiatároló nélküli arányos tag

Ebben az esetben az átviteli tényező a következőképpen alakul:

$$A_p = \frac{U_k}{U_b} = \frac{R_2}{R_1 + R_2} < 1 \quad (2.15)$$

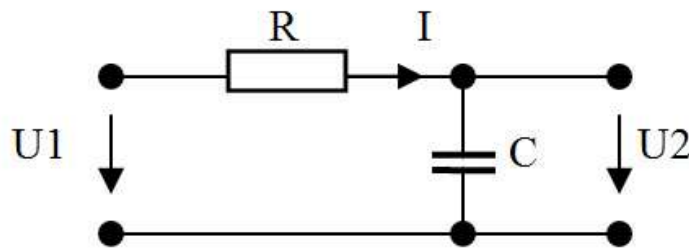
#### 2.4.2. Egytárolós tag

Valóságos körülmények között a szabályozási körben a jelek megváltozása a legtöbb esetben energiaváltozással vagy anyagáramlással jár. Ezek a változások viszont nem mehetnek végbe egyik pillanatról a másikra, így a jeleknek időkéésésük van. Pl. a nyugalomban lévő test tömegének tehetetlenségének köszönhetően az ugrásszerűen ráható erő hatására, nem tudja sebességét ugrásszerűen megváltoztatni. Időre van szüksége, míg újra egyensúlyi helyzetbe kerül és egyenes vonalú egyenletes mozgást fog végezni. A tárolók jelenléte viszont negatívan befolyásolhatja a szabályozó viselkedését, ezért szükséges az elemzése.

Differenciál egyenletét a 2.2 alapján a következő összefüggés (2.16) adja meg:

$$T_p \frac{dy(t)}{dt} + y(t) = A_p \cdot [x(t)] \quad (2.16)$$

Látható, hogy ebben az esetben az átmeneti tényező meghatározása már bonyolult matematikai műveleteket feltételez, ezért más úton közelítjük meg a feladatot. Egy ilyen esetet a legegyszerűbben egy RC aluláteresztő szűrővel (2.15. ábra) modellezhetünk. Az energiatároló szerepét a C kondenzátor képviseli.



2.15. ábra. Egytárolós tag modellezése

A feladatunk ebben az esetben is az átmeneti tényező meghatározása ( $A_p$ ), de itt már az időállandóval ( $T_p$ ) is számolnunk kell. Eddigi ismereteink alapján felírhatjuk a következő összefüggéseket:

$$A_p = \frac{U_2}{U_1} = \frac{\frac{I}{j\omega C}}{R + \frac{I}{j\omega C}} = \frac{I}{I + j\omega RC}; T_p = RC \quad (2.17)$$

Az előző összefüggés alapján látható, hogy az egyébként időbeli folyamat frekvenciafüggő lesz. Ebből következik, hogy szükséges lesz frekvenciatartományban is vizsgálni, vagyis az átviteli függvényt is meg kell határoznunk. Felhasználva a 2.1.1. fejezetben használt jelölést ( $s = j\omega$ ) megkapjuk a frekvenciatartománybeli átviteli függvényét.

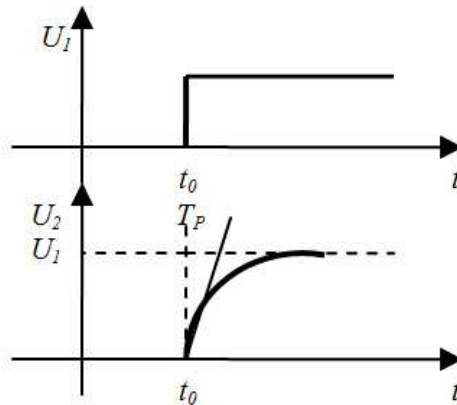
$$Y(s) = \frac{A_p}{1 + sT_p} \quad (2.18)$$

Ahol  $A_p = I$  és  $s = j\omega$ .

Az egytárolós tag átmeneti függvénye a tároló nem lineáris viselkedésének köszönhetően már nem olyan egyszerű (2.19 összefüggés).

$$U_2 = U_1 \left( 1 - e^{-\frac{t-t_0}{T_p}} \right) \quad (2.19)$$

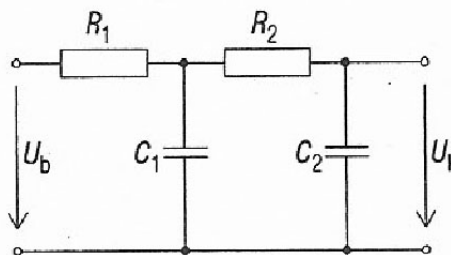
Az átmeneti függvény változását a 2.16. ábra szemlélteti. A függvény exponenciálisan változik. A görbe kezdőpontjában húzott érintő és a bementi érték metszéspontja, megadja az időállandó ( $T_P$ ) grafikus értelmezését.



2.16. ábra. Az egytárolós tag átmenti függvénye

### 2.4.3. Kéttárolós tag

Minél több tárolót tartalmaz egy szabályozó tag, annál bonyolultabb lesz a viselkedése, és a stabilitása is jelentősen csökken. A valóságban azonban számolnunk kell ezekkel a hatásokkal és tudnunk kell kezelni azokat. Kéttárolós tagot legegyszerűbben úgy kapunk, hogy két egytárolós tagot sorba kapcsolunk egymással. Ha az előbbi RC tagból álló modellnél maradunk, akkor a kéttárolós tag elvi felépítése a 2.17. ábra szerinti lesz.



2.17. ábra. Kéttárolós tag (forrás [1])

A 2.2. fejezetben leírtak alapján a soros tag átviteli tényezőjét az egyes tagok átviteli tényezőinek szorzata adja meg.  $A_P = A_{P1} \cdot A_{P2}$ ; az időállandók pedig  $T_1 = R_1 \cdot C_1$  és  $T_2 = R_2 \cdot C_2$ . A tag differenciálegyenlete a 2.2. összefüggés alapján már másodrendű differenciált is tartalmazni fog (2.20).

$$T_1 T_2 \frac{d^2 U_k(t)}{dt^2} + (T_1 + T_2) \frac{dU_k(t)}{dt} + U_k = A_P U_b(t) \quad (2.20)$$

Az imént említett soros kapcsolat alapján pedig felírhatjuk a következő egyenleteket:

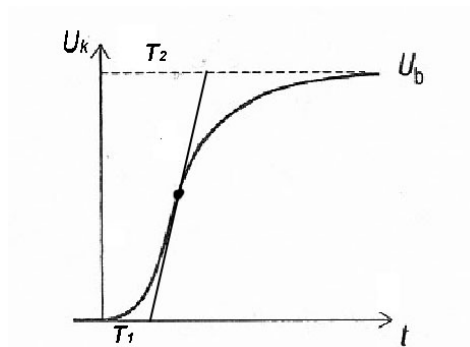
$$A_p = A_{p1} \cdot A_{p2} = \frac{1}{1 + j\omega R_1 C_1} \cdot \frac{1}{1 + j\omega R_2 C_2} \quad (2.21)$$

$$A_p = \frac{1}{1 + j\omega(R_1 C_1 + R_2 C_2) - \omega^2 R_1 C_1 R_2 C_2}$$

Figyelembe véve a tag időállandóinak az értékeit, ezeket felhasználva és a Laplace transzformációt valamint az  $s = j\omega$  jelölést alkalmazva megkapjuk a tag frekvenciatartománybeli átviteli függvényét.

$$Y(s) = \frac{A_p}{T_1 T_2 \cdot s^2 + (T_1 + T_2) \cdot s + 1}; \text{ ahol : } A_p = 1 \quad (2.22)$$

A kéttárolós tag átmeneti függvényét a 2.18. ábrán láthatjuk.

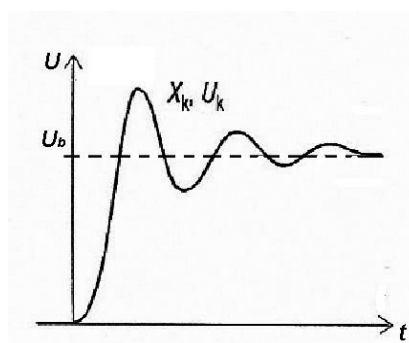


2.18. ábra. Kéttárolós tag átmeneti függvénye

A  $T_1$  és  $T_2$  időállandók grafikus értelmezését a görbe inflexiós pontjába húzott egyenesnek az időtengellyel és a bementi értékkel történő metszéspontja adja meg.

#### 2.4.4. Lengő tag

A szabályozók bemenő jelének ugrásszerű változása sok esetben a kimenő jel belengéséhez vezet. Egy ilyen helyzet látható 2.19. ábrán.

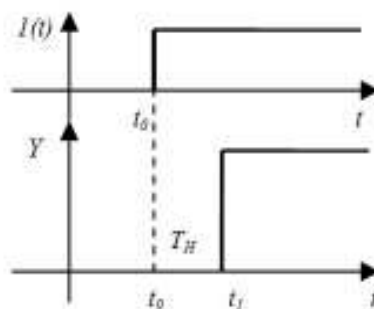


2.19. ábra. A lengőtag átmeneti függvénye (forrás [1])

A jelenséget a rendszer különböző jellegű, egymástól független energiatárolóik jelenléte okozza. Ez nagymértékben befolyásolhatja a rendszer stabilitását. Ezért egy jó szabályozónál alapvető követelmény a stabilitás, amely kezelni tudja ezeket az átmeneti állapotban fellépő jelenségeket. A szabályozó stabilitásának köszönhetően a belengés rövid időn belül megszűnik, és egyensúlyi helyzetbe kerül.

#### 2.4.5. Holtidős tag

A szabályozott berendezésekben az energiatárolókon kívül nagyon gyakran az anyagáramlás alacsony sebessége okoz problémát. A valóságos fizikai folyamatok mind tartalmaznak kisebb-nagyobb időkéséseket. Ezt nevezzük *holtidőnek*. Ilyen jelenséggel találkozunk például, amikor egy szállítószalag elején megindul az anyagáramlás, de a szalag végén csak bizonyos idő eltelte után jelenik meg. Ugyanez a helyzet akkor is, amikor egy hosszú csővezeték kezdetén megnyitjuk a szelepet, a folyadék késve jelenik meg a cső végén. Az átmeneti függvény ilyenkor követi a bemeneti jel változását, de csak a holtidő eltelte után (2.20. ábra).



2.20. ábra. Holtidős tag átmeneti függvénye

A holtidő nagysága könnyen meghatározható, kiszámítható vagy akár mérhető is, de a szabályozónak mindenképpen ismernie kell, hogy különbséget tudjon tenni a holtidő által okozott jelenség és a jelhiány között.

$$T_H = t_1 - t_0 \quad (2.23)$$

#### 2.5. Integráló tag

A proporcionális szabályozók, noha igyekeznek a beavatkozást az alapjelnek megfelelően alakítani ez a valóságban nem mindig vezet kellő eredményre. Sőt vannak olyan esetek, amikor kimondottan káros következményeket okoz. Gondoljuk csak el, egy nagy teljesítményű motorfordulatszám szabályozás esetében, amikor az alapjel indításkor hirtelen nulláról akár a maximális értékre is növekedhet, ez a változás a kimeneten súlyos károkat okozhatna, nemcsak a motornak, hanem a

hozzákapcsolt mechanikai hajtásnak is. Éppen ezért szükséges olyan tagoknak a használata is, amelyek a hirtelen, ugrásszerű bemeneti változás csillapítják, lassítják. Ilyen megoldást biztosít számunkra az integráló tag.

Az integráló tag kimenő jele a bemenő jel integráljával arányos, azaz a kimeneti jel lineárisan változik, ha a bemeneti jel állandó. A bemeneti ugrásszerű változások, a tag kimenetén lineárisan növekvő mennyiséget eredményez. Vagyis az alapjelnek megfelelő kimeneti értéket csak egy bizonyos idő eltelte után fogja elérni. Ezt az időt *integrálási időnek* vagy *integrálási állandónak* nevezzük. A tagot leíró differenciálegyenlet a 2.2 relációnak megfelelően a következő lesz:

$$T_I \frac{dy(t)}{dt} = A_I x(t) \quad (2.24)$$

Ahol  $T_I$  az integrálási időállandó,  $A_I$  pedig az átviteli tényező.

A tag átviteli tényezője a 2.24 alapján :

$$A_I = \frac{\Delta Y}{\Delta X}; hA_I T_I = 1 \quad (2.25)$$

A hatásvázlatban az integráló tag jelölésére a következő szimbólumok egyikét alkalmazhatjuk.



2.21. ábra. Az integráló tag jelölési módjai

A tag átmenti függvényét a 2.26 relációval írhatjuk le.

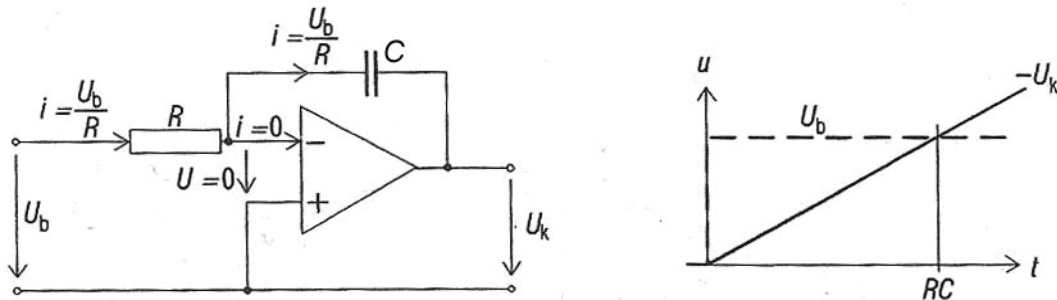
$$y(t) = \frac{A_I}{T_I} \int x(t) dt = \frac{A_I}{T_I} \int_{t_0}^t I(t) dt = \frac{A_I}{T_I} (t - t_0) \quad (2.26)$$

Frekvenciatartománybeli átviteli függvénye pedig:

$$Y(s) = \frac{A_I}{T_I} \frac{1}{s} \quad (2.27)$$

Ahol a már eddig is ismert jelölés szerint:  $s = j\omega t$ .

Az integráló tag gyakorlati megvalósítására kiválóan alkalmas a műveleti erősítő. A 2.22. ábrán az  $U_b$  egyenfeszültség hatására az  $R$  bementi ellenálláson folyó  $i$  áram a műveleti erősítő nagyon nagy bementi ellenállása miatt teljes egészében a kondenzátort tölti. A kondenzátor a töltést halmozza, integrálja aminek következtében kimeneti feszültsége lineárisan fog növekedni.



2.22. ábra. Integráló tag műveleti erősítővel (forrás [1])

Az így kapott integráló tag kimeneti feszültségét ( $U_k$ ) előző tanulmányaink ismeretei alapján, a következőképpen írhatjuk fel:

$$U_k = -\frac{1}{C} \int i dt = -\frac{1}{RC} \int U_b dt \quad (2.28)$$

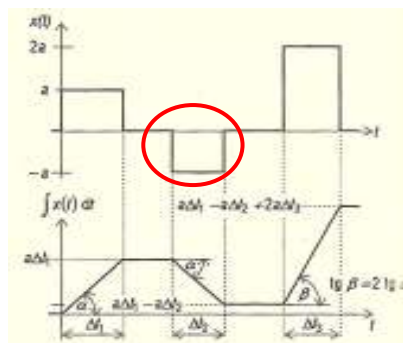
Az előző kimeneti feszültségértéket ha összehasonlítjuk a 2.26-os alap relációval, láthatjuk, hogy teljes mértékben eleget tesz az integrálási követelményeknek, ha  $A_I = 1$  és  $T_I = R \cdot C$ .

Megjegyzés: A 2.22 ábrán a kimeneti feszültség lineárisan növekszik ugyan, de negatív irányban ( $-U_k$ ), mert a műveleti erősítő invertáló bementét használja, a bementi feszültség pedig pozitív.

Az integráló tag legfontosabb jelátviteli tulajdonságai a következők:

- Egységugrás jelből lineáris jelet kapunk,
- Négyszögjelből háromszögjelet,
- Szinuszos jelből koszinuszosan változó jelet  $90^\circ$ -os késéssel, mert szinusznak az integrálja mínusz koszinusz.

Felhasználva ezeket a tulajdonságokat, figyeljük meg a 2.23. ábrán látható bemeneti mennyiségre kapott kimeneti változásokat!



2.23. ábra. Az integráló tag válasz a különböző bementi változásokra (forrás [1])

Az ábra alapján a következő fontos következtetéseket vonhatjuk le:

- Az integráló tag igyekszik a hirtelen megjelenő bemeneti változásokat lassítani;
- A kimenő jel állandó értéke (állandósult állapot) csak akkor tud kialakulni, amikor a bemeneti mennyiség:  $x(t) = 0$ ;
- Negatív hatást kiváltó zavaró tényező hatása nem jelenik meg a kimenten.

Ez utóbbi azt jelenti, hogy az integráló tagot tartalmazó szabályozó zajvédeltsége is magasabb, mert az integrálás során az alapjelre tevődő nemkívánatos villamos zajok nem befolyásolják a kimeneti változás.

## 2.6. Differenciáló tag

A szabályozók esetében nem csak hirtelen változások lehetségesek, hanem olyan esetek is előfordulhatnak, amikor az alapjel nagyon lassan változik. A jó szabályozónak a lassú változásokat is tudnia kell megfelelően kezelni. Ezért szükséges ilyen esetekben a lassú változások gyorsítása, dinamikusabbá tétele. A differenciáló tag ebben segít a szabályozókban. Felhasználva a differenciálás tulajdonságait, a következő jellemzőket sorolhatjuk fel:

- A kimenő jel a bemenő jel differenciáljával arányos;
- Csak az állapotváltozások idejére ad jelet, mert egy konstans érték differenciálja nulla;
- A gyakorlatban viszont csak arányos taggal együtt értelmezhető, mert egységugrás függvényre adott válasza a Dirac impulzus lenne, ami egy elméleti függvény.
- Differenciálegyenlete:

$$y(t) = A_D \cdot T_D \frac{dx(t)}{dt} \quad (2.29)$$

- Átviteli tényezője:

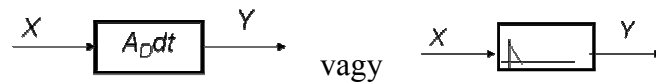
$$A_D = \frac{\Delta Y}{\frac{\Delta X}{\Delta t}}; \quad \text{ha } T_D = 1 \quad (2.30.)$$

- Átmenti függvénye:

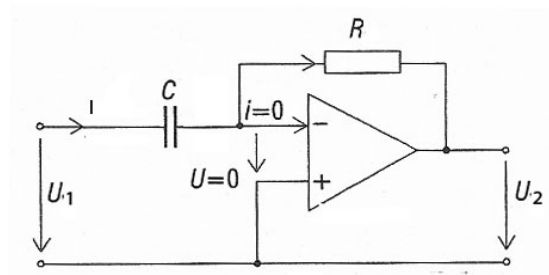
$$y(t) = A_D \cdot T_D \frac{dx(t)}{dt} = A_D \cdot T_D \frac{I(t)}{dt} = 0 \quad (2.31)$$

- Frekvenciatartománybeli átviteli függvénye:  $Y(s) = A_D T_D s \quad (2.32)$

- Hatásvázlati jelölése:



Akárcsak az integráló tag esetében, a differenciáló tagot is a gyakorlatban a leghatékonyabban műveleti erősítővel tudjuk megvalósítani. Ez látható a 2.24. ábrán.



2.24. ábra. Differenciáló tag műveleti erősítővel

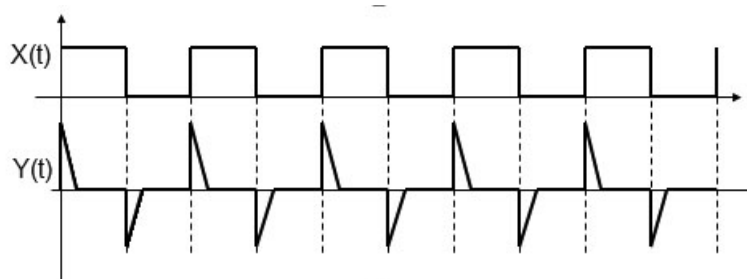
Felhasználva az eddigi ismereteinket, a 2.24. ábrán látható differenciáló tag kimeneti feszültségét ( $U_2$ ) a következőképpen írhatjuk fel:

$$U_2 = I \cdot R = -C \cdot \frac{dU_1}{dt} \cdot R = -T_D \cdot \frac{dU_1}{dt} \quad (2.33)$$

Ahol  $T_D = R \cdot C$  a differenciálási idő.

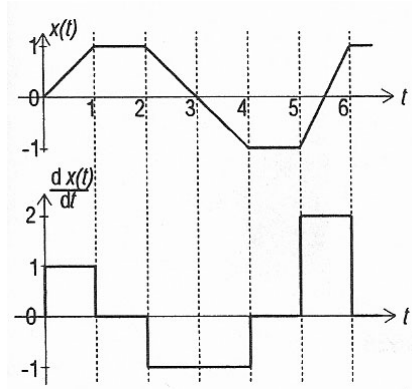
A differenciálásból adódó fontosabb jelátviteli tulajdonságai:

- Egységugrás jelre tűimpulzus keletkezik (2.25. ábra),
- Háromszögjelből négyszögjel lesz,
- Szinusz jelből pedig koszinuszosan változó jel ( $+90^\circ$  fázissal, mert szinusz deriváltja plusz koszinusz).



2.25. ábra. Differenciáló tag válasza a négyszögjelre.

Figyeljük meg a következő ábrán látható kimeneti változásokat!



2.26. ábra. A differenciáló tag hatása

Érdemes megfigyelni, hogy ha a bemenet lineárisan változik, a kimenet azonnal, ugrásszerűen reagál. Az ábra alapján további fontos következtetéseket vonhatunk le:

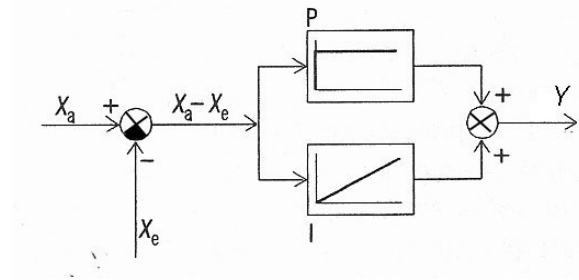
- Igyekszik a lassú változásokat gyorsítani,
- Csak az átmeneti állapotokban fejt ki hatását,
- Állandó értékű bemeneti mennyiség mellett kimenete nulla.

## 2.7. Összetett szabályozók

Az összetett szabályozók célja az optimális szabályozás elérése. Egy ilyen szabályozóval szemben támasztott követelmények a stabilitás, vagyis a bemenő jel változása után rövid időn belül újra stabil állapotba kerüljön és a nagy hurokerősítés, de ennek hatására ne romoljon a stabilitás. Az előző fejezetekben bemutatott P, I, D tagok mindegyike jól meghatározható tulajdonságokkal rendelkezik. A proporcionális tagot leginkább a gyorsaság jellemezte, az integráló tagot a pontosság és a hirtelen változások kezelése, míg a differenciáló tag a lassú változásokat emelte ki. Logikusnak tűnik ezeknek a szabályozó tagoknak a kombinálása annak érdekében, hogy minél hatékonyabb és minden körülmények között jól működő szabályozót kapjunk. A megvalósítás során szükségessé válik a tagok paramétereinek megfelelő kiválasztása és az átviteli függvények optimalizálása. Az eljárást kompenzálásnak nevezzük, és a kompenzálási hatás szerint pedig PI, PD és leggyakrabban PID szabályozókkal találkozhatunk.

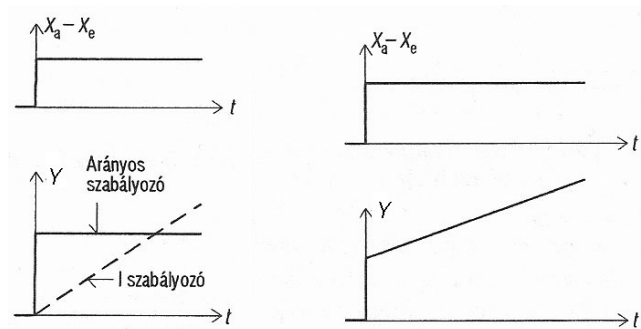
### 2.7.1. PI szabályozó

A PI szabályozó egyesíti a proporcionális és integráló tagok hatásait. Legegyszerűbben a P és I tagok párhuzamos kapcsolásával valósítható meg, mert a 2.2. fejezetben láthattuk, hogy a tagok közötti párhuzamos kapcsolat összegzi a hatásokat. Ez látható a 2.27. ábrán.



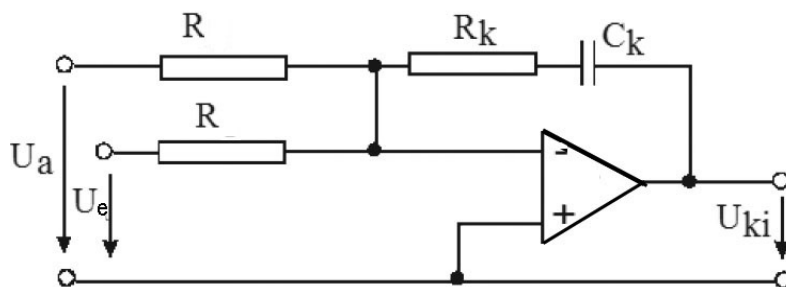
2.27. ábra. P és I tagok párhuzamos kapcsolata (forrás [1])

Az így kapott szabályozó átmeneti függvénye (2.28. ábra) egyesíti a tagok hatásait, ezáltal az arányos tag a gyorsaságot, az integráló tag a pontosságot biztosítja.



2.28. ábra. A PI szabályozó átmeneti függvénye (forrás [1])

Megvalósítása a legtöbb esetben műveleti erősítővel történik (2.29. ábra). Az erősítő invertáló bemenetén létrejövő  $U_r$  hibajel a kimeneten az átmeneti függvény ( $U_{ki}$ ) a proporcionális és az integráló hatást összegzi (2.34).



2.29. PI szabályozó műveleti erősítővel (forrás [3]).

$$U_r = U_a - U_e$$

$$U_{ki} = -\frac{R_k}{R} \left( U_r + \frac{1}{R C_k} \int U_r dt \right) = -A_p \left( U_r + \frac{1}{T_I} \int U_r dt \right) \quad (2.34)$$

$$A_p = \frac{R_k}{R}; T_I = R C_k \quad (2.35)$$

A 2.35 relációkban  $A_p$  a proporcionális tag átviteli tényezője, míg  $T_I$  az integrálási időt jelenti.

Frekvenciatartománybeli átviteli függvényét a Laplace transzformáció alapján a 2.36 egyenlet fejezi ki.

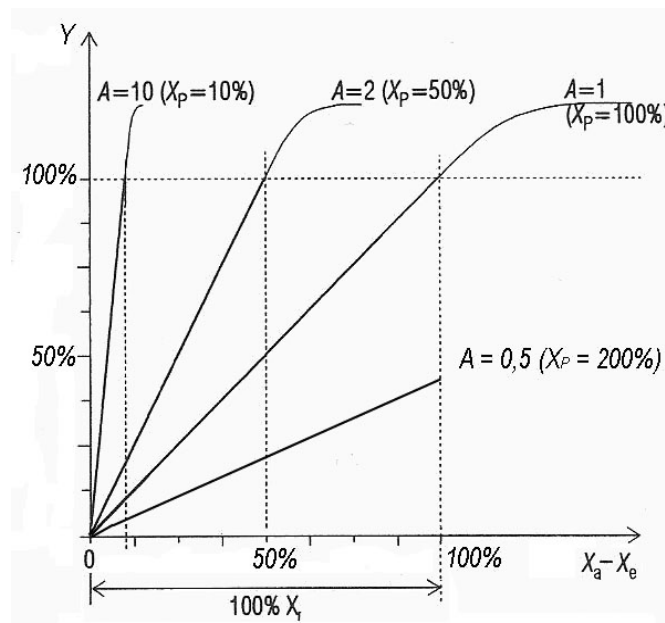
$$Y(s) = A_p \left( 1 + \frac{I}{T_I \cdot s} \right) \quad (2.36)$$

### 2.7.2. A P tag kompenzálása [1]

Az összetett szabályozók arányos tagjának jellemzésére, vagyis a kompenzáció mértékét a tag a százalékosan kifejezett  $X_p$  arányossági tartományát szokás megadni. Ez nem más, mint az arányos tag átmeneti tényezőjének ( $A_p$ ) vagy más néven az a tagot megvalósító műveleti erősítő feszültegerősítése.

$$X_p = \frac{I}{A_p} 100\% \quad (2.37)$$

Az arányossági tartomány a bemenőjel (hibajel) működési tartományának az a hányada, százalékban kifejezve, amelynek hatására a szabályozó a teljes kimenőjel tartományát lefedi. A 2.30. ábrán a kompenzálás sztatikus karakterisztikáját láthatjuk.

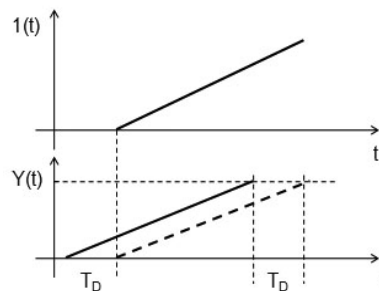


2.30. ábra. A szabályozó arányossági tartománya

Jelentése: a bemenőjel  $x_r = x_a - x_e$  változási tartománya hány százalékát képezi a kimenő jelnek.

### 2.7.3. PD szabályozó

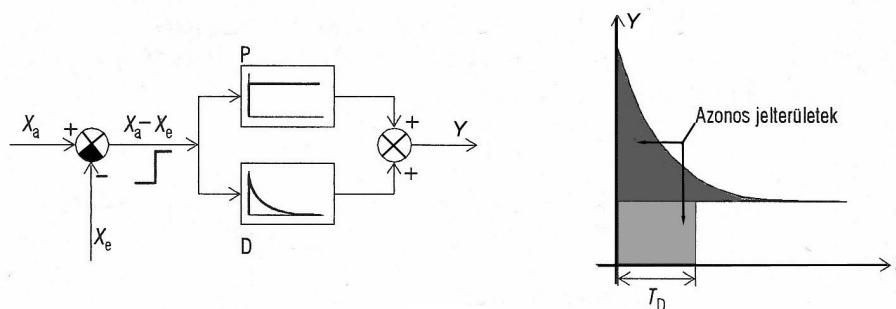
A 2.6. fejezetben azt láttuk, hogy a D tag valós hatása az átmeneti függvényt illetően túimpulzus formájában nyilvánul meg, ami meglehetősen kedvezőtlen a szabályozás szempontjából de mégis jelentősége lehet a lassú változások szempontjából. Ezért a PD szabályozót nem az egységugrás függvénnyel vizsgáljuk, hanem az egységsebesség függvénnyel. Ennek megfelelően a PD szabályozó válasza az egységsebesség függvényre igen érdekes hatást eredményez (2.31. ábra).



2.31. ábra. A PD szabályozó válasza az egységsebesség függvényre

Az ábrán feltüntetett  $T_D$  differenciálási idő meghatározó lehet a szabályozás szempontjából, mert azt fejezi ki, hogy mennyi idővel hamarabb éri el a kimenet a kívánt értéket. Olyan mintha ennyi idővel hamarabb kezdődött volna a hatás.

Megvalósítása ebben az esetben is a proporcionális és a differenciáló tag párhuzamos kapcsolatával érhető el (2.32. ábra).

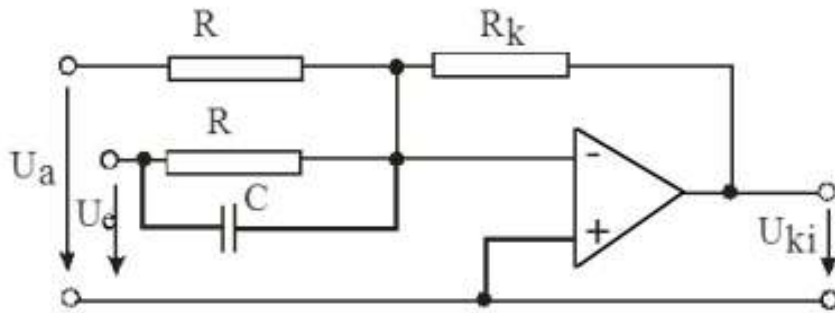


2.32. ábra. PD szabályozó elvi megvalósítása párhuzamos kapcsolással

A differenciálási időt úgy is értelmezhetjük, hogy az az idő, amely alatt az arányos csatorna jelterülete a D csatorna teljes jelterületével egyenlővé válik (2.32. ábra). A gyorsító hatás abból látszik, hogy a P szabályozónak  $T_D$  idővel a hibajel megjelenése előtt kellett volna indulnia ahhoz, hogy ugyanazt az eredményt érje el mint a PD szabályozó. Ez olyan mintha egy gépkocsit úgy gyorsítanánk fel 60 km/h sebességre, mintha 100 km/h-t szeretnénk elérni. A gázpedált mélyebbre

nyomjuk, mint azt a 60 km/h indokolná, majd a megfelelő pillanatban visszavesszük, hogy ne haladjuk túl a kívánt értéket [1].

A PD szabályozó műveleti erősítővel történő megvalósítását a 2.33. ábra szemlélteti.



2.33. ábra. PD szabályozó műveleti erősítővel

Az erősítő invertáló bementén létrejövő  $U_r$  hibajel a kimeneten az átmeneti függvény ( $U_{ki}$ ) a proporcionális és a differenciáló hatást összegzi.

$$U_r = U_a - U_e$$

$$U_{ki} = -\frac{R_k}{R}(U_r + R_k C \frac{dU_r}{dt}) = -A_p(U_r + T_D \frac{dU_r}{dt}) \quad (2.38)$$

$$A_p = \frac{R_k}{R}; T_D = R_k C \quad (2.39)$$

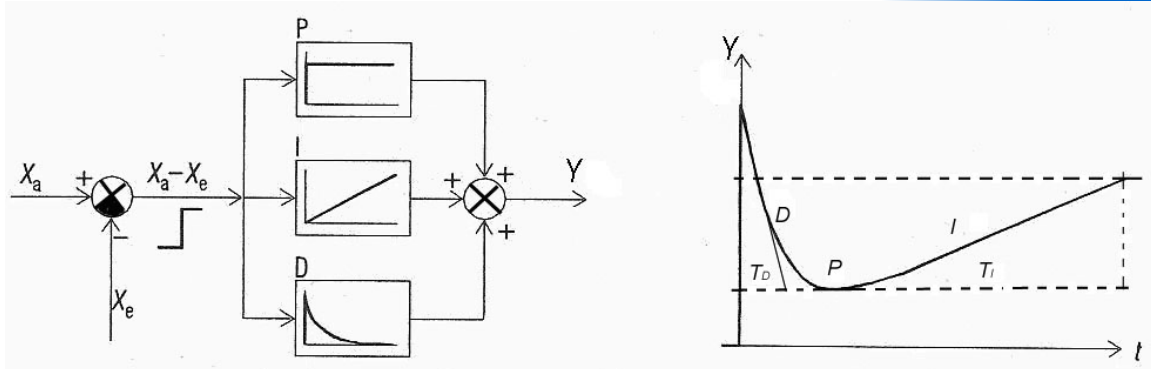
A 2.39 relációkban  $A_p$  a proporcionális tag átviteli tényezője, míg  $T_D$  a differenciálási időt jelenti.

A PD szabályozó frekvenciatartománybeli átviteli függvényét a Laplace transzformáció alapján a 2.40 egyenlet fejezi ki.

$$Y(s) = A_p(1 + T_D s) \quad (2.40)$$

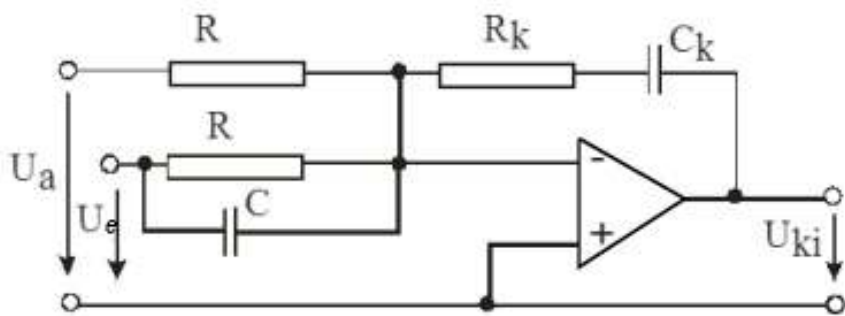
#### 2.7.4. PID szabályozó

Az integráló tag által nyújtott pontosságot és a differenciáló hatásban szereplő gyorsítási lehetőséget egyidejűleg a PID szabályozókban valósíthatjuk meg. Az ilyen típusú szabályozó egyidejűleg egyesíti a három alaptípus tulajdonságait egy szabályozón belül. Elvi megvalósítása a három alaptag párhuzamos kapcsolatával érhető el (2.34. ábra).



2.34. PID szabályozó elvi megvalósítása

Gyakorlati kivitelezési formája a legegyszerűbb módon műveleti erősítővel valósítható meg. Ez látható a 2.35. ábrán.



2.35. ábra. PID szabályozó műveleti erősítővel

Az így megvalósított szabályozóban egyidejűleg érvényesíthető mindhárom alapvető szabályozó hatás. A szabályozó átmeneti függvényét a 2.41 reláció írja le.

$$U_r = U_a - U_e$$

$$U_{ki} = -\frac{R_k}{R}(U_r + \frac{1}{R C_k} \int U_r dt + R_k C \frac{dU_r}{dt}) = -A_p(U_r + \frac{1}{T_I} \int U_r dt + T_D \frac{dU_r}{dt}) \quad (2.41)$$

$$A_p = \frac{R_k}{R}; T_I = R C_k; T_D = R_k C \quad (2.42)$$

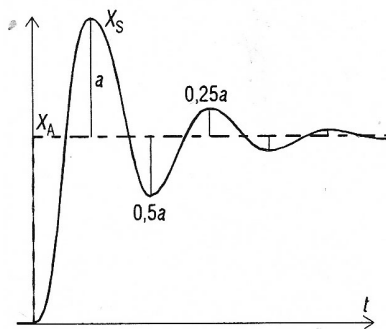
A 2.42 relációkban  $A_p$  a proporcionális tag átviteli tényezője,  $T_I$  az integrálási időt, míg  $T_D$  a differenciálási időt jelenti.

A szabályozó átviteli függvényét a 2.43 összefüggés adja meg.

$$Y(s) = A_p(1 + \frac{1}{T_I s} + T_D s) \quad (2.43)$$

### 2.7.5. A szabályozók hangolása [1]

A legjobb szabályozási eredmény elérése érdekében a szabályozót mindig az adott folyamathoz kell hangolni. Ez azt jelenti, hogy a szabályozó  $X_P$ ,  $T_I$  és  $T_D$  értékeit a folyamat dinamikájához kell igazítani. A megoldás többféleképpen érhető el, közülük a gyakorlatban is leginkább használható az, amelyik a zárt szabályozási kör átmeneti függvényében a második túllövésnek a mértékét az első túllövés felében határozza meg. És így tovább: a harmadik túllövés a második fele kell legyen (2.36. ábra).



2.36. ábra. Az optimális szabályozás átmeneti függvénye

Az optimális beállítás egyik ma is használatos eljárását *Ziegler* és *Nichols* dolgozta ki üzemi kísérletek és mérések alapján. Az eljárás lényege:

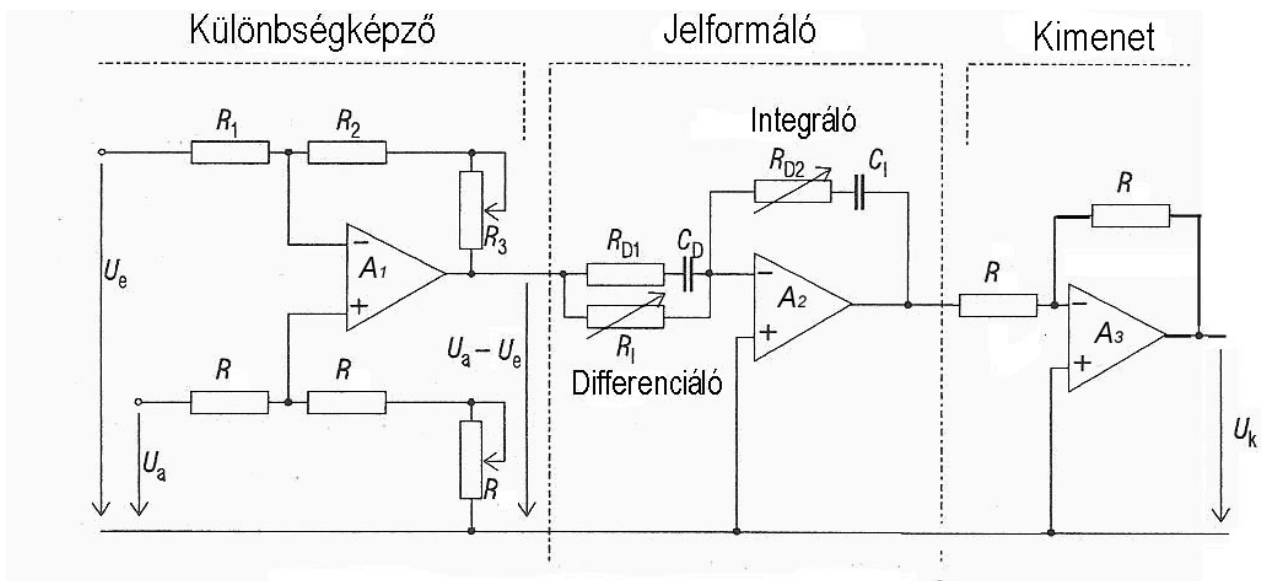
- A szabályozási körben kiiktatjuk az  $I$  és  $D$  tagokat, csak a proporcionális tagot állítjuk be. Növeljük a hurokerősítést, vagyis csökkentjük az  $X_P$ -t mindaddig, amíg a szabályozási hurok lengési állapotba kerül. Ez lesz a stabilitás határeset.
- Ezt követően megmérjük a lengés periódusidejét ( $T_{per}$ ) és feljegyezzük az arányossági tartomány kritikus értékét ( $X_{pkrit}$ ), amely mellett a szabályozó állandósult lengéseket végez.
- E két érték alapján kiszámítjuk a szabályozó optimális beállítási értékeit a 1. táblázat alapján.

1. táblázat (forrás [1])

Szabályozó	$X_P$	$T_I$	$T_D$
<b>P</b>	$X_P \geq 2 X_{pkrit}$	$\infty$	0
<b>PI</b>	$X_P \geq 2,2 X_{pkrit}$	$T_I \geq 0,8 T_{per}$	0
<b>PID</b>	$X_P \geq 1,7 X_{pkrit}$	$T_I \geq 0,5 T_{per}$	$T_D < 0,125 T_{per}$

### 2.7.6. Egy PID szabályozó gyakorlati megvalósítása

A 2.36. ábrán egy műveleti erősítőkkel épített PID szabályozó kapcsolási vázlatát és hangolási lehetőségei láthatjuk. A szabályozó első fokozata az  $A_1$  erősítővel megvalósított különbségképző rész. Ebben a fokozatban tudjuk beállítani az előző fejezetben meghatározott  $X_P$  optimális értékét az  $R_3$ -as ellenállás segítségével. Az  $R$  ellenállással a műveleti erősítő offset beállítását végezhetjük (lásd Elektronika elektrotechnika tárgy idevonatkozó fejezetét).



2.36. ábra. PID szabályozó elvi kapcsolási rajza (forrás [1])

A jelformáló fokozatban valósul meg az előző fokozat által létrehozott hibajel ( $U_a - U_e$ ) integrálása és differenciálása. Az  $R_I$  ellenállással az integrálási időt ( $T_I = R_I C_I$ ), míg az  $R_{D2}$  ellenállással a differenciálási időt ( $T_D = R_{D2} C_D$ ) állítjuk be. Mivel a jelformáló fokozat a műveleti erősítő invertáló bementét használja, ezért szükséges a kimeneti fokozat, amely tulajdonképpen egy újabb invertálást végez erősítés nélkül.

### 3. SZÁMÍTÓGÉPES FOLYAMATIRÁNYÍTÁS

Az ipari folyamatok számítógépes irányítása jóformán egyidős a számítógéppel. Jelentős előrelépés azonban itt is csak a mikroszámítógépek, mikroprocesszoros rendszerek megjelenésével és elterjedésével történt. Az utóbbi évtizedekben az analóg technikát szinte teljes mértékben a digitális technika váltotta fel. A teljesen hardveres megoldások helyett a szoftver jelentősége került előtérbe. Ez azt eredményezte, hogy adott hardver (mikroprocesszor, mikrokontroller, PLC) akár többféle folyamathoz is eredményesen használható, azaz programozható. Ezáltal a termelés is rugalmasabbá válik, a vevői igényekhez maradéktalanul tud igazodni, mert a technológia módosítása miatt nem kell lecserélni az irányítórendszert csak átprogramozni. Egy másik jelentős változás, hogy a centralizált kábelezés helyébe a helyi informatikai hálózat került, annak számtalan előnyeivel. A központi feldolgozást az elosztott intelligencia követte, a matematikai algoritmusok mellett pedig megjelentek a mesterséges intelligencia módszerei.

#### 3.1. Alapfogalmak

A számítógépes folyamatirányítás az ipari gyártási folyamatok és technológiai rendszerek számítógépes vezérlését, irányítását és felügyeletét jelenti. A fejlődés során az irányítórendszerek elemei és struktúrái sokat változtak, több generációjuk ismert. Jelenleg a korszerű folyamatirányító rendszerek az ötödik és hatodik generációhoz tartozó megoldásokat alkalmazzák.

Az ötödik generációs rendszerek kialakulását az *intelligens (SMART) távadók*, valamint az *intelligens beavatkozók* és a *terepi buszok* megjelenése váltotta ki. Ezek az eszközök egy – a nagy hálózatoktól (pl. ETHERNET) eltérő – kommunikációs csatornán a feladat jellegéhez igazodó protokoll szerint kezelhetők. A kommunikációs vonalra (szokásos szóhasználat szerint fieldbuszra) korlátozott számú intelligens eszköz fűzhető fel [4].

A hatodik generációs rendszer az előzőhöz hasonló felépítésű. Lényegi különbség, hogy a terepi buszok jelentős részét az ipari Ethernet hálózat váltja fel és eredményesen használható a vezeték nélküli VLAN technológia is.

A számítógépes folyamatirányítás fontosabb elemei a következők:

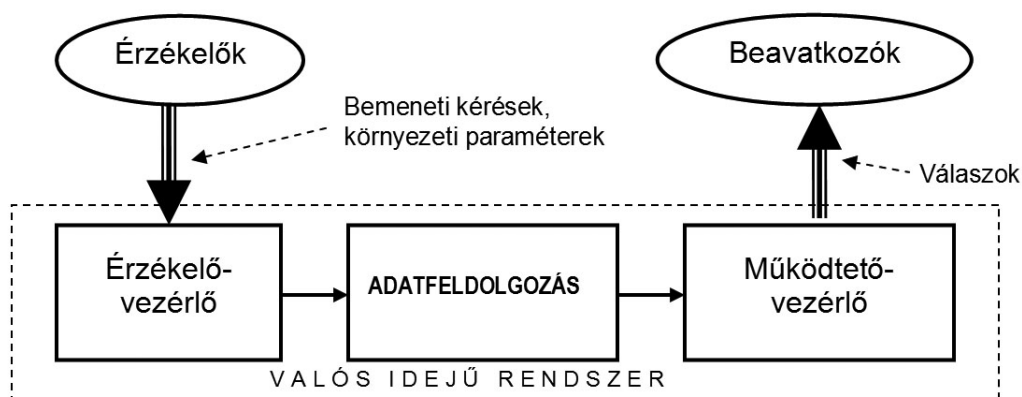
- intelligens távadók, érzékelők, digitális szabályozók;
- ipari számítógépek, PLC-k, PXI-k, mikrokontrollerek, stb.
- számjegyvezérlésű szerszámgepek (NC, CNC), megmunkáló központok;
- robotok, robotkarok;
- szállítópályák, manipulátorok;
- ipari kommunikációs hálózatok.

A számítógépes folyamatirányítás megvalósítást a *valós idejű (Real-Time)* számítógépes rendszerek és *osztott intelligenciájú*, hierarchikus (DCS) rendszerek és a közöttük kiépített kommunikációs kapcsolatok teszik lehetővé.

### 3.2. Valós idejű (Real-Time) rendszerek

*Valós idejűnek tekinthetjük azt a rendszert, amelyek specifikációjában a logikai és számítási funkciókon túlmenően valamilyen időbeli viselkedésre vonatkozó előírás is szerepel a külső, valós időskálához kötötten.* A valós idejű rendszerekben az események feldolgozásában az idő alapvető szerepet játszik. A szigorúan időkritikus rendszereknél a technológiai időkorlátok be nem tartása épp olyan hibának számít, mint egy hibás számítási eredmény. Az alapprobléma, hogy egy vagy több külső eszköz előre nem ismert időpontban valamilyen esemény által kiváltott információt küld a számítógépnek, amelyre az eseményt kiváltó októl függő, előírt időn belüli választ kell küldeni. A környezetből egyidejűleg több eseményhez kapcsolódó kérés is érkezik a számítógéphez és ebben az esetben is teljesülnie kell az előírt időn belüli válaszadásnak [5].

Egy valós idejű rendszer általános felépítését a 3.1. ábrán láthatjuk.



3.1. ábra. Egy valós idejű rendszer általános modellje (forrás [5])

A környezetéből érkező jeleket érzékelők szolgáltatják. Ezeknek feladata a működtetett (irányított) rendszer, környezet állapotáról információkat adni. A feldolgozás eredményét (választ) előírt idő alatt kell a beavatkozó eszközre juttatni. (3.1. ábra).

Egy valós idejű rendszer fontosabb általános jellemzői a következők:

- a válaszidők soha sem haladhatják meg a számukra meghatározott időkorlátot;
- a számítógép az adatokat időlépték szerint gyűjti: a gyorsabb folyamatokról sűrűbb mintavételezéssel, míg a lassúbb folyamatokról ritkább mintavételezéssel történik az adatgyűjtés;
- a számítógép a folyamattal A/D, D/A és digitális interfészekon keresztül tart online kapcsolatot;
- ciklikus működés;
- ha a számítógép a rendelkező jel kiadásával a folyamatba is beavatkozik, akkor a folyamat időléptékéhez kell igazodnia;
- az előírt *időkorlát* (határidő, deadline) az irányított technológiai berendezés, folyamat időviszonyaiból határozható meg.

A valós idejű rendszer adatfeldolgozó programjának nagysága az általa megvalósítandó feladatot leíró algoritmustól függ, de funkcionálisan tartalmazza a 3.1. ábra szerinti három részfunkciót is, amelynek alapján folyamat jól meghatározható. Ez szekvenciális programokban nem mindig kivitelezhető. Ezeket a folyamatokat a valós idejű rendszer specifikusan kialakított operációs rendszere kezeli. A folyamatok futtatása rendszerint *ciklikus*, ebből következik, hogy egyik alapvető időtényezője a rendszernek éppen a *ciklusidő* nagysága.

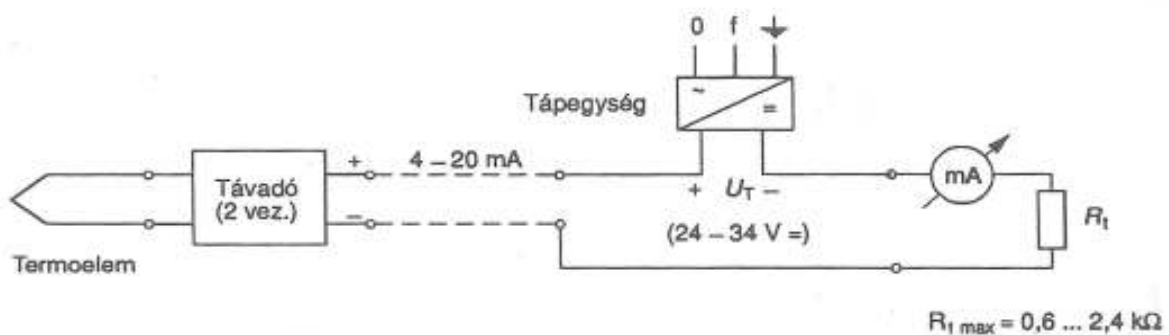
A valós idejű rendszerek általában két csoportra oszthatók:

- **szigorúan valós idejű (hard real-time)** rendszerek, amelyek specifikációja egyértelműen rendszerhibának tekinti valamely időkövetelmény be nem tartását. Ez azt jelenti, hogy a szigorúan valós idejű rendszerek normál működése során nem engedhető meg, hogy valamilyen időkövetelmény teljesítése elmaradjon. Ha mégis bekövetkezne ilyen esemény, a rendszernek a kidolgozott stratégiának megfelelően kell reagálnia, semmiképp sem szabad határozatlan állapotba kerülnie.
- **lazán valós idejű (soft real-time)** rendszerek, amelyek körében az ilyen mulasztást csak a rendszer működőképességének vagy teljesítményének csökkenéseként fogjuk fel, és erre az esetre is specifikáljuk a rendszer viselkedését.

### 3.3. Intelligens (smart) távadók

A klasszikus ipari irányítórendszerek esetében a technológia helyszínén alkalmazott hagyományos távadók az érzékelt fizikai mennyiségeket szabványos analóg villamos jelekké alakítják (0 – 10 V, 0 – 20 mA vagy 4 – 20 mA) és ezeket közvetlenül továbbítják szabályozó megfelelő bementére. A leggyakrabban használt jeltartomány a 4 – 20 mA-es tartomány, egyrészt azért nem a 0 – 20 mA-es, mert így lehetőség van különbséget tenni a minimális érzékelési szint (ebben az esetben ez 4 mA) és az érzékelt jel hiánya (0 mA) között, másrészt pedig ezek a távadók csak két vezetékot igényelnek. A tápvezeték egyben jelvezeték is. Ezeket a távadókat a szakirodalom *élőnullás* távadóként ismeri.

A 3.2. ábrán egy ilyen kétvezetékes 4 – 20 mA-es hőmérséklet távadó elvi vázlatát láthatjuk [6].

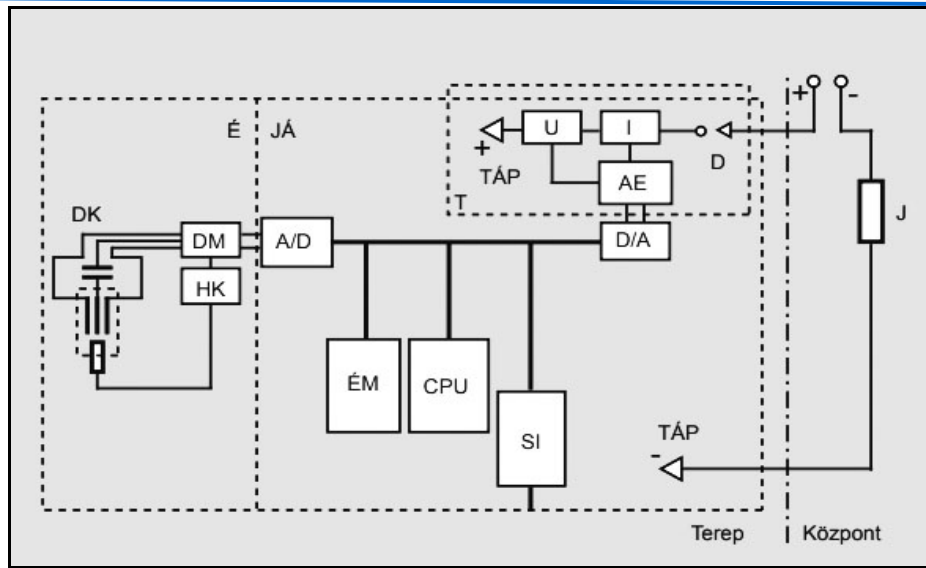


3.2. ábra. Kétvezetékes hőmérséklet távadó

Az egyre korszerűbb irányítóberendezések mind több és több távadót igényelnek. Egy bonyolultabb rendszer esetében ezek a távadók nagyon sok vezetékot igényelnének. Az intelligens távadók az analóg jelszintek mellett információikat digitális formában is képesek továbbítani az ipari kommunikációs hálózaton (Hart, CAN, Profibus, Profinet, stb.) keresztül. Így kevesebb vezetékre van szükség, mert közvetlenül a buszra csatlakoztathatók. Intelligenciájuk abban is megmutatkozik, hogy míg az analóg távadókat az irányítórendszer kénytelen minden pillanatban lekérdezni, az intelligens távadók képesek kommunikációt kezdeményezni a vezérlővel, amikor állapotváltozás történik.

#### 3.3.1. D-generációs távadók

A továbbiakban néhány ilyen távadó működési elvét tárgyaljuk. Intelligenciájukat tekintve a távadókat generációs csoportokba szokás sorolni. A *D-generációs* távadók jellegzetessége a digitális alapú belső működés és az analóg jelfelülettel való kapcsolat. Ezek a távadók ugyan digitális jelfeldolgozásra képesek, de csak analóg kimenettel rendelkeznek. A klasszikus távadókat képesek helyettesíteni. A digitális működésnek köszönhetően LCD kijelzője is lehet és konfigurálhatóak valamilyen soros interfészen keresztül (3.3. ábra [7]).



3.3. ábra. D-generációs nyomáskülönbség távadó (forrás [7])

A 3.3. ábra jelölései a következők:

É – kapacitív nyomáskülönbség-érzékelő; JÁ – digitális (D-generációs) jelátalakító; DK – differenciálkondenzátor; DM – demodulátor; HK – hőmérséklet-kompenzáció; A/D – analóg-digitál átalakító; D/A – digitál-analóg átalakító; CPU – mikroprocesszor vagy mikrokontroller; ÉM – a távadó adatait tartalmazó memória; SI – soros vonali illesztő; J – jelfogadó (a készülékek és a jelvezetékek együttes ellenállása 500...1600 Ω); U – feszültségstabilizátor; I – áram stabilizátor (4 mA); D – védődióda; AE – analóg erősítő; T – tápenergia-átalakító egység.

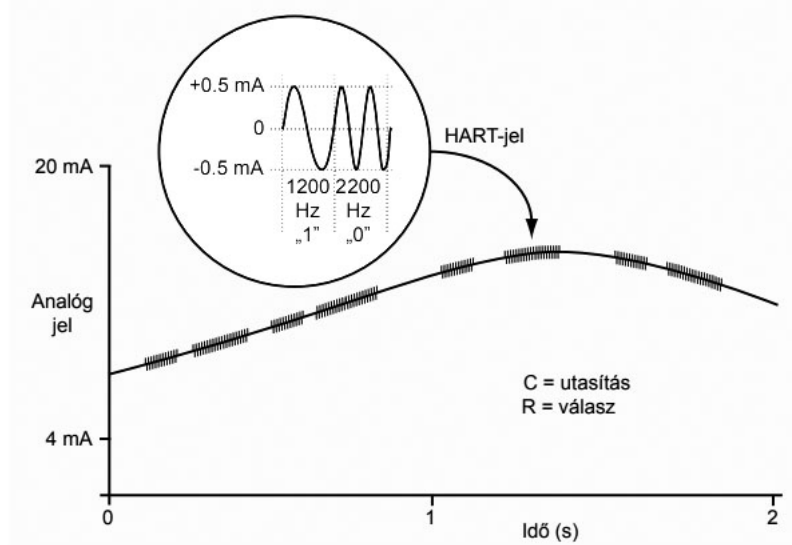
Látható, hogy az érzékelés analóg módon történik, a feldolgozás viszont digitális működésű, a soros interfészen keresztül programozható, konfigurálható, a szabályozó felé pedig szintén analóg 4 – 20 mA-es áramjelet továbbít. A távadóban külön egységet képez az érzékelő és a jelátalakító. Az érzékelő általánosságban analóg elvű, ezért az itt keletkező primer jelek analóg villamos jelek és A/D átalakítással jutnak be a digitális jelátalakító fokozatba. Ugyancsak analóg az esetleges korrekciót biztosító jel és természetesen az esetenként szükséges tápfeszültség is.

### 3.3.2. Kommunikáció a távadóval. A HART protokoll működése [7]

A D-generációs távadó programozása, konfigurálása jelentős előny az előző generációs, klasszikus távadókhoz képest. A D-generációs a távadó esetében már említett mérés határ-változtatása, a paraméterek megváltoztatása és konfigurálása valósítható meg. Ennek megoldására a távadóval digitális jelformában kommunikálni kell. A kommunikáció vagy közvetlenül a jelvezetéken, az

analóg jelre szuperponált, impulzus- vagy szinuszos jelhordozójú jellel, vagy külön kábelen lehetséges.

A HART (Highway Addressable Remote Transducer) protokoll a terepi kommunikációs céllal a Rosemount cég dolgozta ki 1991-ben az intelligens nyomás-távadók számára. Az analóg jelátvitelt egészíti ki, a kétvezetékes, 4 – 20 mA-es. Tipikus alkalmazása a digitális elven működő, analóg jelű (D-generációs) kétvezetékes távadók és helyzetbeállítók távolsági lekérdezése, az alapértékek és paraméterek távbeállítása és a távolsági készülékdiagnózis. A HART-protokoll master-slave topológiájú, pont-pont kapcsolatú. Az üzenetet a központi irányítóegység (pl. PC) vagy a hordozható kézi kommunikátor indítja, és a megcímzett terepi készülék az üzenetet fogadja és megválaszolja. Az üzenet 3 byte ellenőrző információt, mintegy 2 byte címezést és 23 – 37 byte adatmezőt tartalmaz. Az átviteli sebesség 1200 bit/s. Egy üzenetváltás ciklusideje kb. 500 ms.



3.4. ábra. A HART-protokoll jelalakja (forrás [7])

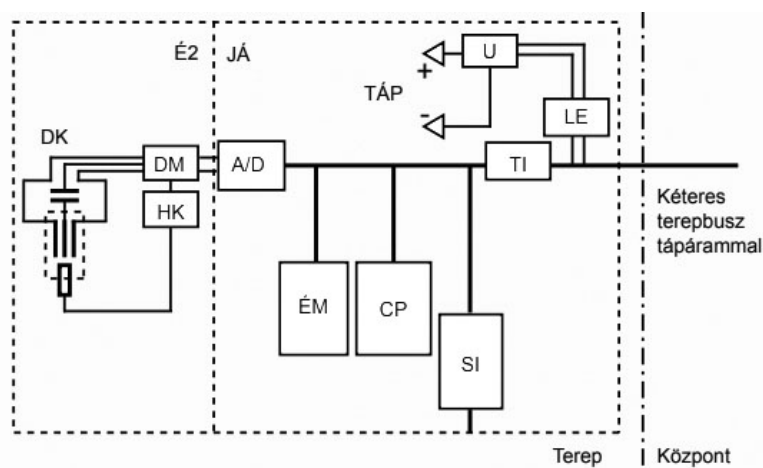
Az átvitel elve az analóg áram- vagy feszültségjelre szuperponált szinuszos jel (3.4. ábra). A Bell 202-eljárás szerint a „0” bitinformációt 2200 Hz frekvenciájú, míg az „1” bitinformációt 1200 Hz frekvenciájú szinuszos jel hordozza. Az amplitúdó feszültségátvitel esetén 400 – 600 mV, áramjelátvitelkor pedig 0,8 – 1,2 mA.

Az üzenetet a központi irányítóegység (pl. PC) vagy a hordozható kézi kommunikátor állítja elő. Más protokoll (pl. RS232) szerint előállított üzenetet *modem* alakítja át a HART-protokoll formájára. A fogadókészülékben is található egy modem (ún. HART-chip), ami a jelvezetéken

érkező szuperponált szinuszos jelet leválasztja és átalakítja a készülék belső protokolljának megfelelő formára, majd ugyanezt visszafelé is megteszi a válasz továbbítása céljából.

### 3.3.3. E- és F-generációs távadók

Ezek a távadók digitális működésűek (csak az érzékelés analóg) és a kimenőjelük is digitális. Illeszthetők a szabványos ipari terepi buszrendszerekhez (HART, PROFIBUS, MODBUS, CAN stb.), de az F-generációsok akár ipari VLAN hálózathoz is. Analóg jelrendszerekben már nem alkalmazhatók, mint a C-generációs vagy D-generációs távadók. A 3.5. ábrán egy ilyen távadó elvi vázlatát láthatjuk.



3.5. ábra. E-generációs távadó elvi felépítése (forrás [7])

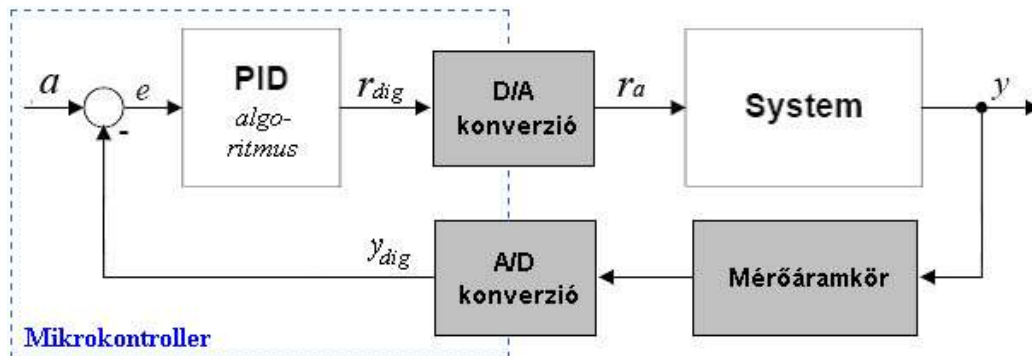
A 3.5. ábrán használatos jelölések a következők:

É2 – kapacitív nyomáskülönbség-érzékelő, JÁ – digitális (E-generációs) jelátalakító, DK – differenciálkondenzátor, DM – demodulátor, HK – hőmérséklet-kompenzáció, A/D – analóg-digitális átalakító, TI – terepbusz-illesztő, CP – mikroprocesszor vagy mikrokontroller, ÉM – az érzékelő adatait tartalmazó memória, SI – soros vonali illesztő, U – tápfeszültség-stabilizátor, LE – tápáram-leválasztó elektronika.

Az E-generációs távadó táplálásának 2 változata van. A 3.5. ábrán a kétvezetékes terepi buszos változatot láthatjuk, amelynél a táplálás a jelvezetéken keresztül történik. Létezik négyvezetékes megoldás is, ahol külön tápegység biztosítja a távadó energiaellátását. Ebben a megoldásban a távadó energia-felvétele nem korlátozott. Ilyenkor gyakran több érzékelő is csatlakozik egy jelátalakítóhoz.

### 3.4. Digitális PID szabályozók [8]

A számítógépes folyamatirányítás sok esetben megköveteli a rendszer szereplőitől a digitális működést. Nincs ez másképp a szabályozókkal sem. Azok a szabályozók, amelyek a számítógépes folyamatirányítás közvetlen felügyelete alatt működnek, digitális felépítésűek, programozhatóak, konfigurálhatóak. A 3.6. ábrán egy digitális PID szabályozó elvi működési vázlatát láthatjuk.



3.6. ábra. Egy digitális PID szabályozó elvi felépítése (forrás [8])

A korszerű irányítórendszerek digitális berendezései rendszerint valamilyen mikroszámítógép alapúak, amelyeket a célberendezés funkciónak ellátására programoznak. Így van ez a digitális PID szabályozó esetében is. A digitális szabályozó megvalósítását szem előtt tartva (3.6. ábra) könnyen rájöhethetünk, hogy magát a PID szabályozót és a szabályozó algoritmust egy mikrokontroller, illetve az azon futó program fogja realizálni. Egyszerű esetben az  $a$  alapjelet is képezhetjük magával a programmal, vagy például megadhatjuk a mikrokontrollernek valamelyik interfészén keresztül egy digitális számként. Tovább gondolkodva arra juthatunk, hogy ha már a szabályozandó rendszert (System) realizáltuk, akkor gondoskodni kell a szabályozott mennyiség ( $y$ ) méréséről, illetve a mért érték analóg-digitális konverziójáról és a szabályozó algoritmus által kiszámolt  $r_{dig}$  beavatkozó jel digitális-analóg konverziójáról. Szerencsés esetben e két utóbbi művelet is megvalósítható a mikrokontroller saját A/D illetve PWM fokozatával, így csak egy mérőáramkört (érzékelő és átalakító) kell még illesztenünk.

Vagyis a digitális PID szabályozó nem más, mint egy mikroszámítógép PID algoritmussal felprogramozva. Az illesztést az analóg környezettel az A/D és D/A átalakítók végzik.

Az algoritmus elkészítésének alapját a PID szabályozó átmeneti függvénye adja (3.1).

$$y(t) = K \left[ r(t) + \frac{1}{T_I} \int_0^t r(t) dt + T_D \frac{dr(t)}{dt} \right] \quad (3.1)$$

Az előző összefüggésből kiindulva a kimeneti mennyiséget digitalizáljuk  $n$ -re, majd  $n+1$ -re és a kettő különbségéből a következő rekurzív formulát kapjuk:

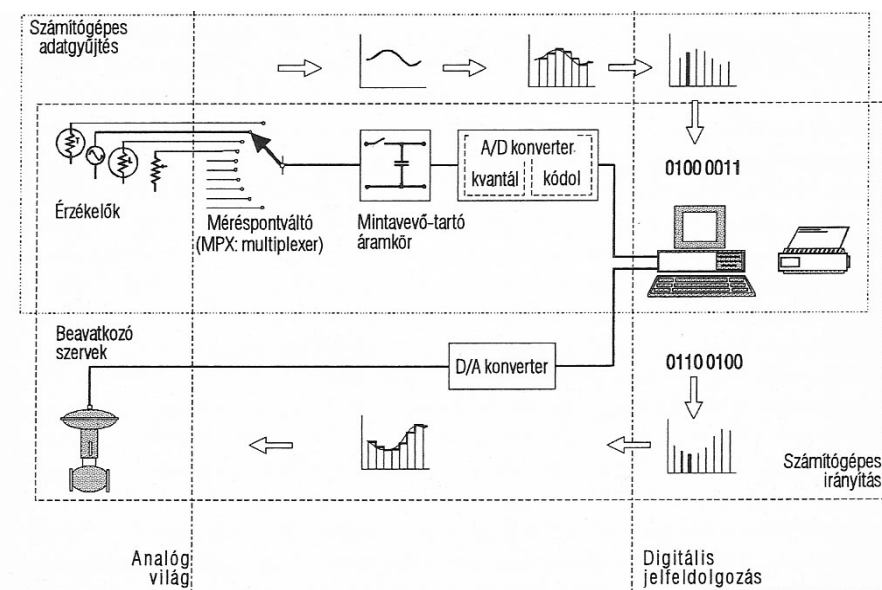
$$\begin{aligned} y_n &= K(r_n + \frac{1}{T_I} \sum_{i=1}^n r_i \Delta t + T_D \frac{r_n - r_{n-1}}{\Delta t}) \\ y_{n-1} &= K(r_{n-1} + \frac{1}{T_I} \sum_{i=1}^{n-1} r_i \Delta t + T_D \frac{r_{n-1} - r_{n-2}}{\Delta t}) \\ y_n &= y_{n-1} + K \left[ (r_n - r_{n-1}) + \frac{\Delta t}{T_I} r_n + \frac{T_D}{\Delta t} (r_n - 2r_{n-1} + r_{n-2}) \right] \end{aligned} \quad (3.2)$$

A 3.2 relációkban a  $K$ ,  $T_I$ ,  $T_D$  értékeket a szabályozott folyamat jellegéhez kell megválasztani figyelembe véve a 2.7.5. fejezet szerinti elveket.

### 3.4. Mérésadatgyűjtő rendszerek

A számítógépes adatgyűjtés az irányításnak az a formája, amikor mérjük a folyamat jellemzőket, digitálisan feldolgozzuk és adott esetben akár beavatkozást is kezdeményezhetünk a számítógépes folyamatirányító rendszerrel. Akár csak adatgyűjtésről, akár irányításról van szó a folytonos értékészletű és folyamatos analóg jelekből diszkrét, időben szakaszos digitális jelet kell előállítani a számítógép számára. (Az átalakítást végző A/D és D/A átalakítók a tananyag első részében már foglalkoztunk.)

A 3.7. ábrán egy számítógépes adatgyűjtő és irányító rendszer elvi vázlatát láthatjuk.



3.7. ábra. Számítógépes adatgyűjtő és irányító rendszer (forrás [1])

Összefoglalva az előbbieket, meghatározhatjuk a számítógépes adatgyűjtő és folyamatirányító rendszer fontosabb feladatait. Ezek a következők:

- Mérés, adatgyűjtés;
- Kiértékelés;
- Döntéshozatal;
- Beavatkozás;
- Dokumentálás;
- Archiválás;
- Folyamatállapot vizualizálás (ember-gép kapcsolat, HMI, SCADA rendszer).

Figyeljük meg, hogy az első négy feladat, amelyek egyben a klasszikus irányítás részműveletei, kiegészül további fontos eseményekkel. A számítógépes folyamatirányító rendszer a mérési eredményeket képes tárolni és archiválni. Ez több szempontból is fontos. Egyrészt nyomon követhető a gyártási folyamat, másrészt pedig bármikor visszakereshetők az adatok, ha utólag valamilyen gyártási probléma jelentkezik. Ezeknek az adatoknak az elemzése alapján kideríthető a hiba oka. Ennek alapján pedig kiküszöbölhető, hogy a jövőben ne fordulhasson elő ilyen hiba.

#### ***3.4.1. Korszerű mérésadatgyűjtő rendszerek. NI DAC rendszer***

A mérésadatgyűjtést tehát számítógép végzi. Ez lehet akár PC alapú számítógép vagy laptop, de lehet valamilyen speciálisan erre a célra fejlesztett célszámítógép. A nagy ipari irányítórendszereket gyártó cégek (Pl. Siemens, National Instruments, stb.) mérésadatgyűjtés céljára készítenek ilyen számítógépeket. Bármilyen is legyen ez a számítógép, képesnek kell lennie az érzékelők felől érkező jeleket kezelni. E célból a PC típusú számítógépekhez mérőmodulokat kell csatlakoztatni. Ilyen típusú mérőmodulok például a PC-LAB vagy a National Instruments által gyártott NI DAC modulok. Egyes mérésadatgyűjtő rendszerek az ipari kommunikációs hálózaton keresztül is csatlakozhatnak a számítógéphez, vagy akár a vezeték nélküli megoldás (VLAN) is kivitelezhető.

A mérésadatgyűjtő számítógépek természetesen programozhatóak. Programozásuk többféleképpen is lehetséges:

- gépi kódban,
- assembly nyelvben,
- magas szintű programozási nyelven,
- grafikus programozással

Újabban a legkorszerűbb rendszerek a programozása már szinte csak grafikus programozási fejlesztőkörnyezetben történik. Ilyen rendszerek például a *LabView*, *Test Point*, *Visual Designer*,

stb. A továbbiakban részletesebben a National Instruments által fejlesztett *NI DAC* mérésadatgyűjtő rendszerek és a *LabView* alkalmazással foglalkozunk.

Az *NI DAC* adatgyűjtő modulok szabványos csatlakozási kialakításuknak köszönhetően közvetlenül illeszthetők PC-hez vagy laptophoz. A használatos slotok, illetve csatlakoztatási módok: PCI, PCI Express (PCK-hez), PXI (NI számítógépekhez), PCMCIA (laptophoz), USB, WLAN, Ethernet. Az adatgyűjtő rendszer fontosabb elemei a következők:

- érzékelők és átalakítók,
- jelkezelés, jelkondicionálás,
- DAQ hardver modul(ok),
- eszközközelők és szoftvermodulok,
- FPGA modulok (programozható hardverelemek),
- Compact RIO (folyamatirányító rendszer).

A 3.8. ábrán egy *NI PXI* alapú mérésadatgyűjtő rendszert, a 3.9. ábrán pedig egy *Compact RIO* folyamatirányító egységet láthatunk.



3.8. ábra. NI PXI mérésadatgyűjtő rendszer



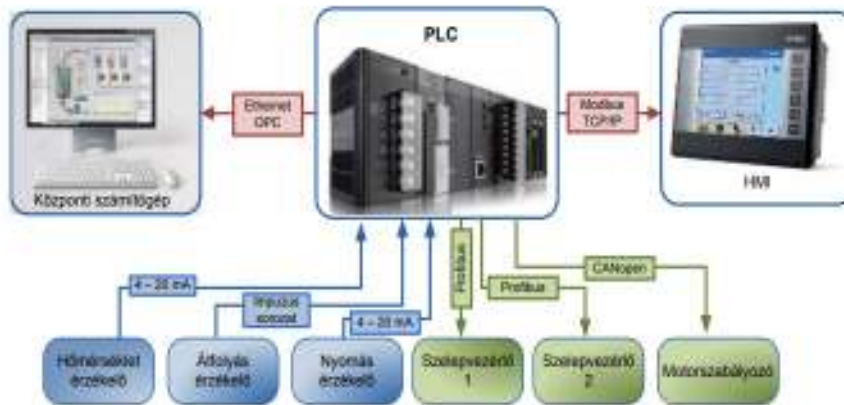
3.9. ábra. Compact RIO rendszer

### 3.5. SCADA rendszerek

Elnevezésük a **Supervisory Control and Data Acquisition** angol rövidítésből származik. Feladata elsősorban a különböző ipari folyamatok vizualizálása. Ahhoz, hogy ez megvalósítható legyen a SCADA rendszer mérés-adatgyűjtési funkciókat is el kell látnia: a jelek összegyűjtése az érzékelőktől, a begyűjtött információk feldolgozása, adatbázisok létrehozása és a folyamatok felügyelete. Vagyis tulajdonképpen egy mérésadatgyűjtő rendszer, felügyeleti és vizualizálási funkciókkal kiegészítve. A jelenlegi korszerű rendszereknél a mérésadatgyűjtő rendszer SCADA feladatokat is ellát.

A SCADA rendszerek képesek együttműködni a Microsoft operációs rendszerekkel (Windows Server, ODBC, OPC, SQL). Felügyeleti szoftverei kompatibilisek az IBM PC platformokkal. Folyamatirányítást nem végez, (az a PLC-k feladata) csak vizualizálási és beavatkozási lehetőséget biztosít.

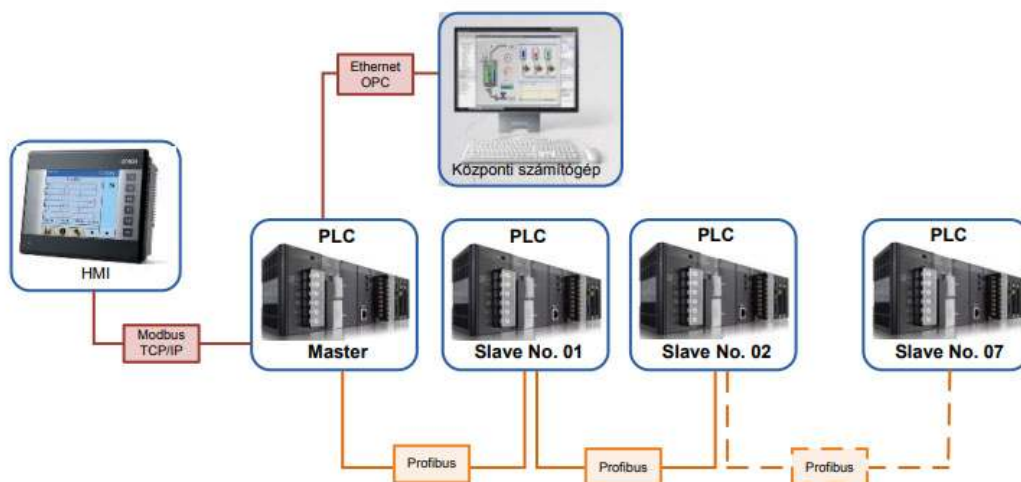
Architektúrájában a legegyszerűbb SCADA rendszer állhat egy ipari számítógépből, amely valamilyen adatgyűjtő modul segítségével információt gyűjt a folyamatról. Egyszerű SCADA struktúrának számít az egy PLC-ből, egy HMI-ből és egy központi számítógépből álló rendszer is, ahogyan azt a 3.10. ábrán is láthatjuk [22].



3.10. ábra. Egyszerű SCADA rendszer.

A 3.10. ábra szerinti megoldásban a folyamat alsó szintjén elhelyezkedő érzékelőkkel és beavatkozókkel a PLC tartja a kapcsolatot. A PLC elsődleges feladata, hogy elvégezze a folyamatirányításhoz szükséges tevékenységeket. A terepi eszközök felé a kapcsolat valamilyen szabványos áramjel, feszültségjel, vagy kommunikációs interfész segítségével történik. A PLC szolgáltatja az érzékelők adatait, mint a központi számítógép, mint pedig a HMI irányába. A példában a számítógép felé OPC-n keresztül valósul meg az adatcsere, míg a HMI felé Modbuson. A PC illetve a HMI lekérdezi a PLC memóriából az érzékelőkhöz tartozó változók értékeit, majd ezeket felhasználva feldolgozzák, illetve megjelenítik azokat.

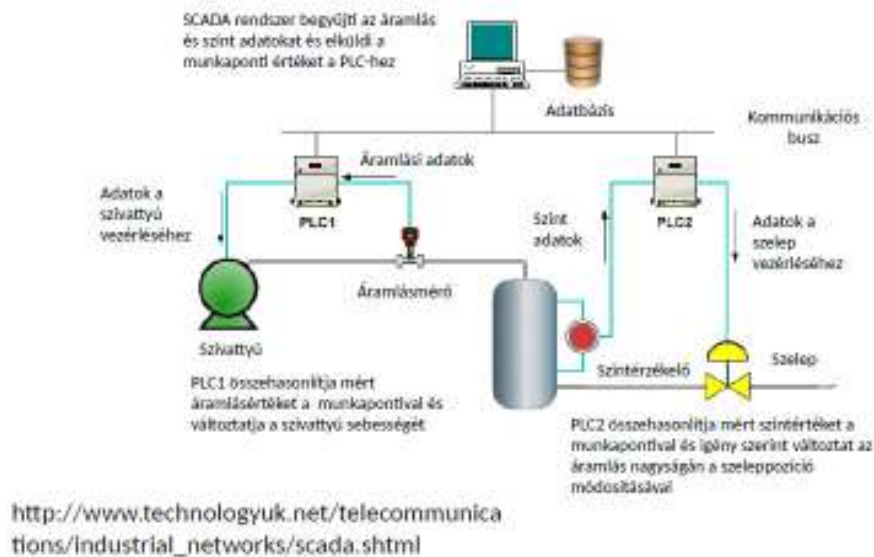
Sok esetben a folyamatirányító számítógép akár több eszközhöz is csatlakozhat, például ha egy terepi buszt használó (Profibus) rendszerhez csatlakozik. Egy ilyen rendszer látható a 3.11. ábrán.



3.11. ábra. SCADA rendszer Profibus hálózatban (forrás [22])

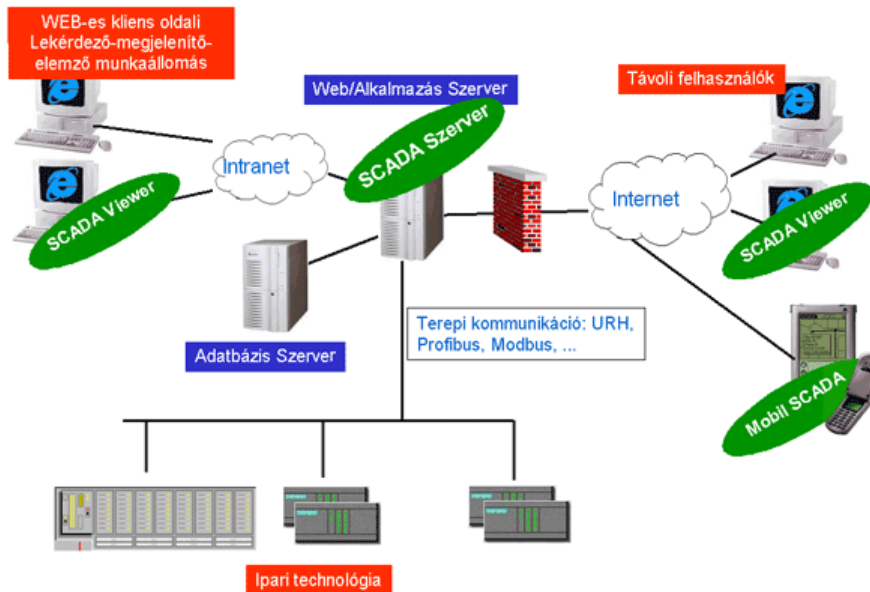
Ebben az esetben a folyamatirányító számítógép (PC) közvetlenül a Master PLC-re csatlakozik, amely azt kiszolgálja a szükséges adatokkal. Hozzá kapcsolódnak továbbá a terepi buszon (Profibus) keresztül a szolga PLC-k is. Így amikor a központi számítógép (PC) lekérdezné az adatokat a „Slave No. 1” PLC-től, akkor azt a „Mester” PLC-től kérdezi le, amely pedig lekérdezi az említett PLC-től. Sok esetben a mester PLC folyamatosan kéri az adatokat a szolga eszközöktől, így amikor a számítógép hozzáfordul az adatokért, akkor az azonnal már az aktuális adatokkal válaszolhat [22].

A továbbiakban bemutatunk néhány gyakorlatban megvalósított SCADA rendszert. A 3.12. ábrán egy tartály folyadékszint felügyeleti rendszerét láthatjuk.



3.12. ábra. Tartály folyadékszint felügyeleti rendszer.

Napjaink egyik legnépszerűbb folyamatirányító és felügyeleti szoftvere webSCADA (3.13. ábra). A kifejlesztett rendszer minden olyan vállalatnál segíti az üzemirányítást, ahol adatokat kell összegyűjteni, feldolgozni és vizualizálni, valamint kritikus gyártási folyamatait vezérelni, szabályozni és irányítani.



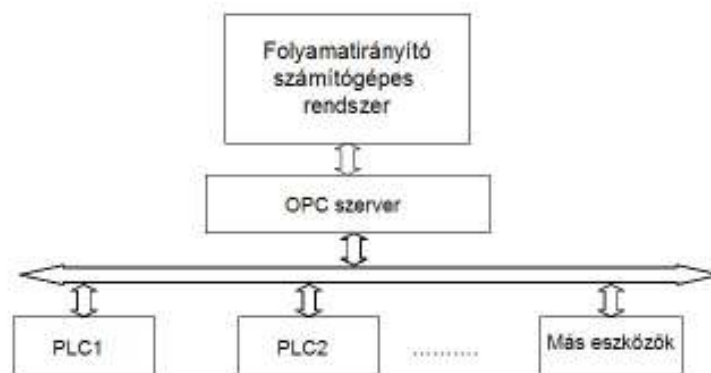
3.13. ábra. A webSCADA rendszer (forrás <http://www.webscada.hu/Termekkek>)

A rendszer központi része a SCADA szerver, amely egyben a folyamatirányítás mérésadatgyűjtő egysége is. Kapcsolatban áll az adatbázis szerverrel és a technológiai berendezések PLC-ivel. Ezen kívül az Ethernet hálózaton keresztül kapcsolódhatnak a különböző kliens oldali lekérdező állomások, operátori és mérnöki munkaállomások. Megfelelő biztonsági tűzfalon keresztül távoli lekérdezések is lehetségesek, sőt akár mobil applikációval is kapcsolódhatunk a rendszerhez.

### 3.6. OPC szerver (Open Platform Communication) [22]

Egy ipari létesítményben számos olyan eszköz található, melyek egymás között információcserét valósítanak meg. Lehet itt szó PLC-PLC kapcsolatról, PLC-HMI kapcsolatról, PLC-SCADA rendszer kapcsolatról, komplex mérő és adatgyűjtő rendszerekről valamint egyéb magasabb szintű adatáramlásról. Az ipari eszközöket forgalmazó cégek gyártóspecifikusan saját eszközeikhez fejlesztenek kommunikációs protokollokat. Ezek a protokollok sok esetben csakis kizárólag egy gyártóhoz tartozó eszközök ismerik. A korszerű automatizálás elengedhetetlen része a számítógépes mérésadatgyűjtő és felügyeleti rendszerek (SCADA, HMI, stb.). Ezek a rendszerek az irányítási rendszerek legmagasabb szintjén helyezkednek el. Ahhoz hogy az információ a különböző gyártóktól származó, alsóbb szinteken elhelyezkedő eszközöktől (pl. PLC, adatgyűjtő egység) eljusson a legfelsőbb szinten lévő felügyeleti rendszerekhez, megfelelő szabványos kommunikációs protokoll alkalmazása szükséges. Külön nehézséget jelent, amikor a legfelső szintről több rendszer is szeretne az alsó szinten lévő információkhoz hozzájutni. Az összetettebb rendszereknél ilyen esetekben komoly anyagi és időráfordítást igényel a kommunikációs hálózat kiépítése. Az OPC szabvány az ilyen problémákra kínál megoldást.

Az OPC célja, hogy az ipari infrastruktúra esetében, szabványos formában szolgáltatssa a folyamatirányítás adatait. Az OPC alkalmazásával egy új réteg jelenik meg az automatizálási rendszer hierarchiájában. Az új réteg a PLC és fölötté elhelyezkedő felügyeleti rendszerek (SCADA, HMI, stb.) közé illeszkedik be, voltaképpen, mint egy kommunikációs hidat képezve közöttük. Ezzel megszűnik a közvetlen kapcsolat a PLC és a felügyeleti rendszerek között. Az OPC szerver tehát összekapcsolja a folyamatirányító rendszert a PLC-kkel és más ipari szabvány szerinti intelligens vezérlőkkel (3.14. ábra).



3.14. ábra. OPC szerveren keresztül összekapcsolt intelligens vezérlők

Az OPC szerver egyéb feladatai:

- Kezeli az adatbázist,
- Valós idejű adatgyűjtés,
- Grafikus megjelenítést biztosít (SCADA feladatokat is elláthat)
- Diagnosztikai funkciókat is képes biztosítani.

Az OPC igazi előnye a szerver-kliens közötti kapcsolatban rejlik. Ugyanis, egy szerverre több kliens is csatlakozhat, így többen hozzá tudnak férni a PLC-k, vagy más alsóbb szinten lévő eszközök által szolgáltatott adatokhoz.

Az OPC szerverek és kliensek is több kategóriába oszthatóak. Ezek a következők:

- Data Access (DA)
- Alarm & Event (A&E)
- Historical Data Access (HDA)

A legelterjedtebb adatmodell a *Data Access*, mely az alsó szinten elhelyezkedő eszközökön (PLC-en) valós idejű írást-olvasást tesz lehetővé. Az ipari OPC alkalmazások 99%-ban a DA-t alkalmazzák. Az OPC DA segítségével PLC memória területéhez tartozó változókat írhatunk, olvashatunk és figyelhetünk meg. Az adatfrissítés a kliens által definiált időközönként történik. Ha

valami kommunikációs hiba miatt a szerver nem tudná elérni az eszközt, megtörténhet, hogy nem a legfrissebb adatot szolgáltatja a kliens felé. Ezt a problémát az OPC DA az adathoz csatolt időbélyeggel tudja orvosolni.

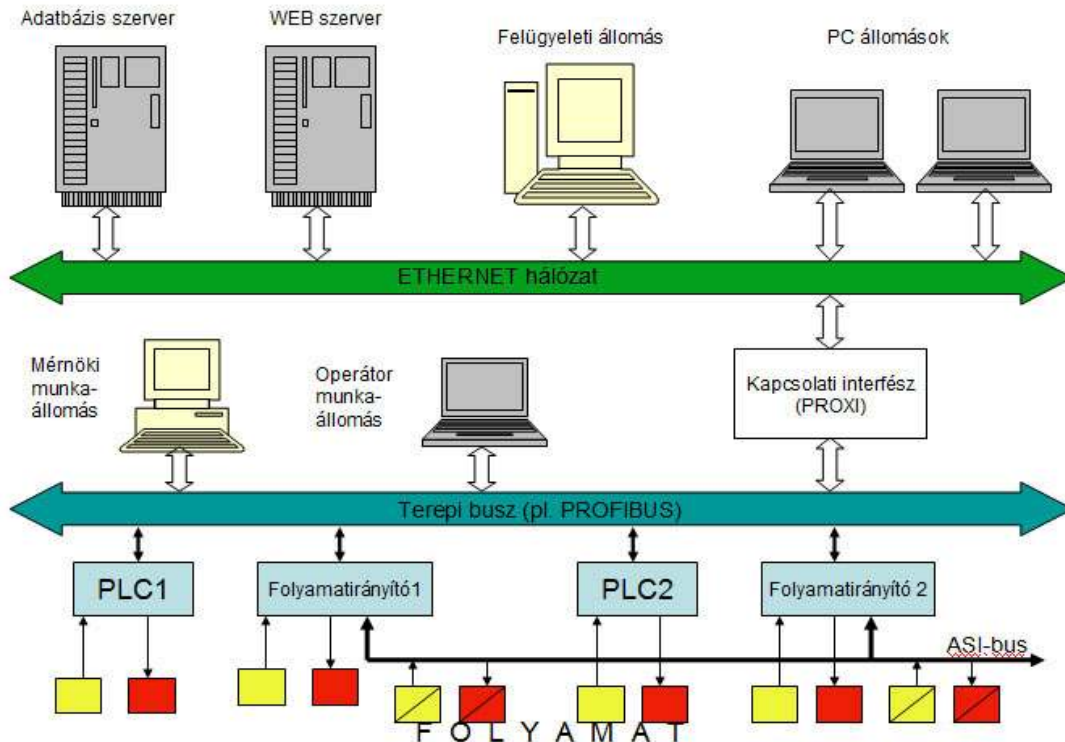
Az *Alarm & Event* (A&E) és a *Historical Data Access* (HDA) adatmodellek is a DA-n alapulnak. Az *Alarm & Event* interfész az OPC szabvány azon része, amely értesítések és riasztások generálására használható. Előre definiált események alkalmával a szerver üzeneteket küld a kliens felé, értesítve azt a megtörtént eseményről. Az esemény karakterisztikájától függően lehet értesítés vagy riasztás. A riasztás nagyobb prioritást, nagyobb fontosságú események jelzésére használják. A riasztás típusától függően lehetőség van nyugtázás kérésére is.

A *Historical Data Access* a harmadik adatmodell, amely szintén az OPC DA szerverre kapcsolódva szolgáltatja a már eltárolt adatokat. A kliens ebben az esetben a már tárolt adatokat kérheti le egy meghatározott időperiódusra. Ezek mellett lehetőség van lekérni egy vagy több változó értékét, amelyek egy bizonyos időbélyeghez tartoznak. Továbbá lehetőség van egy vagy több változóról származtatott adatok (pl. átlagérték, maximum, minimum, stb.) időszerinti lekérdezésére is.

### **3.7. Osztott intelligenciájú (DCS) rendszerek.**

A DCS (Distributed Control Systems) rendszerek, nagy és összetett rendszerek, folyamatok esetében alkalmazott folyamatirányító és felügyelő megoldások. A DCS rendszerek jelen vannak a folyamatautomatizálás és a gyártásautomatizálás terén egyaránt. Alkalmazásuk jellemző a kőolaj-finomítók, vegyi üzemek, villamos energiatermelés, a gyógyszer-gyártás, élelmiszerek és italok gyártása, acélgyártás és papír gyártás esetében. Ötödik generációs irányítási rendszernek is szokás nevezni [4]. A decentralizált rendszerekben több, egymással kommunikációs kapcsolatban lévő irányítóberendezés (számítógép, PLC, stb.) látja el a folyamat irányításához, felügyeletéhez kapcsolódó funkciókat. Egy-egy számítógép vagy PLC feladata jól meghatározható. Ezek technológia közvetlen közelében vannak telepítve és az ipari kommunikációs buszrendszeren keresztül kommunikálnak egymással. A vezérlési adatokat hagyományos vagy intelligens (SMART) távadóktól kapják. Ezeket feldolgozzák, kiértékelik és továbbítják a központi adatbázis felé. Sok esetben az irányításhoz szükséges alapadatokat szintén egy központi adatbázistól kapják.

Az irányítási folyamatban részt vevő berendezések a rendszertervező által meghatározott részfeladatokat megfelelő hierarchiába szervezve látják el, amelynek három szintje terjedt el: *alsó szint (Device Level)*, *középső szint (Control Level)* és *felső szint (Process Control Level)*. (3.15. ábra)



3.15. ábra. Ötödik generációs DCS rendszer

Az *alsó szint* vagy más néven *technológiai szint* a folyamathoz legközelebb álló berendezéseket foglalja magába. Az itt jelen lévő eszközök (PLC-k vagy folyamatirányítók) hagyományos érzékelőkön vagy intelligens távadókon keresztül gyűjtik a folyamat információkat. Ugyancsak ezek az eszközök küldik a beavatkozó jeleket a végrehajtó szervek felé. Az intelligens távadók és beavatkozók az *ASI* (Actuator Sensor Interface) buszon csatlakoznak a vezérlőkhöz. Így jóval kevesebb vezeték szükséges a kapcsolat megvalósításához. Ezeknek az eszközöknek tehát a legfontosabb feladatuk az érzékelők jeleinek valós idejű feldolgozása, valamint a beavatkozó jelek kiadása.

A *középső szint* vagy *automatizálási szint* feladata az intelligens gyártóberendezések PLC-k, CNC-k, robotok, szállítópályák közötti kommunikációs kapcsolat megvalósítása. A kommunikáció megvalósulhat hagyományos terepi buszrendszeren (MODBUS, PROFIBUS, stb.) keresztül is, de a jelenlegi korszerű DCS rendszerek már az ipari Ethernet (PROFINET, EtherCAT, stb.) hálózatot használják. Ugyancsak ezen a szinten vannak az operátori és mérnöki munkaállomások. A kettő között az a különbség, hogy míg az operátori munkaállomáson elindíthatók, megállíthatók, lekérdezhetők, vizualizálhatók bizonyos folyamat állapotok, addig a mérnöki munkaállomásról a gyártási paraméterek is módosíthatók. Ha ezen a szinten a hagyományos terepi buszrendszeren keresztül történik a kommunikáció, akkor az adatoknak a felső szintre való továbbításához szükséges egy *kapcsolati interfész* vagy más néven *PROXI*.



**EFOP-3.5.1-16-2017-00017**

**„NYE-DUÁL- Új utakon a duális felsőoktatással a Nyíregyházi Egyetemen,  
az Északkelet-Magyarországi térség felemelkedéséért”**

A hierarchia felső szintje a *termelésirányítási szint*. Ezen a szinten vannak a különböző szerverek (adatbázis szerver, OPC szerver, WEB szerver), a központi felügyeleti állomás vagy irányító központ és a PC munkaállomások. Ezek az eszközök vezetékes vagy vezeték nélküli Ethernet hálózaton keresztül kommunikálnak. Itt a technológia felől érkező, részben feldolgozott adatok koncentrálódnak. Ezek alapján az optimális (vagy optimum közeli) üzemeléshez szükséges beavatkozások valamilyen algoritmus alapján generálódnak. További feladatuk a kezelők folyamat-felügyeleti tevékenységéhez szükséges adatok, valamint a vállalatirányítási rendszerhez (készletnyilvántartás, rendelésállomány stb.) szükséges csatlakozási lehetőségek biztosítása.

A DCS rendszerek fő jellemzője a nagy megbízhatóságot biztosító redundáns vezérlők, I/O egységek, valamint hálózati eszközök alkalmazása. A redundancia biztosítása nagyon fontos azoknál a termelési rendszereknél, amelyek leállása valamilyen meghibásodás esetén óriási veszteségeket okozna, vagy biztonsági kockázatot jelentene.

## **4. VALÓS IDEJŰ IPARI ETHERNET RENDSZEREK**

Az ipari irányító- és kommunikációs rendszerek gyártói számos ipari Ethernet technológiára épülő megoldást fejlesztettek ki, amelyek jelentős szerepet játszanak a mai, korszerű ipari elosztott vezérlési és folyamatirányítási alkalmazásokban. A hagyományos terepi buszos kommunikáció korlátozott adatátviteli sebessége gátat szab azon korszerű technológiai megoldások alkalmazásának, amelyek nagyobb információmennyiséget és átviteli sebességet igényelnek. A közös Ethernet platform mellett, hogy kétirányú, 100 Mb/s-os bitsebességet képes szolgáltatni, egy lépéssel közelebb hozta egymáshoz a különböző gyártók specifikus termékeit. Az általunk ismer irodai Ethernet azonban nem minden esetben alkalmas ipari kommunikációra.

### **4.1. Az irodai Ethernet alapjai [11]**

A hagyományos irodai Ethernet, a buszon kialakuló ütközések miatt képtelen a valós idejű adattovábbításra. Az irodai Ethernet nem determinisztikus, ami azt jelenti, hogy a válaszidők véletlenszerűek, nem kiszámíthatóak. Márpedig az irányítástechnikában nem elfogadható például az a tény, hogy egy érzékelő által jelzett figyelmeztetés vagy hibaesemény ne legyen kellő időben lereagálva. Az ipari vezérlőeszközök különleges, *valós idejű* kommunikációs körülményeket igényelnek.

Az Ethernet hálózat ötlete 1976-ban a Xerox cég mérnöke, Robert Metcalf nevéhez fűződik. Ő volt az, aki egy számítógépes konferencián (National Computer Conference) bemutatta az első Ethernet hálózatra vonatkozó elképzelésének vázlatát. A Xerox-féle Ethernet olyan sikeres volt, hogy 1978-ban szabadalmaztatták, majd az IEEE (Institute of Electrical and Electronics Engineering) elfogadta és 1983-ban kidolgozta a máig is használatos Ethernet hálózati szabvány alapját az IEEE 802.3-at [9]. Az azóta eltelt időben az Ethernet sokat fejlődött, főleg az átviteli sebességet illetően (10, 100, 1000 Mb/s, 10 Gb/s hálózatok), de a vezeték nélküli hálózat is jelentős előrelépést jelentett (IEEE802.11 WLAN protokoll).

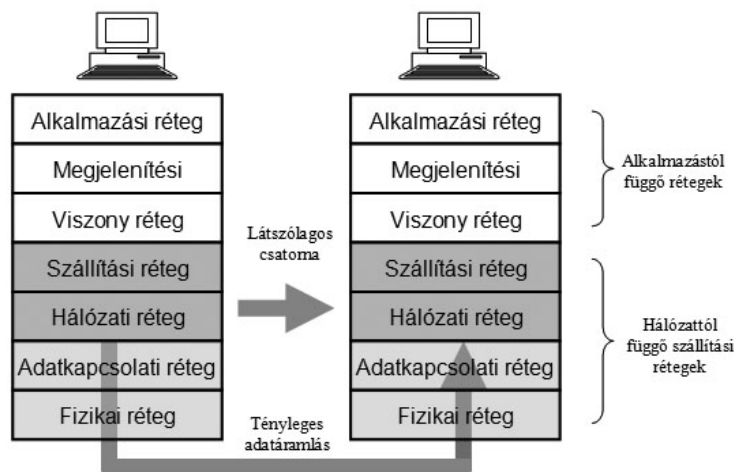
A vezeték nélküli világ egyre nagyobb teret hódít az ipari kommunikációs rendszerek területén is. Lehetővé vált olyan helyeken is érzékelőket és távadókat elhelyezni, ahol egyébként a vezetékek

használata miatt ez eddig lehetetlen volt vagy csak nagyon nehezen és költséges megoldások mellett lehetett megvalósítani.

#### 4.1.1. OSI és TCP/IP modellek

Az Ethernet hálózati kommunikáció a szabványosított hétrétegű OSI modellen alapszik. Az OSI modell a különböző protokollok által nyújtott funkciókat egymásra épülő rétegekbe sorolja. Minden réteg csak és kizárólag az alsóbb rétegek által nyújtott funkciókra támaszkodhat, és az általa megvalósított funkciókat pedig csak a felette lévő réteg számára nyújthatja. A kezdeményezés legfontosabb eredménye az volt, hogy meghatározták azokat a specifikációkat, amelyek pontosan leírták, hogyan léphet egy réteg kapcsolatba egy másik, azonos szintű réteggel. Ez gyakorlatilag azt jelenti, hogy egy adott gyártó által készített réteg programja együtt tud működni egy másik gyártó által készített programmal. A végeredmény egy olyan rendszer lett, mely két alapelvből állt össze (4.1. ábra):

- kialakult a 7 rétegből álló struktúra,
- meghatározták az egyes résztvekenységeket ellátó protokollokat.

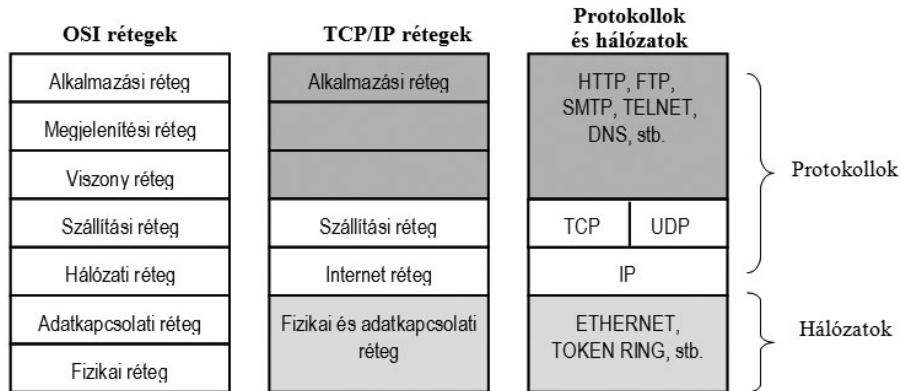


4.1. ábra. Az információ OSI modell szerinti látszólagos és tényleges áramlási útja

Megjelenése után az OSI modellt igencsak sok kritika érte. Egyrészt azért, mert túl bonyolultnak tartották, másrészt azért, mert protokolljai nem teljesen tökéletesek. A viszony réteget alig használja néhány alkalmazás, a megjelenítési réteg pedig szinte teljesen üres. Az időzítés sem volt túl szerencsés. Mire az OSI protokollok megjelentek, addigra a Berkeley-féle UNIX TCP/IP protokolljai már széles körben elterjedtek a kutatóegyetemeken [10].

A TCP/IP modell lényegében az OSI modellnek egy egyszerűsített változata, amely csak négy réteget tartalmaz, de megvalósítja mindazokat a funkciókat, amelyek szükségesek a biztonságos

kommunikációhoz. A két modell közötti különbséget és az alkalmazott protokollokat a 4.2. ábra mutatja.



4.2. Az OSI és a TCP/IP hivatkozási modell és protokolljai *(forrás [11])*

#### 4.1.2. A CSMA/CD protokoll működése

Az IEEE 802.3 szabvány szerinti Ethernet hálózatba kapcsolt állomások mindegyike egyenlő jogokkal rendelkezik a közeg használatát illetően. A hozzáférés ellenőrzése és vezérlése a CSMA/CD (Carrier Sense Multiple Access with Collision Detection) közegjelzési módszer szerint történik, amelynek röviden a lényege a következő.

Mindegyik állomás állandóan figyeli a hálózati forgalmat. Ha talál olyan adatcsomagot, amelyet neki címeztek leveszi, a többit figyelmen kívül hagyja. Az adatküldésre készülő állomás is hallja a közegen zajló forgalmat, és ha azt érzékeli, hogy valaki már adásban van, azaz vivőt érzékel (Carrier Sense), akkor várakozik az adás megszűnéséig. Amikor elcsendesül a közeg, megkezd az adást, amelyet egyedül csak a megcímezett állomás fog átvenni. Minél több állomás van az adott hálózati szegmensre kapcsolva, annál nagyobb az esélye annak, hogy egy időben egyszerre két vagy több állomás is küldeni akar. Ilyenkor a közegen a villamos jelek összeadódnak, az adatok sérülnek, az információ használhatatlanná válik. Ezt az állapotot érzékelni kell (Collision Detection), sőt figyelmeztetni kell rá a többi állomást is, hogy ütközés történt. Az ütközést az adó érzékeli, amikor visszaolvasáskor nem ugyanazt az adatot kapja. Azért, hogy minden állomás értesüljön az ütközésről, az ütközést észlelő egy rövid ideig egy 32 bit hosszúságú un. JAM jelsorozatot helyez a hálózatra, amelynek hatására mindegyik állomás azonnal beszünteti az adást, majd véletlenszerű késleltetési idő után újból próbálkozik.

A véletlen időzítésnek köszönhetően valamelyik adó elsőként lefoglalja és megkezdheti az adását, ezt a többiek érzékelni fogják, mielőtt megkezdenék adásukat. A véletlen várakozási idő N ütközés után maximum a következő lehet:

$$t_w = t_s \cdot (2^N - 1); \text{ ahol: } 1 \leq N \leq 16 \quad (4.1)$$

A fenti relációban szereplő  $t_s$  résidő (slot time), vagy körbejárási késleltetés, az az idő mialatt a keret a szegmens két legtávolabbi pontja között körbejár ( $t_s = 5,12 \mu s$  Fast Ethernet hálózatban). Ennyi idő szükséges ugyanis ahhoz, hogy mindegyik állomás biztonsággal érzékelje az ütközést.

A várakozási idő egy algoritmus szerint (Binary Exponential Backoff) kerül meghatározásra. Az első ütközés után 0 vagy 1 a szorzó, a második után 0, 1, 2, 3, a harmadik után 0, 1, ..., 7 és így tovább. A 10-edik ütközés után (1023) a szorzó nem nő tovább, a 16.-odik ütközés után az eszköz tovább már nem próbálkozik, és hibaüzenettel jelzi az átvitel sikertelenségét.

#### **4.1.3. Az IEEE 802.3 Ethernet keret felépítése**

Az Ethernet hálózat szereplői az információt (adatokat) szabványosított keretek formájában továbbítják. Az IEEE 802.3 keretstruktúráját a 4.3. ábrán láthatjuk.

Előtag	Keret-határoló	Célcím	Forráscím	Adat-hossz	Adatok	Ellenőrző összeg
7 bájtt	1 bájtt	6 bájtt	6 bájtt	2 bájtt	46-1500 bájtt	4 bájtt

4.3. ábra. Az IEEE 802.3 keretformátuma

Az előtag és a kerethatároló igazából nem része a keretnek. A MAC alréteg (az adatkapcsolati réteg egyik alrétege) azért fűzi a keret elejére, hogy biztosítsa a vevő hardverjének ráhangolódását az adó órájára. Tulajdonképpen egy 10101010 mintájú jel 7 bájton keresztül, amelynek a Manchester kódja egy 10 Mhz-es, 5,6  $\mu s$  időtartamú négyszögjelnek felel meg 10 Mb/s bitsebességnél. Ezt követi a kerethatároló bájtt, amely hasonló, kivéve az utolsó bitet, amely ugyancsak 1-es. Ez jelzi, hogy itt kezdődik a keret. A célcím és forráscím az adó illetve a vevő MAC címét tartalmazza, amelyet az adó alrétegének ARP protokollja képez le az IP címből. A vevőnél pedig a RARP éppen az ellenkezőjét teszi, a MAC címből visszaállítja az IP címet a hálózati réteg számára

A kétbájtos adathossz mező, az adatmezőben található bájtok számát adja meg, amely 0 és 1500 közötti érték lehet. Ha 46 bájtnál kisebb ez a szám, a keret kiegészül töltelékbytekkel, azért, hogy a 64 bájtos minimális keretméret meglegyen. Ismervén az adatmező hosszát, a vevő a vételnél leválasztja ezeket.

Az ellenőrző összeg vagy FCS (Frame Check Sequence) tulajdonképpen egy CRC (Cyclic Redundancy Check) algoritmus szerint kapott osztási művelet 32 bites maradéka, amelyet a keret végéhez fűz hozzá. A vételnél a teljes keretet a CRC-vel együtt, ha elosztják ugyanazzal az értékkel

akkor, ha hibátlan a keret maradék nélküli értéket kapunk. Akár már egyetlen egy bit megváltozását is érzékeli, de 99,9%-ban alkalmas csoportos bithibák felderítésére is.

A keret tartalmaz még az adatmezőbe beágyazva három bájtnyi LLC (Logical Link Control) alréteg információt. Ebből kettő, a DSAP (Destination Service Access Point) és az SSAP (Source Service Access Point) a címzett illetve a forrás protokoll azonosítója, a CONTROL blokk pedig az LLC szolgáltatás módját mutatja.

Ezek után pedig újra azzal szembesülünk, hogy van egy szabvány szerinti keretünk, amit gyakorlatilag senki sem használ. Még a Windows alapú operációs rendszerek is az eredeti Ethernet II elnevezésű, természetesen egyszerűbb keretformátumot használják alapértelmezésben. Ebből következik, hogy az ipari Ethernet rendszerek is az utóbbit használják. Az *Ethernet II* keret formátuma annyiban tér el a 802.3 formátumtól, hogy nem az LLC mezők azonosítják a protokollt, hanem az adathossz mező helyett, az ugyancsak 2 bájtos Ethertype mezőt használják [12].

Az előzőekben bemutatott, igencsak nagyfokú bizonytalanságot tükröző közeg-hozzáférési algoritmus ismeretében jogosan feltehetnénk magunknak a kérdést, egyáltalán hol használnak ilyen kiszámíthatatlannak tűnő kommunikációs kapcsolatot? A válasz nagyon egyszerű: mindenütt, kivéve ahol nincsenek időhöz kötött, determinisztikus folyamatok, ott ahol egy esetleges ütközés után kialakult véletlenszerű várakozási idő a valós idejű kapcsolatokat kedvezőtlenül befolyásolná. Az Ethernet hálózat alkalmazása viszont számos előnnyel rendelkezik. Világméretű kiterjedése, az Internet közvetlen elérhetősége, a hardvereszközök és szoftverek viszonylag alacsony ára, a nagy átviteli sebesség (100 Mb/s, 1 Gb/s, stb.) mind olyan tényezők, amelyek az ipari automatizálás fejlesztői számára célul tűzték ki azt a feladatot, hogy hogyan lehetne az Ethernet kommunikációt determinisztikussá tenni?

#### **4.1.4. Követelmények egy ipari Ethernettel szemben**

Ahhoz, hogy az Ethernet hálózat minden körülmények között alkalmas legyen valós idejű ipari kommunikációs adattovábbításra, a következő követelményeket kell teljesítenie:

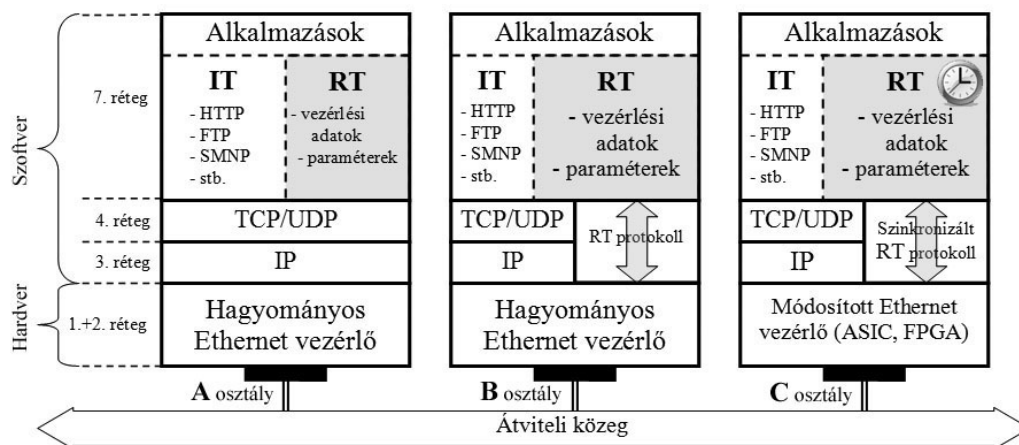
- Az Ethernet szabvány szerinti nyitottság megőrzése,
- Full duplex kommunikáció,
- Ciklikus működés biztosítása,
- Valós idejű kapcsolatok támogatása,
- Biztonságos adatátvitel,
- 100 %-os IT kompatibilitás,
- Diagnosztizálási lehetőségek,

- Szinkronizálás biztosítása az ezt megkövetelő folyamatok számára,
- A már meglévő terepi buszrendszerek egyszerű integrálhatósága.

Tulajdonképpen a ciklikus működés az, ami kiszámíthatóvá (valós idejűvé) teszi az Ethernet hálózatot. Ez úgy valósítható meg, hogy az IEEE802.3 szabvány szerinti keretbe olyan mezőket viszünk be, amelyek megkülönböztetik a valós idejű kereteket, a hagyományos internet keretektől. Ezáltal, a ciklusidő alatt egyszer mindenképpen egy valós idejű keret kerül prioritással továbbításra a többi kerethez képest. A fennmaradó időben, pedig bármilyen egyéb keret zavartalanul továbbítható.

#### 4.2. Fontosabb ipari Ethernet rendszerek [13]

A jelenleg elterjedt ipari Ethernet megoldásokat a gyakorlatban három osztályba sorolhatjuk, attól függően, hogy milyen kommunikációs rétegeket használnak és milyen valós idejű követelményeket képesek teljesíteni. A 4.4. ábrán ezen osztályok rétegstruktúráit láthatjuk.



4.4. ábra. Ipari Ethernet osztályok struktúrái

Az *A osztályú* ipari Ethernet gyakorlatilag ugyanazt a hardver felépítést és TCP/IP vagy UDP/IP protokollkészletet használja, mint az irodai Ethernet. A valós idejű követelményeket az RT alkalmazások a TCP/IP rétegek felett valósítják meg. Az ütközések elkerülése a valós idejű alkalmazásokban szoftveres úton forgalomfinomítással, vagy kapcsolt Etherneten keresztül történik. Meglehetősen korlátozott valós idejű működést képes biztosítani, általában 50-100 ms-os kézbesítési idővel és legalább ekkora jitterrel. Alkalmazási területe: Főleg monitoring rendszerek, folyamatirányítás, épületautomatizálás, stb. Ebbe az osztályba sorolhatók például a hagyományos *Ethernet/IP*, a *Modbus/TCP* vagy a *Profinet CBA* [14, 15, 16].

A *B osztály* ugyancsak a szabványos Ethernet hardveres felületet használja, de már jobb valós idejű támogatást biztosít, egyrészt mert a prioritással ellátott RT csomagokat képes előnyben részesíteni a

nem valós idejű csomagokkal szemben, másrészt az egyes RT protokollok képesek kezelni közvetlenül az alsóbb rétegek felől érkező kereteket. Ilyenkor azonban a kommunikáció csak egy szegmensben belül valósítható meg. Az ide tartozó rendszerek általában kielégítik a gyártásautomatizálás követelményeit: 5-10 ms-os kézbesítési idők, 10-20 ms-os válaszidők, 2-4 ms jitter. Főleg a PLC-s vezérlésű DCS alapú ipari automatizálás terén alkalmazzák. Az osztály egyik meghatározó megoldása a Siemens támogatottságú *Profinet IO* vagy a Bernecker & Rainer cég által fejlesztett *Ethernet Powerlink*

A harmadik, *C osztályba* a legigényesebb valós idejű működést igénylő hajtásszabályozás és mozgásvezérlés tartozik, amely rendszerint szinkronizált, ciklikus működést feltételez. A ciklusidő 1 ms, vagy ennek tört részei (0,5 ms, 0,25 ms), a jitter pedig a legtöbb esetben kisebb, mint 1  $\mu$ s. Mindenképpen 100Mb/s-os *Full duplex Ethernet* hálózatot feltételez. A hardverfelület módosított (nem szabványos Ethernet), speciális ASIC-t vagy FPGA-t tartalmaz, amelyben a kapcsoló a TDMA technikát alkalmazva mindig szabad utat biztosít a szigorúan valós idejű kereteknek, ugyanakkor a fennmaradó időben a nem valós idejű forgalom is biztosított. A szinkronizált működést az adatkapcsolati rétegbe implementált IEEE 1588 PTP (Precision Time Protocol) protokoll [17] biztosítja. Az eszközöket a MAC címek alapján azonosítják, routolás itt sem valósítható meg. A csoport meghatározó képviselője a Backhoff cég által fejlesztett EtherCAT, de jelentős részt foglal el a Profinet IRT és a Sercos III is.

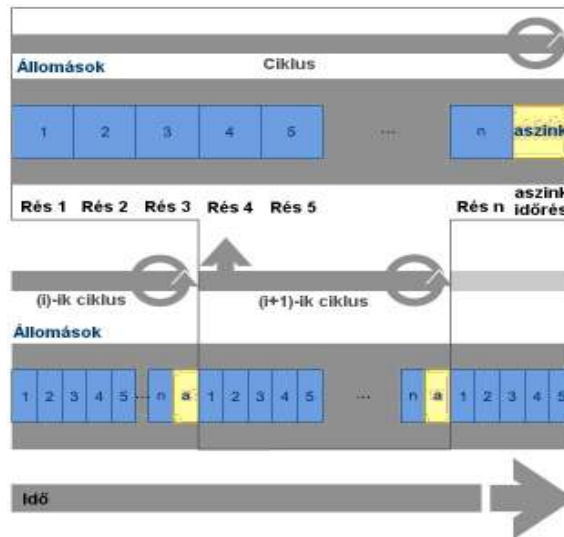
A továbbiakban röviden az *Ethernet Powerlink*, a *Profinet* és az *EtherCAT* rendszer fontosabb jellemzőivel és működési elveivel ismerkedhetünk meg.

#### **4.2.1. Ethernet Powerlink**

Az osztrák B & R Automation cég által kifejlesztett Ethernet PowerLink (EPL) kifejezetten gyors és pontos, C osztályú átviteli sebességet biztosít akár 200  $\mu$ s-os ciklusidővel és 1  $\mu$ s-os jitterrel. Gyorsaságát annak köszönheti, hogy az adattovábbítást akárcsak két fázisban hajtja végre. A szinkron fázisban mindegyik állomás rögzített szélességű időrést kap a valós idejű adatok továbbítására. A telegramok azonosítását a keretbe ágyazott EPL azonosító biztosítja. Az aszinkron fázisban kerül sor a hagyományos IP alapú információ továbbítására (4.5. ábra). A vezérlési adatok küldésekor egy időben mindig csak egy állomás férhet hozzá a hálózathoz, így ütközés nem fordulhat elő. Fontosabb jellemzői:

- IEEE 802.3 szabványra épül;
- A nyitottság megőrzése mellett valós idejű, determinisztikus adatátvitelt biztosít;
- Időkritikus adatok továbbítása gyors és pontos ciklusokkal;

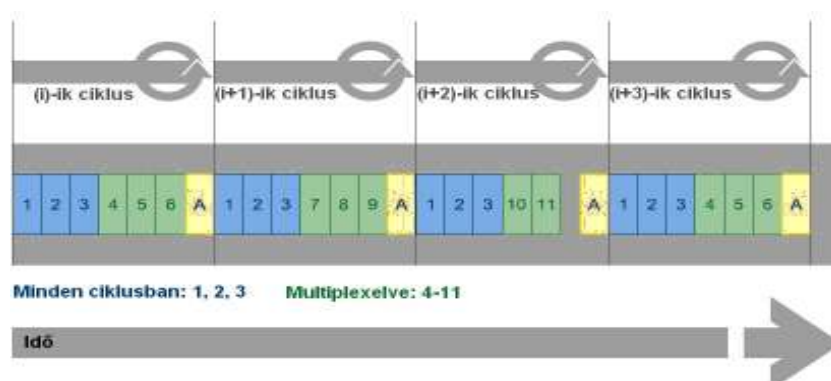
- Ciklusidő akár 200  $\mu$ s;
- Az állomások nagypontosságú szinkronizálása:
  - 1  $\mu$ s-os szinkronizálási lehetőség
- Nem valós idejű adatok továbbítása aszinkron csatornán keresztül.



4.5. ábra. Az Ethernet Powerlink cikluson belüli időosztása

A szinkron fázisban minden állomás rögzített szélességű időablakot kap (1, 2, ..n, EPL azonosítóval). Aszinkron fázisban a hagyományos IP alapú kommunikáció zajlik, TCP/I címzési mód szerint. A valós idejű adatok küldésekor egy időben legfeljebb két állomás férhet hozzá a hálózathoz. Az ütközés tehát kizárt.

A 4.5. ábrán azt láthatjuk, hogy mindegyik állomás mindegyik ciklusban kommunikál. Ez nem minden esetben szükséges. Lehetnek olyan állomások, amelyek megelégednek nagyobb kommunikációs ciklussal. Ennek megfelelően az Ethernet Powerlink multiplexeléssel skálázható. Ez látható a 4.6. ábrán.



4.6. ábra. Az Ethernet Powerlink optimalizálása

A 4.6. ábrán látható, hogy az 1, 2, 3 állomások minden ciklusban, míg a 4, 5, 6 állomások, csak minden harmadik ciklusban kommunikálnak. A 7-11 állomások lehet, hogy csak a negyedik vagy ötödik ciklusonként kommunikálnak (az ábrán ez már nem látható).

#### **4.2.2. PROFINET**

A PROFINET szabványos (IEC 61158 és IEC 61784 [18]), nyílt ipari Ethernet hálózat, amely kielégíti az automatizálási technológiák és ipari folyamatirányító rendszerek minden követelményét. A PROFINET segítségével kivitelezhetők a decentralizált terepi eszközök közötti biztonságos adatkapcsolatoktól a gyors válaszidőt igénylő motorhajtásokig minden irányítási tevékenység. A valós idejű adatátviteli lehetőségei mellett a teljesen IT technológiára épülő előnyei játszanak fontos szerepet abban, hogy egyre inkább kezd elterjedni az ipari kommunikációban. A PROFINET lehetőséget biztosít a már meglévő terepi buszrendszerek: PROFIBUS, INTERBUS, ASI megoldások kiváltására, anélkül, hogy a meglévő eszközöket lecserélnénk.

A PROFINET rendszer előnyei elsősorban a gyártásautomatizálás osztott rendszereinek (DCS) valós idejű ipari Ethernetes kommunikációjában nyilvánulnak meg. A rendszer szinte teljes egészében az információs technológia elemeire épül:

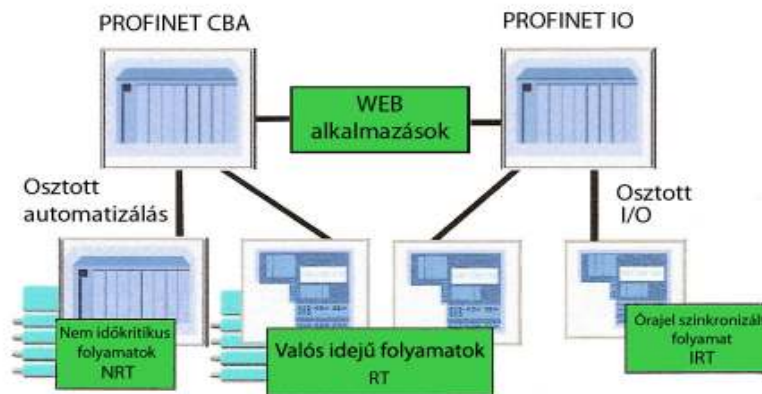
- szabványos csatlakozókat (RJ45) és hálózati elemeket (hub, switch, stb.) használ,
- gyakorlatilag az összes hálózati topológiát támogatja,
- biztosítja a hálózaton keresztül a diagnosztizálás lehetőségét,
- biztonságos kommunikációt tesz lehetővé,
- a WLAN segítségével a különleges mozgást végző gépelemek intelligens érzékelői is könnyen kapcsolódhatnak a vezérlőhöz.

A PROFINET használata egyben csökkenti az eszközök telepítési költségeit, kevesebb mérnöki és felügyeleti munkát igényel, de biztosítja a rendszerfejlesztés rugalmasságát és bővíthetőségét.

A PROFINET rendszer elemeit, funkcionalitásukat tekintve két külön kategóriába sorolhatjuk attól függően, hogy az automatizálás melyik szintjén teljesítenek szolgálatot, de természetesen közöttük is kiépíthető Ethernet kapcsolat. Két alapvető kivitelezési forma használatos: a *Profinet CBA* és a *Profinet IO* (4.7. ábra).

A PROFINET CBA-t komponens alapú, elsősorban programozható funkcionalitású intelligens terepi eszközök valamint kontrollerek számára alakították ki. A komponens modell alatt gépek és technológiai modulok egymástól függetlenül működő részegységeit kell érteni. A fix funkcionalitású komponensek mellett számos programozható funkcióval rendelkező komponens is részt vesz. Ez nagymértékben megkönnyíti a gépek, gépsorok valamint gyártóegységek

modulrendszerű objektumainak kialakítását, amelyek jelentősen lecsökkentik a mérnöki költségeket.



4.7. ábra. A teljes Profinet rendszer kommunikációs kapcsolatai (forrás [15])

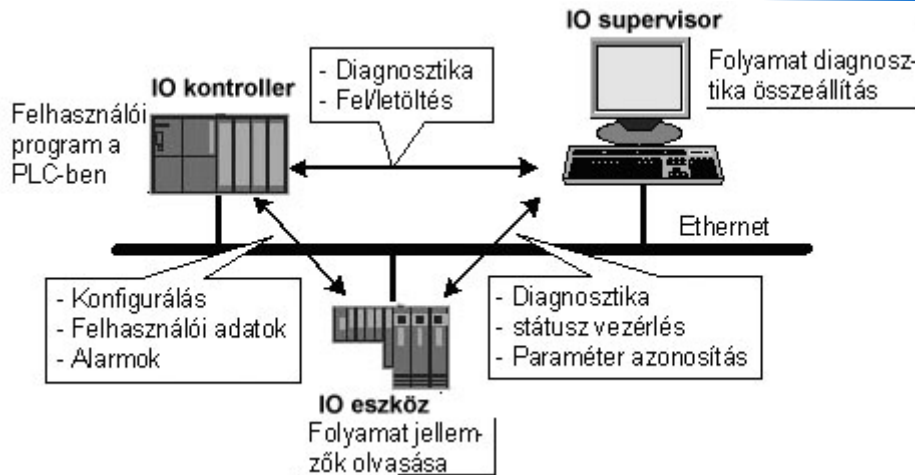
A Profinet CBA leginkább a folyamatautomatizálás területén tölt be jelentős szerepet. Időkapcsolatai nem kritikusak. A válaszidők akár a 100 ms-ot is elérhetik. TCP/IP vagy UDP/IP Ethernetes kommunikációt használ.

Az Ethernetes kommunikáció szempontjából a *PROFINET IO* a legkritikusabb. Itt lehet ugyanis megvalósítani az órajel szinkronizált, mozgásvezérlőknél alkalmazott, meglehetősen időkritikus *PROFINET IRT* kommunikációt is. Architektúrája a jól ismert Profibus rendszer filozófiáját követi azzal a meghatározó különbséggel, hogy az eszközök közötti kommunikáció az ipari Ethernet nyújtotta lehetőségeken alapul. Ebből következik, hogy a *master-slave* modellt az *termelő-fogyasztó* megoldás váltja fel, amely a kommunikációs kapcsolatok szemszögéből egyenrangú felekként tekinti a kapcsolatban résztvevő eszközöket.

A PROFINET IO rendszeren belül a nyílt internetes alkalmazások mellett mindhárom, a fejezet elején már bemutatott kommunikációs osztálybeli kapcsolat megvalósítható. Támogatja az összes lehetséges hálózati topológiát, beleértve a redundancián alapuló kapcsolatokat is. A rendszer felépítését tekintve alapvetően három elemre épül (4.8. ábra). Ezek a következők:

- *PROFINET kontroller* (IO kontroller): ez egy PLC, amelyen a felhasználói program (vezérlési program) fut,
- *PROFINET IO eszközvezérlő* (IO eszközök): decentralizált terepi eszközvezérlők, amelyek Ethernet hálózaton keresztül kommunikálnak a IO kontrollerrel,
- *PROFINET IO supervisor*: programozó vagy diagnosztikai eszköz

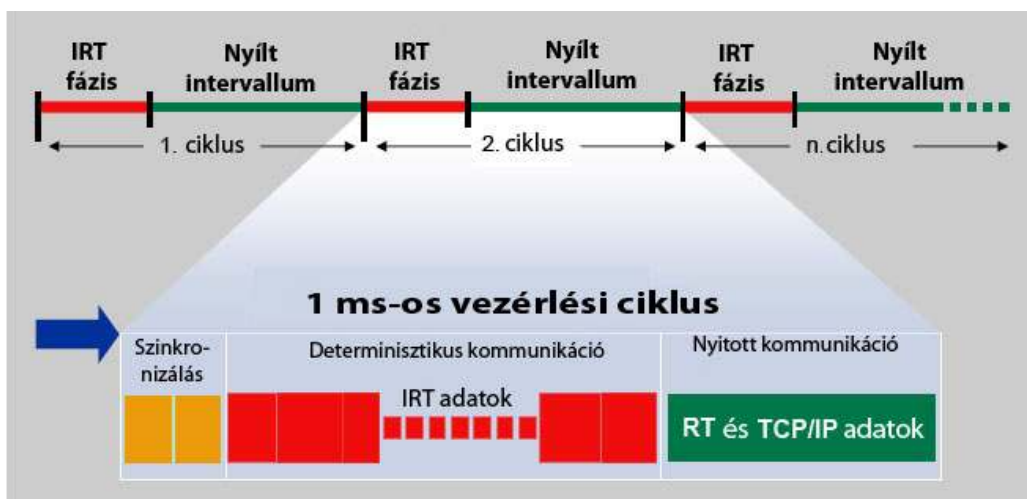
Megjegyzés: A 4.8. ábrán ugyan csak egy IO eszközvezérlő látható, de természetesen több (gyakorlatilag korlátlan számú) ilyen eszközvezérlő is csatlakoztatható.



4.8. ábra. A PROFINET IO elemei és kommunikációs kapcsolatai.

A motorvezérléseknél, ahol időben akár 2-3 vagy annál több tengely összehangolt, pontos mozgására van szükség, az eddig ismertetett szinkronizálás nélküli, valós idejű üzemmód nem alkalmazható. Az ilyen, időszinkronizált feladatok Ethernet hálózati kommunikációja a *PROFINET IRT* (Isochronous Real-Time) segítségével valósítható meg, amely röviden az alábbiakkal jellemezhető:

- A kommunikáció kizárólag egy hálózati szegmensben belül történik, mert a kapcsolat csak a fizikai és az adatkapcsolati OSI rétegeket használja [15].
- A busz ciklus időben két részre osztható (4.9. ábra): *IRT fázisra* (un. piros intervallum), és egy azt követő *nyílt intervallumra* (un. zöld intervallum).
- 1 ms-nál kisebb ciklusidők és 1  $\mu$ s-os, vagy annál kisebb eltéréssel (*Jitter*),
- A küldési intervallumok szinkronizálása.

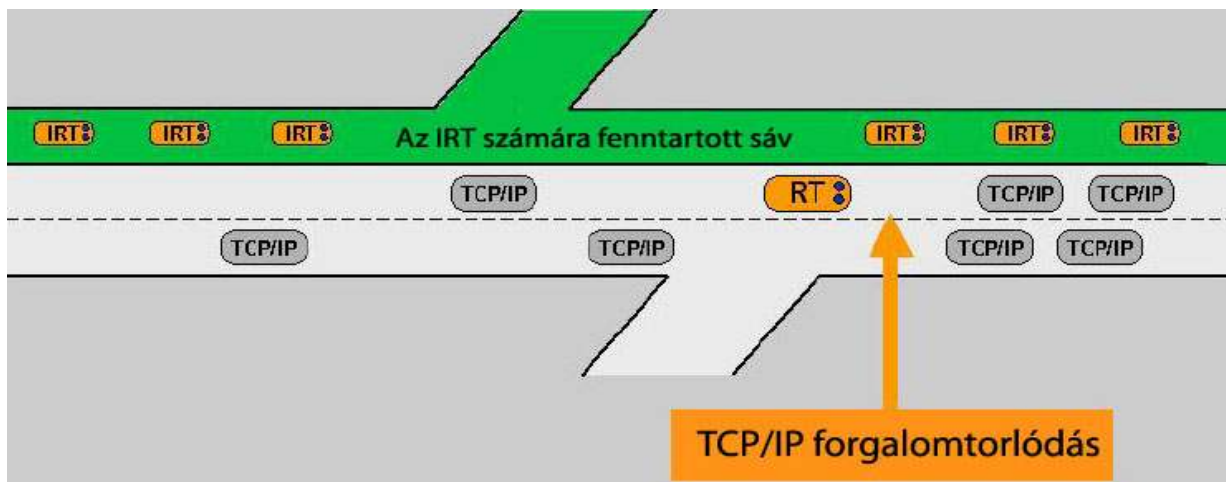


4.9. ábra. Az IRT kommunikáció időosztása (forrás [15])

A terepi eszközök kommunikációs időintervallumai rugalmasan állíthatók a felhasználó által. A piros intervallum (IRT fázis) nagysága főleg attól függ, hogy hány IRT eszköz csatlakozik a kontrollerhez. A kontroller konfigurálása során, amikor megadjuk az IRT eszközök számát, ez az intervallum automatikusan lefoglalásra kerül az IRT keretek számára. Az IRT keretekkel azonos időben érkező nem időkritikus kereteket az IRT kapcsoló (ASIC – Application Specific Integrated Circuit) pufferelemi a zöld intervallumig, vagyis mindig szabad utat biztosít az IRT adatok számára. A zöld intervallumban a valós idejű (RT), prioritással ellátott, IEEE 802.3Q szabvány szerinti, valamint a nem valós idejű (NRT) keretek továbbítása valósul meg.

A zöld intervallumból (nyílt intervallum) a pirosba (IRT fázis) való átmenetet, a szinkronizálás fázisa alatt (narancssárga intervallum) a determinisztikus kommunikáció alapjául szolgáló hardver (ERTEC 400 vagy ERTEC 200) felügyeli. A küldési ciklus mindig a szinkronizálással kezdődik, amikor a kontroller órájához igazodnak az IRT eszközök órái.

A kommunikáció során az IRT adatok mindig prioritást élveznek a számukra fenntartott sávban. A ciklus kezdetekor, a kommunikáció alapjául szolgáló IRT kapcsoló mindig szabad sávot biztosít a szigorúan időkritikus adatoknak. Olyan mintha egy többsávos autópályán az előzési sáv mindig a Profinet IRT adatok számára lenne fenntartva (4.10. ábra).



4.10. ábra. Az IRT kommunikáció számára fenntartott sáv

#### 4.2.3. EtherCAT

A szigorúan valós idejű ipari Ethernet kommunikációs rendszereknél (C osztály) a vezérlési adatok továbbítása két módon valósítható meg. Az egyik megoldás, amikor a kontroller (master) mindegyik, vele Ethernet kommunikációs kapcsolatban lévő IO eszköznek (slave) külön-külön címzi a kereteket, amelyek a megfelelő vezérlési adatokat tartalmazzák. Ezt a módszert alkalmazza az előző fejezetben bemutatott PROFINET IRT rendszer. Láthattuk, hogy ebben az esetben 36



**EFOP-3.5.1-16-2017-00017**

***„NYE-DUÁL- Új utakon a duális felsőoktatással a Nyíregyházi Egyetemen,  
az Északkelet-Magyarországi térség felemelkedéséért”***

bájtól kisebb vezérlési adatok gyakorlatilag nem befolyásolják a továbbításhoz szükséges időt, mert így a teljes keretméret nem haladja meg a minimális, 64 bájtot. Hátránya ennek a megoldásnak, hogy az előbb említett esetben a keretterhelési tényező a legjobb esetben is csak 56% lehet. Igazi előnye az 1000 Mb/s-os Gigabit Ethernet hálózaton mutatkozik meg, amikor az adattovábbításhoz szükséges idő gyakorlatilag már nem függ a keret méretétől [13].

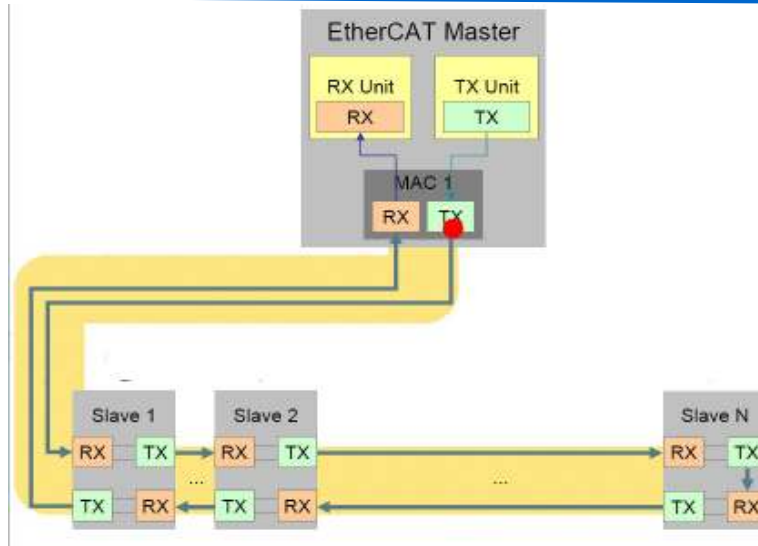
A másik megoldás, amikor arra törekszünk, hogy kihasználjuk a maximális keretterhelést, amely az IEEE 802.3 szerint 1500 bájt lehet. Ebben az esetben az IO eszközök számára továbbítandó vezérlési adatokat (telegramokat) igyekszünk egy vagy több kereten belül úgy elhelyezni, hogy minél nagyobb legyen a keretek kihasználtsága. Ez a módszer kétségtelenül előnyt jelent kisméretű (36 bájtól kisebb) telegramok továbbítása esetében. Így egy Ethernet kereten belül egyszerre továbbíthatja akár az összes csomópontban lévő eszközök vezérlési adatait. Ezt a megoldást alkalmazza a Beckhoff cég által fejlesztett EtherCAT (Ethernet for Control Automation Technology) [19].

Az EtherCAT rendszer jellemzői:

- rövid ciklusidő ( $\leq 1\text{ms}$ ),
- zavarmentes kommunikáció,
- nagy pontosságú szinkronizáció (jitter  $< 1\mu\text{s}$ ),
- korlátlan számú IO eszköz (max. 65535),
- 100 Mbps full duplex Ethernet hálózat,
- „menet közben” adatcsere,
- rövid,  $1,35\mu\text{s}$ -os IO eszköz késleltetési idő,
- korlátlan (max. 60 kbájt) adatmennyiség,
- busz, csillag, fa, vegyes topológia.

Az erre a célra kialakított Ethernet hardvernek köszönhetően a csomópontokon belüli késleltetési idő rendkívül rövid, mindösszesen  $1,35\mu\text{s}$  full duplex hálózatban [20]. A rendszer felépítése és az adattovábbítási elve a 4.11. ábrán látható.

Az EtherCAT kontroller a feladat elvégzéséhez nem szükséges, hogy rendelkezzen semmilyen extra kommunikációs processzorral. Ebből következik, hogy bármilyen kontroller, amelyik rendelkezik szabványos Ethernet interfésszel alkalmas erre a feladatra, függetlenül attól, hogy milyen operációs rendszert vagy alkalmazást használ.



4.11. EtherCAT adattovábbítás

A rendszer a vezérlési adatok nagysága szempontjából sem mondható korlátozottnak. Noha a motorvezérléseknél nincs szükség nagyméretű telegramok továbbítására, a gyártó 60 kilobájtban határozta meg a maximális adatmennyiséget, amely továbbítható a csomópontoknak. (Ebben az esetben természetesen több keretre van szükség) [21].

## 5. ROBOTTECHNIKA ÉS ROBOTIZÁCIÓ

A robot szó szláv eredetű „robota” szóból ered, aminek jelentése szolgamunka a cseh, szlovák és lengyel nyelvekben, valamint a munkát jelenti az oroszban. A szó elterjedését Karel Čapek 1921-es *R.U.R. (Rossum's Universal Robots)* című színdarabjának köszönhetjük. Az idők során számtalan definíció jelent meg a robotokkal kapcsolatosan, mint például Ivan M. Havel szerinti meghatározás: *„A robot egy automatizált, vagy számítógéppel vezérelt integrált rendszer, amely az ember utasításai alapján képes önálló és célorientált interakcióra a környezetével. Az interakció a környezet észlelése és felismerése, tárgyak kezelése és a környezetben történő mozgás.”* [23]

Általános megfogalmazásként elmondhatjuk, hogy *a robot egy intelligens programozható gép, amely az ember mintájára van megszerkesztve, információkat fogad el a környezetétől és értelmezi őket, elmozdít vagy kezel objektumokat és műveleteket végez.*

A robotok elméletével, leírásával és működésükkel kapcsolatos kérdéseket a *robotika* tárgyalja, amely egy olyan *tudományág, amely a robotok működését tanulmányozza, szerkesztésével és alkalmazásával foglalkozik az emberi tevékenység különböző területein.*

Napjainkban nagyon sokféle robot létezik. Ha osztályozni szeretnénk őket, akkor két nagy csoportba oszthatjuk: *humanoid robotok* és *ipari robotok*. De ide sorolhatjuk a *manipulátorokat* is, amelyeket ugyancsak ipari tevékenységeket végeznek. Ebben a jegyzetben csak ipari robotokkal fogunk foglalkozni.

Sokan félnek attól, hogy a robotok és a mesterséges intelligencia egyszer csak saját célokat tűz ki magának, hagyja a gazdáját, szembe fordul az emberiséggel. Ezért nem árt szem előtt tartani a robotika három alaptörvényét:

1. *A robotnak nem szabad kárt okoznia emberi lényben vagy tétlenül tűrnie, hogy emberi lény bármilyen kárt szenvedjen.*
2. *A robot engedelmeskedni tartozik az emberi lények utasításainak, kivéve, ha ezek az utasítások az első törvény előírásaiba ütköznének.*
3. *A robot tartozik saját védelméről gondoskodni, amennyiben ez nem ütközik az első és a második törvény előírásaiba.*

## 5.1. Ipari robotok

Az első ipari robotokat a hatvanas évek elején veszélyes környezetben történő munkavégzéseknél alkalmazták. Leginkább mérgező, radioaktív, egészségre ártalmas vegyi anyagok vagy forró munkadarabok kezelésére használták. Napjainkban, a robotizáció korában a fejlett ipari létesítmények szinte mindegyikében, de főleg az autóiparban használnak ipari robotokat. Külön iparág fejlődött a robotok gyártására. Élenjáró ipari robotgyártó cégek például a KUKA, a YASKAWA, a FANUC vagy az ABB (5.1. ábra).



5.1. ábra. Különböző ipari robotok (forrás: Internet)

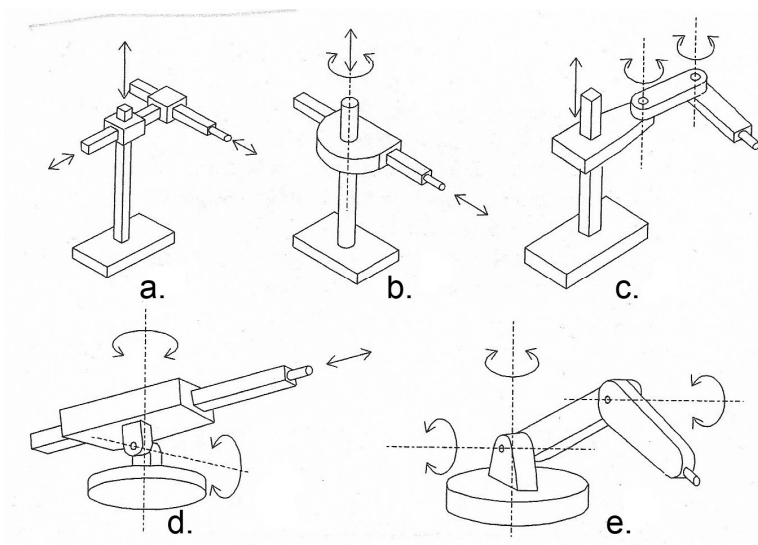
Különböző meghatározások alapján az ipari robot egy olyan mechatronikai szerkezet, amely nyílt kinematikai láncú mechanizmust és intelligens vezérlést tartalmaz, automatikus működésre és irányított mozgásokra képes. Betanítható vagy programozható az előírt feladatra.

### 5.1.1. Az ipari robotok szerkezete [1]

Amikor az ipari robotok felépítéséről beszélünk, akkor az emberi felső végtaghoz hasonlítva: *robotkarról, csuklóról és robotkézről, azaz megfogó szerkezetről* beszélünk. A mechanikai részek, akár csak az emberi kar csontjai, *ízületekkel* csatlakoznak egymáshoz. Az ízületeket közvetlenül vagy áttételeken keresztül különböző segédenergiájú (legtöbbször elektromos vagy hidraulikus) hajtások mozgatják. Az egymástól függetlenül mozgatható ízületek (tulajdonképpen tengelyek) önálló hajtással rendelkeznek. Ezeknek a száma adja meg a robot *szabadságfokát*. Ahhoz, hogy a robot megfogó szerkezetét tetszőleges térbeli pozícióba és irányba vezéreljük hat szabadságfok szükséges.

## Robotkar

Az ipari robot mechanikai rendszerének első három tagját, amelyek a megfogó szerkezet pozicionálását végzik, *robotkarnak* nevezik. Leggyakrabban négy részből tevődik össze, ennek megfelelően három ízülettel. Ezek a robotkar elemek ízületeik révén csúszó (*Transzlációs*) vagy forgó (*Rotációs*) mozgást végezhetnek. A robotkar kialakítása egyben a robot *munkaterét* is meghatározza. Az elemek és az ízületi kombinációk sokféle robotkar konstrukciót eredményezhetnek, de közülük a gyakorlatban öt megoldást alkalmaznak (5.2. ábra).

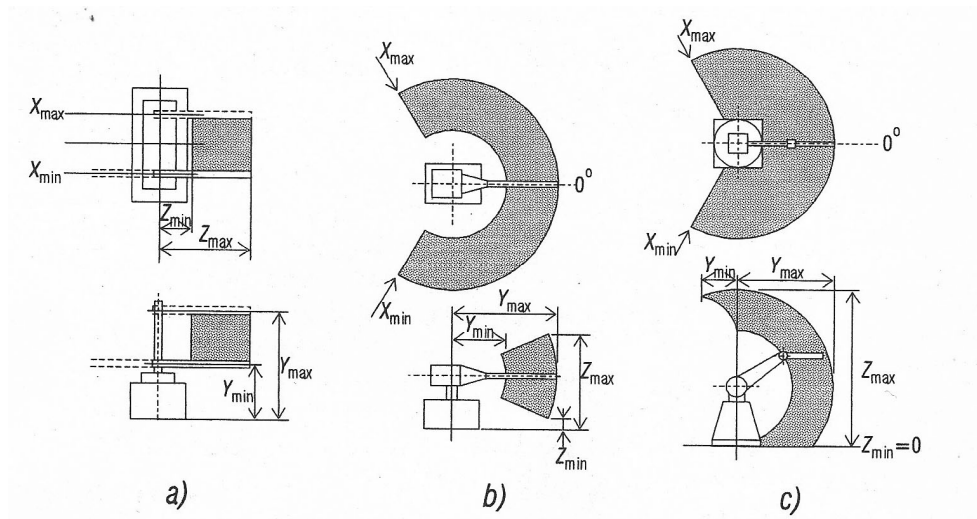


5.2. ábra. Robotkar típusok (forrás [1])

Az 5.2. ábrán látható robotkar típusok mindegyike jól meghatározott mozgásokat végeznek az alábbiak szerint [1]:

- Hasáb-munkaterű, derékszögű koordinátás robotkar (TTT)*, három csúszó ízülettel. Egyszerűsége és irányíthatósága miatt ez az egyik legkedveltebb típus.
- A hengerkoordinátás munkaterű robotkar (RTT)* két csúszó és egy forgó ízülettel rendelkezik.
- Kettős hengeres munkaterű robotkar (TRR)* vagy más néven *SCARA* robot két forgó és egy egyenes vonalú mozgást végző tengelyből áll. Elsősorban finommechanikai vagy elektronikai szerelési technológiáknál használják.
- Gömbkoordinátás robotkar (RRT)*; munkaterét a csúszó ízület maximális és minimális kinyúlási hosszával azonos sugarú gömbszeletek határolják.
- Csuklókoordinátás robotkar (RRR)*; kizárólag csak forgó ízületeket tartalmaz. Viszonylag nagy munkatér és rugalmasság jellemzi, de a bemutatott megoldások közül ez igényli a mozgása számára a legtöbb számítást a vezérlő részéről.

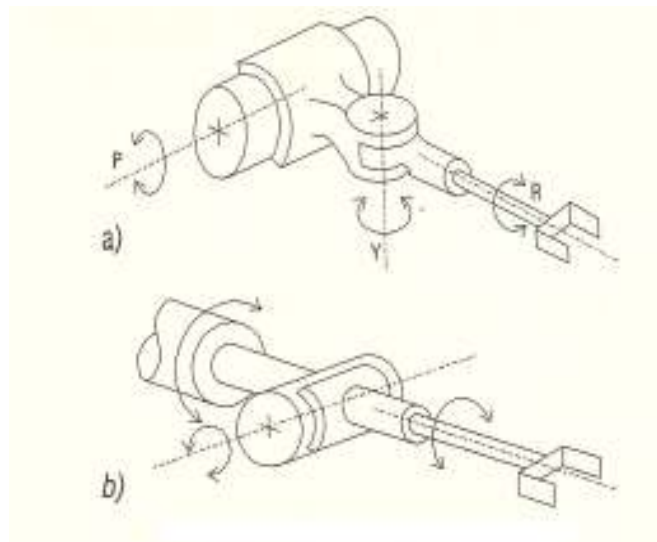
Az 5.3. ábrán a derékszögű (a), a gömb- (b) és a csuklókoordinátás (c) robot munkaterét láthatjuk, bejelölve az  $X$ ,  $Y$  és  $Z$  abszolút koordinátákat.



5.3. ábra. Munkaterék. a) derékszögű; b) gömb; c) csuklókoordinátás robot [1]

### Robotcsukló

A robotcsukló biztosítja a robotkar által elért helyen a tetszőleges térbeli irányból történő megfogást vagy munkavégzést. A csuklók különböző konfigurációjú forgó ízületeket tartalmaznak. A gyakorlatban két fő csuklótípust alkalmaznak (5.4. ábra):



5.4. ábra. RPY és Euler robotcsukló [1]

- Az *RPY* csukló szomszédos tengelyei egymásra merőlegesek. Az elnevezést a hajózásból vett szakkifejezések kezdőbetűi alkotják. **Roll**: dülöng – a hajótest hossz tengelye körül; **Pitch**: bugdácsol – a hajóorr emelkedik vagy süllyed; **Yaw**: cselleng – oldalirányú legyezőmozgást végez.

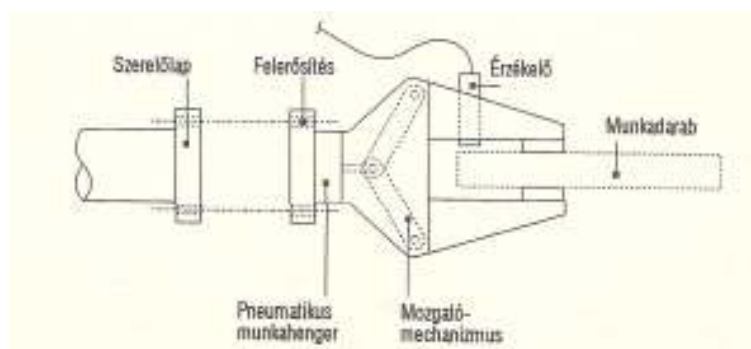
- b. Az *Euler-csukló* első és harmadik tengelye alaphelyzetben egymással párhuzamos, a középső pedig ezekre merőleges.

### *Megfogó szerkezet*

A megfogó szerkezet az ipari robotnak az a része, amely a robot csuklót és a robot által megfogott tárgyat vagy anyagot kapcsolja össze. A manipulálandó tárggyal közvetlen kapcsolatban van, azt megfogja, biztosan megtartja, az előírt helyzetbe hozza, majd leteszi (elengedi). Az anyagok tárgyak sokféleségének megfelelően a megfogó szerkezetek is igen változatosak. Az 5.5. ábrán a kétujjas mechanikus megfogó szerkezetet láthatjuk.

A megfogási elv és a megfogó szerkezet mozgatásához szükséges energiaforrás szerint a megfogás lehet:

- Mechanikus,
- Pneumatikus (vákuumos szívókorong),
- Elektromos (elektromágneses).



5.5. ábra. Kétujjas megfogó [1]

A mozgató mechanizmus működtetése pneumatikus munkahengerrel vagy villamos szervomotorral történik. A pneumatikus megoldás előnye, hogy olcsó és lágy megfogást eredményez. Amikor a megfogáshoz nagy erő kifejtés szükséges hidraulikus megoldást is alkalmaznak. Ilyenkor hidraulikus munkahenger szorítja meg az anyagot. A megfogó szerkezet tartalmaz még rendszerint valamilyen szenzort, amely a megfogott munkadarab jelenlétét érzékeli.

Ha a robotot valamilyen munkavégzés céljára akarják használni, akkor a megfogó szerkezet helyett az adott célra alkalmas szerszámot használnak. Ilyen például a festékszóró, motoros fűrő, csavarhúzó, csavarkulcs, ponthegeztő, vágószerszám, stb.

### 5.1.2. A robotok hajtása [26]

Az ipari robotok hajtási rendszereinek feladata az, hogy a robot un. *TCP* (Tool Center Point: szerszám középpont) pontját az előírt pontossággal, az irányítórendszer utasításainak megfelelően az előírt pályán mozgassa, illetve a megadott pozícióba juttassa. A roboton alkalmazott hajtások száma megegyezik a robotmechanika szabadságfokainak számával. Másként fogalmazva: az ipari robothajtás feladata a robotváz mozgatása. A hatások lineáris vagy forgómozgást valósítanak meg.

Minden robottengelynek külön hajtása van, ennek elemei:

- motor/tápegység,
- hajtómű,
- mozgás-átalakító,
- vezérlő/szabályozó egység.

A motor az elsődleges energia-átalakító. A robothajtások a robotok, illetve azok egységeinek, ízületeinek mozgatását teszik lehetővé.

A robotok irányítása a hatáslánc alapján lehet *nyílt* (visszacsatolás nélküli) vagy *zárthurkú* (negatív visszacsatolásos) irányítás. A visszacsatolás jellegétől függően és a működési módjuk alapján az ipari robotok, akár csak a CNC szerszámgépek, lehetnek *pontvezérlésűek* vagy *pályavezérlésűek*.

- *Pontvezérlés (PTP – Point to Point)* alatt azt a vezérlést értjük, amelynél a robot *TCP* pontját a munkatér adott pontjaiba kell eljuttatni. A pozicionálás útvonala, sebessége tetszőleges, csak a célhelyzeteket, a kijelölt pontokat kell nagy pontossággal elérnie.
- *Pályavezérlés (CP – Continuous Path)* esetében a robot *TCP* pontját a robotkarok a munkatérben úgy mozgatják, hogy a *TCP* pont a programban előírt síkbeli, vagy térbeli görbe mentén haladjon. A pozicionálás útvonala tehát nem tetszőleges, hanem egyértelműen determinált. Csak a szervó irányítású robotok képesek pályavezérlésre.

A hajtások legjellemzőbb műszaki paraméterei a következők:

- Az elmozdulás vagy szögelfordulás mértéke,
- Húzó- illetve nyomóerő nagysága vagy a forgatónyomaték,
- Sebesség, szögsebesség,
- Gyorsulás, szöggyorsulás,
- Pontosság.

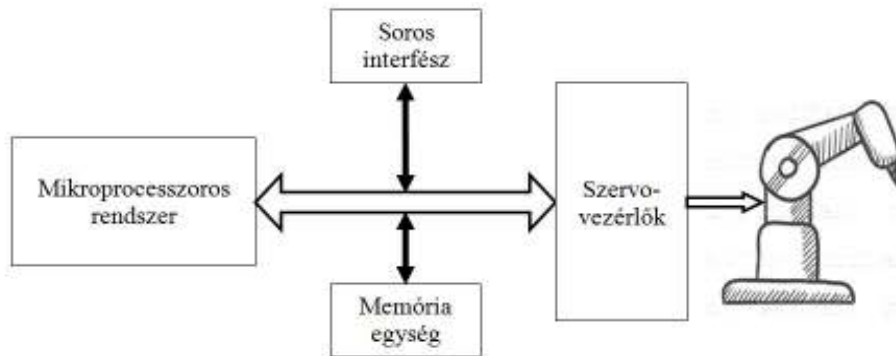
Az ipari robotok TCP pontjának nagy pontosságú irányítása a helyzetérzékelők és beavatkozók együttes közreműködése révén valósul meg. A szervó irányítású ipari robot beavatkozó szervének vezérlését az ízület érzékelt pozíciója és a kívánt pozíció közötti hiba alapján számítja ki az irányítórendszer. Ennek érdekében mindegyik tengely (ízület) saját helyzetérzékelővel kell rendelkezzen. Erre a célra a leggyakrabban optika inkrementális jeladót ún. enkódert használnak. Az érzékelőket általában közvetlenül a beavatkozó szerv (szervomotor) tengelyére szerelik. Ezt a megoldást lehet a legegyszerűbben irányítani feltételezvé, hogy a hajtórendszer kotyogásmentes.

Az erő- és nyomatékérzékelés, ha szükséges, erőmérő cellák segítségével valósítható meg. A legtöbb megoldás *félvezetős* vagy *fémfóliás nyúlásmérő bélyegeket* alkalmaz. Az érzékelőt a mérendő erő vagy nyomaték közvetlen közelébe helyezik el. Ahhoz, hogy a robottal finom szerelési munkákat is lehessen végezni, a szerszámnak az emberi kézhez hasonló tulajdonságokkal kell rendelkeznie. Biztos de nem roncsoló megfogásokhoz a megfogó erőt a tapintásérzékelők mért jele alapján számítja ki az irányító rendszer. A fejlettebb tapintásérzékelők nyomás érzékeny *piezoelektromos cellákból* állnak. A nyomóerő növekedésével a cellák árama növekszik, és ezekből a jelekből előállítható a megfogási felület nyomáseloszlása [1].

### **5.1.3. A robotok irányítórendszere**

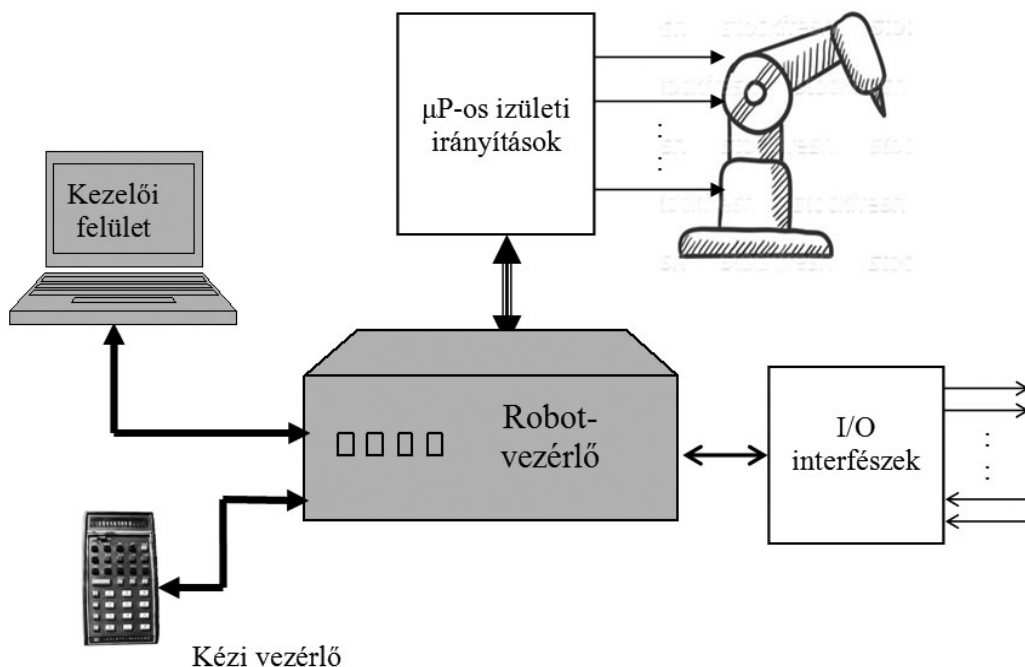
A korszerű ipari robotok irányítórendszere valamilyen intelligens eszköz, legtöbbször mikroprocesszoros berendezés. Az irányítórendszer egyrészt parancsokat küld a robot egyes ízületi végrehajtó szerveinek, másrészt kapcsolatot tart fenn a robot feladatát meghatározó külső vezérlőkkel, számítógéppel, más hasonló eszközökkel például PLC-kkel. Az irányítórendszer egyszerűbb esetben lehet egy egyprocesszoros mikroszámítógép, de bonyolultabb esetben akár egy többprocesszoros, osztott intelligenciájú rendszer is.

Az egyszerűbb ipari robotok irányítórendszerének többsége a végrehajtó szerveket mozgató analóg szervó rendszereken alapszik. Ez a rendszer egy olyan mikroszámítógép köré épül, amely előállítja a tengelyek szervó vezérlőinek alapjeleit. A számítógép indítja a mozgásokat és ellenőrzi a leállási feltételeket [1].



5.6. ábra. Egyszerű mikroprocesszoros robotvezérlő

Az ipari robotok irányítási feladatainak ellátására az esetek többségében túlságosan szűk határokat szab az 5.6. ábrán bemutatott egyszerű irányítási rendszer. Az ízületek (tengelyek) irányításán kívül a vezérlőnek ki kell szolgálnia az ember-gép kapcsolat eszközeit (karbantartás, beállítás, betanítás, programozás) is. Ezek az eszközök csak korlátozott módon kapcsolódhatnak a soros interfészen keresztül. Az ipari robotok irányítórendszerének kialakítására ma már egyre inkább jellemző a többprocesszoros rendszer. Ennek egyik formája az 5.7. ábrán látható.



5.7. ábra. Többprocesszoros robotvezérlő

Az 5.7. ábrán látható robotvezérlő nagyteljesítményű mikroprocesszora hajtja végre a kezelői felületről betöltött vezérlési programot. A feldolgozott utasítások alapján koordinálja az ízületi irányításokat és kezeli a külső eszközöket (kézi vezérlő, I/O interfész). Az ízületi irányítások is

mikroprocesszoros vezérlésűek. Minden egyes tengely mozgását külön mikroprocesszoros vezérlő koordinálja az alapjel (amit a robotvezérlőtől kap) és a helyzetérzékelők jelei alapján.

Az I/O interfész rendszerint analóg és digitális be- illetve kimeneti csatornákat is tartalmaz. Ide kapcsolódhatnak olyan elemek vagy vezérlők (PLC-k), amelyek a robot környezetében valamilyen egyéb feladatot látnak el.

A kézi vezérlő segít a robot mozgatásában, pozicionálásában vagy betanításában.

#### **5.1.4. A robotok programozása**

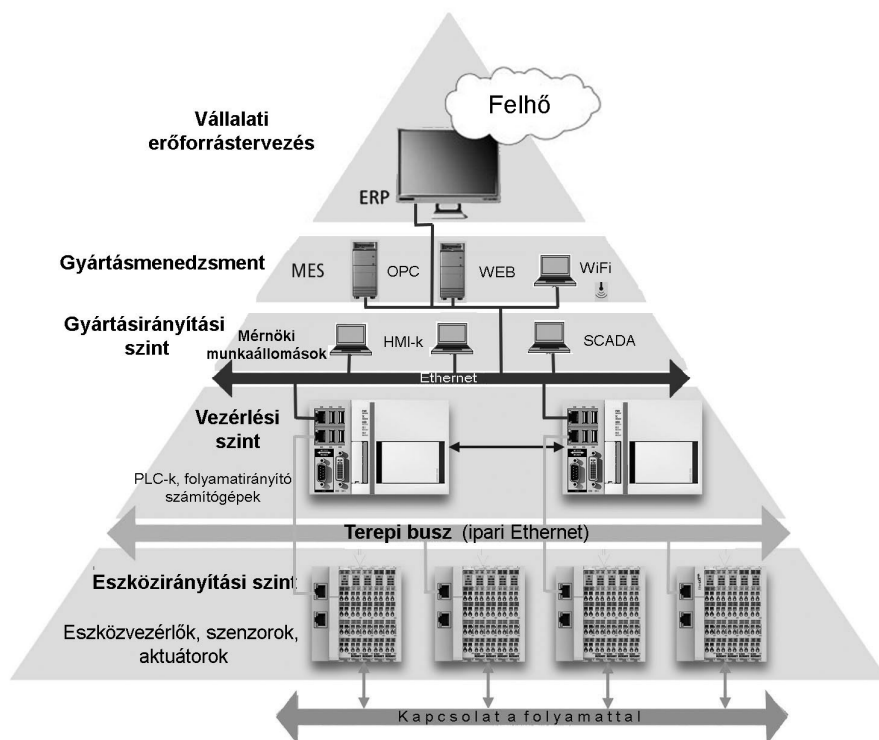
Egyik legrégebbi robotprogramozási eljárás legfőbb eszköze az úgynevezett *betanítás* volt. Az eljárás szerint a *betanítási üzemmódban* a robotkar kezelője kézzel mozgatja a robot tengelyeit a kívánt pontokba, miközben az irányítórendszer érzékelői folyamatosan méri a pálya koordinátáit és ezeket a robotvezérlő eltárolja a memóriában. Az így eltárolt információk alapján a robot mindaddig ezt a feladatot fogja végezni, ameddig új betanítás nem történik. Hasonló eredményt érünk el, ha fizikai mozgatás helyett a kézi vezérlővel irányítjuk a robotot a betanítás idején.

Napjainkban egy gyártósoron egyre több és több robotot alkalmaznak. Ezeknek a kézi betanítása már nagyon sok időt venne igénybe. Nem beszélve arról, hogy ezeknek a robotoknak kommunikálniuk is kell egymással vagy más intelligens irányítórendszerekkel. Egy ilyen, minden szempontból előnyös megoldás a robotok *offline programozása*. Egyik legnagyobb előnye ennek a módszernek, hogy a programozás idejére a robotot nem kell kivenni a termelésből. A vezérlési program a robot nélkül is elkészíthető. Az informatika fejlődésével a programozási fejlesztőkörnyezet egyre felhasználóbarátabbá vált, sok esetben nem is szükséges a programozónak magas szintű programozási ismeretekkel rendelkezniük. Így akár a technológusok is beprogramozhatják a robotokat a technológiai feladatnak megfelelően.

## 6. IoT AZ IPARBAN. IPAR 4.0

Napjainkban egyre többet halljuk az Ipar 4.0 kifejezést, amely a negyedik ipari forradalom kialakulását jelenti. Az Ipar 4.0 az új ipari forradalom elterjedését jelzi az intelligens gyártás területén. Az „ipari tárgyak internete” (*IIoT Industrial Internet of Things*) kifejezés az automatizálás eszközeinek az összekapcsolódását jelenti más analitikai eszközökkel a hatékonyabb működtetés érdekében.

A klasszikus ötödik generációs háromszintű hierarchián alapuló intelligens irányítási rendszer a piramis csúcsán kiegészült további szintekkel: gyártásmenedzsment szint és vállalati erőforrás-tervező szint (6.1. ábra).



6.1. ábra. Korszerű intelligens irányítórendszer (forrás [25])

Az ipar 4.0 és az IoT stratégia szigorú követelményeket támaszt a különböző szinteken elhelyezkedő elektronikus eszközök, illetve a szolgáltatások hálózati és adatkapcsolati képességeivel szemben. A piramiselvű szemlélete szerint nagy adatmennyiségeket kell cserélni a

kihelyezett érzékelők és a rendszer magasabb szintű rétegei között, ugyanakkor a modern gyártóüzemekben létfontosságú a PLC-alapú vezérlőrendszerek közötti horizontális kommunikáció is. A korszerű PC-alapú vezérlési technológiák univerzálisan alkalmasak az ilyen jellegű adatkapcsolatok kezelésére is, egyre inkább fontos szerephez jutnak a korszerű automatizálási projekteknél [25].

### 6.1. Az Ipar 4.0 fontosabb területei

Az Ipar 4.0-hoz tartozó technológiák három nagyobb területre oszthatók fel:

- a felhő alapú számítástechnikai rendszerek, felhőszolgáltatók,
- kiber-fizikai rendszerek (*CPS*)
- az intelligens gyár (*Smart Factory*), továbbá azok jellemző technológiái.

A *felhő alapú rendszerek* (*Cloud Computing*) nagy mennyiségű valós idejű, adattárolást (*Big Data*) és kommunikációt biztosítanak különböző intelligens alkalmazások (*Smart Product*) segítségével. A gyártási adatok tehát nem egy saját adatbázis szerveren vannak, hanem egy úgynevezett *felhőszolgáltató* tárhelyén, amelyhez hozzáférést biztosít a szolgáltató. Ez több szempontból is jelentős előnnyel jár főleg a multinacionális vállalatok esetében. A különböző földrajzi helyekre kihelyezett gyáregységek mindegyike ugyanahhoz az adatbázishoz tud hozzáférni.

A nyilvános felhőszolgáltatók, például a *Microsoft Azure* vagy az *Amazon Web Services* (AWS) bőséges szolgáltatást kínálnak saját, nagykapacitású adatközpontjaikból. Így akár virtuális gépet is igényelhetünk, ahol a felhasználó választhatja ki az operációs rendszert és az arra telepített a kommunikációs és adatszolgáltatásokat, amelyeket a felhasználó beintegrálhat az alkalmazásaiba.

A *kiber-fizikai rendszerek* az intelligens szenzorhálózat és jeladók felől érkező információkat applikációikon keresztül továbbítják a felhőszolgáltatás felé. Ugyancsak ezek a rendszerek valósítják meg a gépek közötti automatikus információcserét. A *Machine-to-Machine* (*M2M*) technológia olyan adatáramlást jelent, mely emberi közreműködés nélkül, gépek között zajlik. A kommunikáció minden olyan gép között létrejöhet, amely a megfelelő technológiával van ellátva ahhoz, hogy bekapcsolható legyen a rendszerbe.

Az *intelligens gyár* (*Smart Factory*) olyan termelési környezetet vizionál, melyben a gyártó- és vezérlő berendezések messzemenően saját maguk, emberi beavatkozás nélkül hangolják össze és szervezik meg a működésüket. Ennek technológiai alapját az imént ismertetett kiber-fizikai rendszerek alkotják, melynek elemei az *IoT* segítségével kommunikálnak egymással. Ide tartozik

még az is, hogy a termék (pl. a munkadarab) és a gyártó berendezés is kommunikál egymással. A termék a gyártóberendezés által olvasható, értelmezhető formában maga szolgáltatja a gyártási információt, pl. vonalkód vagy RFID chip segítségével. Ezeknek az adatoknak az alapján halad végig a termék a gyártósoron, s ezekre az adatokra épül a gyártási fázisok vezérlése.

## **6.2. Prediktív diagnosztika (adatközpontú karbantartás)**

A gyártók folyamatosan keresik azokat a módszereket, amelyekkel kihasználhatják az okos eszközök és az IoT adta lehetőségeket a létesítmények karbantartása és jobb üzemeltetése terén anélkül, hogy emberi beavatkozásra kellene hagyatkozniuk. Az üzemekben komoly problémát okoznak a váratlan leállások, amelyek közvetlen termeléskiesést okoznak. Az állásidő minimalizálása érdekében a gyártóknak helyben kell karbantartási szakembereket alkalmazniuk az ilyen váratlan problémák megoldásához.

Az adatgyűjtés elengedhetetlen a modern ipari cégek számára a gyártás és a karbantartás során. A dolgok internete (IoT) és az ipari dolgok internete (IIoT) megoldásai folyamatosan fejlődnek, mellette pedig egyre több lehetőséget kínálnak a megelőző karbantartás tökéletesítéséhez.

Az Omron cég olyan intelligens közelítéskapcsolókat fejlesztett, amelyekkel harmadára csökkenthető a váratlan üzemleállások kockázata. Az eszközök érzékelik a leállásokra utaló figyelmeztető jeleket is, és a hálózaton keresztül értesítik a felhasználókat. A váratlan leállások megakadályozásával és az állásidő csökkentésével jelentős mértékben javítják a létesítmények üzemteljesítményét.

Az ABB robotika cég a prediktív karbantartás bevezetésével, öt kontinensen keresztül figyelemmel kíséri a robotok karbantartási igényeit, és időben biztosítja a szükséges alkatrészellátást. Ezzel nagymértékben csökkenthetőek a nem tervezett leállások.

A Fanuc robotgyártó cég is komolyan mérlegeli az ipari létesítmények leállásának csökkentését. A robotikában használt érzékelők adatainak felhőalapú elemzéssel párhuzamosan a vállalat előre jelezheti, amikor egy komponens, például a robotrendszer vagy a folyamatirányító berendezés meghibásodása várható.

## FELHASZNÁLT SZAKIRODALOM

- [1] Dr. BALÁZS László: *Automatika*, Műszaki Könyvkiadó, Budapest, 2000, ISBN 963 16 1683 5.
- [2] Dr. TÓTH János: *Automatika*, TERC Kft. Budapest, 2013, ISBN 978 963 9968 57 8
- [3] Dr. NEMES József: *Irányítástechnika*, Nyugat-magyarországi Egyetem, 2012.
- [4] Dr. AJTONYI István: Dr. GYURICZA István, *Programozható irányítóberendezések hálózatok és rendszerek*, Műszaki könyvkiadó, Budapest, 2002, ISBN 963 16 1897 8.
- [5] KONDOROSI Károly: SZIRMAI-KALOS László, LÁSZLÓ Zoltán: *Objektum orientált szoftverfejlesztés*, 5. fejezet, <http://www.tankonyvtar.hu/hu/tartalom/tkt/objektum-orientalt/ch05.html#id517610>.
- [6] BÁNHIDI László, OLÁH Miklós: *Automatika mérnököknek*, Nemzeti tankönyvkiadó, 2001, ISBN 963 19 2322 3.
- [7] Dr. TELKES Zoltán: *Érzékelők, távadók és jelfeldolgozók 2*, Magyar Elektronika, 2009.11. 44 – 47 oldal.
- [8] SZŰCS Zoltán: *Digitális PID szabályozó (oktatási segédanyag)*, Budapesti Műszaki és Gazdaságtudományi Egyetem, <https://www.hobbielektronika.hu/forum/getfile.php?id=252405>, 2008.
- [9] IEEE 802.3 Part 3: *Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Method and Physical Layer Specifications*, 2008.
- [10] Andrew S. TANNENBAUM: *Számítógép hálózatok*, ISBN 978-963-545-529-4, 2011.
- [11] Dr. FERENCZI István: Út az ipari Ethernetig, <http://zeus.nyf.hu/~elat/ethernet.pdf>, 2017.
- [12] PETRÉNYI József: *TCP/IP alapok*, <https://mek.oszk.hu/08300/08374/pdf/tcpip1.pdf>, 2009.
- [13] Dr. FERENCZI István: *Ipari Ethernet rendszerek*, [http://zeus.nyf.hu/~elat/Ipari\\_Ethernet.pdf](http://zeus.nyf.hu/~elat/Ipari_Ethernet.pdf), 2017.
- [14] Mark CHAFFEE: *CIP Motion Implementation Considerations*, CIP Networks Conference, február 2009.
- [15] PROFINET: *System Description – Technology and Application*, Version June 2011.



EFOP-3.5.1-16-2017-00017

**„NYE-DUÁL- Új utakon a duális felsőoktatással a Nyíregyházi Egyetemen,  
az Északkelet-Magyarországi térség felemelkedéséért”**

- 
- [16] ARCOMAG INC: *Introduction to Modbus TCP* Wixom, USA, 2005.  
[www.acromag.com/sites/default/files/Acromag\\_Intro\\_ModbusTCP\\_765A.pdf](http://www.acromag.com/sites/default/files/Acromag_Intro_ModbusTCP_765A.pdf)
- [17] Dirk MOHL: *Precise Time Synchronization as the Basis for Real Time Applications in Automation*, IEEE 1588, 2004.
- [18] Max FESLER, Thilo SAUTER: *Standardization of Industrial Ethernet – the next battlefield?* Proceedings IEEE International Workshop on Factory Communication Systems, 2004.
- [19] EtherCAT Master – *Application Developers Manual*, Doc. Nr. P.4500.21/Rev. 1.4, 2011.
- [20] Dr. FERENCZI István: *Beágyazott telegramokat tartalmazó ipari Ethernet keretek ciklusideje*, GÉP, Vol. 63/5, pp. 51-54. ISSN 0016-8572, 2012.
- [21] EtherCAT – *The Ethernet Fieldbus*, 2009, [www.ethercat.org](http://www.ethercat.org)
- [22] Dr. PETL Szilveszter, Dr. KINCSES Zoltán: *PLC és SCADA rendszerek, egyetemi jegyzet*, Szegedi Tudományegyetem, 2004.
- [23] Petr BLECHA, Zdenec KOLIBAL és mások: *Robotika*, jegyzet, Brno-i Műszaki Egyetem, 2005.
- [24] MKIK Informatikai Kollégium: *Ipar 4.0 második szakértői tanulmány*, 2017,  
<https://www.gymskik.hu/hu/letoltes/80867/ee73d>
- [25] PORGÁNSZKI Éva: *Megoldások az Ipar 4.0 és az IoT számára*, Backhoff Automation, 2016.
- [26] Dr. KODÁCSY János, Dr. PINTÉR József: *Szerszámgépek és gyártórendszerek*, Széchenyi István Egyetem, 2011.