Tartalomjegyzék

Bevezetés

Az „Automatika alapjai példatár” nem helyettesíti a Gecsey László – Neszveda József, „Automatika I. laboratórium” ÓE KVK 2042 [1] és a Neszveda József, „Automatika I.” ÓE KVK 2044/1 [2] jegyzeteket.

Az „Automatika alapjai példatár” célja, hogy segítse az elméleti ismeretanyag megértését és az automatika alapfogalmainak készségszintű alkalmazását a szaktantárgyakban. Ugyancsak segíti a hallgatókat az „Automatika I.” tantárgy zárthelyi dolgozataira, valamint a vizsgájára felkészülni. Továbbá cél - az [1] jegyzetet kiegészítve - bemutatni a MATLAB és a SIMULINK programok néhány szolgáltatását, amelyek megkönnyítik a mérési feladatok kiértékelését és az elméleti ismeretanyag elsajátítását.

Az „Automatika alapjai példatár” elsődlegesen segíti a hallgatókat felkészülni az „Automatika I. laboratórium” tantárgy méréseire. Az „Automatika I. laboratórium” tantárgy méréseinek menete az alábbi:

- Számítógépes teszt formájában a mérésre való felkészülés ellenőrzése.
  Megjegyzés: Az „Automatika I. laboratórium” tantárgy 2 – 3 héttel később kezdődik, mint az „Automatika I.” előadás. Így az előadáson már elhangzanak a felkészülés ellenőrzésére szolgáló kérdések megválaszolásához szükséges elméleti ismertek, valamint értelemszerűen az [1] és [2] jegyzetekből is megkereshető a helyes válasz.
- A méréseken a hallgatók az „Automatika alapjai példatár”-ban megadott feladatsorhoz hasonló felépítésű és erősségű, de attól és egymásétól eltérő feladatsort oldanak meg önállóan, majd a saját munkájukat dokumentálva számítógépes formátumban adják le.
- A mérésvezető oktató magyarázata, amely a következő alkalommal elvégzendő mérésre történő felkészülést segíti.
  Megjegyzés: Ez alól az első mérés kivétel, amikor is az aznapi mérés elvégzéséhez ad útmutatást, és ezért megelőzi a mérést.

A példatár felépítése

Az „Automatika alapjai példatár” 1. fejezete a kétállapotú jeleket feldolgozó vezérléstechnikai feladatok leírási technikáit, illetve szövegesen megfogalmazott feladat algoritmizálási módszereit mutatja be.

Az „Automatika alapjai példatár” 2. – 3. és 5. – 7. fejezeteinek felépítése elsődlegesen az „Automatika I. laboratórium” tantárgy méréseire való felkészüléshez igazodik. Ebből adódóan a fejezetek tartalma a következő felépítésű:

- Számítógépes teszt kérdései.
  Megjegyzés: Értelemszerűen a tényleges tesztkérdésekben lehet némi fogalmazásbeli eltérés, az ábrák (átmeneti függvények, Bode, illetve Nyquist diagramok) paramétereiben biztosan vannak eltérések.
- Az adott mérésen megoldandó feladatsor minta.
  Megjegyzés: A hallgatók a mintafeladathoz jellegre, erősségre hasonló, de paramétereiben és esetleg a feladatsorrendben eltérő feladatsort oldanak meg.
- A feladatsor minta megoldása.
  Megjegyzés: Egy adott feladatot számos esetben többféle módszerrel oldhatunk meg. Ezekre a lehetőségekre a teljesség igénye nélkül a megoldásokban utalunk.

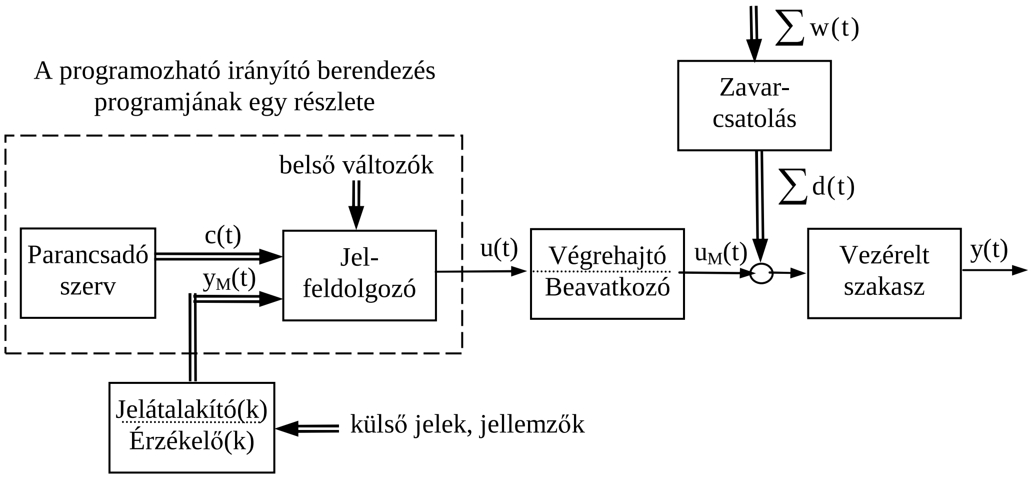
Az „Automatika alapjai példatár” 4. fejezete kiegészítése az [1] jegyzetben ismertetett MATLAB és SIMULINK programok kezelési leírásainak.

Az „Automatika alapjai példatár” 8. fejezete példákat mutat be, hogy hogyan lehet az összetett jelátviteli tagok átmeneti függvényéből, vagy az átviteli körfrekvencia függvényének Bode vagy Nyquist diagramjából meghatározni a jelátvivő tag jellegét, konkrét vagy közelítő paramétereit.

Az „Automatika alapjai példatár” 9. fejezete az egyhurkos szabályozási kör - a MATLAB III mérésben nem tárgyalt - kompenzálási technikáit mutatja be példákon keresztül.

1. Bevezetés a vezérléstervezésbe

Vezérlést akkor alkalmazhatunk, ha az összes zavarjellemző ($\sum w(t)$) együttes hatása, vagyis az összes zavarjel ($\sum d(t)$), a megengedett hibahatáron belül befolyásolja a vezérelt jellemző (y(t)) értékét. Ezen cél eléréséhez elegendően pontos matematikai modell kell, ami mindig megalkotható, ha a folyamat leírható kétállapotú jelekkel.



1.1. ábra A programozható vezérlés egy kimenetének elemei, jelei, jellemzői

Ha a konkrét vezérlési lánchoz tartozó algoritmus ilyen, akkor az algoritmus által feldolgozott belső változók és mért jelek vagy eleve kétállapotúak, vagy analóg jeltartomány esetén csak arra van szükség, hogy a jel egy adott értéket meghalad-e, vagy nem.

Az 1.1 ábrán a napjainkban leggyakrabban alkalmazott programozható irányító berendezéssel megvalósított egy bemenetű egy kimenetű vezérlési lánc felépítése látható. A jelfeldolgozó ez esetben a mikroprocesszoros rendszer CPU-ja. A parancsadó szerv a felhasználói program egy részlete. A belső változók (markerek) a memóriában, a mért jelek a bemeneti regiszterben tárolt értékek. A végrehajtó jel (u(t)) kétállapotú. A mért jelek, jellemzők ($y_M(t)$) vagy kétállapotúak (helyzetérzékelők, állapotkapcsoló, stb. jelei) vagy analógok (nyomás-, hőmérséklet, stb. távadók), de ez esetben a CPU képez belőlük kétállapotú jelet.

1.1. Az algoritmus megadása

A kétállapotú jelekkel leírható folyamat algoritmusa mindig definiálható a BOOL algebra segítségével. Az ipari irányító berendezések programnyelvei a BOOL algebra logikai műveletei közül az ÉS (AND), a VAGY (OR), a kizáró VAGY (XOR), és a negáció (N) műveleteit alkalmazzák.

A vezérlési algoritmus logikai állításokkal fogalmazható meg. Például:

1. HA (kézi start jel VAGY automatikus start jel érkezett) ÉS nincs motorhiba jel, AKKOR működjön a motor ÉS jelezzen a motor üzemállapot lámpája.
   Megjegyzés: A kézi start és az automatikus start jeleket kizáró VAGY kapcsolatba kell tenni. A gyakorlatban általában van egy kézi / automatikus kapcsoló, és ennek a jelével van megakadályozva, hogy egyszerre legyenek hatásosak.
2. A zsalu nyitva jel mellett HA letelt a töltési idő VAGY megtelt a tartály AKKOR zárjon a zsalu, KÜLÖNBEN villogjon a tartálytöltés lámpa.

Az 1. példa feladata megadható BOOL algebrai logikai függvényekkel is. Az áttekinthetőség kedvéért ehhez célszerű a szöveges megnevezés helyett rövid változó neveket bevezetni.

1.1 táblázat. Az 1. mintafeladat változói

kézi start	I1
automatikus start	M1
motorhiba	I2
motor működjön	Q1
üzemállapot lámpa	Q2

Megjegyzés: A korszerű ipari irányító berendezések fejlesztő szoftvere megengedi a hosszabb, de „beszédesebb" szimbolikus változó nevek használatát.

Miután két végrehajtójelünk van, két logikai függvényt kell felírni:

$$\begin{aligned} Q1 &= (I1 + M1) \cdot I2 \\ Q2 &= Q1 \end{aligned} \qquad (1.1)$$

I1	I2	M1	Q1	Q2
0	0	0	0	0
0	0	1	0	0
0	1	0	0	0
0	1	1	1	1
1	0	0	0	0
1	0	1	0	0
1	1	0	1	1
1	1	1	1	1

(1.2)

Kétállapotú jelek közötti logikai összefüggést megadható táblázatosan (igazság tábla 1.2 kifejezés) és négy vagy több logikai jel esetén Karnough táblával.

Megjegyzés: A Karnough táblának többféle ábrázolási módja van.

Az 2. példában leírt feladat nem adható meg pusztán BOOL algebrai logikai függvényekkel, mert időmérést tartalmaz. Az ipari irányító berendezések fejlesztő szoftvere a gyakran előforduló feladatokra kész szubrutinokat kínál, ami a felhasználói programban funkció blokként vagy függvényként [2] hívható meg. A funkció blokk bemeneti és kimeneti értékei számára a rendszer szoftver memória területet foglal le. A kimenet csak olvasható, a bemenet írható és olvasható. Ha vannak olyan szubrutinok, hogy:

- az „S” bemeneti jel (logikai 1) magas értékét adott idővel később követi a „T” kimenet magas értékre váltása és az „S” bemeneti jel (logikai 0) alacsony értékénél a „T” kimenet értéke alacsony
- az „S” bemeneti jelének magas értékénél a „T” kimenet értéke előírt módon alternál és az „S” bemeneti jel (logikai 0) alacsony értékénél a „T” kimenet értéke alacsony

akkor az 1.2 táblázattal megadott kétállapotú változókkal megadhatók a 2. mintafeladat logikai függvényei (1.2 kifejezés).

1.2 táblázat. Az 2. mintafeladat változói

zsalu nyitva	I1
töltésidő mérése	S1_TON
töltésidő letelt	T1_TON
tartály tele	I2
zsalu nyisson	Q1
töltés jelzése	S1_BLINK
villogtatás	T1_BLINK
villogó lámpa	Q2

Megjegyzés: A szubrutinoknak van típusa és egy szubrutint többször felhasználhatunk. A deklarációban jelezni kell a típust, és hogy melyik adatbázisból (memória területről) dolgozzon. A szimbolikus nevek használata megengedett.

A logikai függvények felírásakor beleütközünk abba a problémába, hogy a feladat aluldefiniált, mert nincs megadva, hogy milyen összefüggésre nyit ki a zsalu. Használjunk tartó-relét a zsalu (Q1 kimenet) vezérlésére. Az ismeretlen indító parancs magas értékbe billenti, és a 2. mintafeladatban megadott feltétel állítja vissza a Q1 kimenetet. Ez alapján a logikai függvények:

$$\begin{aligned} &\mathrm{S1\_TON} = \mathrm{S1\_BLINK} = \mathrm{I1} \\ &\mathrm{RESET\,Q1} = \overline{\mathrm{I1}} \cdot (\mathrm{I2} + \mathrm{T1\_TON}) \qquad (1.2) \\ &\mathrm{Q2} = \mathrm{T1\_BLINK} \end{aligned}$$

Amikor Q1=0 értékre vált a zsalu elkezd záródni, így I1 alacsony értékű lesz, és ezért a két időzítő alaphelyzetbe áll és megszűnik a villogás.

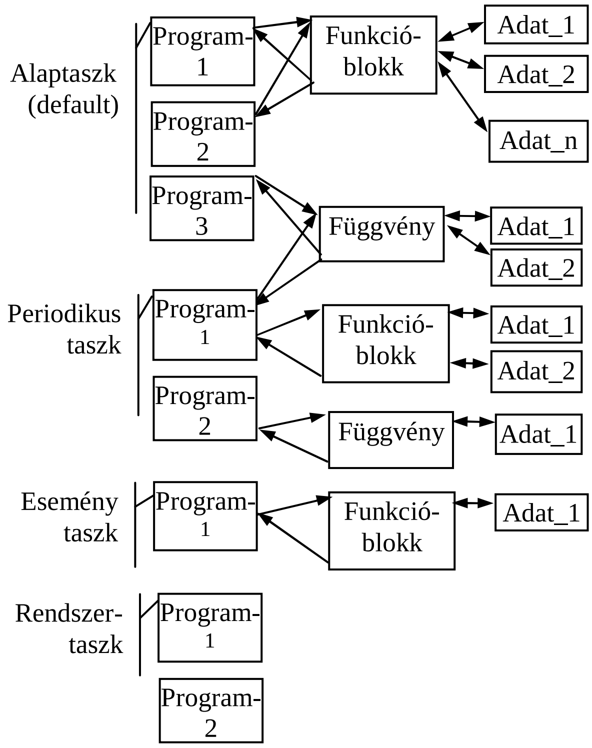
A SET és RESET műveleteket tartalmazó logikai függvényekre nem alkalmazható az igazság-tábla.

1.2. Irányítástechnikai programozási nyelvek

Az MSz EN 61131-3 szabvány adja meg az ipari irányító berendezések (PLC-k, folyamatirányítók, ipari számítógépek) programozási nyelveinek felépítését (taszkok, programok, funkcióblokkok, függvények [2]).

A 1.2. ábra mutatja a felhasználói program szerkezetét. A rendszertaszk programjai az ipari irányító berendezés rendszerszoftvere. A rendszertaszk programjai nem hívhatnak meg funkcióblokkot vagy függvényt, mert mint legmagasabb prioritású nem adhatja át az irányítást.

Az MSz EN 61131-3 szabvány nem korlátozza, hogy egy taszkban hány program futhat, és hogy egy program hány funkcióblokkot és függvényt hívhat meg. Periodikus és esemény taszkból nem szabad időszámlálást alkalmazó funkcióblokkot meghívni.



a. 1.2. ábra. A felhasználói program szerkezete

Terjedelmi okokból az 1.2 ábra nem mutat példát arra, hogy funkcióblokkból is meghívható függvény.

Megjegyzés: Az egyes gyártók csak a legnagyobb teljesítményű eszközükben szolgáltatják a teljes palettát. Minél olcsóbb az eszköz, annál kevesebbet tud. A rendszertaszk és az alaptaszk egy programmal mindenképp kell.

Az MSz EN 61131-3 szabvány adja meg az ipari irányító berendezések programozási nyelveinek szintaktikáját, és azokat a funkcióblokkokat, függvényeket, amelyeket kötelezően tartalmaznia kell a fejlesztő szoftvernek.

Megjegyzés: A felhasználói szoftver egyes programjai (1.2 ábra) megírhatók különböző nyelveken. Az assembler szintű nyelvek egymásba átkonvertálhatók, a magas szintű nyelvek assembler szintűre átkonvertálhatók. A felhasználó is írhat funkcióblokkokat vagy függvényeket.

A felhasználói programok (alkalmazások) az 1.2, ábrának megfelelő szerkezetűek. Az 1.2. ábrán látható programrészek, funkcióblokkok, függvények írhatók különböző szabványos programnyelven. Az MSz EN 61131-3 szabványban definiált programozási nyelvek és főbb jellemzőik:

Strukturált szöveg (ST: Structural Text)

Magas szintű, szöveges programozási nyelv. Szintaktikája a PASCAL nyelvével rokon.

- Az 1.2. ábrának megfelelően engedélyezi a szubrutinhívásokat. A változólistában kell megadni a szubrutin (funkcióblokk, függvény) szimbolikus nevét, típusát (pl: TON, CTU, stb.) és ezután a szimbolikus név egyben szubrutinhívó is.
- A logikai műveleteket (AND, OR, NOT, stb.) vagy függvényeket (pl: SIN(változó). SQRT(változó), stb.) az IF-THEN-ELSE vagy CASE feltételes szerkezetekben kell megadni.
- Engedélyezi a REPEAT-UNTIL; WHILE-DO ciklusszervező utasítások alkalmazását.
  Megjegyzés: Ez hasznos lehet például memória terület áthelyezésekor, de az alaptaszkban elhelyezett programban nem célszerű alkalmazni.
- A „legyen egyenlő” (:=) értékadó utasítással rendelhetők a szimbolikus változó nevekhez konstans értékek, funkcióblokk be, és kimenetek, stb.

Funkciótérkép (FC: Function Chart)

Magas szintű, grafikus programozási nyelv. Ez két különböző nyelv: Sorrendi funkciótérkép (SFC Sequential Function Chart) és Folytonos funkciótérkép (CFC Continuous Function Chart). Mindkettő összetett technológiák, gépcsoportok összehangolt működésének programozására hatékony. Az ismertetésük meghaladja a jegyzet célkitűzéseit.

Utasítás soros (IL: Instruction Line)

Alacsony szintű, szöveges programozási nyelv. Szintaktikája az CPU assembler nyelvekkel rokon. Alapegysége a **szekvencia**. A szekvencia egy logikailag zárt műveletsor, ami egy vagy több utasítássorral írható le.

Egy utasítássorból álló szekvencia a feltétel nélküli szubrutinhívás (CALL) és a feltétel nélküli ugrás (JMP).

A több utasítássorból álló szekvencia mindig szekvencia kezdő LD (Load: Töltsd be) vagy LDN (Load Not: Töltsd be negáltját) utasítással kezdődik. A szekvencia záró utasítások: ST (Store: Tárold), STN (Store Not: Tárold a negáltját), SET, RESET. Ugyancsak szekvencia záró utasítások: JMC (Jump Conditional: Feltételes ugrás logikai „1” értékre), JMCN (Jump Conditional Not: Feltételes ugrás logikai „0” értékre), CALLC (Call Conditional: Feltételes szubrutinhívás logikai „1” értékre), CALLCN (Call Conditional Not: Feltételes szubrutinhívás logikai „0” értékre). Szekvencia záró utasításból több is lehet egy szekvencia végén. Miután az utasításokat egymás után hajtja végre a rendszerprogram, ugró utasítások esetén nem mindegy a szekvencia záró utasítások sorrendje.

Megjegyzés: Az átugrott utasításokat nem hajtja végre a program.

A szekvencia kezdő és záró utasítások között lehetnek logikai utasítások: AND, ANDN, OR, stb., komparáló utasítások LT (Less Than Kisebb, mint), GT (Greater Than Nagyobb, mint), stb., valamint aritmetikai utasítások: ADD (Additional Összegzés), SUB (Subtraction Különbségképzés), stb.

Létra diagram (LD: Ladder Diagram)

Alacsony szintű, grafikus programozási nyelv. Szintaktikája a kontaktus logikán alapul. A kontaktus logikában a kontaktusok és a tekercsek kétállapotú változók. A kontaktusokon átfolyó áram gerjeszti a tekercset és a gerjesztett tekercs meghúzza (a nyitottat zárja, a zártat nyitja) a hozzátartozó kontaktusokat. Alaphelyzetben a nyitott kontaktus a ponált értéket és a zárt kontaktus negált értéket jelent. Az MSz EN 61131-3 szerinti grafikus jelölések:

—| |— : Ponált bináris változót reprezentáló kontaktus

—|/|— : Negált bináris változót reprezentáló kontaktus

—( )— : Ponált bináris változót reprezentáló tekercs

—(/)— : Negált bináris változót reprezentáló tekercs

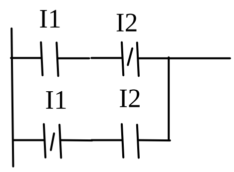
—(S)— : SET művelet

—(R)— : RESET művelet

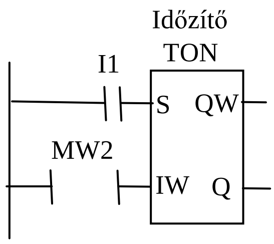
Programozáskor a változók szimbolikus neve a változók felett jelenik meg. A sorba kötött kontaktusok AND kapcsolatba, a párhuzamosan kötött kontaktusok OR kapcsolatban vannak. Kizáró vagy (EXOR) kapcsolat megvalósításához a változót reprezentáló ponált és negált kontaktusra is szükség van (1.3.a ábra).

A szubrutinhívás nem része az eredeti kontaktus logikának. MSz EN 61131-3 LD programozási nyelvben a szubrutinhívást egy téglalap jelképezi, amelynek baloldalán a bemeneti változók (1.4. ábra S, IW), jobboldalán a kimeneti változók (1.4. ábra Q, QW) csatlakoznak. A téglalap felett van a típus, és a felett a szimbolikus név. A szubrutin memória területet foglal le a be és kimeneti változói részére, ami a szimbolikus név alapján címezhető (pl.: Időzítő.S).

A szubrutin bemeneti változóihoz hozzá kell rendelni a feladat be- és kimeneti, vagy belső változóit. A nem kétállapotú változókat széjjelhúzott kontaktus jelképezi (1.4. ábra). A szubrutin kimeneti változóihoz csak akkor kell tekercset hozzárendelni, ha fizikai kimenethez akarjuk csatlakoztatni.



1.3. ábra EXOR kapcsolás



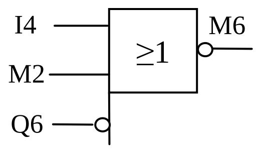
1.4. ábra TON típusú időzítő

Funkcióblokk diagram (FBD: Function Block Diagram)

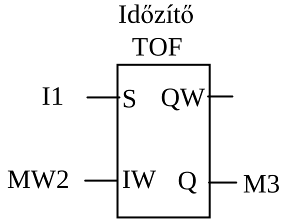
Alacsony szintű, grafikus programozási nyelv. Szintaktikája a digitális alapáramkör logikán alapul. A digitális alapáramkör logika több bemenetű AND, OR, stb. logikai kapukkal írja le a megvalósítandó feladatot.

Az MSz EN 61131-3 FBD programozási nyelvben a logikai műveleteket négyzet jelképezi, amelynek baloldali hosszabbított éléhez a logikai függvény argumentum változói (1.5. ábra I4, M2, Q6), jobboldalához az eredményváltozó (1.5. ábra M6) csatlakozik. A négyzetben van jelezve a logikai függvény típusa. A négyzetbe írt „&” az AND, a „≥1” az OR és a „=1” az EXOR logikai kapcsolatot jelenti. A négyzet csatlakoztatási pontjainál elhelyezett üres karika a változó negált értékét jelzi.

Az MSz EN 61131-3 FBD programozási nyelvben a szubrutinhívást szintén egy téglalap jelképezi. A szubrutin be és kimeneti változóihoz hozzá kell rendelni a feladat be és kimeneti, vagy belső változóit. Nem szükséges a szubrutin minden be és kimeneti változójához értéket rendelni.



1.5. ábra EXOR kapcsolás

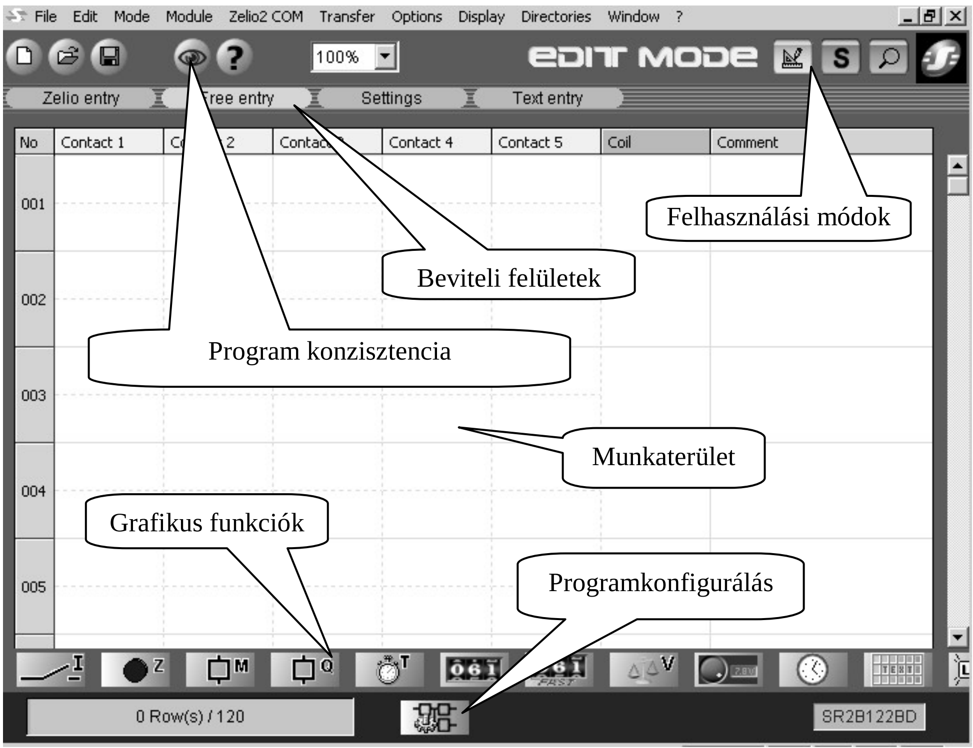


1.6. ábra TOF típusú időzítő

1.3. Vezérlő relé és programozása

A vezérlő relé készülékcsaládot piaci megfontolások alakították ki. Az egyszerűen átlátható feladatokhoz, a huzalozott logika helyett olyan programozható eszközt kínálnak a gyártók, ami olcsó és a programozása szakmunkás szinten is elsajátítható. Az ipari vezérlők 1.2. ábrán látható felhasználó programszerkezetéből csak a rendszer taszk és az alaptaszk fut. Az alaptaszkhoz csak egy korlátozott méretű program rendelhető. A funkció blokkok típusa és száma a felhasználó által nem befolyásolható és erősen korlátozott. A vezérlő relék fejlesztő szoftverében beágyazva található a programozást segítő, a vezérlő relé működését szimuláló funkció. A legtöbb vezérlő relé kontaktus logikával programozható, de a programozás szintaktikája nem azonos a szabványos MSz EN 61131-3 LD programozási nyelvvel és gyártófüggő.

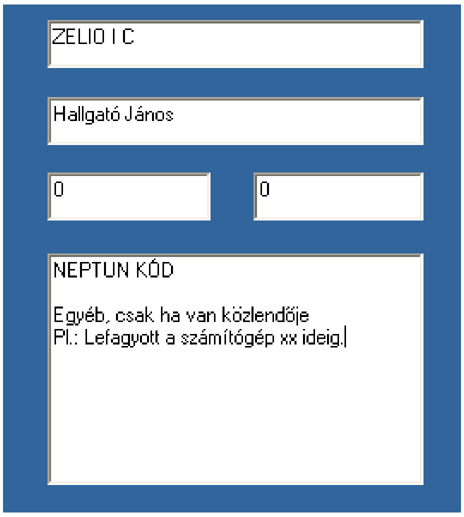
Az Automatika I. laboratóriumi mérésekben a ZelioSoft fejlesztő szoftver használják, amelynek az „Automatika I. Laboratóriumi mérések” jegyzetben található leírását az alábbiakkal egészítjük ki.



1.7. ábra ZelioSoft programozói felülete

Kezdje az önálló feladatának megoldását a „Programkonfigurálás” ikon (1.7. ábra) megnyitásával és csak a „Properties” fülön megjelenő adatmezőket töltse ki az 1.8 ábrán látható módon. A programmal kapcsolatos egyéb megjegyzéseit a „Munkaterület” „Comment” oszlopába írja (1.7. ábra). Ha a „Comment” oszlopban több sorba tördelve akar írni, akkor használja a Ctrl Enter billentyű kombinációt.

Megjegyzés: A feladat sorszámát és a feladatban nem definiált közbenső változó célját mindenképpen írja be a „Comment” oszlopba.



1.8 ábra Programkonfigurálás

Discrete inputs		
01	I1	VészLe
02	I2	Start
03	I3	KapuFent
04	I4	KapuLent
05	IB	OptoKint
06	IC	
07	ID	
08	IE	

1.9. ábra Text entry

Folytassa a feladatot a „Text entry” fül felnyitásával és kitöltésével (1.9. ábra).

Megjegyzés: Amikor egy változót lehelyez a „Munkaterületre”, akkor megjelenik mellette egy boríték, amit klikkeléssel kinyitva láthatóvá válik a változó szimbolikus neve. A szimbolikus név segít elkerülni a hibás (pl.: I3 helyett I4) változó alkalmazását. A változó szimbolikus neve a „Munkaterületről” is definiálható a változót kijelölve, majd jobb egér gomb, a legördülő menüből a „Properties” sor, majd a „Comment” fül választása.

A vezérlő relé programozás logikája, hogy ha egy változó egy logikai függvény argumentumában van, akkor kontaktus, és ha a logikai függvény eredménye, akkor tekercs szimbolizálja.

Tekercs lehet fizikai kimenet Q, vagy belső M memória terület, valamint szubrutinnal megvalósított grafikus objektum (időzítő T és R, számláló C, D, és R, szövegblokk TX, RX stb.) engedélyező, törlő, vagy egyéb bemenete.

No						Comment
01	M1	[	]	S	R	
02	M2	[	]	S	R	
03	M3	[	]	S	R	StartVolt
04	M4	[	]	S	R	KocsitÉrzett
05	M5	[	]	S	R	Áthaladt
06	M6	[	]	S	R	AutKézi
07	M7	[	]	S	R	
08	M8	[	]	S	R	
09	M9	[	]	S	R	
10	MA	[	]	S	R	
11	MB	[	]	S	R	

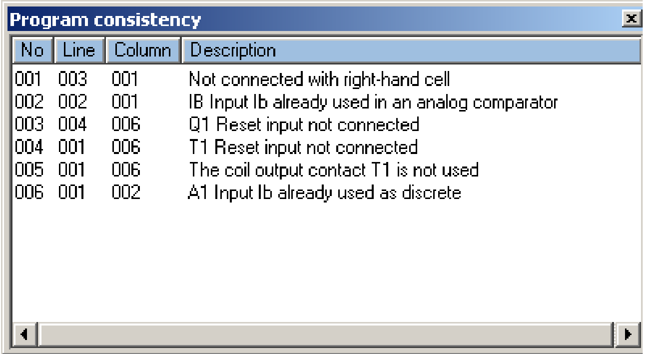
1.10. ábra Tekercs típusok

Kontaktus lehet fizikai bemenet I, vagy belső M memória terület, valamint szubrutinnal megvalósított grafikus objektum (időzítő T, számláló C, stb.) kimeneti eredmény változója. Továbbá vannak szubrutinnal megvalósított speciális kontaktusok (számláló komparátor V, analóg komparátor A, stb.).

Megjegyzés: Az időzítő, számláló, szövegblokk engedélyező tekercsének lehelyezése jelenti a szubrutinhívást. Az időzítő, számláló stb., tekercseinek vagy kontaktusának lehelyezési sorrendje tetszőleges. Ugyancsak szubrutinhívás a számláló komparátor, analóg komparátor, stb. speciális kontaktusok használata.

A kontaktusok alapállapota lehet „Normally open” záró (ponált változó) és „Normally closed” bontó (negált változó). A lehelyezett kontaktust kijelölve, a jobb egér klikk felhoz egy menü ablakot, amelyből a kívánt állapot kiválasztható.

Programírás közben egy-egy részfeladat befejezésekor hasznos szintaktikailag ellenőrizni a megírt programrészletet a „Program konzisztencia” ikon (1.7. ábra) segítségével. Az 1.11. ábrán látható, hogy a „Program konzisztencia” hibaüzenete megmutatja a hiba helyét a „Munkaterületen” (sor, oszlop).



Program consistency

No	Line	Column	Description
001	003	001	Not connected with right-hand cell
002	002	001	IB Input Ib already used in an analog comparator
003	004	006	Q1 Reset input not connected
004	001	006	T1 Reset input not connected
005	001	006	The coil output contact T1 is not used
006	001	002	A1 Input Ib already used as discrete

1.11. ábra A „Program konzisztencia” ikon hibaüzenete

Szimulációs üzemmódban a kontaktusokra klikkelve válthatja azok állapotát. A nyomógombok (pl.: Z1), vagy a speciális kontaktusok (pl.: A1) csak egy pillanatra változnak a klikkelés hatására. Ha nem átmenetileg akarja a kontaktust zárni vagy nyitni, akkor a „Force and maintain” funkciót kell választania. A „Force and maintain” hatását az előbbi párbeszédablakban található „Release” funkció oldja fel. Rövid, nyomógombszerű átkapcsolást a „Momentary forcing” füllel valósíthat meg.

Fontos: Minden olyan változó, amit tekercshez lehet rendelni, figyelembe vehető kontaktusként egy másik, esetleg ugyanabban a szekvenciában. Fordítva az állítás nem igaz, mert vannak olyan kontaktusok, amelyekhez tartozó változó nem lehet tekercs.

2. Egyszerű vezérlési láncok

A ZELIO I. mérés célja, hogy a hallgatók logikai függvénnyel, igazság táblázattal vagy Karnough táblával, továbbá szövegesen megfogalmazott néhány szekvenciából álló egyszerű feladatokon keresztül megismerjék a vezérlő relé programozásának technikáját, a vezérlő relékben alkalmazott legfontosabb funkció blokkokat. Az egyes feladatok egymástól függetlenek, nem egy felhasználói alkalmazás részfeladatai.

2.1. Ellenőrző kérdések

Az ellenőrző kérdések ábráit számítógép szolgáltatja.

1. Válassza ki a felsorolásból a vezérlés működési vázlatát alkotó elemek nevet.
2. Válassza ki a felsorolásból azon jelek, jellemzők neveit, amelyek csak a vezérlés működési vázlatában fordulnak elő.
3. Jelölje be azokat az állításokat, amelyek a vezérlésre igazak.
4. A felsorolt állítások közül melyiknek vagy melyeknek kell teljesülnie, hogy irányítási feladatot vezérléssel lehessen megvalósítani?
5. A felsorolt állítások közül melyik a szabályozás alapelve?
6. A felsorolt értékek közül melyek tartoznak a szabványos ipari jeltartományok közé?
7. Az MSz EN 61131-3 LD programozási nyelvben hogyan történik a szubrutinhívás?
8. Az vezérlő relé kontaktus logikában hogyan történik a szubrutinhívás?
9. Az ábrán látható idődiagramok közül melyek tartoznak a TON (Time On Delayed) időzítőhöz? (Több változatban)

10. Az ábrán látható idődiagramok közül melyek tartoznak a TOF (Time Off Delayed) időzítőhöz? (Több változatban)

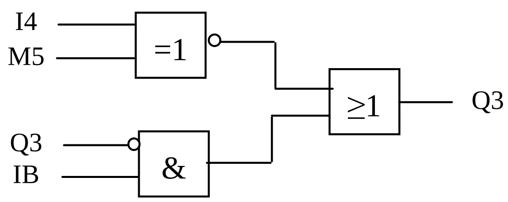
11. A felsorolt kontaktushoz rendelt változók közül melyek nem rendelhetők tekercshez? (Több változatban)

2.2. A minta feladatsor

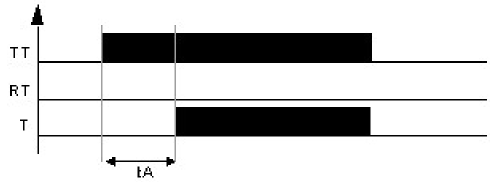
Programozás közben töltse fel a szimbolikus nevek listáját. Használja a feladatokban megadott neveket, ha van ilyen.

1. Ha az I1 (érzékelő1), I2 (érzékelő2), és MB (érzékelő3) közül bármelyik kettő egyszerre jelez, akkor az M1 (riasztás) marker billenjen magasba. Az M1 markert a Z1 (nyugtázó) nyomógomb törölje, amennyiben nem áll fenn a riasztás feltétele.
2. Amennyiben az M1 marker magas, akkor a Q1 (lámpa) kimeneten 2:1 kitöltési tényezőjű és T = 3 sec. periódusidejű villogás (T1: villogó) legyen. (A 2.1. ábrán láthatók a választható idődiagramok.)
3. Az IE (szint) bemenetre érkező 8 V-nál magasabb jel állítsa logikai „1” értékre az MB belső változót, és az IE bemenetre érkező 3 V-nál kisebb jel állítsa ismét logikai „0” értékre!
4. Oldja meg az alábbi logikai függvényekkel megadott feladatokat. Az M5 marker az F1 logikai függvény minden 0 → 1 átmenetére váltson értéket!

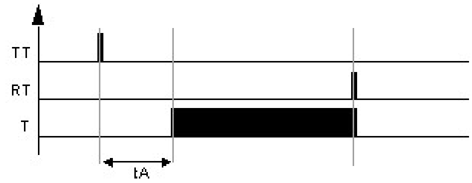
   $F1 = (I3 + \overline{M4}) \cdot I4$
5. Konvertálja át az alábbi funkció blokkokkal megadott logikai kifejezést létra diagramos leírássá!



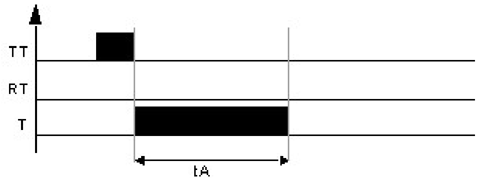
6. Számolja meg az M1 marker magas szintjeinek számát! Amíg nem éri el a 4-et a Q2 (normál) kimenet legyen magas szinten! Négy vagy a feletti érték esetén a Q2 kimenet 2 másodperces periódusidővel villogjon! A számlálót a Z2 (törlő) nyomógomb állítsa alaphelyzetbe!



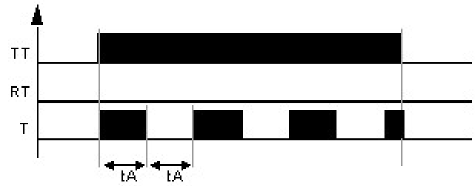
a) Késleltetve meghúzó



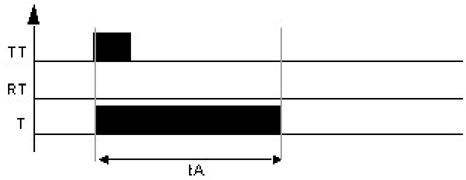
b) Start-stop késleltetve meghúzó



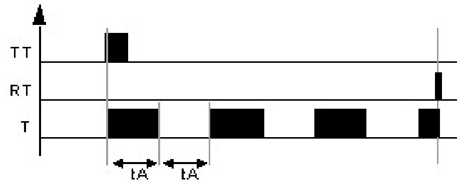
c) Adott hosszú pulzus lefutó élre



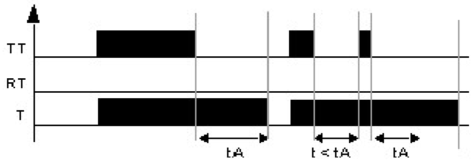
d) Szimmetrikus villogó



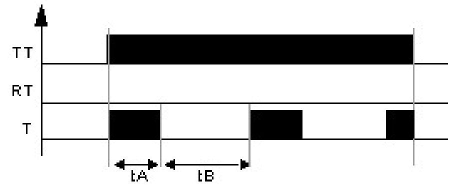
e) Adott hosszú pulzus felfutó élre



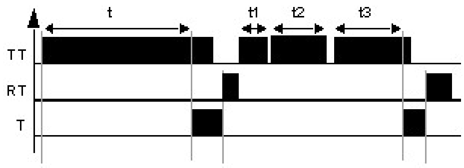
f) Start/Stop villogó



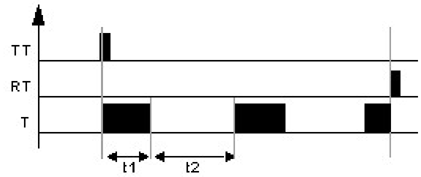
g) Késleltetve elengedő



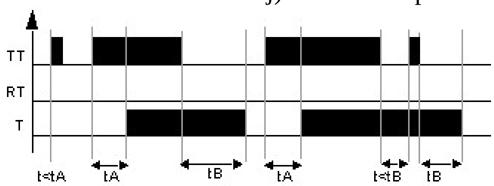
h) Különböző periódus idejű villogó



i) Összegzett idő



j) Különböző periódus idejű Start/Stop villogó



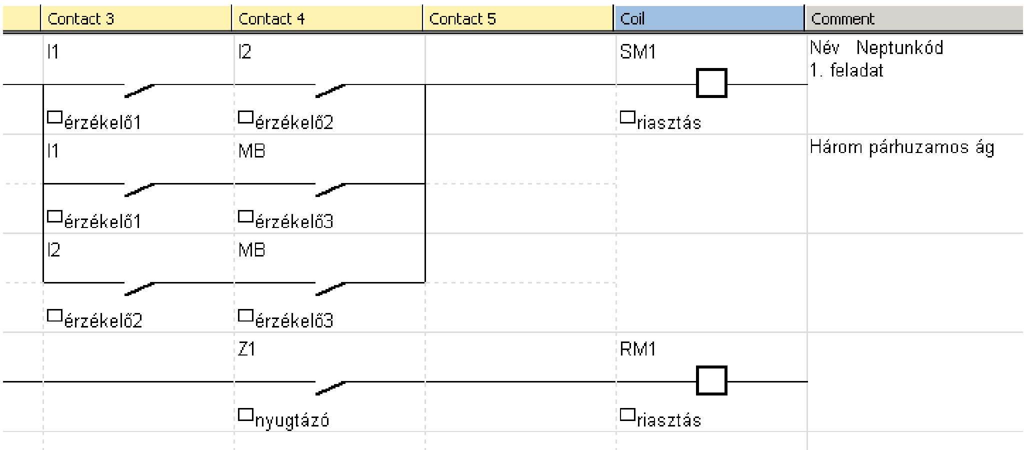
k) Késleltetve meghúzó és elengedő

2.1. ábra A ZELIO vezérlő relé időzítőinek idődiagramjai

2.3. A minta feladatsor megoldása

1. Az első feladatba három jelből háromféleképp lehet két jelet kiválasztani. Ennek megfelelően három párhuzamos ág gerjeszti az M1 markert, és mindegyik ágban két-két sorosan kapcsolt kontaktus van.
   $\text{If } (\mathrm{I1} \cdot \mathrm{I2} + \mathrm{I1} \cdot \mathrm{MB} + \mathrm{MB} \cdot \mathrm{I2}) \text{ Than Set M1}$

   Fontos: Ha tartó relét alkalmazunk, akkor a Set és Reset tekercseket mindig párban kell alkalmazni!



2.2. ábra Az első feladat létra diagramja

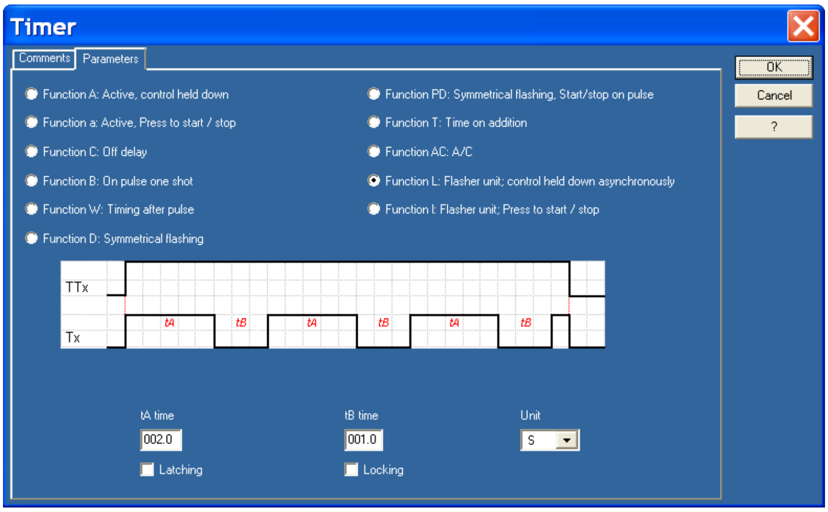
A „Program konzisztencia” gomb piros! Ráállítva a kurzort kizöldül. Felnyitva a következő szöveg olvasható: „The coil output contact M1 is not used” (Nincs használva az M1 tekercs kontaktusa.).

Megjegyzés: Ez nagyobb összetett feladatírás közben természetesen előforduló jelenség. Egy alkalmazás teljes programjának elkészültekor azonban hibára utal.

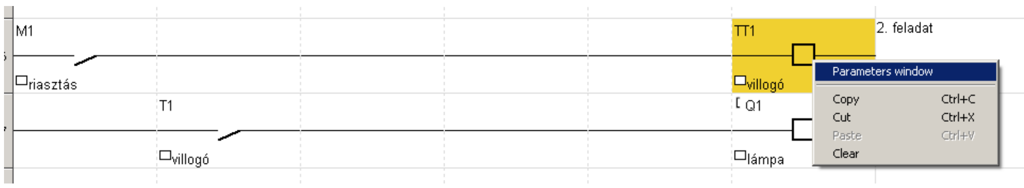
Fontos: Amikor szimulációs üzemmódba vált át (a „Simulation” ikonnal 1.7 ábra), és a „Run” ikonnal indítva a szimulációt teszteli az áramkört, akkor azt tapasztalja, hogy a Set / Reset funkció nem működik! Ez Zelio sajátosság. A tervezők úgy gondolták, hogy ne imitáljon helyes működést a program, amíg nincs használva az M1 kontaktus. Ha lehelyez egy szabad sorba egy M1 kontaktust, akkor helyesen működik. Ilyenkor „Program konzisztencia” a „No connected with right-hand cell” üzenetet küldi.

Megjegyzés: Teszteléskor a létra diagram az aktuális logikai állításnak megfelelő színt mutatja. Ez tartó- és impulzus relé alkalmazásakor nem feltétlen azonos a valóságos állapottal. Alkalmazza az alsó ikonsort!

2. A 2:1 kitöltési tényező és a 3 sec. periódus idő azt jelenti, hogy 2 másodpercig világít és 1 másodpercig sötét a Q1 kimenet.



2.3. ábra A második feladat időzítőjének paraméterezése



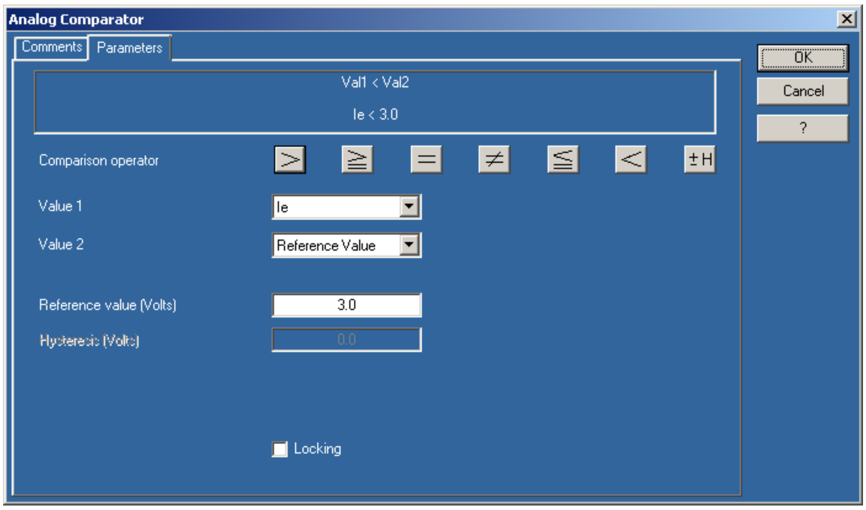
2.4. ábra A második feladat létra diagramja

Fontos: Az időzítő T1 eredmény kontaktusát követi a Q1, ezért csak normál relét szabad alkalmazni. Az időzítő paraméterezése akár a tekercs, akár a kontaktus letételekor megtörténhet. Ha az időzítő nincs paraméterezve, akkor a „Program konzisztencia” a „Time out value zero” hibaüzenetet küldi.

Teszteléskor használja a szimulációs képernyő alján levő állapotjelző ikonokból a funkció blokkok állapotait mutatót (2.5. ábra)!

2.5. ábra
A funkcióblokkok állapotát mutató ikon

3. A Zelio vezérlő relé speciális szubrutinhívó kontaktusokat kínál fel az „Analog comparators” ikon alatt. Az analóg komparátor szubrutin függvény típusú, mert eredménye azonnal felhasználásra kerül. A paraméterező ablaka (2.6 ábra) jobb egérgomb klikkel válaszható ki. A 2.6. ábrán az alsó határérték beállítása látható.

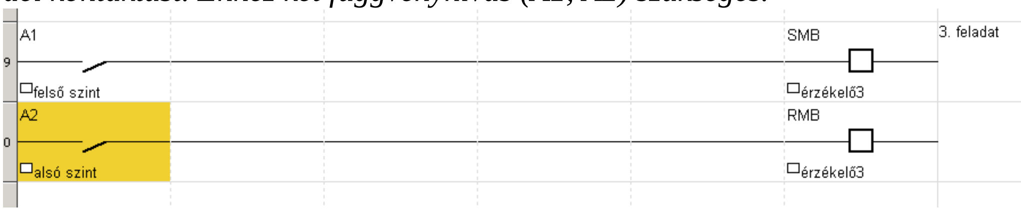


2.6 ábra Az analóg komparátor paraméterező ablaka

Fontos: Ne feledje a logikai relációt megfelelően beállítani!

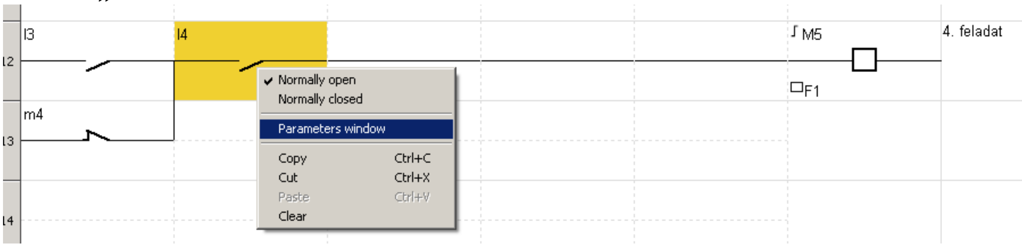
Az alkalmazott vezérlő relé típus sajátossága, hogy négy bemenete (IB, IC, ID, IE) lehet kétállapotú és analóg bemenet is, de csak vagy ilyen, vagy olyan! Ha egy program ugyanazt a bemenetet diszkrét és analóg bemenetként is megcímzi, akkor a „Program konzisztencia” a „IB Input Ib already used in an analog comparator” és „A1 Input Ib already used as discrete” hibaüzeneteket küldi a sor és oszlop pozíció megjelölésével.

Megjegyzés: Ugyanaz az analóg bemenet használható több analóg komparátorban. A példa az IE bemenetre érkező analóg jel két pontjához rendel kontaktust. Ehhez két függvényhívás (A1, A2) szükséges.



2.7 ábra A harmadik feladat létra diagramja

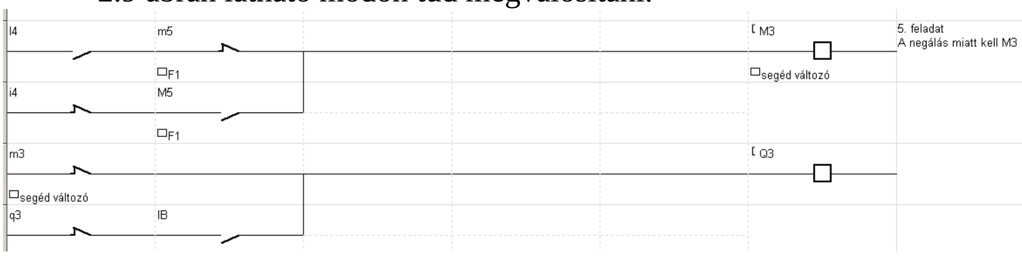
4. A 0 → 1 átmenetre történő értékváltás az impulzus relék sajátossága. A létra diagramon (2.8 ábra) látható a jobb egér klikkel felhozható párbeszéd ablak, amin beállítható a változó ponált vagy negált értéke és a „Parameter window” fül választásával elnevezhető a változó.



2.8 ábra A negyedik feladat létra diagramja

Emlékeztető: Teszteléskor használja az alsó ikonsort (1.7. ábra), mert a tartó és az impulzus relé állapota nem mindig egyezik meg a gerjesztés pillanatnyi állapotával!

5. Az ötödik feladat egyik érdekessége, hogy az I4 és a M5 változók „kizáró vagy” logikai kapcsolatban vannak, amit a kontaktus logika csak a 2.9 ábrán látható módon tud megvalósítani.



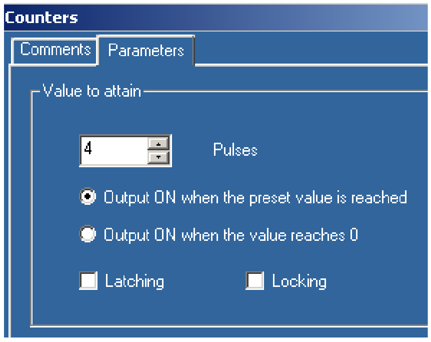
2.9 ábra Az ötödik feladat létra diagramja

A feladat másik érdekessége, hogy közbenső (M3) segéd változót kell definiálni a negálás megvalósítása érdekében.

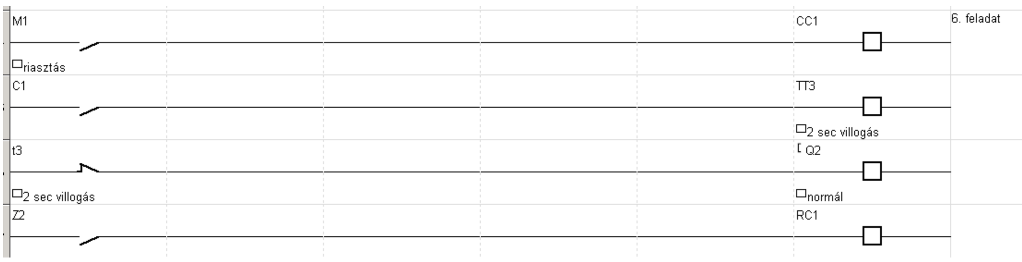
Megjegyzés: A negált kimenetű logikai funkcióblokkok De Morgan szabály szerinti átalakítása időigényes és hibaforrás. A huzalozott logikáktól eltérően a programozható eszközökben nincs kemény korlátja a belső változók alkalmazásának, ezért ez a legkézenfekvőbb megoldás.

6. A feladat megoldásához (2.11. ábra) szükség van egy számlálóra. Elegendő felfelé számolni és alaphelyzetbe állni, vagyis a „D” irányváltó bemeneti tekercsre nincs szükség.

Amíg a számláló eredmény kontaktusa „0” értékű, addig villog a Q2-es kimenet. Ehhez újabb időzítőre (T3) van szükség. Az időzítőt a számláló eredmény kontaktusának negált értéke engedélyez, mert a 2.8 ábrán látható, hogy a számláló eredmény kontaktusa vagy a beállított érték elérésekor (Output On when preset value is reached), vagy „0” értéknél (Output On when value reached 0) magas értékű.



2.10 ábra A számláló paraméterezése



2.11. ábra A hatodik feladat létra diagramja

A feladat érdekessége, hogy T3 kontaktusa alaphelyzetben negált. Ezt a szimmetrikus villogó (2.1. ábra) idődiagramja teszi lehetővé.

3. Összetett vezérlési láncok

A ZELIO II. mérés célja egyszerű, szövegesen vagy Karnough táblázattal definiált feladatok önálló megoldása.

3.1. Ellenőrző kérdések

1. A felsoroltak közül melyik az alapegysége az MSz EN 61131-3 LD és IL irányítási nyelveknek?
2. A felsoroltak közül melyek a szabványos kétállapotú és analóg ipari jeltartományok?
3. Az ábrákon látható kapcsolások közül melyik valósítja meg a megadott logikai függvényt? (Több változatban)
4. A felsorolt Zelio vezérlő relé változók közül melyek alkalmazhatók reléként és kontaktusként is? (Több változatban)
5. A felsorolt Zelio vezérlő relé változók közül melyek alkalmazhatók csak kontaktusként? (Több változatban)
6. A felsorolt bemenetek közül melyek alkalmazhatók diszkrét és analóg bemenetként is?
7. Válassza ki, hogy a Zelio vezérlő relé melyik változó paraméterező ablakát látja a képen? (Több változatban)
8. Az ábrán látható idődiagramok közül válassza ki az állapotvezérelt időzítőket? (Több változatban)
9. Az MSz EN 61131-3 LD programozási nyelvben melyik változó generál a szubrutinhívást! (Több változatban)
10. Az ábrán látható idődiagramok közül válassza ki az élvezérelt időzítőket! (Több változatban)

3.2. A minta feladatsor

Programozás közben töltse fel a szimbolikus nevek listáját! Használja a feladatokban megadott neveket, ha van ilyen!

1. Amennyiben nincs motorhibajelzés az I1 bemeneten és I4 és IB indító nyomógombok egy időben legalább két másodpercig magas értéken vannak, akkor induljon a motor! A motort a Q1 kimenet „1” értéke vezérli. A motor azonnal álljon le motorhibajelzés, az I3 stop bemenet, továbbá az M1 belső változó hatására!
2. Az M1 belső változó a jobb oldalon látható Karnough táblának megfelelő értékeknél legyen magas értékű!
   Az A1 változó az IE bemenetre érkező 9 V-nál nagyobb jelérték esetén legyen magas értékű!

Q1

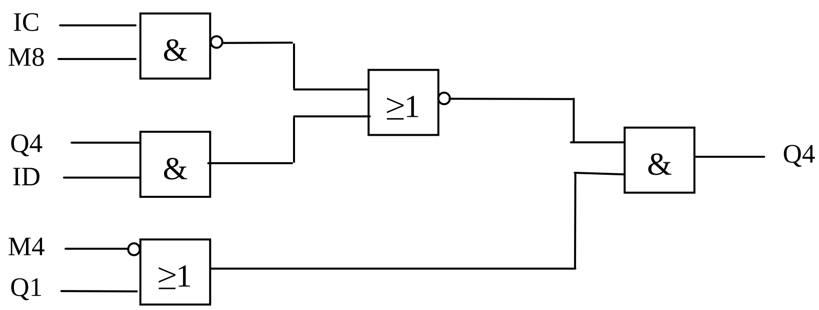
MC

	00	10	01	11
00	0	0	0	0
10	1	1	0	0
01	0	0	1	0
11	0	0	1	0

I3

A1

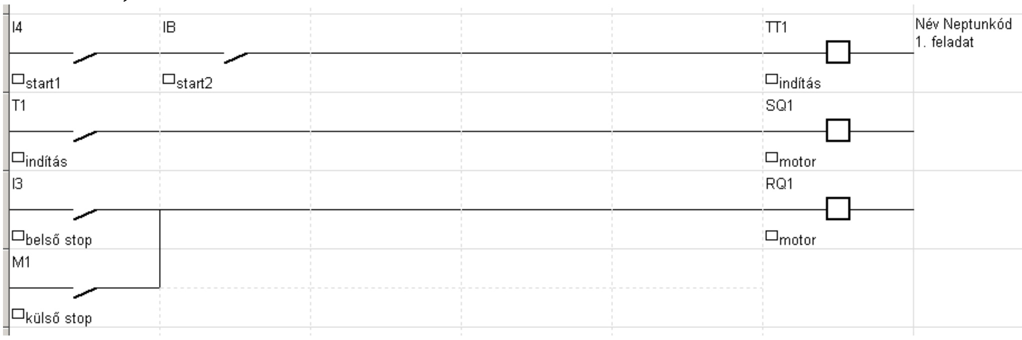
3. Ha a motor már legalább 5 másodperce működik, akkor két másodperces szünetekkel háromszor nyolc másodpercre nyisson ki a töltő szelep! A Q2 magas szintje vezérli a szelepet.
4. Számolja a töltő szelep működtetésének számát! A hatodik működtetés után, de a hetedik előtt billentse magasba az M2 markert! A markert és a számlálót a Z2 nyomógomb törli.
5. Ha az M2 magas értékű, akkor 3 másodperc periódusidejű villogás legyen a Q3-mas kimenete.
6. Konvertálja át az alábbi funkció blokkokkal megadott logikai kifejezést létra diagramos leírássá!



3.3. A minta feladatsor megoldása

Adjon szimbolikus neveket a változóknak!

1. Az első feladat egy két kezes indítás. Legyen start1 (I4) és start2 (IB) sorba kötve egy késleltetve meghúzó indítás időzítő (T1) engedélyező jele Ha 2 másodpercnél rövidebb ideig gerjesztik az időzítő engedélyező tekercsét, akkor az időzítő nulla bemeneti jelre nullázódjon le! A külső stop (I3) és a belső stop (M1) jeleket párhuzamosan kell kapcsolni, hiszen külön-külön is le kell állítaniuk a motort.



3.1. ábra Az első feladat létradiagramja

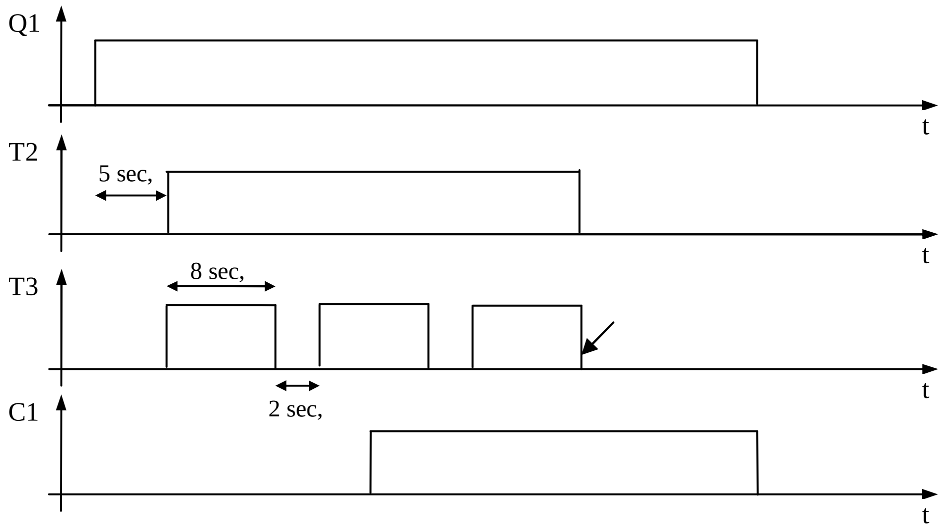
Megjegyzés: A „Program konzisztencia" hibátlan programot jelez. Ennek ellenére a szokásos módon ellenőrizze a programrészletet szimulációs üzemmódban, mert a „Program konzisztencia" csak a szintaktikai hibákat deríti fel, a logikai hibákat (a program nem az elvárások szerint működik) nem.

2. A második feladatban a négyváltozós Karnough táblában össze lehet vonni az egymás melletti cellákat. A második sorban levő két egyes cella együttes értéke $\overline{Q1}\cdot I3\cdot \overline{A1}$ és a harmadik oszlopban levő két egyes cella együttes értéke $MC\cdot Q1\cdot A1$ és így: $M1=\overline{Q1}\cdot I3\cdot \overline{A1}+MC\cdot Q1\cdot A1$.

3.2 ábra A második feladat létradiagramja

Az A1 analóg komparátor paraméterező táblázatát (2.6. ábra) ne felejtse el kitölteni!

3. A feladat megoldáshoz kell egy késleltetve meghúzó időzítő (T2) és egy olyan villogó (T3), amely 8 másodpercig magas és 2 másodpercig alacsony értékű jelet ad. Szükséges továbbá egy számláló, ami a háromszori működést számolja. A feladat megoldásához ajánlott a jelek idődiagramjának megszerkesztése.



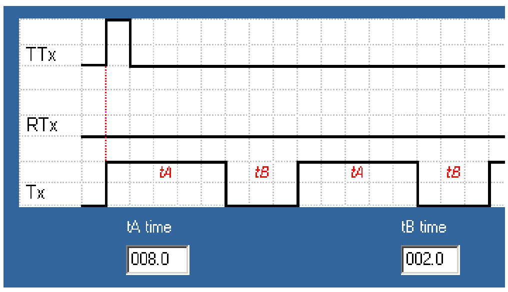
3.3. ábra A harmadik feladat idődiagramja

Megjegyzés: A feladat nem egyszerű. Ha elakad, lépjen tovább! Nem szükséges sorrendben megírni a feladatokat. A felhasználói program ciklikus feldolgozása következtében csak a SET, RESET utasításpárok létraágainak sorrendje kritikus.

A T3 időzítő állapotvezérelt (egytekercsű) időzítő, és a T2 engedélyezi a működését. Nem magától értetődő, de a T2 időzítőnek felfutó éllel vezérelt (kéttekercsű TT, TR) időzítőt célszerű választani! Így oldható meg. hogy a Q1 felfutó éle indítsa az időzítőt és a magas szint léte ne indítsa újra, valamint hogy a $\overline{T3}\cdot C1$ kombináció törölje.

A C1 számlálót a $\overline{Q1}$ értéke állítja alaphelyzetbe.

Megjegyzés: Ügyeljen arra, hogy a T3 időzítő paraméterezésekor (3.4. ábra) a 8 másodperces magas periódussal kezdjen!



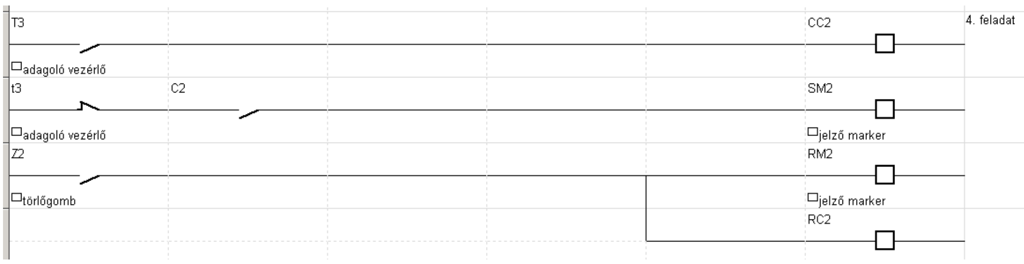
3.4. ábra A T3 időzítő paraméterezése

Megjegyzés: Ha a Q1 stop vagy hibajel hatására a hármas ciklus előtt válik „0” értékűvé, akkor a 3.5. ábra szerinti működés hibás lehet. Valóságos feladat megoldáskor ezeket a jeleket is figyelembe kell venni a T2 időzítő és a C1 számláló törlésébe. Az órai feladatban ennek kihagyása nem hiba.

3.5 ábra A harmadik feladat létradiagramja

Fontos: Az előírt működés szerint teszteljen! Kezdetben Q1 alacsony értékű. Használja a „Force and maintain” funkciót és folyamatos Q1 magas értékkel tesztelje a számlálást. A negyedik feladattal együtt is tesztelje a működést. Ekkor kétszer kell indítani a C1 számlálót. A Q1 alacsony értékre állításához a „Release” funkciót használja!

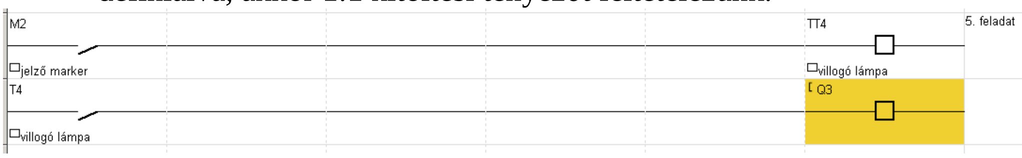
4. Ha a 3.3 ábrára úgy tekint, hogy a nyíllal jelölt helyen ért véget a hatodik impulzus, akkor a feladatmegoldáshoz csak még egy további C2 számláló kell. Az M2 markert a $\overline{T3}\cdot C2$ jel állítja magasba a C2 számlálót és az M2 markert a Z2 nyomógomb törli.



3.6 ábra A negyedik feladat létradiagramja

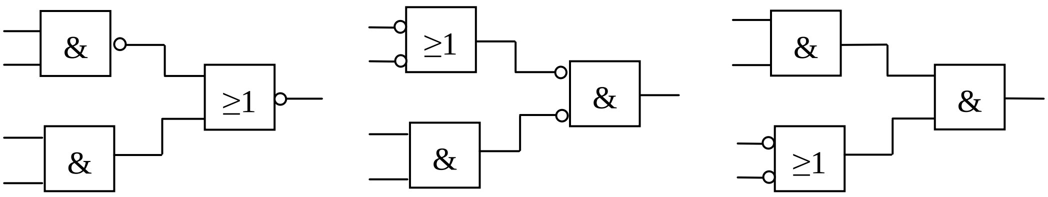
Emlékeztető: Ennél a feladatnál is zavaró lehet, hogy tesztelésekor nem működik tökéletesen, mert az M2 kontaktusa még nem volt felhasználva. A kontaktus letétele lehetővé teszi a pontos tesztelést vagy az 5. feladattal együtt célszerű tesztelni.

5. Ehhez a feladathoz csak egy újabb (T4) időzítőre van szükség. Ha nincs definiálva, akkor 1:1 kitöltési tényezőt feltételezünk.



3.7 ábra Az ötödik feladat létradiagramja

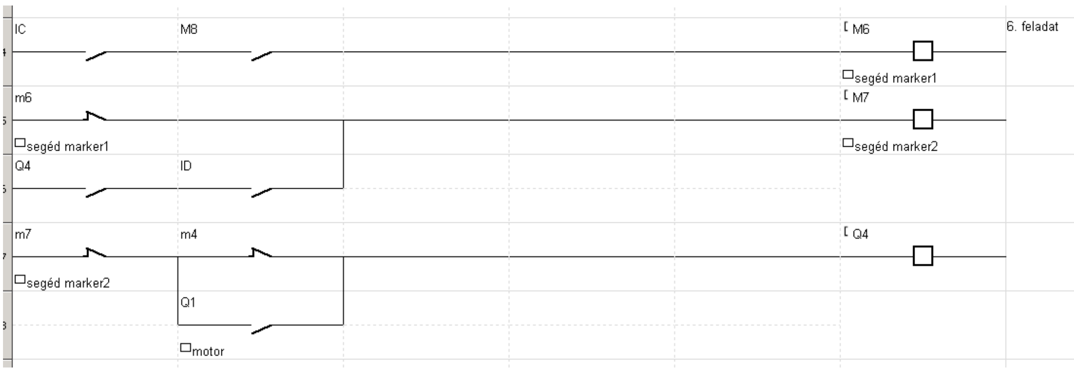
6. A feladat egyik megoldása, hogy az invertáló kimenetű funkció blokkokat a De Morgan szabállyal átalakítjuk.



3.8 ábra Az invertáló funkció blokkok átalakítása

$$Q4=\overline{\overline{IC\cdot M8}+Q4\cdot ID}\cdot(\overline{M4}+Q1)=IC\cdot M8\cdot(\overline{Q4}+\overline{ID})\cdot(\overline{M4}+Q1)$$

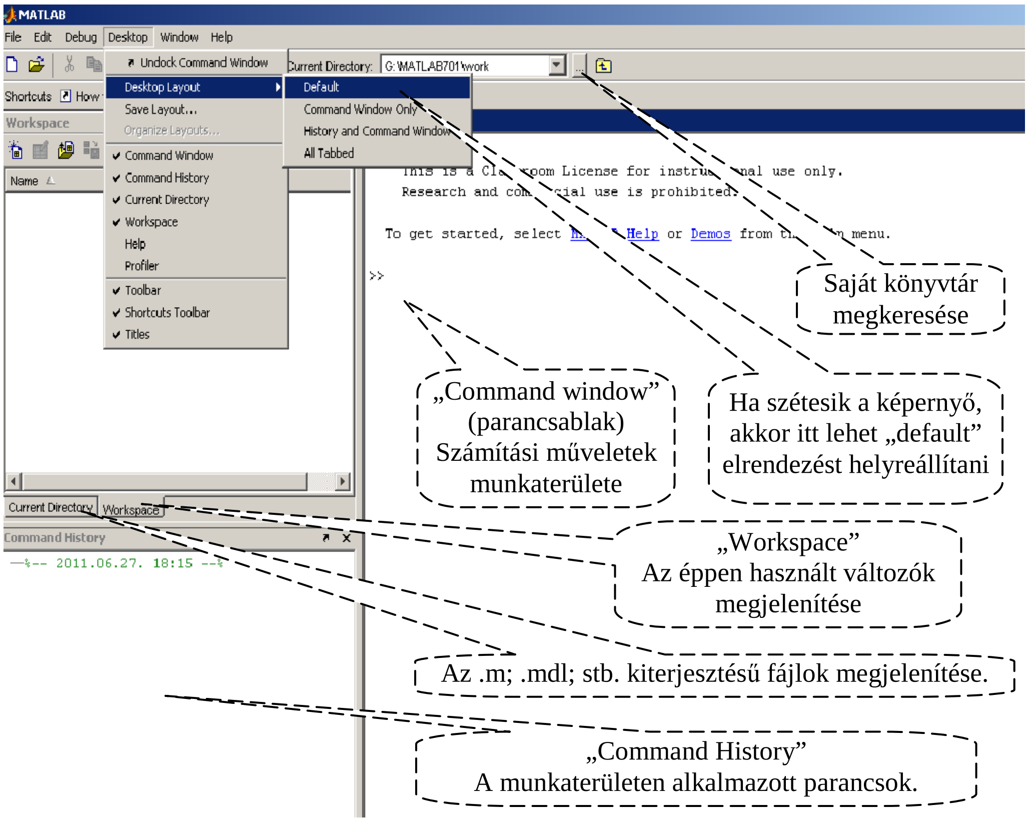
Kézenfekvőbb azonban további segéd markerek felvételével leprogramozni a 6. feladatot.



3.9 ábra A hatodik feladat létradiagramja

4. Bevezetés a MATLAB használatába

A MATLAB betűszó. A „Matrix Laboratory” rövidítése. Ez a legelterjedtebb matematikai program az egyetemi, kutatási, fejlesztési területen. Az alaprendszer számos, különböző tudományterületre kidolgozott „Toolbox”-szal és programozási feladatot kiváltó grafikus szerkesztő felületekkel (GUI: Grafical User Interface) egészíthetők ki. Az Automatika I. laboratórium méréseiben az alaprendszeren túl csak a „Control System Toolbox”-ot és a „SIMULINK” GUI-t alkalmazzuk. A MATLAB 7.0.1 nyitó felülete az alábbi:



4.1. ábra A MATLAB 7.0.1 program nyitóképernyő elrendezése

A MATLAB 2008-as programverzióban ugyanezek a képernyő funkciók vannak egy kicsit más elrendezésben.

4.1. Változók, változók adattípusai, algebrai műveletek

A változó nevek legfeljebb 31 karakter hosszúak lehetnek. Csak az **angol ABC betűit, számokat, és a _ jelet** szabad alkalmazni! Ez a megkötés a fájlnevekre és a mentés útvonalára is vonatkozik. **Figyelem**: A kis és a nagy betűk különböző karakternek számítanak!

A MATLAB a változók értékadására az = jelet alkalmazza. A parancsablakban (4.1. ábra), a >> promptjel után beírva a változó értékadó utasítását, a megadott számformátumtól függően, a MATLAB automatikusan deklarálja az adattípusokat.

>> k1=2	egész		
>> k2=2.0	valós	vagy	k4=3.6e-5 (0.000036 helyett!)
>> k3=2+1.2*i	komplex	vagy	k5=1.3e(i*pi/2)

Fontos: A MATLAB-ban tizedes pont van és nem tizedes vessző, valamint az egység képzetes vektor i és j egyaránt lehet, de mindig i jelenik meg!

A MATLAB logikájában a legtöbb változó mátrix. A vektor egy dimenziójú (egyszer n vagy n-szer egy), a skalárváltozó egy egyszer egyes mátrixnak felel meg.

Skalárváltozókra és függvényekre megkötés nélkül, vektorokra és mátrixokra a dimenzió szabályok betartásával érvényes a jobboldalon felsorolt hat művelet.

+	összeadás
-	kivonás
*	szorzás
/	osztás
\	baloldali osztás
^	hatványozás

Megjegyzés: komplex számok használata esetén van különbség.

’	konjugált transzponálás
.’	transzponálás

További speciális tömbműveletek láthatók a jobboldalon:

.*	elemek szerinti szorzás
./	elemek szerinti osztás
.^	elemek szerinti hatványozás
.\	elemek szerinti baloldali osztás

A mátrix értékadásának szintaktikája:

$$>> \mathrm{A} = [2,2,3;4,5,5] \tag{4.1}$$

ami a $\mathbf{A} = \begin{bmatrix} 2 & 2 & 3 \\ 4 & 5 & 5 \end{bmatrix}$ mátrixnak felel meg.

A szintaktika: szögletes zárójel, szám, vessző, szám, stb., pontos vessző, szám, stb. fontos!

A mátrix szintaktikából következik, hogy a sorvektor elemei vesszővel, az oszlopvektor elemei pontos vesszővel vannak elválasztva.

A parancsokat a parancsablakba írva az ENTER billentyűvel lehet érvényesíteni. A további szöveges leírásban a parancsablakba beírt parancsok baloldalon, a parancsablakban megjelenő eredmény a jobboldalon lesz.

Egy mátrix bármely sorát, például az **A** mátrix **a1** első sorát:

```
>>   a1=A(1,:)                                    (4.2)
          Eredmény:   a1=
                          2.0000    2.0000    3.0000
```

vagy bármely oszlopát, például az **A** mátrix $\mathbf{a2}^{\mathbf{T}}$ második oszlopát:

```
>>   >>a2=A(:,2)                                  (4.3)
          Eredmény:   a2=
                          2.0000
                          5.0000
```

sorvektorként (4.2. kifejezés), illetve oszlopvektorként (4.3. kifejezés) előállíthatjuk. Ha több sorból vagy oszlopból áll, akkor egymás melletti sorok vagy oszlopok esetén 4.4. kifejezés szerint, és egymástól független sorok vagy oszlopok esetén 4.5. kifejezés szerint kell megadni a parancsot.

Például ha az **A** mátrix 2. és 3. oszlopából akar generálni egy új **A23** mátrixot, akkor a:

```
>>   >>A23=A(:,2:3)                               (4.4)
          Eredmény:   A23=
                          2.0000    3.0000
                          5.0000    5.0000
```

szintaktikájú parancssort kell használni.

Ha az **A** mátrix 1. és 3. oszlopából akar generálni egy új **A13** mátrixot, akkor pedig a:

```
>>   >>A13=A(:,[1,3])                             (4.5)
          Eredmény:   A13=
                          2.0000    3.0000
                          4.0000    5.0000
```

szintaktikájú parancssort kell használni. Tetszőleges sorok kiválasztása a 4.2. és a 4.5. kifejezések összevetéséből logikusan adódik.

Számpéldák mátrix műveletekkel a MATLAB felhasználásával:

```
>>   v1=[1,2,3+ 0.5i]
          Eredmény:   v1=
                          1.0000    2.0000    3.0000+0.5000i
```

```
>>  v2=[3- 0.5i,4,5]
            Eredmény:   v2=
                          3.0000-0.5000i    4.0000    5.0000
>>  v1+v2
            Eredmény:   ans=
                          4.0000-0.5000i    6.0000    8.0000+0.5000i
>>  v3=v1-v2
            Eredmény:   v3=
                         -2.0000+0.5000i   -2.0000   -2.0000+0.5000i
```

Mátrixok és vektorok szorzására és osztására számos megkötés van. Például: csak azonos hosszúságú sorvektort és oszlopvektort lehet összeszorozni és nem mindegy a vektorok sorrendje.

Példa: A v1 változó konjugált transzponálása:

```
>>  v1=v1'
        Eredmény:   v1=
                      1.0000
                      2.0000
                      3.0000-0.5000i
```

Oszlopvektor szorozva sorvektorral mátrixot eredményez:

```
>>  v1*v2
        Eredmény:   ans=
                      3.0000-0.5000i    4.0000    5.0000
                      6.0000-1.0000i    8.0000  10.0000
                      8.7500-3.0000i  12.0000-2.0000i  15.0000-2.5000i
```

Sorvektor szorozva oszlopvektorral skalárt eredményez:

```
>>  v2*v1
        Eredmény:   ans=
                      26.0000-3.0000i
```

A mátrixok elemek szerinti szorzása és hatványozása értelemszerű. Az elemek szerinti osztásnál van különbség a bal és jobboldali osztás között. Elemek szerint sorvektort sorvektorral és oszlopvektort oszlopvektorral lehet szorozni, balról, jobbról osztani, és hatványozni.

A polinom alakban felírható függvényeket speciális vektorváltozóként értelmezi a MATLAB.

Például az $f(x) = 3x^2+2x+x$ polinom függvény MATLAB alakja:

```
>>f=[3 2 1];
```

Megjegyzés: A parancssort lezáró pontos vessző szerepe, hogy nem írja ki az eredményt a parancsablakba. Fontos szintaktikai szabály, hogy nincs se vessző, se pontos vessző az együtthatók között! Ezért értelmezi polinom függvénynek.

A polinom függvény vizsgálatához a MATLAB számos parancsot tartalmaz. Ezek közül az Automatika I. laboratórium méréseiben a **roots** és az **abs** parancsokat használjuk.

A függvény gyökeit számítja ki a **roots** parancs.

```
>>  roots(f)
                                   Eredmény:  ans=
                                                 -0.3333 + 0.4714i
    (Megadható roots([3 2 1]) alakban is!)       -0.3333 -  0.4714i
```

A komplex gyökök abszolút értékét számítja ki az **abs** parancs.

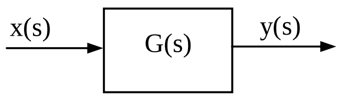
```
>>  abs(-0.3333 + 0.4714i)
                                   Eredmény:  ans=
                                                 0.5773
```

4.2. Bevezetés a Control System Toolbox alkalmazásába

A „Control System Toolbox” a lineáris, időben állandó együtthatós szabályozási modell (LTI: Linear Time Invariant System) vizsgálatához nyújt hathatós segítséget.

Az LTI modell jellemezhető az időtartományban differenciál egyenlettel, az operátoros körfrekvencia tartományban az átviteli függvénnyel, és az idő és frekvenciatartományban egyaránt alkalmazható az állapotteres leírási mód.

LTI modellel jellemezhető jelátviteli tag (4.2 ábra) operátoros átviteli függvénye megadható polinom tört (4.1.a kifejezés) és gyöktényezős (4.1.b kifejezés zérus, pólus, erősítés) alakban is.



4.2. ábra Jelátviteli tag

$$G(s) = \frac{b_m s^m + b_{m-1} s^{m-1} + \cdots b_1 s + b_0}{a_n s^m + a_{n-1} s^{m-1} + \cdots a_1 s + a_0} \tag{4.6.a}$$

$$G(s) = \frac{K \cdot (s - z_m) \cdot (s - z_{m-1}) \ \ (s - z_0)}{(s - p_n) \cdot (s - p_{n-1}) \ \ (s - p_0)} \tag{4.6.b}$$

Példa: Definiáljuk a $G(s)=\dfrac{2.5}{s^3+5s^2+8s+4}$ polinomtört formátumú átviteli függvényt, mint „G1” MATLAB változót és a G(s) gyöktényezős alakját $G(s)=\dfrac{2.5}{(s+2)^2(s+1)}$, mint „G2” MATLAB változót

A szükséges MATLAB parancsok a **tf** („transfer function”) és a **zpk** (zeros poles gain).

A **tf** parancs szintaktikája tf(num,den), ahol num a számláló, den a nevező polinomja.

```
>>  G1=tf([2.5],[1 5 8 4])
```

Eredmény:

```
Transfer function
          2.5
-------------------------          (4.7.a)
s^3 + 5 s^2 + 8 s + 4
```

A **zpk** parancs szintaktikája: zpk([zeros],[poles],gain), ahol a zeros a számláló, a poles a nevező gyökei, és a gain az átviteli függvény erősítése..

```
>>  G2=zpk([],[-2 -2 -1],2.5)
```

Eredmény:

```
Zero/pole/gain:
      2.5
----------------          (4.7.b)
(s+2)^2 (s+1)
```

A G(s) gyöktényezős alakja előállítható polinomtört alak konvertálásával: G2=zpk(G1), illetve a polinomtört alak előállítható a gyöktényezős alak konvertálásával: G1=tf(G2).

Ha nem a gyöktényező alak, hanem csak a gyökök és az erősítés érték a megismerni kívánt paraméter, akkor

```
>>  [z,p,k]=tf2zp([2.5],[1 5 8 4])
```

Eredmény:

```
z =
   Empty matrix: 0-by-1
p =
  -2.0000
  -2.0000          (4.8)
  -1.0000
k =
   2.5000
```

Megjegyzés: A z, p, k értékek kiolvashatók a gyöktényezős alakból is.

Megjegyzés: A 4.7.a., és 4.8. kifejezésekben a skalárváltozót (a számláló ilyen) nem szükséges szögletes zárójelbe tenni.

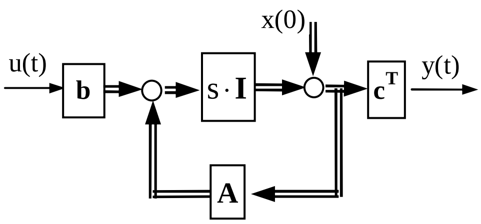
Ha csak a pólusok kellenek, akkor használható a **roots** parancs:

\>> roots([1 5 8 4])

Eredmény:
```
-2.0000
-2.0000
-1.0000
```

A G1, illetve a vele azonos G2 változóval reprezentált G(s) átviteli függvény operátoros állapotteres modellje a 4.3. ábrán látható.

Akár a polinom törtes, akár a gyöktényezős alak átkonvertálható állapotteres („state space") alakká.



4.3. ábra Az állapotteres modell

\>> G3=ss(G1), (vagy G3=ss(G2))

Eredmény:
```
a =
       x1  x2  x3
   x1  -2   1   0
   x2   0  -2   1
   x3   0   0  -1
b =
          u1
   x1      0
   x2      0
   x3  1.581
c =
          x1     x2     x3
   y1  1.581      0      0
d =
       u1
   y1   0
 Continuous-time
model.
```

Ha az a, b, c, d mátrixok (vektorok) ismertek, akkor a G3 változó ezekből is generálható:

\>> G3=ss(a,b,c,d)

Ha ismert a „z" transzformált G(z) átviteli függvény polinomtört, gyöktényezős, vagy állapotteres alakja, akkor a megfelelő LTI modellt generáló parancs (tf, zpk, ss) bármelyikével létrehozható úgy, hogy a T_0 mintavételi időt hozzáfűzzük a modellt generáló parancs argumentumához. Például:

\>> Gz=tf(num,den,T_0).

Ismert átviteli függvényből a **c2d** ("continuous to discrete") paranccsal konvertálható át mintavételezett „z" transzformátoros alakká az operátoros átviteli függvény. A **c2d** parancs szintaktikája:

\>> c2d(Gx, T_0) (4.9)

ahol Gx tetszőleges LTI modellel megadott átviteli függvény és a T_0 a mintavételi idő.

A 4.9. kifejezés 0-ad rendű („zoh: zero order holder") tartószervet alkalmazva készül. Ha 1-es rendű („foh: first order holder") tartószervvel akarjuk, akkor a konvertáló parancsot következőképp kell kiegészíteni: G4=c2d(G1,0.2,'foh')

```
>>  G4=c2d(G1,0.2)
```

Eredmény: Transfer function:

```
0.002603 z^2 + 0.008132 z + 0.001579
----------------------------------------------
  z^3 - 2.159 z^2 + 1.547 z - 0.3679

Sampling time: 0.2
```

Az LTI modell vizsgálata

Ha adott egy jelátvivő tagot reprezentáló MATLAB változó, akkor az **impulse** paranccsal előállítható a jelátvivő tag súlyfüggvénye; a **step** paranccsal előállítható a jelátvivő tag átmeneti függvénye; a **bode** paranccsal kirajzolható a jelátvivő tag Bode diagramja; a **nichols** paranccsal kirajzolható a jelátvivő tag Nichols diagramja; a **nyquist** paranccsal kirajzolható a jelátvivő tag Nyquist diagramja.

Megjegyzés: A Bode, a Nichols, és a Nyquist diagram a koordináta tengelyek megválasztásában tér el egymástól.

- ♦ A Bode diagram felbontja a komplex számokat amplitúdó átvitelre és fázistolásra, majd ezeket a körfrekvencia függvényében (logaritmikus léptékben) ábrázolja.
- ♦ A Nichols diagram a fázistolás függvényében ábrázolja az amplitúdó átvitelt. Fontos a rácsozatot („grid") bekapcsolni, mert a koordináta rendszer szintvonalai nem derékszögűek!
- ♦ A Nyquist diagram a komplex számérték sorozatot a komplex számsíkon ábrázolja és a görbe pontjaihoz rendeli a körfrekvencia értékeket.

A G1, G2, G3 változók teljesen azonosak, hiszen ugyanazon jelátvivő tag folytonos modelljét szolgáltatják. A G4 változó tartalmazza a mintavételezésből származó torzítást (4.4. ábra).

A **step** parancs szintaktikája:

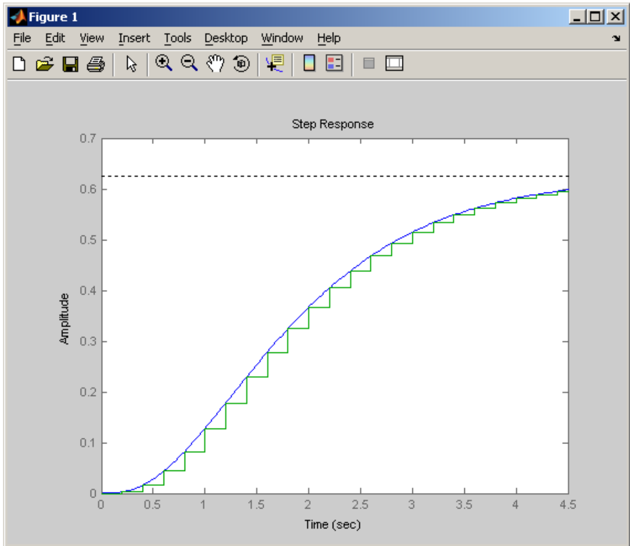
a) >>step(LTIsys1, LTIsys2,⋯,LTIsysN)

Az LTIsys: lineáris, idő invariáns, egy be- és egy kimenetű modell. A parancs hatására feljön a **figure** GUI. Az N darab függvénygörbe közös ábrában jelenik meg (4.4. ábra). Természetesen egy darab LTIsys modell is elegendő a **step** parancs argumentumában.

Megjegyzés: A 4.4. ábrán jól látszik, hogy bár a G1 átviteli függvény is mintavételezett értékek sorozata, csak a mintavételezés olyan sűrű, hogy *a vizsgálatok szempontjából folytonosnak tekinthető! A G1 átviteli függvény kvázi folytonos.*

>> step(G1,G4)

Eredmény:



4.4 ábra. A folytonos és a mintavételes átmeneti függvények

Az a) pontban megadott szintaktikájú **step** parancs esetén belső algoritmus döntötte el, hogy az LTI modell kimeneti értékei mekkora időtartamra lesz meghatározva. Ez nem mindig előnyös. A vizsgálat időtartamát és a mintavételi időt a feladatnak megfelelően célszerű beállítani.

b) >>step(LTIsys1, LTIsys2,⋅⋅⋅,LTIsysN,t)

A „t" időváltozónak már léteznie kell. Legkésőbb a parancs beírása előtt definiálni kell: $t=T_{min}:T_0:T_{max}$; szintaktikával, ahol a T_{min} kezdőérték, a T_0 mintavételi idő, a T_{max} végérték.

*Megjegyzés: Ilyenkor nem lehet folytonos és mintavételezett LTI modell együtt a **step** parancs argumentumában, mert kétféle időléptéket nem tud kezelni. Több mintavételezett LTI modell is csak akkor ábrázolható együtt, ha azonos a mintavételi idejük.*

Ha a 4.4. ábrán levő függvények időléptékére és értékeire van szükségünk, akkor ezt csak külön-külön lehet lekérdezni G1 és G4 változó értékeire, mert egyszerre csak egy LTI modell argumentumai kérdezhetők le.

c) >>[y1,t]=step(G1);

Megjegyzés: A pontos vessző a sor végén azért kell, hogy a MATLAB ne írja ki a parancsablakba a változó több száz értékét.

Az **impulse** parancs szintaktikája teljes mértékben megegyezik a **step** parancséval.

A **bode**, **nichols**, és a **nyquist** parancsok is kiadhatók az a) pont szerinti szintaktikával.

A b) pont szerinti szintaktika úgy módosul, hogy kapcsos zárójelek közt az alsó (w_{min}) és a felső (w_{max}) körfrekvencia értéket kell megadni:

$$>>bode(G1,\{w_{min},w_{max}\}) \quad (4.10)$$

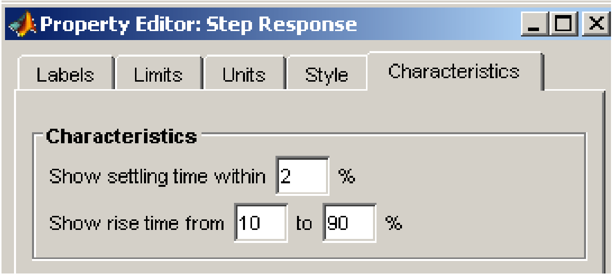
A c) pont szerinti argumentumok kinyerése az alábbi:

>>[magnitude,phase,w]=bode(G1);

A figure GUI szolgáltatásai

A figure GUI számos, az ábrázolt grafikus objektumok kiértékelését segítő szolgáltatást tartalmaz. Ezek mindegyike elérhető a legördülő menüsorból (4.4. ábra). A legfontosabbak elérhetők a menüsor alatti ikonok és az egér kurzorok segítségével is.

Ha az ikonok közül egyik sincs kijelölve, akkor az ábra fehér területére kettőt klikkelve a bal egérgombbal feljön a „Property Editor" párbeszéd ablak. A 4.5. ábrán a párbeszédablak felső fele látszik. Az ábra alján a szokásos Windows gombok vannak elhelyezve.

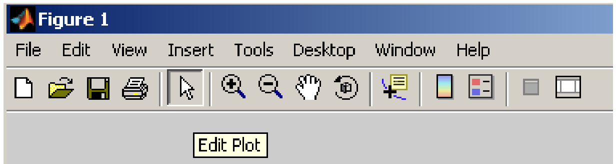


4.5. ábra. Property Editor

- A „Labels" fül alatt átírható az ábra címe és a koordináták elnevezései.
- A „Limits" fül alatt átírhatók a koordináta tengelyek határértékei.
- A „Units" fül alatt átkonvertálhatók a koordináta tengelyek mértékegységei. (Az idő és a mértékegység nélküli tengely nem!)
- A „Style" fül alatt megadhatók a szövegek stílusai. Itt kapcsolható be a rácsozat („Show grid").
- A „Characteristics" fül látható a 4.5. ábrán. A felkínált beállítási lehetőségek az ábrázolt függvénytől vagy diagramtól függ. Az átmeneti függvény esetén a tolerancia sáv mértéke és a felfutási idő határai állíthatók. A 4.5. ábrán nem lettek átírva a default értékek.

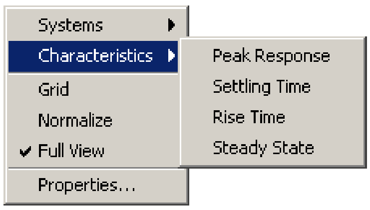
Az Automatika I. laboratóriumi mérések elvégzéséhez a 4.6. ábrán látható „Edit Plot" és a 4.8. ábrán látható „Data Cursor" használatának ismerete szükséges.

Amikor az „Edit Plot” ikon van kijelölve, akkor az egérgombbal legördülő menüből elérhet számos menüsor szolgáltatás, továbbá rajzolhatunk az ábrába az „Insert” menüből választott objektumot, illetve írhatunk a „Text Arrow” és „Textbox” objektumokba.



4.6. ábra. A figure GUI ikonjai

Amikor „Edit Plot” van kijelölve → akkor kettős klikk a bal egérgombbal az ábra fehér területére → feljön az „Property Editor - Axes” párbeszéd ablak, amelyben elvégezhetők az ábrázolt függvény vagy diagram tengelyeihez tartozó beállítások.



4.7. ábra Az átmeneti függvényhez felkínált szolgáltatások

Amikor az „Edit Plot”van kijelölve → akkor jobb egérgombbal klikk az ábra fehér területére → feljön a 4.7. ábrán látható párbeszédablak. A párbeszédablakban felkínált szolgáltatások függnek az ábrázolt függvénytől vagy diagramtól. A 4.7. ábrán látható szolgáltatások az átmeneti függvényhez tartoznak.

A 4.7. ábrán látható, hogy innen is meghívható a „Property Editor”, beállítható a rácsozat, stb.

Fontos: A „Characteristics” fül alatt (4.7. ábra) a „Peak Response” a csúcsértéket, a „Settling Time” a beállási időt, ami zárt szabályozási kör esetén azonos a szabályozási idővel, a „Rise Time” a felfutási időt, a „Steady State” végértéket jelöli meg az ábrázolt függvényen vagy diagramon. Az egér kurzort a megjelölt pontra állítva a feljövő információs ablakban az ábrázolt függvény vagy diagram megjelölt ponthoz tartozó paraméterei láthatók.

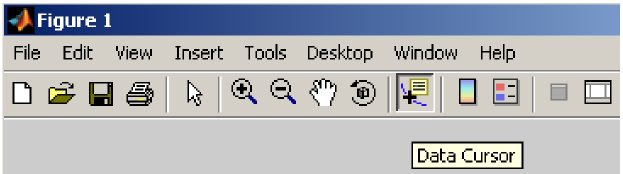
Megjegyzés: A Bode, Nichols, és a Nyquist diagram ábrázolásakor a „Peak Response” mellet a „Minimum Stability Margins” és az „All Stability Margins” fül jelenik meg. Az utóbbiakat választva a fázistartalék(ok) és az erősítés tartalék(ok) helye lesz megjelölve a diagramon.

Fontos: Ha az „Edit Plot”van kijelölve és a bal egérgombbal klikkel az ábra szürke területére, akkor a feljövő párbeszédablakból elérheti a figure GUI forráskódját. Ezt ne írja át!

Az Automatika I. laboratóriumi mérésekben használt másik ikon a 4.8. ábrán látható „Data Cursor”.

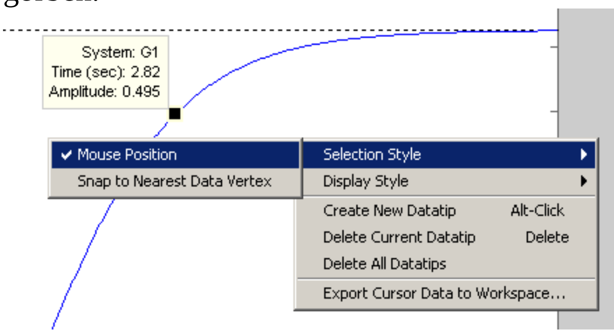
Amikor a „Data Cursor” a kijelölt ikon, akkor az egér kurzort a függvény vagy diagram görbéjére helyezve, a bal egérgomb klikkel felhozható az adott pont paramétereit tartalmazó információs ablak.

Az egér kurzort a kijelölt pontra állítva és a bal egérgombot lenyomva az információs ablak mozgatható a görbén. Az egér kurzort az információs ablakra állítva és a bal egérgombot lenyomva az információs ablak mozgatható a kijelölt pont körül.

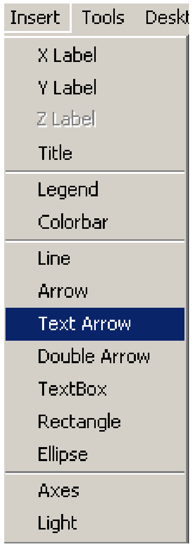


4.8. ábra. A figure GUI ikonjai

A jobb egérgombbal hozható fel a 4.9. ábrán látható párbeszédablak. Ebben lehet az aktuális információs ablakot rögzíteni „Create New Datatip”, illetve törölni. A 4.9. ábrán látható „Mouse Position” választás lehetővé teszi a változó diszkrét értékei közötti mozgást a görbén.



4.9. ábra. A „Data Cursor” szolgáltatásai



4.10. ábra. Az „Insert” menü

A bal egérgombnak sem az ábra fehér, sem a szürke területén nincs funkciója, amikor a „Data Cursor” a kijelölt ikon.

Megjegyzés: Az ábrázolt görbe a változó diszkrét értékeiből és a diszkrét értékeket összekötő interpolációs egyenesekből áll. Alaphelyzetben csak diszkrét értékekre lép rá az egér kurzor. A „Mouse Position” választással az interpolációs egyenesen is végighalad.

Amennyiben rajzzal szerkeszteni vagy szöveget írni akarunk az ábrába, a menüsorból az „Insert” fület (4.6. és 4.8. ábrák) kel kinyitni.

Az Automatika I. laboratóriumi mérésekben leggyakrabban egy adott görbepont mellé kell szöveges megjegyzést fűzni.

A 4.10. ábrán a „Text Arrow" fület választva, az ábra fehér területén a kezdőpontot bal egérgombbal kijelölve és a bal egérgombot lenyomva a kívánt ponthoz húzva a kurzort, a nyíl megjelenik.

A nyíl kezdőpontjánál van az írható szövegdoboz. Ha a már beírt szöveget módosítani szükséges, akkor bal egérgombbal kijelölve a nyilat, a jobb egérgombbal feljövő párbeszéd ablakból a „Edit Text" választással a szöveg ismét szerkeszthetővé válik.

Ha a görbéhez egy egyenest akar illeszteni, akkor a „Line" fület (4.10. ábra) kell választania. Az egyenes vonal lehelyezése ugyanúgy történik, mint a szöveges nyílé. Az egyenes vonal mozgatásához a bal egérgombbal kell kijelölni az egyenest, majd az egér kurzort az végpontjára helyezve, a bal egérgombbal mozgatható a megkívánt pozícióba.

4.3. Bevezetés a SIMULINK alkalmazásába

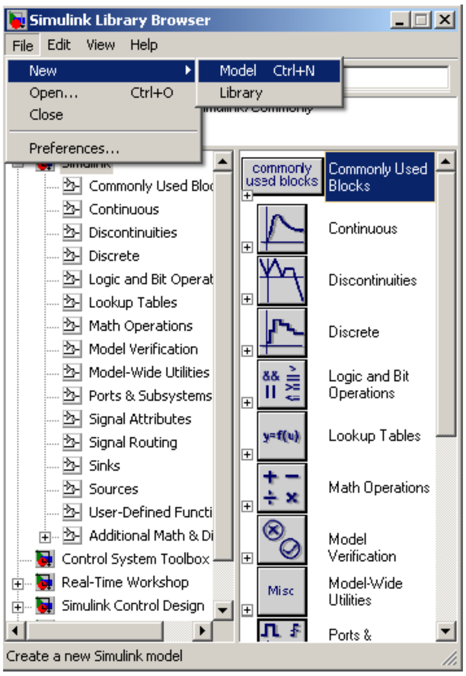
A SIMULINK a szabályozási körök elemzésére szolgáló összetett grafikus interfész, ami a MATLAB programból (4.11. ábra) hívható meg.

Megjegyzés: Ha nem találja az ikont, akkor a „Desktop" menüben a „Toolbar" fület kell kijelölni. Ekkor a SIMULINK ikonja megjelenik. A „Desktop" menüben a „Desktop Layout" → „Default" választással a MATLAB ablakok eredeti elrendezése áll helyre.

4.11. ábra.
SIMULINK ikon

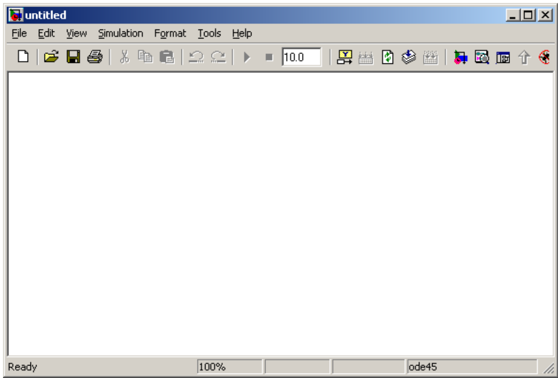
A 4.11. ábrán látható ikonnal a „Simulink Library Browser" nyitható meg (4.12. ábra). A 4.12. ábra baloldalán a grafikus objektumok könyvtárszerkezete, a jobboldalán a kijelölt könyvtárhoz tartozó grafikus objektumok vannak. Egy új szerkesztő felület (4.13. ábra) a „Simulink Library Browser" menüjéből a „File"→"New"→"Model" útvonalon (4.12. ábra) vagy a menü alatti windows-os ikonnal generálható.

A grafikus objektumokat a szerkesztő felületre az ismert „fogd és vidd" technikával lehet lehelyezni. A szerkesztő felületre vitt objektum belsejébe helyezett kurzorral és a bal egérgomb kettős kattintásával megjelenik a grafikus objektum párbeszéd ablaka. A grafikus objektum párbeszéd ablakában az „Apply" gombot alkalmazva, a párbeszédablak nem csukódik be, és a szerkesztő felületre letett grafikus objektumban megfelelő formátumban megjelenik a bevitt adat. Ha nem fér el a bevitt adat a grafikus objektum felületén, akkor a bal egérgombbal nyújtható a grafikus objektum. Fontos szerkesztési segítség, hogy több grafikus objektumot csoportba foglalhat úgy, hogy - a bal egérgombot lenyomva - körbe keríti azokat. A csoportba foglalt grafikus objektumokat együtt lehet mozgatni.



4.12. ábra. A „Simulink Library Browser"

A SIMULINK szerkesztő felületén (4.13. ábra) lineáris és nem lineáris jelátviteli tagok tetszőleges kombinációja valósítható meg. Az időtartományban történő szimulációval vizsgálható az eredő, vagy bármely közbenső jel időbeli lefolyása.



4.13. ábra. A SIMULINK szerkesztő felülete

Megjegyzés: A grafikus objektumok paraméterei definiálhatók körfrekvencia vagy állapotteres leírási móddal, azonban a szimuláció a differencia egyenletek alkalmazásával történik.

A kijelölt szimulált jelek, az általunk beállított struktúrával, MATLAB változóként vissza exportálható[1] a „Workspace" könyvtár területére (4.1. ábra). A „Workspace" területen elhelyezett időtartománybeli változók a **plot** paranccsal kirajzolhatók. A **plot** parancs szintaktikája:

plot(x,y) (4.11)

ahol „x" az X tengely értékei és „y" az Y tengely értékei.

Megjegyzés: Jelátvivő tag időtartománybeli be- és kimeneti jelsorozatából előállítható a jelátvivő tag körfrekvencia tartománybeli átviteli függvénye. Ezt az eljárást nevezik identifikálásnak.[2]

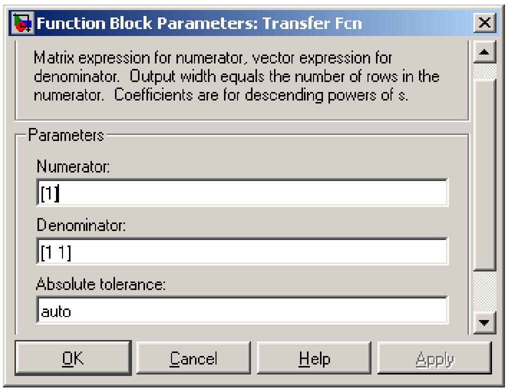
[1] A mintavételi időpontok sorozata és az időpontokhoz tartozó szimulált jel vagy jelek.

[2] Az Automatika I. laboratóriumban a „Simulation" Toolbox az ident GUI-val – amivel elvégezhető az identifikálás - csak a MATLAB 2008 verzióhoz lett megvéve.

Az egyhurkos szabályozási kör vizsgálatához leggyakrabban alkalmazott grafikus objektumok, az objektumok helye a könyvtárban és paraméterezésük a következők:

Transfer Fcn: ***(átviteli függvény)***

A Simulink → Continuous útvonalon érhető el.



4.14. ábra Az átviteli függvény paraméterező ablaka

A **tf** parancsnál megismert módon a számláló polinomját („Numerator”) és a nevező polinomját („Denominator”) kell megadni.

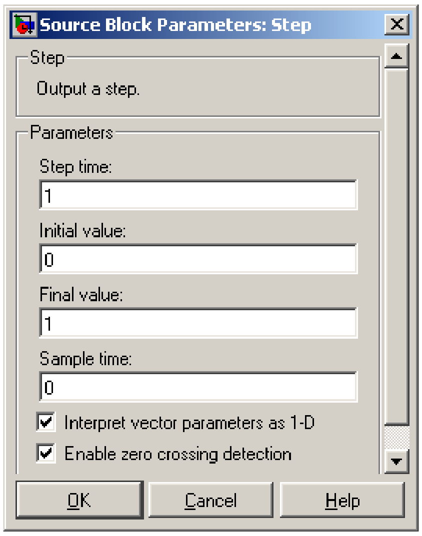
Az „OK” vagy „Apply” gombbal fogadtatható el beírt érték.

Az „Apply” gombot alkalmazva a párbeszédablakban a szerkesztő felületre letett téglalapban megjelenik az átviteli függvény.

Megjegyzés: Az átviteli függvény együtthatója valós szám és a „Workspace” ablakban létező változó valós értékű lehet.

Step: ***(egységugrás)***

A Simulink → Sources útvonalon érhető el.



4.15. ábra Az egységugrás paraméterező ablaka

A „Step time” mezőben megadott érték azt jelenti, hogy a szimuláció kezdő időpontjához képest mikor lép be az egységugrás. A default érték az 1 másodperc. A jel kiértékelésekor fontos ismerni a gerjesztés kezdő pontját.

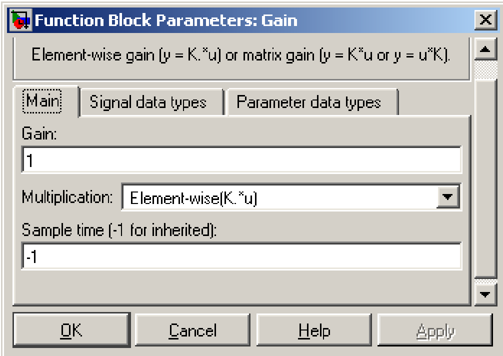
Az „Initial value” mezőben a gerjesztő jel kezdő értékét, a „Final value” mezőben a gerjesztő jel végértékét lehet beállítani.

A „Sample Time” mezőt csak mintavételezett modell esetén kell kitölteni!

Ha az „Interpret…” fül ki van jelölve, akkor a kimenet vektorváltozó, különben folyamatosan konstans értéket szolgáltat.

Gain: ***(erősítés)***

A Simulink → Commonly Used Blocks vagy
a Simulink → Math operations útvonalon érhető el.



4.16. ábra Az erősítés paraméterező ablaka

A „Gain” mezőben adható meg a kívánt erősítés érték.

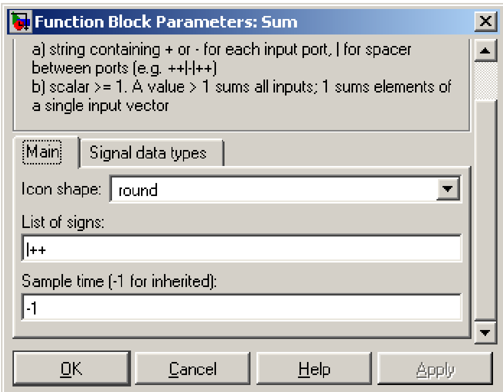
A „Multiplication” mező legördülő menüjében lehet balról vagy jobbról történő mátrixszorzásra váltani.

Fontos: A „Sample Time” mezőt csak mintavételezett modell esetén kell kitölteni!

A „Signal data types” és a „Parameter data types” fülek alatt a kimeneti jel, és a belső változók jeltípusát és a kerekítési szabályt lehet beállítani.

Sum, Subtract: ***(összegzés, kivonás)***

A Simulink → Commonly Used Blocks vagy
a Simulink → Math operations útvonalon érhető el.



4.17. ábra Az összegző paraméterező ablaka

A „Icon shape” mezőben választható ki, hogy a grafikus objektum kerek („round”) vagy négyszögletes („rectangle”) formájú legyen.

A „List of sign” mezőben elhelyezett + és – jelek száma adja meg a bemenő jelek számát és előjelét.

A „Sample Time” mezőt csak mintavételezett modell esetén kell kitölteni! A „Signal data types” fül alatt a kimeneti jel típusát és a kerekítési szabályt lehet beállítani; továbbá, hogy a bemenetek azonos típusúak legyenek-e.

Megjegyzés: A „Sum” és a „Subtract” grafikus objektumok egymásba átalakíthatók, vagy több bemenetűvé módosíthatók.

Egy LTI modell ezekből a grafikus objektumokból már összerakható. Az Automatika I. laboratórium méréseihez szükségesek lehetnek még a következők:

Manual switch: (kézi kapcsoló)

A Simulink → Signal routing útvonalon érhető el.

Nincs paraméterező ablaka. Bal egérgombbal a kapcsolón kétszer klikkelve vált át a morzeérintkező.

Bus Creator (Mux): jel multiplexer

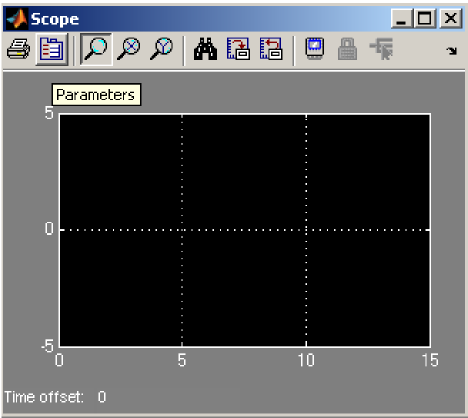
A Simulink → Signal Routing vagy
a Simulink → Commonly Used Blocks útvonalon érhető el.

A feladata több jel egybefoglalása. A paraméterező ablakában csak a bemenetek számát („Number of inputs”) kell megváltoztatni, ha kettőnél több bemenetre van szükség. A jelek felülről lefelé vannak lekérdezve.

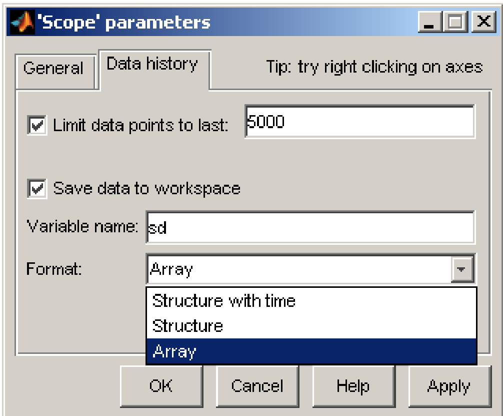
A szimulált jelek megjelenítéséhez vagy a „Workspace” területre küldéséhez szükségesek a további grafikus objektumok:

Scope: (oszcilloszkóp)

A Simulink → Commonly Used Blocks vagy
a Simulink → Sinks útvonalon érhető el.



4.18. ábra Az oszcilloszkóp



4.19. ábra A paraméterező ablak

Az oszcilloszkóp grafikus objektumra kétszer klikkelve először nem a paraméterező ablak, hanem a 4.18 ábrán látható oszcilloszkóp jelenik meg. A „Parameters” ikonra klikkelve jön fel a 4.19. ábrán látható paraméterező ablak. **Fontos**: A „General” fül alatt nincs számunkra lényeges beállítandó paraméter. A „Data history” fül alatt kijelölve a „Save data to workspace” funkciót, adhatunk nevet a változónknak és meghatározhatjuk a változó formátumát. A 4.19. ábrán a név a default „ScopeData” helyett sd és a formátum „Array”.

Az így beállított oszcilloszkóp „sd” néven lement a „Workspace” területre egy mátrixváltozót, amelynek első oszlopába a szimulációs időpontok, második oszlopában a „BusCreator” felső bemenetére vezetett jel, harmadik oszlopában a „BusCreator” alsó bemenetére vezetett jel értékei vannak. Ugyancsak lement „tout” néven egy oszlopvektort a szimulációs időpontok értékeivel.

Fontos: Sajnos itt hibázik a MATLAB. Bár, mint a 4.19. ábrán is látható nem cél, mégis belefűzi az idő paramétert a változóba!

Három oszlopból álló mátrix első oszlopa a 4.4. kifejezéssel az alábbi módon törölhető:

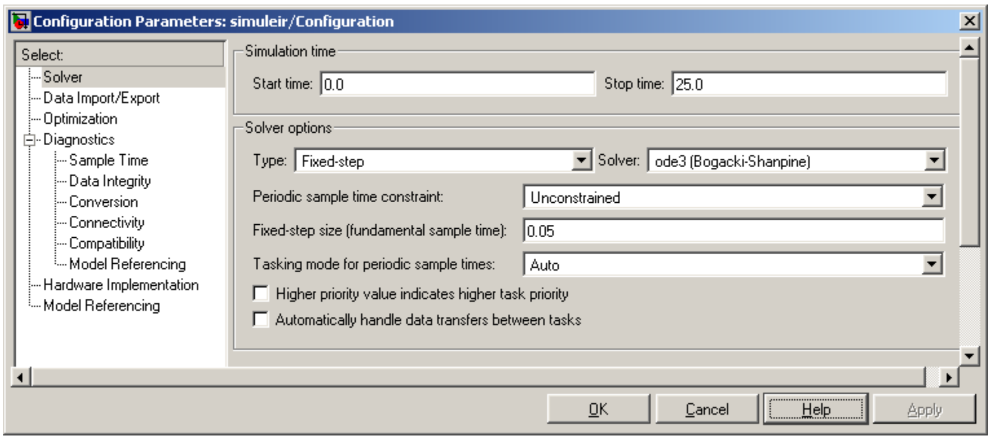
adat1=sd(:,2:3)

Ezután alkalmazható (4.11. kifejezés) a **plot** parancs:

plot(tout,adat1)

A 4.17. ábrán a távcső ikonra klikkelve a koordinátatengelyek automatikusan az ábrázolt jelhez illeszkednek. Mód van a koordinátatengely paramétereket egyedileg beállítani az oszcilloszkóp képernyőjén jobb egérgomb hatására feljövő ablakból az „Axes Properties” választásával.

Az „sd” változó sorainak száma a szimulációs paraméterek beállításától függ. A 4.13. ábrán látható modellszerkesztő ablak „Simulation” menüpontja alatt a „Configuration Parameters..” fület választva feljön egy párbeszédablak (4.19. ábra), amit az ajánlott beállítások betartásával célszerű kitölteni.



4.20. ábra. A szimulációs paraméterek beállítása.

A 4.20. ábra párbeszédablakának ajánlott beállítása:

- „Start time”: A szimulációs adatfelvétel kezdete
- „Stop time”: A szimulációs adatfelvétel vége
- „Type”: A „Fixed step” helyett a másik lehetőség a „Variable step”. Ez utóbbi a jelváltozás függvényében növeli vagy csökkenti a mintavételek

között idõt. Miután ez általában csökkenti a mintavételek számát, nagyon sok (több ezer felett) mintavételi pontból álló mintasorozat felvételekor előnyös, de számos szempontból, pl.: a **c2d** konverzió szempontjából, kifejezetten hátrányos.

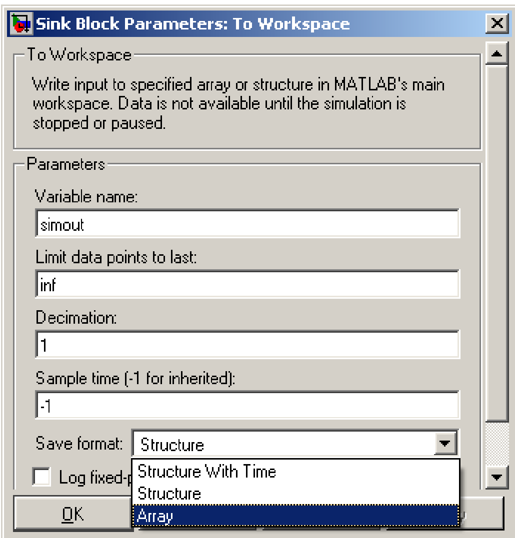
- „Solver”: A szimulációs számítások lineáris, állandó együtthatós differencia egyenleteken („ordinary differential equations: ode”) alapul. A differenciahányados számításának pontosságát növeli, ha kettőnél több pontból van meghatározva. Az „ode3” kelően sűrű mintavétel esetén elegendően pontos. Ha megoldási hibát jelez a MATLAB, akkor próbálja meg az „ode14x (extrapolation)” választással.
- „Periodic sample time constraint”: A „Unconstraint” választása esetén az általunk megadott mintavételi időt veszi figyelembe.
- „Fixed step size fundamental sample time”: Itt kell megadni a mintavételi időt. Ökölszabály, hogy a mintavételi idő a jelátvivő tag beállási idejének kétszázad részénél kisebb legyen! Kerekítési problémák miatt az ezred részénél ne legyen kevesebb.

  Fontos, hogy egész szám legyen a szimulációs adatfelvétel hossza osztva a mintavételezési idővel!

  A többi lehetőséget hagyjuk default értéken!

To Workspace: (A változók könyvtárába)

A Simulink → Sinks útvonalon érhető el.



4.21. ábra A változó képző paraméterező ablaka

Ez a grafikus objektum lehetővé teszi a változók egyedi lementését úgy, hogy a **plot** parancs számára azonnal feldolgozható legyen.

A „Variable name” a lementett változó szimbolikus nevét kell ide beírni.

A „Limit data points to last” maradjon a végtelen (infinitive).

A „Decimation” a tizedespont utáni számjegyek száma.

A „Sample Time” mezőt, mint tudjuk csak mintavételezett modell esetén kell kitölteni!

Fontos: A „Save format”: Az „Array” oszlopvektor formátumot válassza!

A megadott szimbolikus néven a változó, egy „tout” nevű változóval együtt a „Workspace” könyvtárban jelenik meg.

Fontos: A szimuláció végrehajtását követően ellenőrizze, hogy a „tout” és a változó mérete azonos!

4.4. SIMULINK modell szerkesztése

A SIMULINK modell szerkesztését legegyszerűbben egy példán keresztül mutatható be.

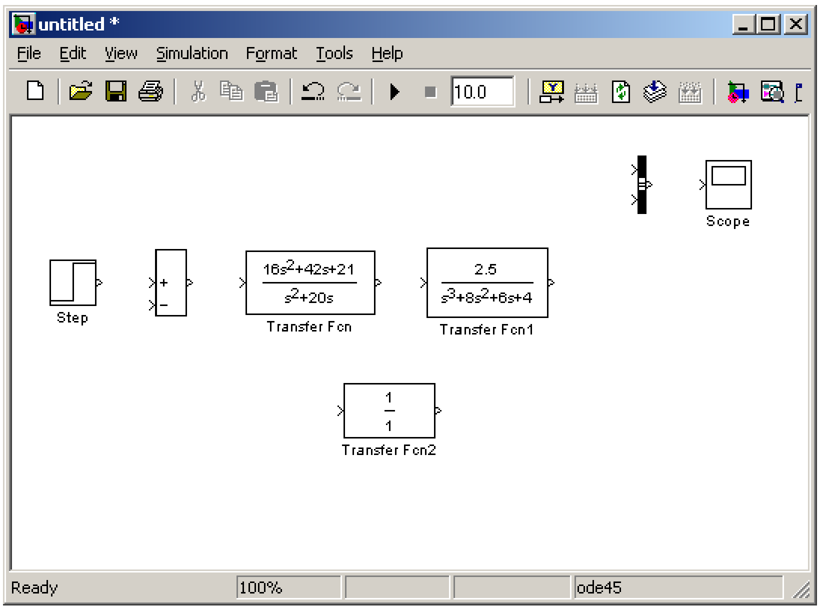
Legyen a szakasz az ismert (4.7.a. kifejezés) G1 átviteli függvény:

$$G1 = \frac{2.5}{s^3 + 5s^2 + 8s + 4} \tag{4.12}$$

Most még magyarázat nélkül a Gpidt kompenzáló tag:

$$G_{pidt} = \frac{16s^2 + 42s + 21}{s^2 + 20s} \tag{4.13}$$

A 4.12. ábrán látható módon megnyitva a 4.13. ábrán látható szerkesztő felületet, a „fogd és vidd” technikával helyezzen el három „Transfer Fcn”, valamint egy-egy „Sum” vagy „Subtract”, „Step”, „BusCreator”, és „Scope” grafikus objektumot az alábbi elrendezésben, majd paraméterezze fel azokat.

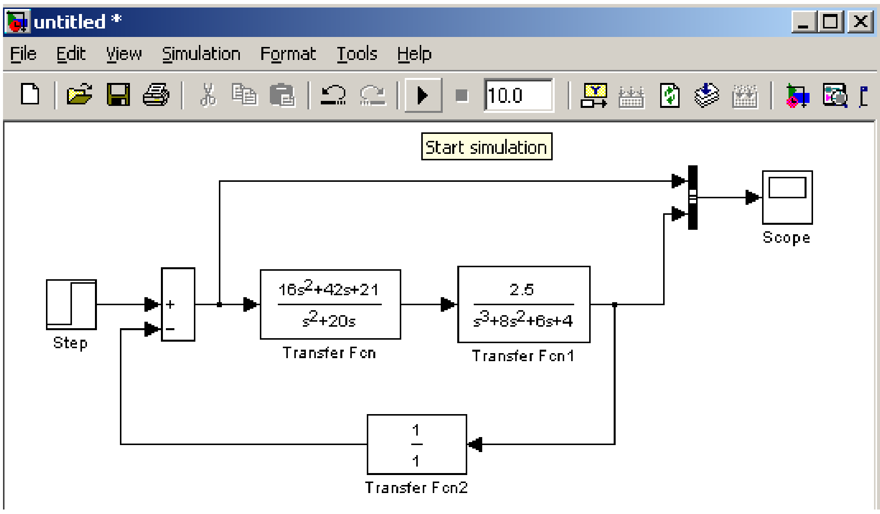


4.22. ábra. Az objektumok felhelyezése a szerkesztő felületre

Megjegyzés: zA egyszeres erősítésű átviteli függvény („Transfer Fcn2”) teljesen felesleges a vizsgálat szempontjából. Arra szolgál, hogy bemutassuk, hogy mi a teendő, ha van a visszacsatoló ágban is jelátvivő tag.

A huzalozáshoz a „Transfer Fcn2” rossz irányba áll. Kijelölve a grafikus objektumot, a jobb egérgombra megjelenő ablakból a „Format” → „Flip Block” választással az vízszintesen tükröződik. Ezután huzalozza össze a grafikus objektumokat. A huzalozás szabályai:

- Mindig a kimenetet kötjük bemenethez!
- A bal egérgombot lenyomva, a grafikus objektum kimenetétől a kívánt bemenethez húzza az egér kurzort. Ha nem a gépre hagyatkozik, akkor derékszögű irányváltást úgy generálhat, hogy egy pillanatra megáll az egér kurzorral.
- Egér kurzorral kijelölve a grafikus objektumot, majd lenyomva a Crtl billentyűt, bal egérgomb klikk a fogadó grafikus objektumra.
  Megjegyzés: Ha a fogadó grafikus objektumnak több bemenete van, akkor fentről lefelé, az első üres bemenetre köti.
- Ha meglevő vonalról akar elágazni, akkor lenyomva a Crtl billentyűt a vonal kívánt pontjától indítható a huzalozás a bal egérgombbal.



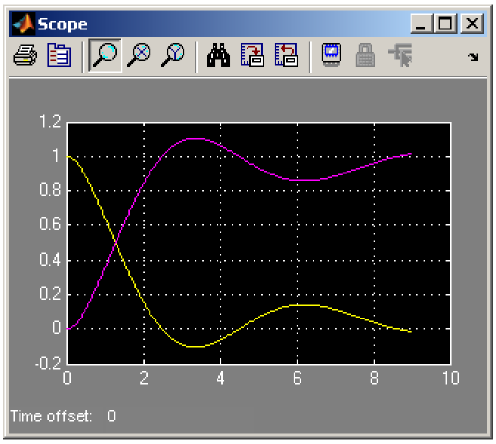
4.23. ábra. Az huzalozott modell

Mielőtt a háromszög ikonnal (4.22. ábra) indítaná a szimulációt, be kell állítani a szimulációs paramétereket, amik a „Simulation” menüpontja alatt a „Configuration Parameters..” fület választva érhetők el (4.20. ábra).

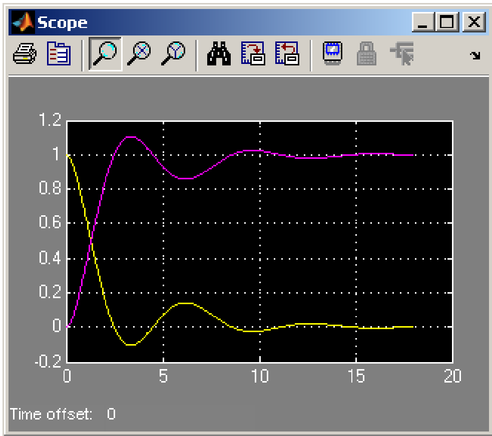
Ehhez ismerni kell, hogy mekkora legyen a szimulációs adatfelvétel időtartama.

A 4.2.b. kifejezés megadja a G1 szakasz pólusait: p1 = p2 = -2, p3 = -1. Ebből az időállandók: T1= T2 = 0.5, T3 = 1. A domináns póluspárok alapján megfogalmazott ökölszabály, hogy az időállandók összegének kb. négy – ötszöröse megfelelő szimulációs időtartam. Ha azonban három vagy több pólus van egy dekádon belül, akkor nem érvényes a domináns póluspár szabály és jobb a hat – hétszeres időtartammal elvégezni a szimulációt.

Legyen a „Stop time” 9.0 sec. és a „Fixed step size fundamental sample time” 0.03 sec.! Jelenítse meg a „Scope” grafikus objektum képernyőjét és indítsa a szimulációt. A képernyőm a 4.23.a. ábra jelenik meg.



4.24.a. ábra Az oszcilloszkóp kép



4.24.b. ábra Az oszcilloszkóp kép

A 9 másodperces szimulációs időtartammal és 0.03 másodperces mintavételi idővel végzett szimuláció eredménye 0 „Start time” időponttal együtt 301 mintavételi érték, ami a „Workspace” könyvtárban megjelenő sd változó „Value” oszlopában ellenőrizhető.

Az oszcilloszkóp ábrán azonban még nem állandósult a jel. A szabályozási idő és a végérték pontos meghatározásához növeljük meg a szimulációs időtartamot a duplájára! Ehhez elég a 4.23. ábrán látható „Start simulation” ikon utáni első szövegmezőt átírni.

Ismét a „Start simulation” ikonra klikkelve megjelenik a 4.24.b. ábra és a „Workspace” könyvtárban az sd változó „Value” oszlopa 601 elemet mutat.

A 4.4 kifejezéssel törölve az sd változó időoszlopát, alkalmazható a **plot** parancs (4.11. kifejezés) és kiértékelhető a zárt szabályozási kör hibajelének időbeli változása és az átmeneti függvénye.

5. Jelátvivő alaptagok és soros, párhuzamos, visszacsatolt eredőik

A MATLAB I. mérés célja, hogy elősegítse a jelátvivő alaptagok jellemzőinek memorizálását; megismertesse a leggyakrabban alkalmazott összetett tagokat; segítse felismerni az összetett tagok függvényeiből, diagramjaiból, hogy milyen alaptagokból és hogyan tevődnek össze. Bármely alap vagy összetett taghoz tartozó kérdésnek több változata van.

5.1. Ellenőrző kérdések

1. Válassza ki a PT1 alaptaghoz tartozó függvényeket és/vagy diagramokat! Lehet több is!

   Megjegyzés: Az ilyen típusú kérdésekben különböző alaptagok átmeneti függvényei és körfrekvencia átviteli függvényinek Bode és Nyquist diagramjai közül kell választani.
2. Válassza ki a PT2 alaptaghoz tartozó függvényeket és/vagy diagramokat! Lehet több is!
3. Válassza ki az IT0 alaptaghoz tartozó függvényeket és/vagy diagramokat! Lehet több is!
4. Válassza ki a DT0 alaptaghoz tartozó függvényeket és/vagy diagramokat! Lehet több is!
5. Válassza ki a DT1 soros összetett taghoz tartozó függvényeket és/vagy diagramokat! Lehet több is!
6. Válassza ki az IT1 soros összetett taghoz tartozó függvényeket és/vagy diagramokat! Lehet több is!
7. Válassza ki a HT1 soros összetett taghoz tartozó függvényeket és/vagy diagramokat! Lehet több is!

8. Válassza ki a HIT0 soros összetett taghoz tartozó függvényeket és/vagy diagramokat! Lehet több is!
9. Válassza ki az PI párhuzamos összetett taghoz tartozó függvényeket és/vagy diagramokat! Lehet több is!
10. Válassza ki az PDT1 párhuzamos összetett taghoz tartozó függvényeket és/vagy diagramokat! Lehet több is!
11. Válassza ki, hogy melyik Bode diagram tartozik olyan átviteli függvényhez, amelyben van integráló hatás!
12. Válassza ki, hogy melyik differenciálegyenlet tartozik a PT1 taghoz!
13. Válassza ki, hogy melyik differenciálegyenlet tartozik az IT0 taghoz!
14. Válassza ki, hogy melyik összefüggés tartozik a jelátvivő tagok soros eredőjének meghatározásához!
15. Válassza ki, hogy melyik összefüggés tartozik a jelátvivő tagok párhuzamos eredőjének meghatározásához!

5.2. A minta feladatsor

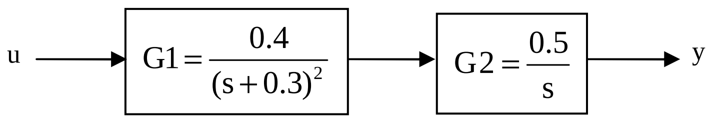
A feladat megoldása közben az ábrákat kep_1, kep_2 vagy abra_1, abra_2 vagy 1kep, stb. néven mentse! Semmiképp ne használjon ékezetes betűket és egyéb írásjeleket! Ne módosítsa a fig kiterjesztést!

A feladatot nem szükségszerű sorrendben megoldani, mert nem épülnek egymásra. A jegyzőkönyvben (az M fájlban) egyértelműen jelezze, hogy melyik feladathoz tartozik a parancssor vagy a kommentár, és hogy milyen néven lett elmentve a feladathoz tartozó kép!

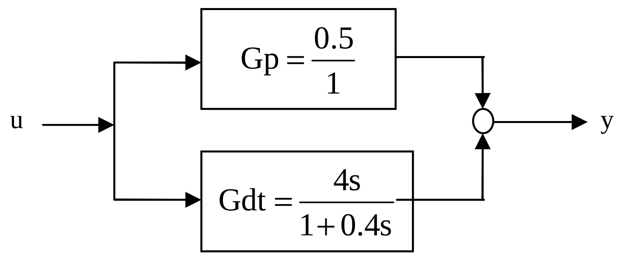
Az „M” fájl olyan szöveges fájl, amely kommentárokat és MATLAB parancssorokat tartalmaz és egybefüggően lefuttatható.

1. Legyen Gi MATLAB változó olyan IT0 alaptag átviteli függvénye, amely integrálási tényezője $K_I = 0.25\left[s^{-1}\right]$! Mutassa meg, hogy hol és hogyan olvasható le a T_I integrálási idő az átmeneti függvényen és a Bode diagramon!

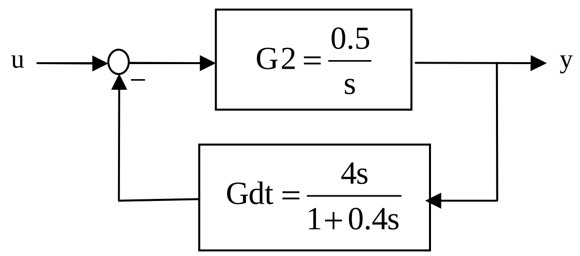
2. Határozza meg az alábbi, sorba kötött jelátvivő tagok eredőjének $G_3(s)$ átviteli függvényét.



3. Határozza meg a $G_3(s)$ eredő jelátvivő tag vágási körfrekvenciáját és jelölje be az eredő jelátvivő tag Bode és Nyquist diagramján!
4. A $G_3(s)$ jelátvivő tag átmeneti függvényén hol és hogyan olvasható le az integrálási idő és a látszólagos T_g időállandó?
5. Határozza meg az alábbi, párhuzamosan kötött jelátvivő tagok eredőjének $G_{PDT1}(s)$ átviteli függvényét!



6. A $G_{PDT1}(s)$ jelátvivő tag átmeneti függvényén hol és hogyan olvasható le a T_D differenciálási idő, a T időállandó, és a K_C erősítés?
7. Határozza meg az alábbi visszacsatoltan kötött jelátvivő tagok eredőjének $G_4(s)$ átviteli függvényét.



8. Határozza meg a $G_4(s)$ jelátvivő tag vágási körfrekvenciáját és saját frekvenciáját! Jelölje be az eredő jelátvivő tag Bode és Nyquist diagramján!

5.3. A minta feladatsor megoldása

Az „M” fájl szerkesztője a „File”→”New”→”M file” útvonalon nyitható meg. Az „M” fájlt matlab_1x néven mentse le, ahol az x a megoldandó feladatsora betűkódja. Ne felejtse beírni a nevét és Neptun kódját az „M” fájlba!

Az „M” *fájl* szerkesztőjének menüsorából a „Debug”, majd a legördülő menüből a „Save and Run” fület választva, az „M” fájl az elmentése után sorban végrehajtja a MATLAB parancsokat.

1. Az IT0 integráló alaptag átviteli függvénye:

$G_I = \frac{K_I}{s} = \frac{1}{sT_I}$. A példában megadott értékkel $G_I(s) = \frac{0.25}{s} = \frac{1}{s4}$,

ahol a K_I az integrálási tényező, T_I az integrálási idő.

A MATLAB parancs

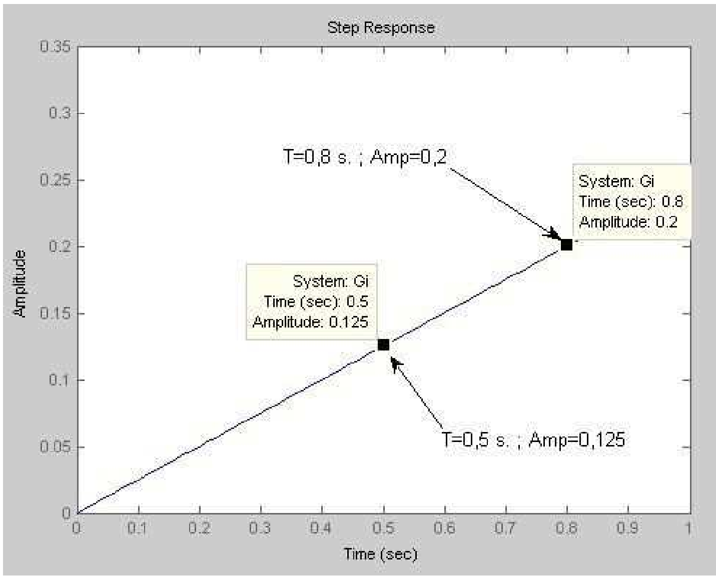
```
>>   Gi=tf(0.25,[1 0])
```

Eredmény:
```
Transfer function:
0.25
------
  s
```

Az átmeneti függvényt a **step** parancs, a Bode diagramot a **bode** parancs szolgáltatja. Először az átmeneti függvény vizsgáljuk:

```
>> step(Gi)
```

Eredmény:



5.1. ábra. A lementett kep1

Megjegyzések:

A kép mentésekor a kiterjesztést nem kell beírni.

A sárgaszínű információs ablak a lementett kep1.fig fájlban nem marad meg.

A leolvasás helyét és a leolvasott értékeket be kell írni a lementett ábrába.

Az eddigi M fájl:

```
% Jó Hallgató XXYYZZ
% MATLAB_1x
% 1. feladat
% Az átviteli függvény:
Gi=tf(0.25,[1 0])
% Az átmeneti függvény
step(Gi)
% Az ábra: kep1
% A kijelölt pontokból:
% Ki=(0,2-0,125)/(0,8-0,5)
% Ki=0,25 1/sec.; Ti=4 sec.
```

A körfrekvencia átviteli függvény Bode diagramja a következő:

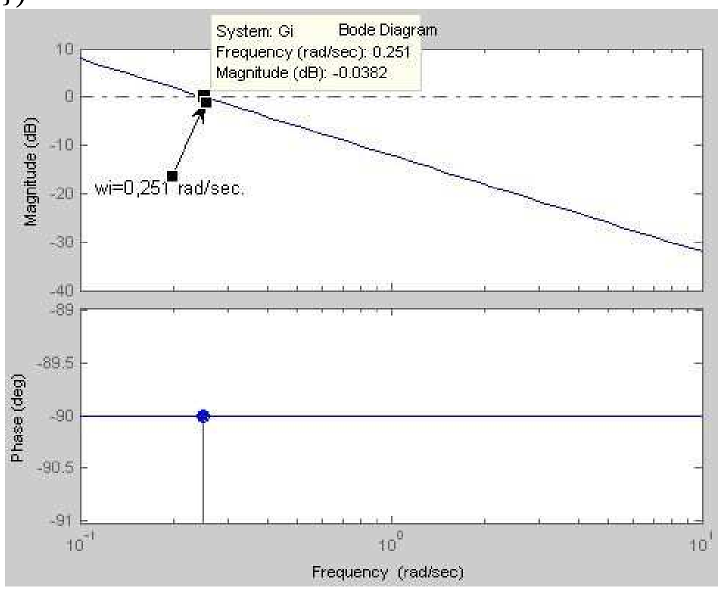
\>> bode(Gi)

Az automatikusan felvett körfrekvencia tartományban (1 – 10 rad/sec.) az amplitúdó átvitel görbéje nem metszi a 0 dB-es tengelyt. Ezért növelni kell a vizsgált körfrekvencia tartományt (4.10. kifejezés).

\>> bode(Gi,{0.1,10})

Eredmény:

Emlékeztető:
A jobb egérgomb → „Characteristics” → stb. segítségével jeleníthető meg szaggatott vonal és a vágási és fáziskereszteződési körfrekvenciák helye.



5.2. ábra. A lementett kep2

Az M fájl folytatása:

```
% A Bode diagram:
% bode(Gi)
% Nem volt megfelelő
bode(Gi,{0.1,10})
% Az ábra: kep2
% A 0dB metszésénél
% wi=0,251, így Ki~0,25
```

Kiegészítés

A DT0 alaptag vizsgálata hasonló az IT0 alaptagéhoz.

A PT1 alaptag kérdése: Hol és hogyan olvasható le az átmeneti függvényen és a Bode diagramon a PT1 alaptag időállandója?

Emlékeztető:

- Az átmeneti függvényen a végérték $h(\infty) \approx 63\%$-nál leolvasható idő a T időállandó.
- A PT1 alaptag időállandója a Bode diagramon a -45°-os fázistolásnál leolvasott ω körfrekvencia érték reciprok értéke.

A PT2 alaptag kérdései: Vajon a D csillapítási tényező nagyobb vagy kisebb, mint 0,5? Hol és hogyan olvasható le a jelátvivő tag erősítése az átmeneti függvényen?

Emlékeztető:

- Ha az Mp% < kisebb 5%-nál, akkor D > 0,5, és ha Mp% > 10%-nál, akkor D < 0,5-nél.
- Az erősítés (átviteli tényező) a jelátvivő tag átmeneti függvény végértéke osztva a konstans gerjesztő jel amplitúdójával.

2. Először a G1(s) és G2(s) jelátviteli tagok átviteli függvényét definiáljuk. A G1(s) jelátviteli tag gyöktényezős alakban van megadva, ezért a **zpk** parancsot célszerű alkalmazni.

```
>> G1=zpk([],[-0.3,-0.3],0.4)
```

Eredmény: Zero/pole/gain:

```
    0.4
-----------
(s+0.3)^2
```

A G2(s) jelátviteli tag gyöktényezős és polinomtört alakja azonos. Bár a gyöktényezős alak és polinomtört alak összeszorozható, ezt is a **zpk** paranccsal definiáljuk.

```
>> G2=zpk([],[0],0.5)
```

Eredmény: Zero/pole/gain:

```
0.5
----
 s
```

A G12 a G1 és a G2 szorzata.

```
>> G3=G1*G2
```

Eredmény: Zero/pole/gain:

```
     0.2
-------------
s (s+0.3)^2
```

Az M fájl folytatása:

```
% 2. feladat
G1=zpk([],[-0.3,-0.3],0.4)
G2=zpk([],[0],0.5)
G3=G1*G2
```

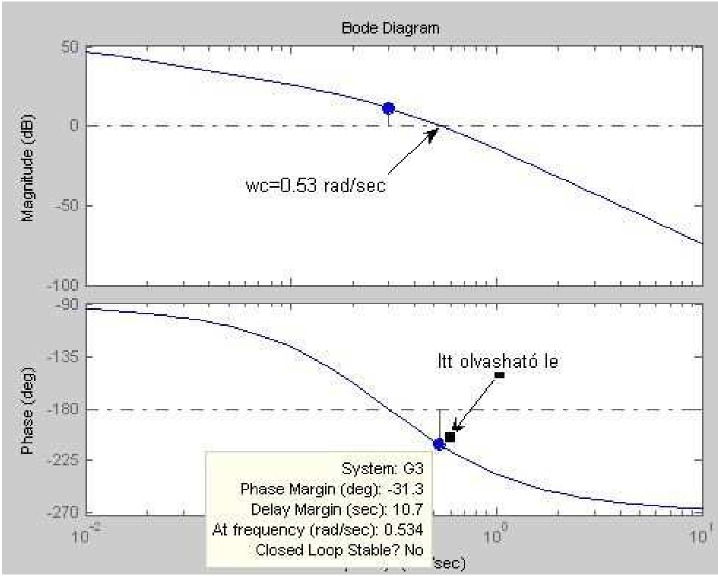
3. A vágási körfrekvencia (ω_c) az a körfrekvencia, ahol az amplitúdó átvitel egységnyi.

- A Bode diagramon ez azonos az amplitúdó kereszteződési körfrekvenciával, vagyis azt a körfrekvenciát kell megjelölni, ahol az amplitúdó menet metszi a 0 dB-es tengelyt.
- A Nyquist diagramon azt a körfrekvenciát kell megjelölni, ahol a Nyquist görbe metszi az egységsugarú kört.
  Emlékeztető: Az 5.4. ábrán a magyarázat érdekében rajta van a jobb egérgomb klikkre minden ikon kikapcsolt állapotában feljövő ablak. Az ábrán a negatív körfrekvencia tartomány kikapcsolása látszik.
  Felette a „Characteristics” → „All Stability Margins” útvonalon állítható be a vágási és a fáziskereszteződési körfrekvencia megjelenítése, valamint a Bode diagramon a -180° fázistolásnál a szaggatott egyenes és a Nyquist diagramon az egységsugarú kör.

A Bode diagramot a **bode** parancs rajzolja ki:

\>> bode(G3)

Eredmény:

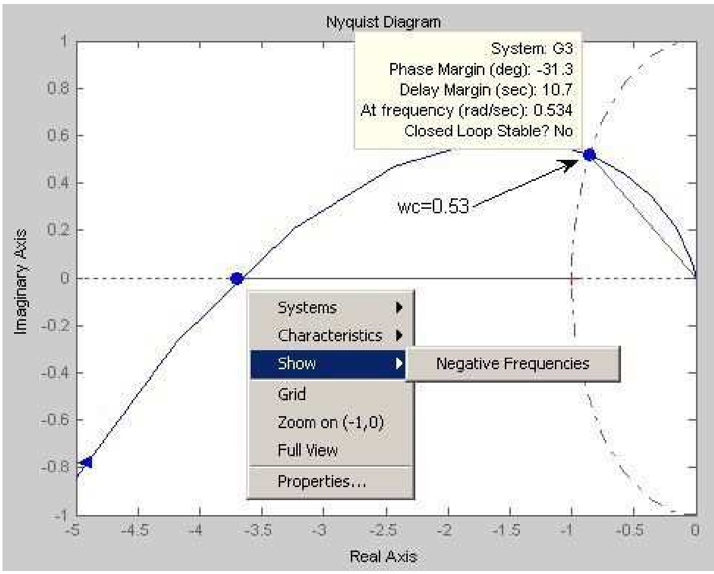


5.3. ábra. A lementett kep3

A Nyquist diagramon át kellett állítani a reális koordináta méretezését!

Emlékeztető: A koordináta tengelyek átméretezésének menete: a dupla klikk a képen (nem lehet a „Data Cursor” bekapcsolva). Az „Edit Plot” ikon állapotától függ, hogy a „Property Editor” vagy a „Property Editor - Axes” jön fel. A már elmentett kép újra nyitásakor csak az utóbbi jelenik meg.

A könnyebb kiértékelhetőség érdekében célszerű kikapcsolni a negatív körfrekvencia ábrázolását.



5.4. ábra. A lementett kep4

Az M fájl folytatása:

```
% 3 feladat
bode(G3)
% Az ábra: kep3
% az wc=0.53 rad/sec.
nyquist(G3)
% Az ábra: kep4
% A wc=0.53 rad/sec.
```

Kiegészítés

Lehet még kérdés a saját körfrekvencia (ω_n) és a fáziskereszteződési körfrekvencia (ω_p).

Emlékeztető:

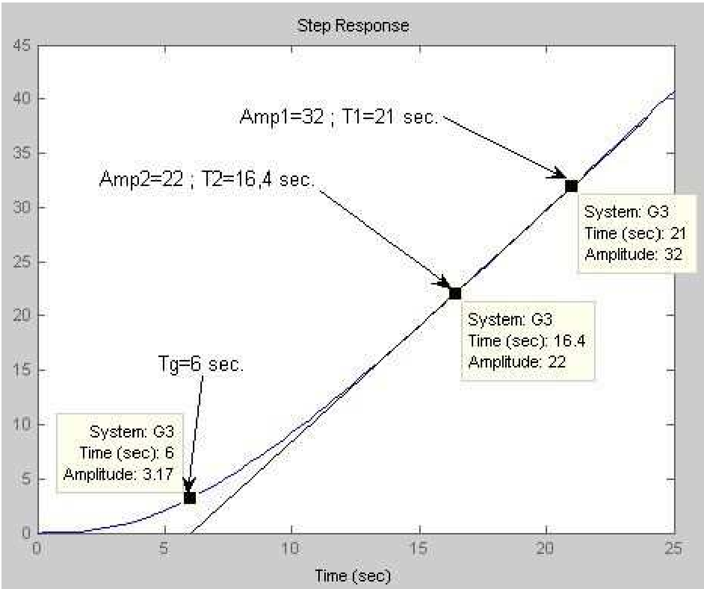
- A -90°-os fázistolásnál van a saját körfrekvencia.
- A -180°-os fázistolásnál van a fáziskereszteződési körfrekvencia.

4. A step paranccsal állítsuk elő a $G_3(s)$ átmeneti függvényét!

\>> step(G_3)

Eredmény:

A T_I integrálási idő meghatározásához az 5.5. ábra két olyan pontja kell, amely az átmeneti függvény egyenes szakaszán helyezkedik el. A T_g látszólagos időállandó az ábra egyenes szakaszához illesztett egyenes és az időtengely metszéspontjánál olvasható le.



5.5. ábra. A lementett kep5

Emlékeztető*: Az átméretezés: Dupla egérklikk → „Property Editor” → „Limits”. A rajz és a szöveg: a menüből „Insert” → a „Line”és a „Text Arrow” fülek választása.*

Az M fájl folytatása:

```
% 4. feladat
step(G3)
%Az ábra: kep5
%az ábra pontaiból
%(21 - 16,4)/(32 - 22)
% Ti=0,46 sec.
%A látszólagos egytároló
% Tg=6 sec.
```

Kiegészítés

Lehet feladat: Határozza meg önbeálló tagon a T_g látszólagos időállandót és a T_u látszólagos holtidőt!

Emlékeztető [1,2]:

- A T_g látszólagos időállandó az átmeneti függvény inflexiós pontjához illesztett érintő és az időtengely valamint az érintő és a végérték metszéspontjai közötti időtartam.
- A T_u látszólagos holtidő a kezdő érték és az átmeneti függvény inflexiós pontjához illesztett érintő és az időtengely metszéspontja közötti időtartam.

Lehet kérdés: Hogyan kell a Bode diagramon meghatározni, hogy hányad rendű energiatárolót tartalmaz a jelátvivő tag?

Emlékeztető:

- Ahányszor -90° a fázistolás végértéke, olyan rendű a tároló hatás. Például: Ha a fázistolás kezdő értéke 0° és a végértéke -270°, akkor T3, vagy ha a fázistolás kezdő értéke -90° és a végértéke 270°, akkor IT2.
  Megjegyzés: A holtidős tag megnehezíti vagy lehetetlené teszi az időállandók számának meghatározását.

Lehet feladat: Határozza meg a Bode diagramon, hogy önbeálló vagy integráló jellegű-e a jelátvivő tag!

Emlékeztető:

- A fázismenet vizsgálatával a legegyszerűbb eldönteni. A -90°-os fázistolás alacsony körfrekvencián utal az integráló jellegre.

5. A MATLAB szintaktikája lehetővé teszi, hogy a P típusú G_p átviteli függvényt értékét valós számként definiáljuk.

```
>> Gp=0.5
```

Eredmény: Gp =
0.5000

>> Gdt=tf([4 0],[0.4 1])

Eredmény: Transfer function:

```
   4 s
---------
0.4 s + 1
```

>> Gpdt1=Gp+Gdt

Eredmény: Transfer function:

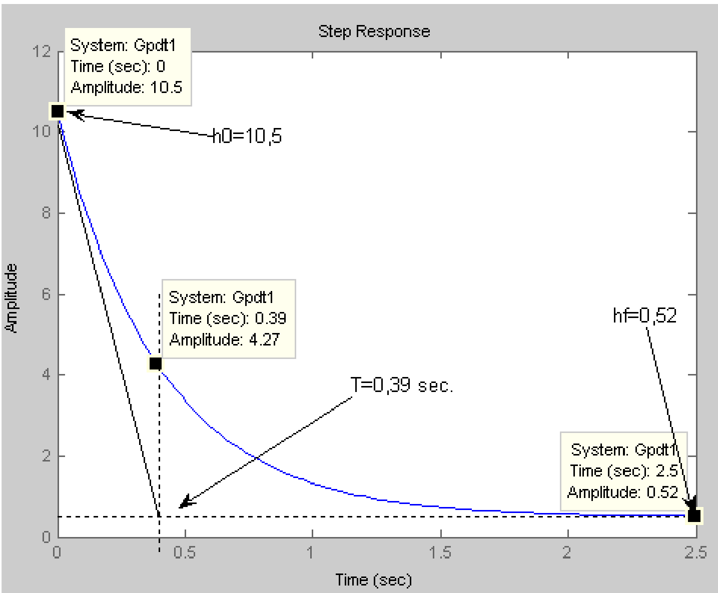
```
4.2 s + 0.5
-------------
  0.4 s + 1
```

Az M fájl értelemszerű.

6. Az 5.6. ábra mutatja a **step** parancs által kirajzolt átmeneti függvényt. Az 5.6. ábráról leolvasott értékek $h_{PDT1}(0) = 10.5 = h_{DT}(0) + h_P(0)$ és $h_{PDT1}(\infty) = 0.52 = h_{DT}(\infty) + h_P(\infty)$, valamint a T=0,39 sec. A T időállandót szerkesztéssel lehet meghatározni. A leolvasási hely a kezdeti meredekséggel húzott egyenes és a hf végérték egyenes metszéspontja.

>> Step(Gpdt1)

Eredmény:



5.6. ábra. A lementett kep6

Segítség: Ha kijelöli a vonalat és jobb egérklikk → „Line Style” választhat vonal stílusok (szaggatott (dash) pontozott (dot), stb.) között.

Szerkesztési segítség: A görbe pontjára helyezze a „Data Cursor”-t és utána húzza szerkesztő vonallal fedett részre!

A T_D differenciálási idő:

$T_D = (h(0) - h(\infty)) \cdot T = (10{,}5 - 0{,}52) \cdot 0{,}39 \approx 3{,}9\,\text{sec}$.

Magyarázat: A t=0 és a t=∞ időpontokban leolvasott számértékeket a határérték tételek segítségével értelmezzük.

A $G_{PDT}(s)$ differenciáló tag átmeneti függvényének az értéke a t=0 időpontban: $h_{DT}(0) = \lim\limits_{s\to\infty} \frac{T_D s}{Ts+1} = \frac{T_D}{T} = A_D$. Esetünkben az A_D differenciális erősítés értéke 10. A t=∞ időpontban $h_{DT}(\infty) = \lim\limits_{s\to 0} \frac{T_D s}{Ts+1} = 0$.

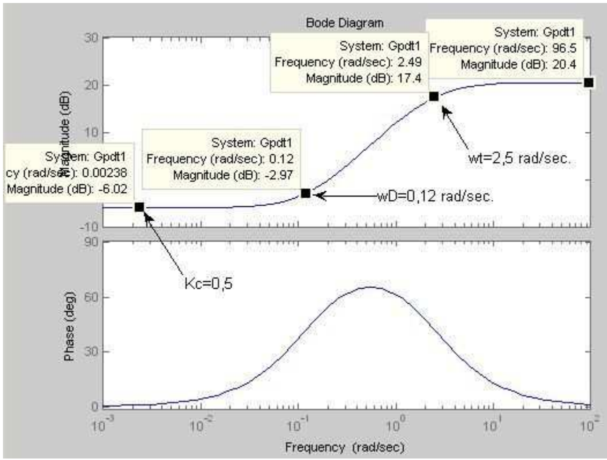
A $G_p(s)$ arányos tag átmeneti függvényének a t=0 és t=∞ időpontban egyaránt $h_P(0) = h_P(\infty) = K_C$. A párhuzamos kapcsolásban ezek az értékek összegződnek.

Kiegészítés

Lehet feladat: A PDT1 kompenzáló tag Bode diagramján a K_C erősítés, a T_D differenciálási idő, és a T időállandó leolvasása.

>> bode(Gpdt1)

Eredmény:



5.7. ábra. A Gpdt1 Bode diagramja

A leolvasás magyarázata érdekében hoztuk létre a Gpdt1 kompenzáló tag Bode diagramját (5.7. ábra).

- A K_C az alacsony körfrekvenciák egyenes szakaszán az erősítés értéke.

  Megjegyzés: Számolja át a dB-ben leolvasott értéket!

- Az ω_D az alacsony körfrekvenciákon az amplitúdó átvitel egyenes szakaszához tartozó erősítés értéknél 3 dB-lel nagyobb amplitúdó átviteli értékhez (az 5.7. ábrán ez -6 + 3 dB = -3 dB) tartozó körfrekvencia.

  A $T_D = \frac{K_C}{\omega_D}(1 - \frac{\omega_D}{\omega_T}) \approx \frac{K_C}{\omega_D}$ (5.1)

Magyarázat: A feladat olyan struktúrájú, mint az amerikai PDT kompenzáló tag (5.8. ábra).

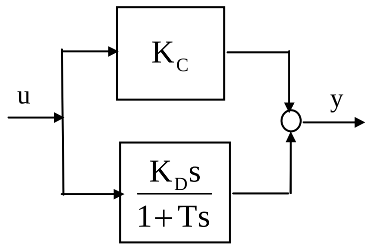
Ebből adódóan:

$$\frac{y(s)}{u(s)} = \frac{K_C + sTK_C + sT_D}{1 + sT}$$

Bode alakra hozva:

$$\frac{y(s)}{u(s)} = K_C \frac{1 + s(T + \frac{T_D}{K_C})}{1 + sT}$$

A mért ω_D érték: $\omega_D = \dfrac{1}{(T + \frac{T_D}{K_C})}$



5.8. ábra Az amerikai PDT struktúra

Ebből levezethető az 5.1. kifejezés ($\omega_T = \frac{1}{T}$ helyettesítéssel).

- ♦ Az ω_T a magas körfrekvenciákon az amplitúdó átviteli egyenes szakaszához tartozó erősítés értéknél 3 dB-lel kisebb amplitúdó átviteli értékhez (az 5.7. ábrán ez 20,4 - 3 dB = 17,3 dB) tartozó körfrekvencia.

A $T = \dfrac{1}{\omega_T}$ (5.2)

A leolvasott értékekkel:

$T = \dfrac{1}{2.5} = 0.4\,\text{sec}$. és $T_D = \dfrac{0.5}{0.12}(1 - \dfrac{0.12}{2.5}) = 3{,}97\,\text{sec}$.

A szerkesztési pontatlanságot figyelembe véve a kiszámolt értékek jók.

Lehet feladat: A PI kompenzáló tag átmeneti függvényén a K_C erősítés és a T_I integrálási idő leolvasása.

Emlékeztető:

- ♦ A K_C erősítés a t=0 időpontban olvasható le.
- ♦ A T_I integrálási idő leolvasása az 1. feladatban megismert módon számolható ki.

Lehet feladat: A PI kompenzáló tag Bode diagramján a K_C erősítés, és a T_I integrálási idő leolvasása.

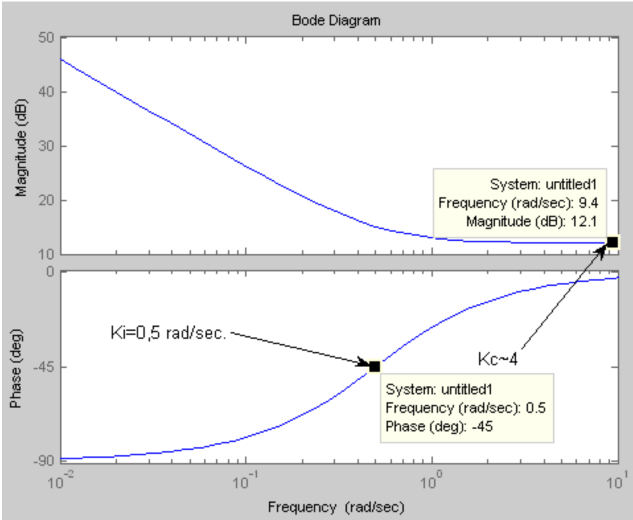
Legyen $G_{PI}(s) = 4 + \dfrac{1}{0.5s} = \dfrac{2s+1}{0.5s}$ (5.3)

```
>> Gpi=tf([2 1],[0.5 0])
        Transfer function:
        2 s + 1
        --------
          0.5 s
```

A Bode diagram:

```
>> bode(Gpi)
```

Eredmény:



5.9. ábra. A Gpi Bode diagramja

A PI kompenzáló tag Bode diagramja az 5.9. ábrán látható.

- A K_C a magas kör-frekvenciához tartozó erősítés érték. A decibelben leolvasott értéket számítsa át!
- Az -45° fázistolásnál van az ω_I. Az $\omega_I = K_I$.

$$T_I = \frac{1}{K_C \omega_I} \qquad (5.4)$$

Magyarázat: Az 5.3. kifejezés olyan struktúrájú, mint az amerikai PI kompenzáló tag.

Ebből adódóan:

$\frac{y(s)}{u(s)} = \frac{sT_I K_C + 1}{T_I s}$, ami Bode alak. (5.9. ábrán a $\frac{K_D s}{1+Ts}$ helyett $\frac{1}{T_I s}$ van.)

A mért ω_I érték: $\omega_I = \frac{1}{K_C T_I}$, amit átrendezve kapjuk az 5.4. kifejezést.

A 6. feladat M fájl részlete:

```
% 6. feladat
step(Gpdt1)
%Az ábra: kep6
%h0=0,52; hf=10,5;
%Kc=0,52; T=0,39 sec.
%TD=(10,5-0,52)*0,39=3,9 sec.
```

7. A negatívan visszacsatolt egyhurkos szabályozási kör alapjel átviteli függvényét a **feedback** parancs állítja elő. Szintaktikája:
feedback(előre vezető ág eredője,visszacsatolt ág eredője.)

```
>> G4=feedback(G2,Gdt)
```

Eredmény: Transfer function:

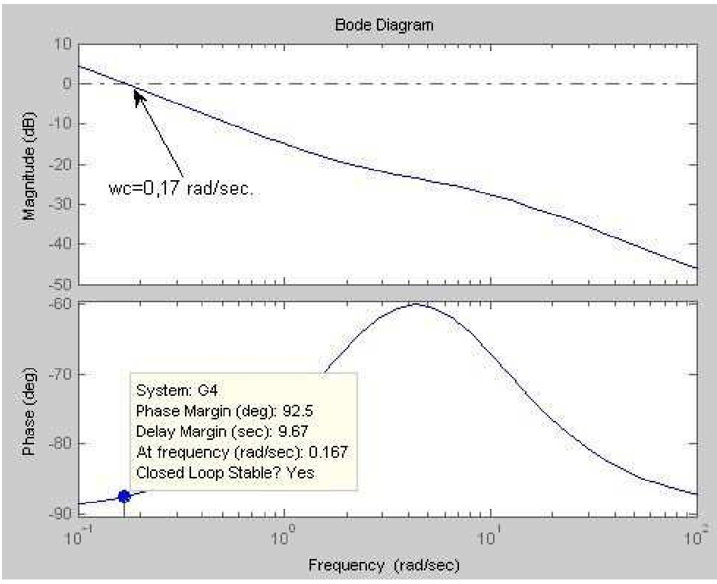
```
  0.2 s + 0.5
---------------
0.4 s^2 + 3 s
```

8. A Bode és a Nyquist diagramok létrehozásának és kiértékelésének menetét a 3. feladatban már bemutattuk.

```
>> bode(G4)
```

Eredmény:

Emlékeztető:
Az ábra fehér területén bal egérgomb klikk → „Characteristics” → „”All Stability Margins” jelöli ki az ω_C vágási körfrekvenciát és a ω_P fáziskereszteződési körfrekvenciát.



5.10. ábra. A Gpi Bode diagramja

Az 5.10. ábrán csak az ω_C vágási körfrekvencia van megjelölve.

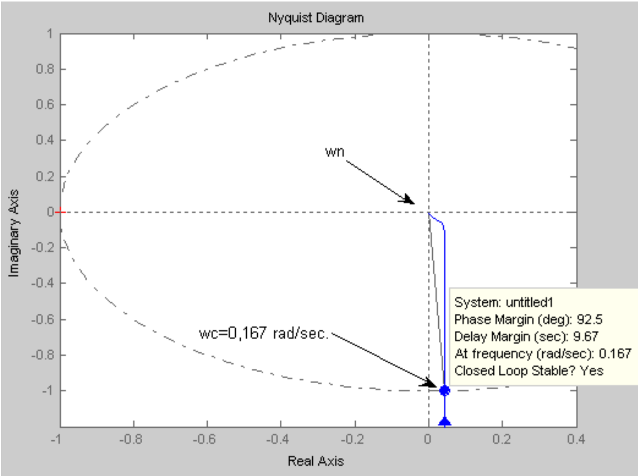
Emlékeztető: Az ω_n saját körfrekvencia -90° fázistoláshoz tartozik.

Az ω_n saját körfrekvencia a 0 rad/sec. és a ∞ rad/sec. körfrekvencia értékekhez tartozik. Egyik sincs bejelölve az 5.10. ábrán.

A Nyquist diagramon a ∞ rad/sec. körfrekvencia értékekhez tartozó ω_n saját körfrekvenciát meg lehetett jelölni (5.11. ábra)

\>> Nyquist(G4)

Eredmény:



5.11. ábra. A Gpi Nyquist diagramja

Az 5.11. ábrán az ω_C vágási és a ∞ rad/sec. körfrekvencia értékekhez tartozó ω_n saját körfrekvencia van megjelölve.

Emlékeztető: Az ábra fehér területén bal egérgomb klikk → „Show” lehet a negatív körfrekvencia ábrázolását megszüntetni.

Emlékeztető: Az ábra fehér területén bal egérgomb kettős klikk hatására jelenik meg a „Property Editor”, amivel a tengelyeket lehet átméretezni.

Az M fájl részlet:

```
% 8. feladat
bode(G4)
% Az ábra: kep7
% wc=0,17 rad/sec.
% wn nincs az ábrán bejelölve,mert
% a 0 és a végtelen rad/sec.
% értékhez tartóznak
nyquist(G4)
% Az ábra kep8
% Az értékek a fentiekkel azonos.
```

Kiegészítés

A $G_4(s)$ átviteli függvény általában nem a már az előző feladatokban deklarált változók visszacsatolt kapcsolása. Ez a 8. feladat megoldását nem befolyásolja. A 7. feladat kiegészül egy vagy két új LTI modell létrehozásával.

6. Az egyhurkos szabályozási kör vizsgálata

A MATLAB II. mérés célja, hogy a hallgatók megismerjék az egyhurkos szabályozási kör átviteli függvényeit és megnevezéseit; az egyhurkos szabályozási kör stabilitás vizsgálati módszereit; az egyhurkos szabályozási kör idő- és körfrekvencia tartománybeli minőségi jellemzőit.

6.1. Ellenőrző kérdések

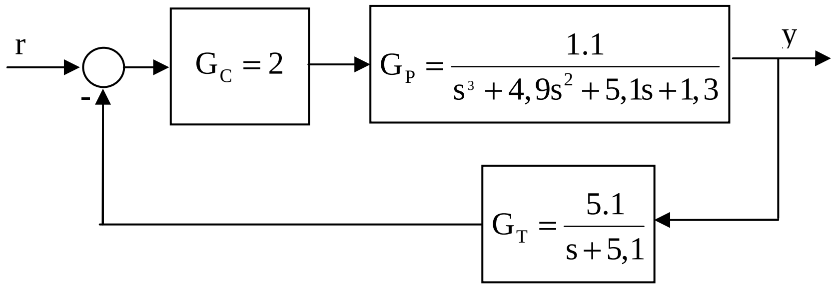
1. Helyezze el az elnevezéseket a szabályozási kör hatásvázlatának megfelelő elemeihez!
2. Helyezze el az elnevezéseket a szabályozási kör hatásvázlatának megfelelő jeleihez, jellemzőihez!
3. Helyezze el az elnevezéseket, ahol azok leolvashatók a szabályozási kör átmeneti függvényén!
4. Jelölje be, hogy a felsorolt jelek, jellemzők, értékek közül melyik szükséges vagy melyek szükségesek a szabályozási kör hibajelének meghatározásához!
5. Jelölje be a szabályozási kör hatásvázlatán, hogy hol olvasható le az alapérték!
6. Jelölje be, hogy a felsorolt jelek, jellemzők, értékek közül melyik szerepel vagy melyek szerepelnek a tolerancia sáv definíciójában!
7. Jelölje be a szabályozási idő (T_a) leolvasási helyét a szabályozási kör átmeneti függvényén!
8. Jelölje be a túllendülés ($M_{P\%}$) leolvasási helyét a szabályozási kör átmeneti függvényén!

9. Válassza ki, hogy a megadott kifejezések közül melyik a felnyitott hurokátviteli függvény!
10. Jelölje be a zárt szabályozási kör fázistartalékának (pm) leolvasási helyét a felnyitott hurok körfrekvencia átviteli függvényének Bode diagramján!
11. Jelölje be a zárt szabályozási kör erősítéstartalékának (gm) leolvasási helyét a felnyitott hurok körfrekvencia átviteli függvényének Bode diagramján!
12. Jelölje be a zárt szabályozási kör erősítéstartalékának (gm) leolvasási helyét a felnyitott hurok körfrekvencia átviteli függvényének Nyquist diagramján!
13. Jelölje be a zárt szabályozási kör vágási körfrekvenciájának ω_C leolvasási helyét a felnyitott hurok körfrekvencia átviteli függvényének Nyquist diagramján!
14. Válassza ki, hogy a megadott kifejezések közül melyik az alapjelre vonatkoztatott átviteli függvény ($G_{yr}(s)$)!

6.2. A minta feladatsor

A jegyzőkönyv részére hozzon létre M fájlt és matlab_2x néven mentse, ahol x a saját feladatának betűkódja! A jegyzőkönyvbe ne felejtse beírni, hogy melyik kép, melyik feladathoz tartozik! Ne felejtse beírni a nevét és Neptun kódját az „M” fájlba.

1. Határozza meg a 6.1. ábrán látható egyhurkos szabályozási kör $G_{yr}(s)$ alapjel átviteli és $G_0(s)$ hurokátviteli függvényeit!
2. Határozza meg a 6.1. ábra szabályozási körének az eredő szakasz ($G_E(s)$) átviteli függvényének időállandóit és az időállandók összegét!



6.1. ábra. A szabályozási kör

3. Állapítsa meg, hogy mekkora K erősítés értékkel kellene megszorozni az eredő szakasz K_E erősítését, hogy -135° legyen az amplitúdó kereszteződési körfrekvencián a fázistolás értéke!

4. Szerkessze meg a 6.1. ábra szabályozási körének SIMULINK modelljét $G_C(s)$ helyett $G_{PIDT}(s)$ kompenzáló taggal úgy, hogy az y(t) szabályozott jellemzőt és az e(t) rendelkező jelet oszcilloszkóppal lehessen vizsgálni, és a két változó a „Workspace” területen is megjelenjen! A modellt matlab_2x néven mentse el, ahol x a feladatának betűkódja!

   A kompenzáló PIDT tag európai struktúrájú, fix mintavételi idejű legyen és paraméterezze fel a következőképp:

- A PIDT kompenzáló tag integrálási ideje legyen $T_I = T_{max}$, ahol T_{max} az eredő szakasz legnagyobb időállandója!
- A PIDT kompenzáló tag differenciálási ideje legyen $T_D = T_{max2}$, ahol T_{max2} az eredő szakasz második legnagyobb időállandója, és a tárolós tag időállandója T legyen a differenciálási idő tizede!
- A PIDT kompenzáló tag erősítése legyen $K_C = K$, ahol K a 3. feladatban kiszámolt erősítés érték!
- Az eredő szakasz gyökeinek ismeretében válassza megfelelő értékre a szimuláció időtartamát és a mintavételi időt!

5. Válassza ki az oszcilloszkóp ábrája alapján, hogy a P, PI, és a PIDT kompenzáló tag közül melyik szolgáltatja a legjobb minőségi jellemző-

ket! A legjobbnak ítélt átmeneti függvényen a minőségi jellemzőket számszerűen is állapítsa meg!

6. A legrosszabbnak ítélt kompenzálási struktúra minőségi jellemzőit is állapítsa meg számszerűen!

7. Hasonlítsa össze a 6.1. ábra látható zárt szabályozási kör idő- és körfrekvencia tartománybeli minőségi jellemzőit!

6.3. A minta feladatsor megoldása

1. Először definiáljuk a $G_C(s)$, $G_P(s)$, $G_T(s)$ átviteli függvények Gc, Gp, Gt MATLAB változóit. Miután a $G_C(s)$ kompenzáló tag P típusú, vagyis konstans, nem szükséges a **tf** paranccsal deklarálni.

```
>>  Gc=2
                Eredmény:  Gc =
                                 2

>>  Gp=tf(1.1,[1 4.9 5.1 1.3])
                Eredmény:  Transfer function:
                                      1.1
                           ---------------------------------
                           s^3 + 4.9 s^2 + 5.1 s + 1.3

>>  Gt=tf(5.1,[1 5.1])
                Eredmény:  Transfer function:
                             5.1
                           --------
                           s + 5.1
```

A $G_0(s)$ felnyitott hurok átviteli függvény a hurokban szereplő tagok átviteli függvényeinek szorzata.

```
>>  G0=Gc*Gp*Gt
                Eredmény:  Transfer function:
                                              11.22
                           ---------------------------------------------------
                           s^4 + 10 s^3 + 30.09 s^2 + 27.31 s + 6.63
```

A $G_{yr}(s)$ alapjel átviteli függvényt a **feedback** parancs állítja elő:

```
>>   Gyr=feedback(Gc*Gp,Gt)
             Eredmény:   Transfer function:
                                        2.2 s + 11.22
                         ----------------------------------------------------
                         s^4 + 10 s^3 + 30.09 s^2 + 27.31 s + 17.85
```

2. Az eredő szakasz a szabályozó felöl nézve, vagyis az u végrehajtójel és az y_M ellenőrzőjel közötti jelátvivő tag.

```
>> Ge=Gp*Gt
             Eredmény: Transfer function:
                                        5.61
                       --------------------------------------------------
                       s^4 + 10 s^3 + 30.09 s^2 + 27.31 s + 6.63
```

Az időállandók meghatározása történhet: vagy a **roots** paranccsal vagy a gyöktényezős alakra való áttéréssel.

Alkalmazzuk most az utóbbit:

```
>> Geg=zpk(Ge)
             Eredmény: Zero/pole/gain:
                                        5.61
                       ------------------------------------------------
                       (s+5.1) (s+3.575) (s+0.9365) (s+0.3883)
```

A pólusok: p1=-5,1; p2= -3,57; p3=-0,94; p4=-0,39

Az időállandók a $T = \frac{1}{-p}$ alapján:

$$T1 = \frac{1}{5.1} = 0.2\ \text{sec}.;\ \ T2 = \frac{1}{3.57} = 0.28\ \text{sec}.;$$

$$T3 = \frac{1}{0.94} = 1.06\ \text{sec}.;\ \ T4 = \frac{1}{0.39} = 2.56\ \text{sec}..$$

Az időállandók összege: $\sum_{k=1}^{4} T_k = 0.2 + 0,28 + 1,06 + 2,56 = 4,1\ \text{sec}.$

Megjegyzés: Mind a négy időállandó egy dekádon belül van.

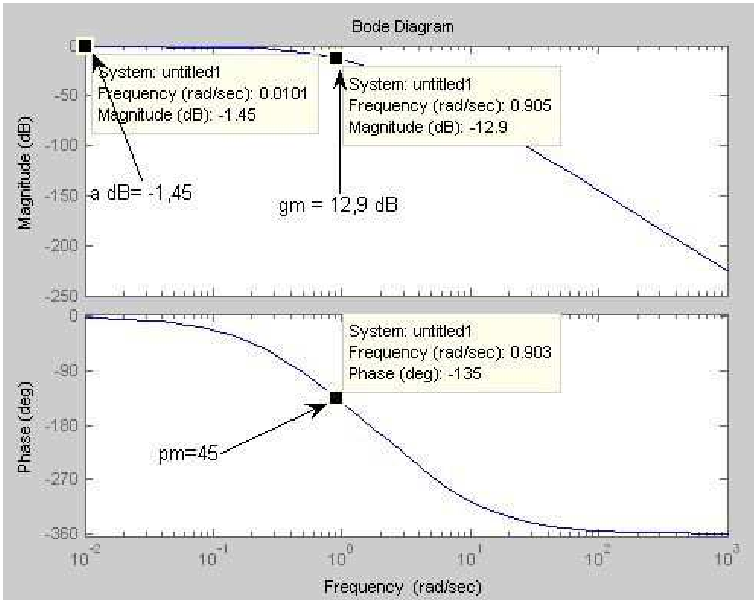
A M fájl részlet:

```
% 2. feladat
Ge=Gp*Gt
Geg=zpk(Ge)
% Zérus nincs, és a gyöktényezős alak erősítése Kzp=5,61
% A pólusok: p1=-5,1; p2= -3,57; p3=-0,94; p4=-0,39
% T=1/(-p) alapján
% T1=0,2 sec.; T2=0,28 sec.; T3=1,06 sec.; T4=2,56 sec.;
% SumTk=4,1 sec.
```

3. A feladat megoldásához a $G_E(j\omega)$ körfrekvencia átviteli függvény Bode diagramja kell.

>> bode(Ge)

Eredmény:



6.2. ábra. Az eredő szakasz körfrekvencia átviteli függvénye

A 6.2. ábra a $G_E(s)$ eredő szakasz körfrekvencia átviteli függvényének Bode diagramja. Az ábra alapján:

- Önbeálló jellegű a szakasz, tehát a K_C a kellően alacsony körfrekvenciákon mérhető. A leolvasott érték: a dB = -1,45. K_C = 0,85
- A -135° fázistolás az ω_x = 0,9 rad/sec. körfrekvencia értéknél van.

Emlékeztető: Minden ω_x körfrekvencián meg lehet határozni a fázistolás $\varphi(\omega_x)$ és a -180° különbségét ($\varphi(\omega_x)+180°$). A pm fázistartalék az ω_C vágási körfrekvenciához tartozó érték ($\text{pm}=\varphi(\omega_C)+180$).

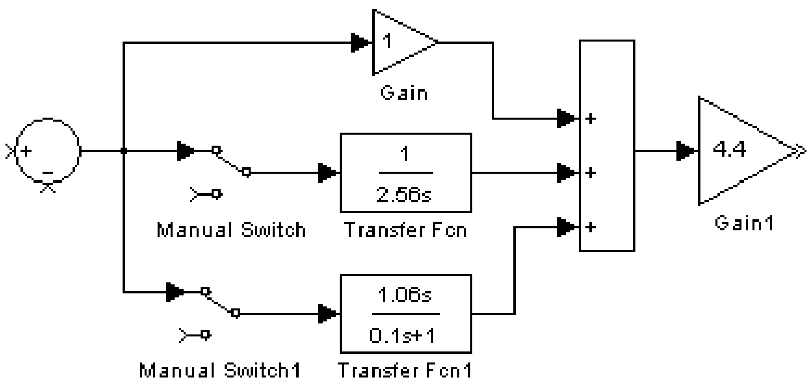
Az ω_x körfrekvencia értéknél kell meghatározni az amplitúdó átvitel értékét, és ennek reciprok értéke az a K erősítés, amivel megszorozva az eredő szakasz K_C erősítését az ω_x körfrekvenciánál lesz egységnyi az amplitúdó átvitel (Bode diagramon metszi a 0 db-es tengelyt).

Az ábráról leolvasott értékek: Az ω_x = 0,9 rad/sec. Az amplitúdó átvitel diagramon a 0,9 rad/sec körfrekvencián a = -12,9 dB, aminek a reciprok értéke az erősítés tartalék. A gm = 12,9 dB, ami átszámítva K = 4,4.

M fájl részlet.

```
% 3. feladat
bode(Gp*Gt)
% Az ábra: abra1
% a0 = -1,45 dB, ebből Kc=0,85;
% -135 foknál wx=0,9 rad/sec.
% a wx körfrekvencián a=-12,9 dB.
% gm=12,9 dB, ebből K=4,4
```

4. Először a PIDT kompenzáló tagot hozzuk létre a SIMULINK szerkesztő felületén. Ehhez két-két „Transfer Fcn", „Gain", „Manual Switch"; „Sum" grafikus objektum kell. Az európai PIDT kompenzáló tag SIMULINK modellje a 6.4. ábrán látható.



6.3. ábra Európai PIDT kompenzáló tag SIMULINK modellje

Megjegyzés: Az arányos ágat fixen bekötjük. Ezért elég két „Manual Switch”.(Így I kompenzációt nem lehet választani.) A „Manual Switch” egyik morzeérintkezőjét szabadon hagyva szimuláció minden futtatásakor figyelmeztetés jelenik meg parancsablakban.

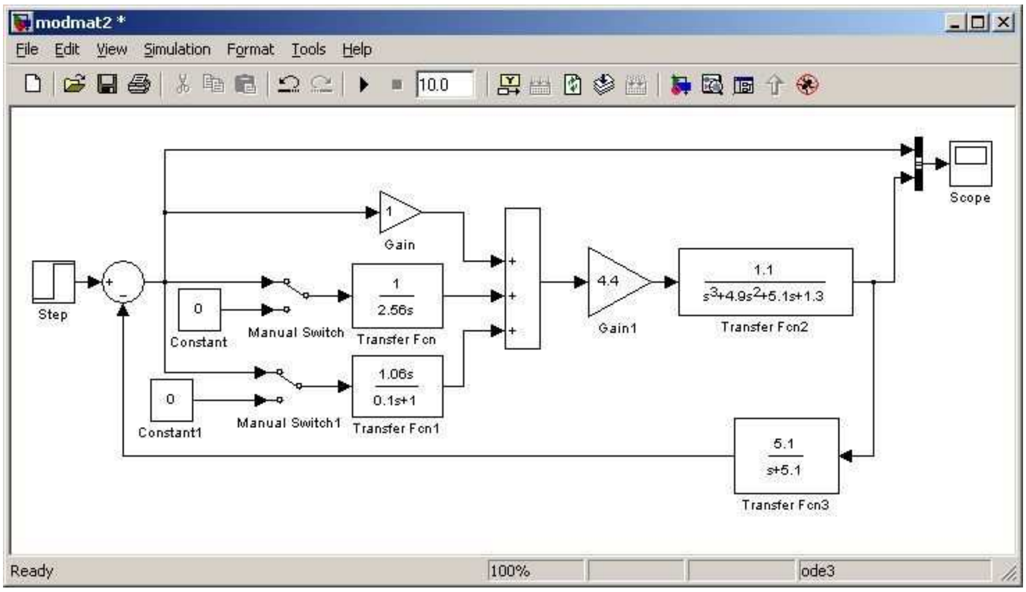
Paraméterezés: A T_I integrálási idő a 2. feladatban kiszámolt legnagyobb időállandó értékével; a T_D differenciálási idő a második legnagyobb időállandó értékével egyezik meg. A DT1 egy tárolós differenciáló jelátvivő tag T (tárolós) időállandója a T_D tizede. A „Gain1” grafikus objektum erősítése a 3. feladatban meghatározott K értékkel azonos.

Ezután hozzuk létre a zárt szabályozási kört a SIMULINK szerkesztő felületén, majd egészítsük ki a jelvizsgálatokhoz szükséges grafikus objektumokkal! Ehhez két „Transfer Fcn” és egy-egy „Step”; „Bus Creator”; és „Scope” grafikus objektum használunk fel.

Emlékeztető: A fenti objektumok paraméterezési szabályai a 4. fejezetben találhatók.

Megjegyzés: Ha nem akarja, hogy a szabadon hagyott morzeérintkező a szimuláció minden futtatásakor figyelmeztetést írjon a parancsablakban, akkor két 0 értékűre állított konstans grafikus objektummal – ami a „Commonly used” könyvtárban található - zárja le a morzeérintkezőket!

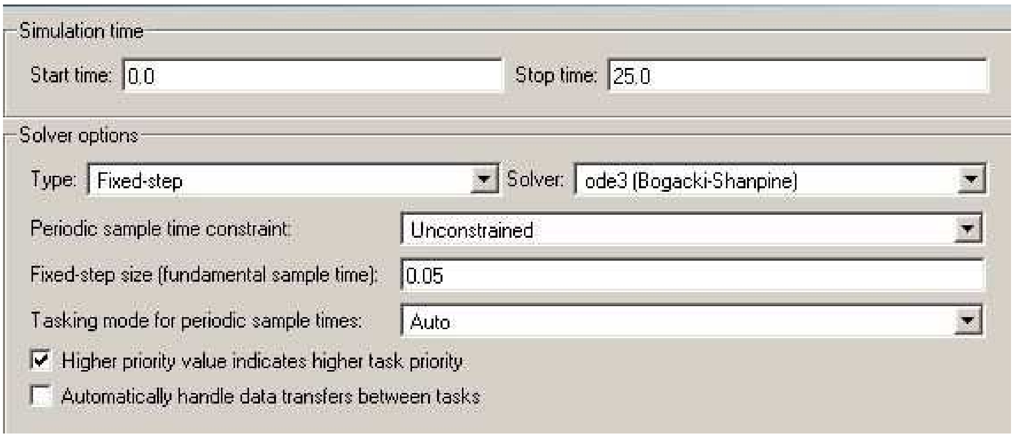
A 6.4. ábra mutatja a kész modellt



6.4. ábra A zárt szabályozási kör SIMULINK modellje

Emlékeztető: Ne felejtse a „Scope” → „Parameters” ikon → „Data history” fület megnyitva a „Workspace” területre menteni a „Scope” ábráit!

A 6.4. ábrán a „Simulation” legördülő menüből a „Configuration Parameters” fület választva a feljövő párbeszédablakban (6.5. ábra) be kell állítani a szimulációs paramétereket.



6.5. ábra A szimulációs paraméterek beállítása

A mintafeladatban az eredő szakasz időállandóinak összege: $T_\Sigma = \sum T_k = 4{,}1\,\mathrm{sec}$. Majdnem mind a négy időállandó egy dekádon belül van, ezért legyen a szimulációs időtartam: 25 sec.

Emlékeztető: a domináns póluspárok alapján megfogalmazott ökölszabály: A megfelelő szimulációs időtartam az eredő szakasz időállandói összegének kb. négy – ötszöröse. Ha azonban az eredő szakasznak három vagy több időállandója van egy dekádon belül, akkor jobb a hat – hétszeres időtartammal elvégezni a szimulációt.

Fontos: Az aluláteresztő jellegű rendszereknél, ha a T_Σ időtartam alatt legalább 200 mintavétel történik, akkor a szabályozási kör kvázi folytonosnak tekinthető.

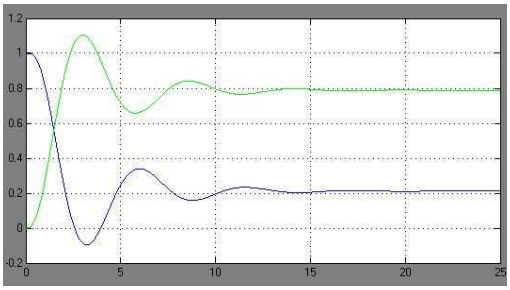
Megjegyzés: A gyakorlatban a T_Σ *időtartam alatt 1000 mintavételnél több már nem növeli, sőt kerekítési pontatlanságok miatt inkább csökkenti a modell pontosságát.*

A mintafeladatban legyen a T_0 mintavételi idő 0,05 sec. (Ez, a t=0 kiindulási értékkel együtt 501 mintavételi pontot jelent.)

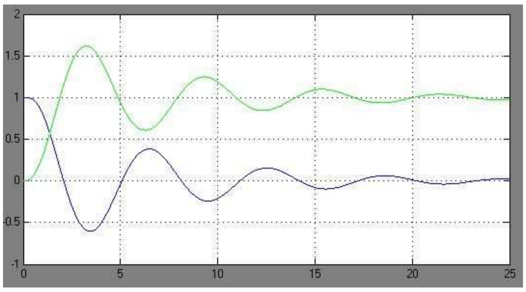
Az M fájl részlet:

```
% 4. feladat
% A modell matlab_2x néven van elmentve
% A Tsum szimulációs idő ~ 6*SumTk= 25 sec.
% A Fix mintavételi idő 0,05 sec.
```

5. A P kompenzáló taggal elvégzett szimuláció oszcilloszkóp képernyője a 6.6. ábrán, a PI kompenzáló taggal elvégzett szimuláció oszcilloszkóp képernyője a 6.7. ábrán látható.

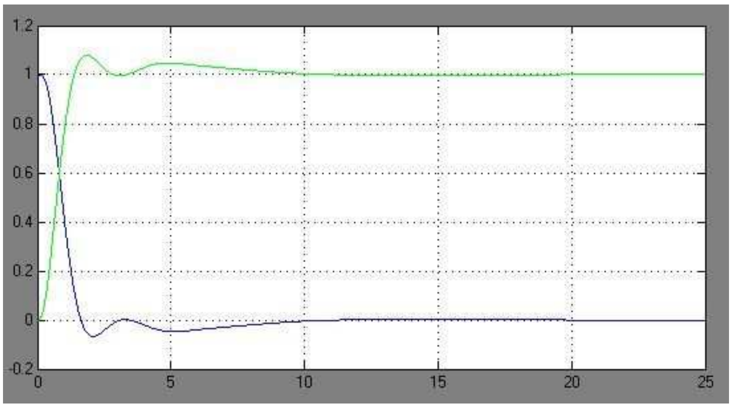


6.6. ábra P kompenzáció esetén a hibajel és a szabályozott jellemző



6.7. ábra PI kompenzáció esetén a hibajel és a szabályozott jellemző

Bekapcsolva a DT1 hatást is az eredmény a 6.8. ábrán látható. PIDT kompenzáló taggal látványosan jobb minőségi jellemzőkkel rendelkezik a szabályozási kör, mint a P vagy PI kompenzáló taggal.



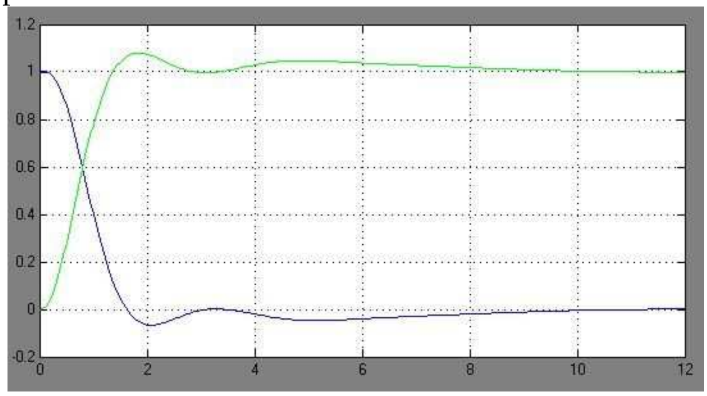
6.8. ábra PIDT kompenzáció esetén a hibajel és a szabályozott jellemző

Megjegyzés: Az eredő szakasz tulajdonságaiból adódik, hogy melyik kompenzáló tag lesz a legmegfelelőbb.

A 6.8. ábrán az is szembeötlő, hogy jóval kisebb a $T_{a2\%}$ szabályozási idő, és így érdemes a szimuláció időtartamát is csökkenteni.

Legyen a szimuláció időtartama 12 sec. A T_0 mintavételi időt nem kell módosítani, mert így is 241 mintavételezési pontunk van.

Lefutatva az új „Stop time” értékkel a szimulációt az oszcilloszkóp képernyő képe a 6.9. ábrán látható.



6.9. ábra PIDT kompenzáció esetén a hibajel és a szabályozott jellemző

Az oszcilloszkóp képernyő nem rendelkezik az ábra kiértékeléséhez szükséges szolgáltatásokkal. Ezért a „Figure” GUI-ban kell megjeleníteni a „Workspace” területre mentett változókat.

A „Bus Creator” bekötési sorrendjének megfelelően az SD változó második oszlopa a hibajel és a harmadik oszlopa a szabályozott jellemző. A két oszlop kinyerése (4.4. kifejezés):

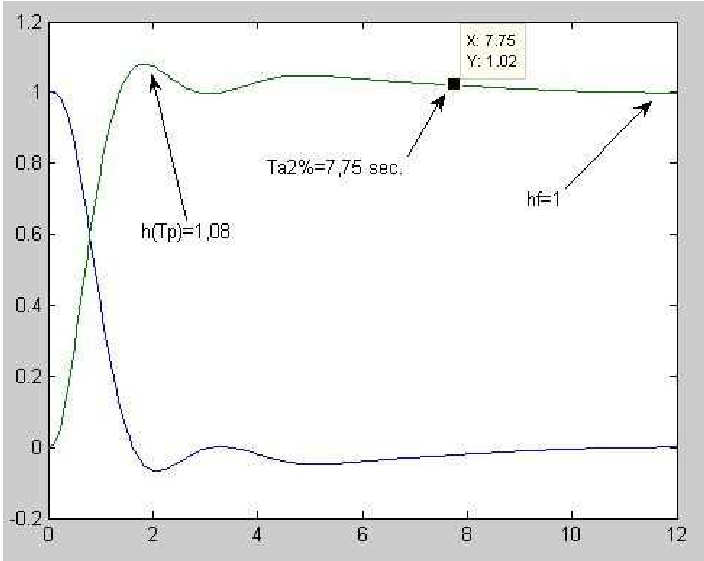
>> adat1=SD(:,2:3);

Ezután a „Figure” GUI meghívható a **plot** paranccsal.

>> plot(tout,adat1)

Fontos: A **plot** paranccsal nem csak átmeneti vagy átviteli függvény jeleníthető meg, ezért az átmeneti vagy átviteli függvények kiértékelését segítő szolgáltatások itt nem működnek. A „Data Cursor” használatakor csak az X, Y koordináta pontok jelenek meg és nem működik a jobb egérgomb klikk. Ellenőrizze, hogy a vízszintes tengelyen az időkoordináta vagy az adatpontok száma van-e ábrázolva!

A 6.10. ábrán látható a hibajel és a szabályozott jellemző. Miután egységugrás alapjel a gerjesztés a szabályozott jellemző függvénye egyben az átmeneti függvény. A kiértékelést a szabályozási kör átmeneti függvényén végezzük el. A 6.10. ábrán az X az időváltozó másodpercben, az Y az aktuális h(t) átmeneti függvény érték.



6.10. ábra A PIDT kompenzálás **plot** parancssal kirajzolt hibajele és szabályozott jellemzője

A $h(\infty)$ végérték és a $h(T_P)$ csúcsérték a „Data Cursor" segítségével közvetlenül leolvasható. E két értékből a túllendülés:

$$M_{P\%} = \frac{h(T_P) - h(\infty)}{h(\infty)} 100 = 8\%$$

A $T_{a2\%}$ szabályozási idő meghatározásához szükség van a Δ tolerancia-sáv határértékeire. Ez most nagyon egyszerű:

$\Delta_{min} = 0{,}98 * h(\infty) = 0{,}98$; és $\Delta_{max} = 1{,}02 * h(\infty) = 1{,}02$.

A 6.10. ábrán jobbról balra addig kell mozgatni a „Data Cursor"-t, amíg a sárga információs ablakban az Y koordináta értéke nem kisebb a Δ_{min} vagy a nem nagyobb a Δ_{max} értéknél. Ennél a pontnál az X koordináta érték mutatja a $T_{a2\%}$ szabályozási idő.

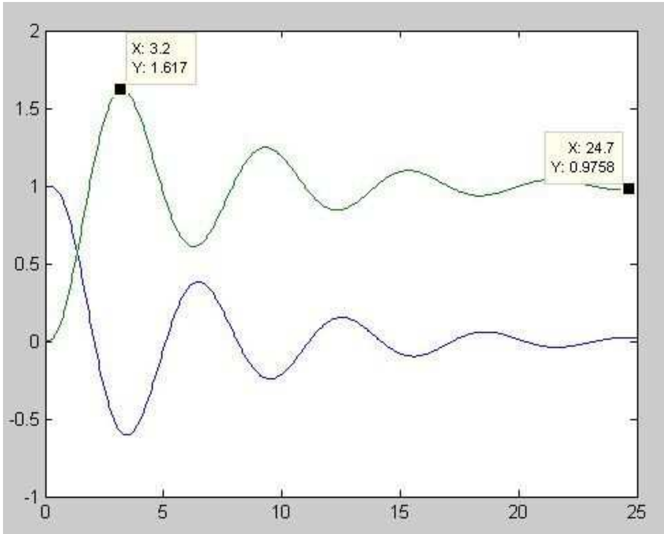
Az M fájl részlete:

```
% 5. feladat
% Egyértelműen a PIDT kompenzácó volt a legjobb
% Elegendő volt Tsum=12 sec. szimulációs időtartam.
adat1=SD(:,2:3);
plot(tout,adat1)
% Az ábra: abra1
% yh=1-hf=0; Ta2%=7,75 sec.; Mp=(h(Tp)-hf)/hf=8%
```

6. Nehéz egyértelműen választani, mert a PI kompenzáló taggal megvalósított szabályozásnak nincs maradó szabályozási eltérése, viszont hosszasan és nagyokat leng. Ezért ezt választjuk.

A 6.11. ábra elkészítéséhez ki kell kapcsolni a PDT ágat; vissza kell állítani a szimulációs időtartamot; le kell futtatni a szimulációt. Ezt követően:

```
>> adat2=SD(:,2:3);
>> plot(tout,adat2)
```



6.11. ábra A PI kompenzálás **plot** paranccsal kirajzolt hibajele és szabályozott jellemzője

A 6.11. ábráról leolvasott értékekből látszik, hogy a kevés a szimulációs időtartam. Ennél a mérésnél nem kell teljesen egzakt kiértékelés.

M fájl részlet:

```
% 6. feladat
% Ábra: abra2
% Nem elegendő a szimulációs időtartam.
% 1 típusú a szabályozási kör, ezért hf=1, így yh=0.
% Mp=(h(Tp)-hf)/hf=61,7%
% Ta2% > 25 sec.
```

Fontos: Ha a PI kompenzáló tag erősítését 0,8 értékre állítjuk, akkor majdnem olyan jó minőségi paramétereket kapunk, mint az 5. feladat PIDT kompenzáló tagjával. A kompenzáló struktúra véletlenszerű megválasztása és a paraméterek Monte Carlo módszerű keresése adhat elfogadható eredményt, de nem megbízhatóan és biztos nem az optimálist.

7. Az idő- és a körfrekvencia tartományban egyaránt a zárt szabályozási kör viselkedésének jellemzésére minőségi jellemzőket vezettek be.

Elméleti áttekintés

Az **időtartományban** a zárt szabályozási kör h(t) átmeneti függvényét vizsgáljuk. Az irányítástechnika számára legfontosabb minőségi jellemzők az időtartományban:

- A y_h maradó szabályozási eltérés, ami az $y_{ideális}$, az alapjel által előírt érték és a $h(\infty)$ végérték különbsége.
- A $T_{2\%}$ szabályozási idő, ami ahhoz szükséges, hogy az y(t) szabályozott jellemző értéke maradjon a $h(\infty)$ végérték körüli $\Delta = \pm 2\%$ tolerancia sávon belül.
- Az $M_{P\%}$ túllendülés, ami az $M_{P\%} = \dfrac{h(T_P) - h(\infty)}{h(\infty)}$ képlettel definiált, ahol $h(\infty)$ a végérték, a $h(T_P)$ a szabályozott jellemző csúcsértéke, és a T_P a csúcsértékhez tartozó idő.

A **szabályozási cél**, hogy minél kisebb (lehetőleg 0) maradó szabályozási eltéréssel, minél gyorsabban, és túllendülés nélkül vagy előírt maximális túllendüléssel kövesse az egységugrás alapjelet a szabályozási kör.

Megjegyzés: Ha a zárt szabályozási körben mechanikai működésű végrehajtó van, akkor fontos lehet:

- *A ν* lengés szám, *ami a T_a szabályozási időn belüli lengési periódusok száma, vagyis a végérték fölé lendülések száma.*
- *A T_r* felfutási idő *az átmeneti függvény $h(\infty)$ végértékének 10%-kától, a végérték 90%-káig jutáshoz szükséges időtartam.*

A **körfrekvencia tartományban** az $G_{yr}(j\omega)$ alapjel körfrekvencia átviteli függvényt vizsgáljuk. Az irányítástechnika számára legfontosabb minőségi jellemzők az körfrekvencia tartományban:

- Az a_h dB maradó szabályozási eltérés, ami 0 dB és az $\omega = 0$ körfrekvenciához tartozó erősítés $|G_{yr}(0)|$ különbsége.
- Az A_{max} kiemelés, ami az amplitúdó átvitel maximális értéke $|G_{yr}|_{max}$ és a $|G_{yr}(0)|$ hányadosa.

- Az ω_C vágási körfrekvencia, ami végső soron a zárt szabályozási kör sávszélességét határozza meg.

A szabályozási cél, hogy maradó szabályozási eltérés nélkül, minél nagyobb sávszélességgel (minél nagyobb ω_C értékkel), minél kisebb kiemeléssel rendelkezzen a zárt szabályozási kör.

A minőségi jellemzők összehasonlítása:

- Azonos dimenzióban összevetve, a maradó szabályozási eltérésre azonos értéket kell szolgáltatnia a két vizsgálatnak.
- A $T_{a2\%} \approx \dfrac{5}{\omega_A}$ (6.1)

  ahol az ω_A a maximális kiemeléshez tartozó körfrekvencia. Ha nincs kiemelés, akkor $\omega_A = \omega_C$ vágási körfrekvenciával.
- A túllendülés mértéke a maximális kiemelés $\dfrac{|G_{yr}|_{max}}{|G_{yr}(0)|}$ értékétől függ.

  Ha $\dfrac{|G_{yr}|_{max}}{|G_{yr}(0)|} \geq 1,5$, akkor $\dfrac{h(T_P)}{h(\infty)} \approx \dfrac{|G_{yr}|_{max}}{|G_{yr}(0)|} - 0,1$ (6.2)

  Ha $\dfrac{|G_{yr}|_{max}}{|G_{yr}(0)|} \leq 1,5$, akkor $\dfrac{h(T_P)}{h(\infty)} \approx \dfrac{|G_{yr}|_{max}}{|G_{yr}(0)|}$ (6.3)

A feladat megoldása

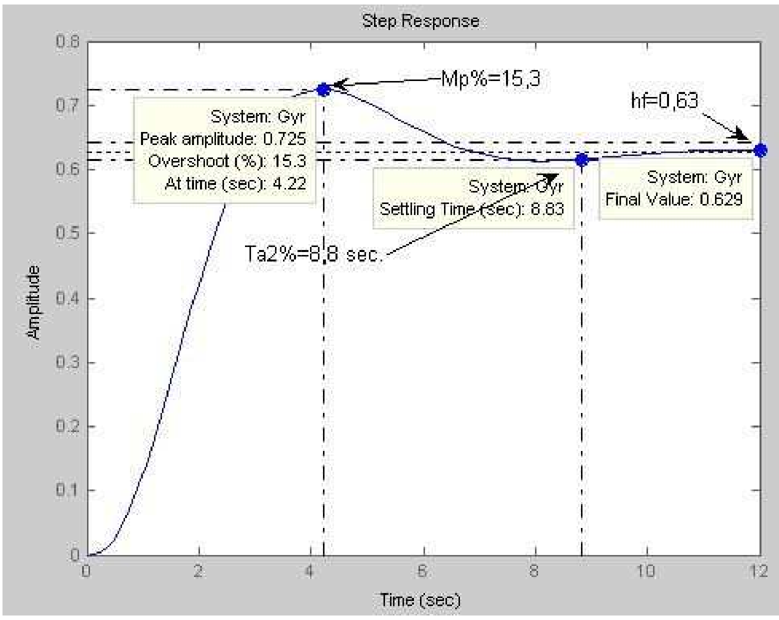
Kezdjük a zárt szabályozási kör átmeneti függvényével. Az átmeneti függvény a 6.12. ábrán látható.

Emlékeztető: A 6.12. ábrán a jobb egérgomb klikk → „Characteristics” → stb. hatására jelöli ki a minőségi jellemzők helyét. Az ábra nem őrzi meg a sárga információs ablakot. Az ábrán jelölje meg a leolvasás helyét és az M fájlba írja be az értékeket!

Megjegyzés: Ha össze akarja hasonlítani a SIMULINK generálta változókkal a ***plot*** *paranccsal kirajzolt ábrákkal, akkor a 4. fejezetben leírt módon létre kell hozni egy a SIMULINK beállításokkal azonos léptékű és terjedelmű „t” időváltozót, és a megfelelő szintaktikájú* ***step*** *paranccsal generálni a zárt szabályozási kör átmeneti függvényét.*

Megjegyzés: A $h(T_P)$ csúcsérték és a $h(\infty)$ végérték arányát az összehasonlítás érdekében érdemes kiszámítani.

A zárt szabályozási kör átmeneti függvényét a **step** parancs szolgáltatja.
>> step(Gyr)
Eredmény:



6.12. ábra. A lementett abra3

Folytassuk a körfrekvencia átviteli függvénnyel

Megjegyzés: A 6.13. ábrán a jobb egérgomb klikk → „Show” útvonalon kikapcsoltuk a fázis diagramot, mert itt nem olvasunk le róla adatot.

Végezzük el az összehasonlítást úgy, hogy a 6.12. ábráról leolvasott értékeket összevetjük 6.13. ábráról leolvasott értékekből kiszámított értékekkel. Az összehasonlításhoz a 6.13. ábráról leolvasott erősítés értékeket át kell számolni: $\left|G_{yr}(0)\right| \approx -4\,\mathrm{dB} \to 10^{\frac{-4}{20}} = 0,63$ és $\left|G_{yr}(0)\right| \approx -3\,\mathrm{dB} \to 10^{\frac{-4}{20}} = 0,71$.

Az y_h maradó szabályozási eltérés:

A 6.12. ábráról leolvasva. $h(\infty) = 0,63$. A maradó szabályozási eltérés: $y_h = 1 - h(\infty) = 0,27$.

A 6.13. ábráról leolvasva. $\left|G_{yr}(0)\right| \approx -4\,\mathrm{dB} \to 10^{\frac{-4}{20}} = 0,63$. A maradó szabályozási eltérés: $y_h = 1 - h(\infty) = 0,27$.

A két érték megegyezik.

A $T_{a2\%}$ szabályozási idő:

A 6.12. ábráról. $T_{a2\%} = 8,8\,\text{sec}$.

A 6.13. ábráról az ω_A = 0,6 rad/sec.

A 6.1. kifejezéssel számolt szabályozási idő: $T_{a2\%} \approx \frac{5}{\omega_A} 8,3\,\text{sec}$.

Az eltérés 6%-os.

Az $M_{P\%}$ túllendülés:

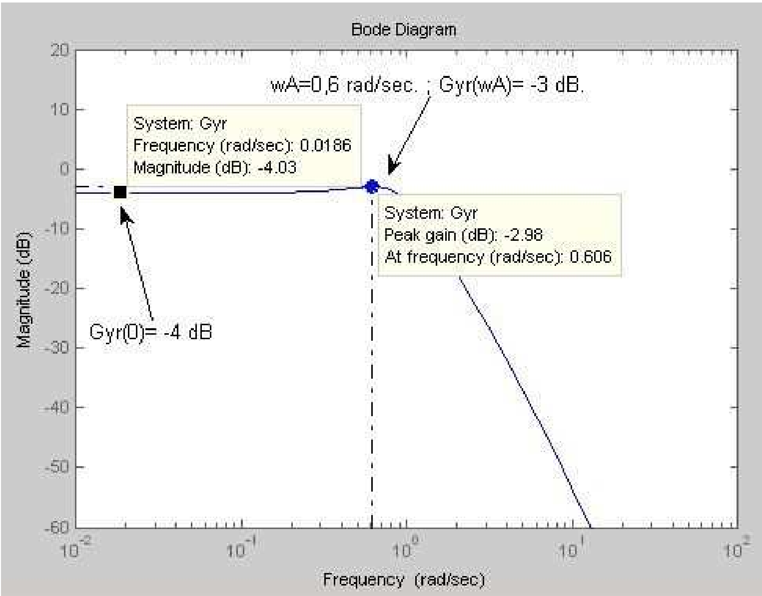
A 6.12. ábráról leolvasott érték 15,3%. A maximális kiemelés: $\frac{|G_{yr}|_{max}}{|G_{yr}(0)|} = \frac{0,71}{0,63} = 1,26$, ami kisebb, mint 1.5.

A 6.3. kifejezés alapján: $\frac{h(T_P)}{h(\infty)} = 1,26$. Átrendezve: $h(T_P) = 0,71$. Behelyettesítve $M_{P\%} = \frac{h(T_P) - h(\infty)}{h(\infty)} = \frac{0,71 - 0,63}{0,63} = 12,7\%$.

Az eltérés 20%-os.

>> bode(Gyr)

Eredmény:



6.13. ábra. A lementett abra4

Nézzük meg, mit kell minimálisan beírni az M fájlba.

```
% 7. feladat

% Időtartománybeli vizsgálat
%step(Gyr)
%Az ábra: abra3
%yh=1-hf=1-0,63=0,37; Ta2%=8,8 sec.;
%Mp%=15,3;

% Frekvencia tartománybeli vizsgálat
% bode(Gyr)
%Az ábra: abra4
% wA=0,6 rad/sec; Gyr(wA)= -3 dB ami 0,71 ; Gyr(0)= -4 dB ami 0,63
% yh=1-0,63=0,37; Ta2%=5/0,6=8,3 sec.;
% h(Tp)/hf=Gyr(wA)/Gyr(0)=1,26; h(Tp)=0,71; Mp%=(h(Tp)-hf)/hf=12,7

% Összehasonlítás:
% Az yh maradó azonos
% A Ta2% szabályozási idő 8,8 sec. és 8,3 sec. ami 6%-os eltérés.
% Az Mp% túllövés 15,3% és 12,7% ami 20%-os eltérés.
```

7. A szabályozási kör kompenzálása

A MATLAB III. mérés két feladatsorból áll. Az első feladatsor célja, hogy a hallgatók gyakorolják, hogyan kell a megfelelő kompenzáló tag struktúrát választani adott eredő szakaszhoz az eredő szakasz átmeneti függvénye alapján, és hogyan kell a választott kompenzáló tag struktúrát paraméterezni.

A második feladatsor célja, hogy a hallgatók gyakorolják, hogyan kell adott struktúrájú kompenzáló tagot illeszteni az eredő szakasz körfrekvencia átviteli függvényéhez, és a kompenzáló tag illesztésének eredményeképpen a kompenzáló tag paramétereit megállapítani.

7.1. Ellenőrző kérdések

1. Jelölje meg, hogy az ábrázolt $G_0(j\omega)$ felnyitott hurokátviteli függvény Bode diagramján a nyilakkal megjelölt pontok közül a felsorolt jellemzők (fázistartalék, erősítéstartalék, hurokerősítés, és valami kakukktojás) hol olvashatók le!
2. Az ábra egy zárt szabályozási kör átmeneti függvénye. Az átmeneti függvényen bejelölt pontok közül a felsorolt jellemzők (maradó szabályozási eltérés, szabályozási idő, csúcsérték, túllendülés, felfutási idő, és valami kakukktojás) hol olvashatók le?

   Megjegyzés: Az 1. kérdés különböző hurokátviteli függvénnyel, a 2. kérdés átmeneti függvénnyel több verzióban fordul elő.

   Figyelem: A négy felsorolt jellemző közül csak kettő megfelelő az ábrákon bejelölt három ponthoz!
3. Válassza ki, hogy a felsorolt állítások közül mely, vagy melyek alkalmasak a PD kompenzálás jellemzésére!
4. Válassza ki, hogy a felsorolt állítások közül mely, vagy melyek alkalmasak a PI kompenzálás jellemzésére!

5. Válassza ki, hogy az ábrázolt átmeneti függvények, valamint a Bode vagy a Nyquist diagramok közül mely vagy melyek tartoznak a PI kompenzáló taghoz!
6. Válassza ki, hogy az ábrázolt átmeneti függvények és a Bode vagy a Nyquist diagramok közül mely vagy melyek tartoznak a PDT1 kompenzáló taghoz!

   Megjegyzés: Az 5. és a 6. kérdés különböző átmeneti függvénnyel, valamint Bode és Nyquist diagrammal több verzióban fordul elő.
7. A megadott időállandók egy $G_E(s)$ eredő szakasz időállandói. Melyiket választja a $G_C(s)$ PIDT kompenzáló elemben (T_I) integrálási időnek és (T_D) differenciálási időnek?

   Megjegyzés: Az 7. kérdés különböző időállandókkal vagy pólusokkal több verzióban fordul elő.
8. A felsorolt állítások közül mely vagy melyek igazak a szabályozási idő és a túllendülés kapcsolatára?
9. A $G_{yr}(s)$ alapjel átviteli függvény gyökeire a felsorolt állítások közül mely vagy melyek igazak?
10. A párhuzamos PDT1 kompenzáló tag Bode diagramján a nyilakkal megjelölt pontok közül mely vagy melyek szükségesek a K_C arányos erősítés meghatározásához?
11. A párhuzamos PDT1 kompenzáló tag Bode diagramján a nyilakkal megjelölt pontok közül mely vagy melyek szükségesek a T_D differenciálási idő meghatározásához?
12. A párhuzamos PDT1 kompenzáló tag Bode diagramján a nyilakkal megjelölt pontok közül mely vagy melyek szükségesek a DT1 ág egy tárolós összetevője T időállandójának meghatározásához?

13. A párhuzamos PI kompenzáló tag Bode diagramján a nyilakkal megjelölt pontok közül hol olvashatók le az arányos erősítés?

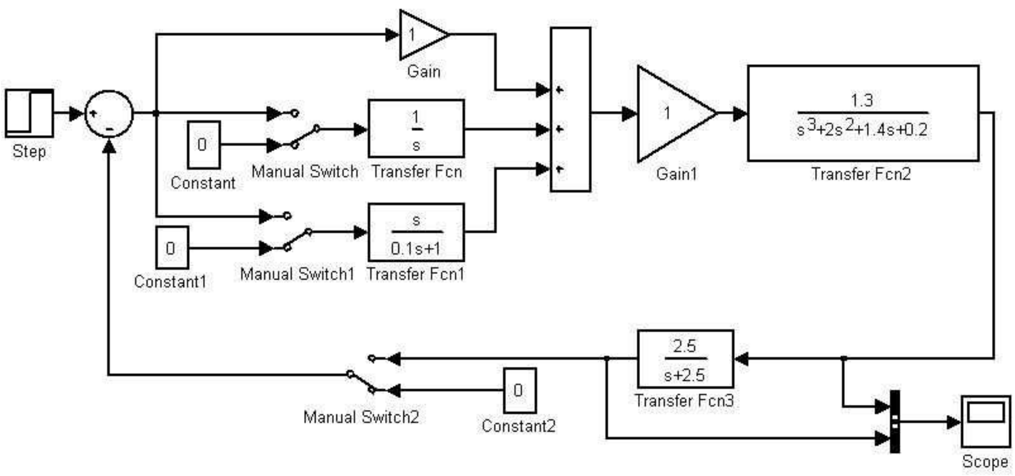
14. A párhuzamos PI kompenzáló tag Bode diagramján a nyilakkal megjelölt pontok közül hol olvashatók le az integrálási átviteli tényező?

15. Válassza ki, hogy a felsorolt szakkifejezések közül mely vagy melyek nem tartoznak a zárt szabályozási kör időtartománybeli minőségi jellemzői közé!

    Figyelem: A négy felsorolt jellemző közül lehet, hogy mind időtartománybeli minőségi jellemző, de lehet, hogy csak kettő!

7.2. Az első minta feladatsor: A $G_E(s)$ eredő szakasz átmeneti függvénye alapján történő kompenzálás

1. A jegyzőkönyv részére hozzon létre M fájlt és matlab_3Tx néven mentse, majd hozza létre a 7.1. ábrán látható SIMULINK modellt és matlab_3x néven mentse! Az x a megoldandó feladata betűkódja.



7.1. ábra A Simulink modell

2. Vegye fel az $G_E(s)$ eredő szakasz átmeneti függvényét és szerkesztéssel határozza meg a releváns közelítő HPT1 vagy IT1 jelátvivő tag paramé-

tereit! Határozza meg a szimuláció időtartamát! A szerkesztett képet mentse el!

3. Válasszon kompenzáló struktúrát a Melléklet 1. segítségével és indokolja az M fájlban a választását!
4. A választott kompenzáló struktúrát a Melléklet 2. segítségével úgy paraméterezze fel, hogy az egységugrás követése túllendüléstől mentes legyen! Az választott értékeket írja be a SIMULINK modellbe és az M fájlba egyaránt!
5. Zárja a szabályozási kört, hajtsa végre a szimulációt és állapítsa meg a szabályozási kör időtartománybeli minőségi jellemzőit! A szerkesztett képet mentse el, és az értékeket írja be az M fájlba is!
6. Indokolja az M fájlban, hogy melyik paramétert melyik irányba módosította az időtartománybeli minőségi jellemzők javítása érdekében. A szerkesztett képet mentse el és az értékeket írja be az M fájlba is!

Megjegyzés: A paraméter táblázatok (Melléklet 2.) ajánlásai az ideális HPT1 vagy IT1 jelátvivő tagra igazak. Minél nagyobb az eltérés a közelítő jelátvivő tag és az eredő szakasz átviteli függvénye között, annál kevésbé optimális a kompenzáló tag beállítása.

Megjegyzés: A lassú (perces időállandókkal rendelkező) rendszerek átmeneti függvénye méréssel könnyen meghatározható. Ezért ezt a módszert főleg az ipari technológiák üzembe helyezésekor alkalmazzák.

Megjegyzés: A MATLAB rendelkezik az eredő szakasz mérési eredményéből bonyolultabb szakasz (pl.: HIT2) közelítésre alkalmas identifikáló „Toolbox”-szal.

7.3. Az első minta feladatsor megoldása

1. A SIMULINK modell a 7.1. ábra.

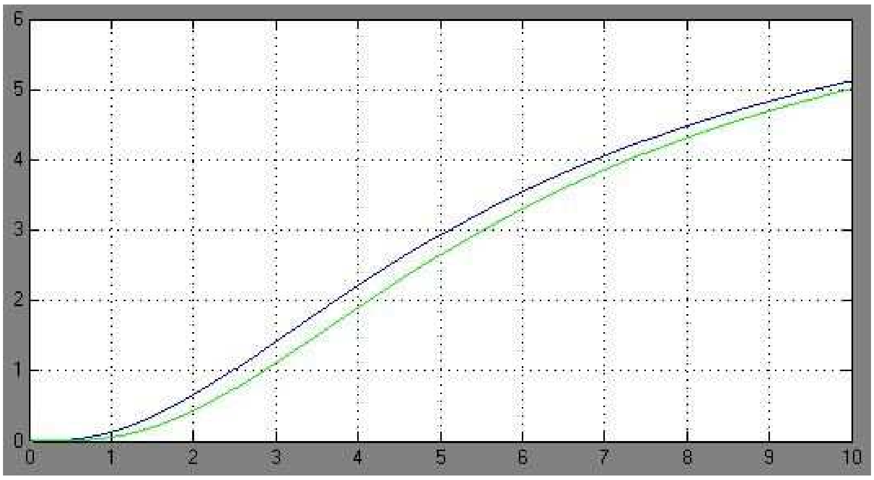
Fontos: Ügyeljen a „Step” és a Scope” grafikus objektumok paraméterezésére (Lásd: 4. fejezet)!

Megjegyzés: Ebben az elrendezésben az oszcilloszkóp „Workspace” területre mentett SD változójában az első oszlop az idő, a második a szabályozott jellemző és a harmadik az ellenőrző jel.

2. Ehhez a 7.1. ábra „Manual Switch” kapcsolóinak állása és a $K_C = 1$ érték éppen megfelelő. Az alapjelképző jele közvetlenül a PIDT szabályozó kimenetére, és így az eredő szakasz bemenetére kerül.

Fontos: A mérés problémája, hogy semmit sem tudunk a szükséges szimulációs időtartamról és a mintavételi időről. Ha egy önbeálló szakaszt túl rövid ideig vizsgálunk, akkor könnyen integráló jellegűnek tűnhet. Ezért az oszcilloszkóp segítségével több szimulációs időtartam mellett ellenőrizze az eredő szakasz átmeneti függvényét és igazítsa hozzá a mintavételi időt! A legmegfelelőbb eredő szakasz átmeneti függvényt használja a szerkesztéshez!

- Első lépésben maradjon a szimuláció időtartama 10 sec. és a mintavételi idő legyen 0.02 sec.
  Megjegyzés: Ez 501 mintavételi pontot eredményez, ezért ha kell a szimuláció időtartamát jelentősen módosítani, akkor módosítani kell a mintavételi időt.



7.2. ábra A szabályozott jellemző és az ellenőrző jel rövid szimulációs időtartammal

Emlékeztető: A „Scope” ábra megjelenése után használja az „Auto scale” ikont a koordinátaméretezésre!

Megjegyzés: A 7.2. ábrán az ellenőrzőjel a lassúbb, hiszen az tartalmazza a távadó időkésleltetését is.

A 7.2. ábra alapján valószínűsíthető az önbeálló szakasz jelleg. Célszerű a szimuláció időtartamát háromszorosra növelni. Ez jelentős változtatás, ezért a mintavételi időt is növeljük a háromszorosára. A „Scope” ábrán ellenőrizve a szimuláció időtartam és mintavételi idő ezen értéke megfelelő.

Fontos: A „Scope” ábra 5000 mintavételi pontot rajzol ki, de a **plot** parancs csak 1000 képes feldolgozni, ezért kell a mintavételi időt is megnövelni.

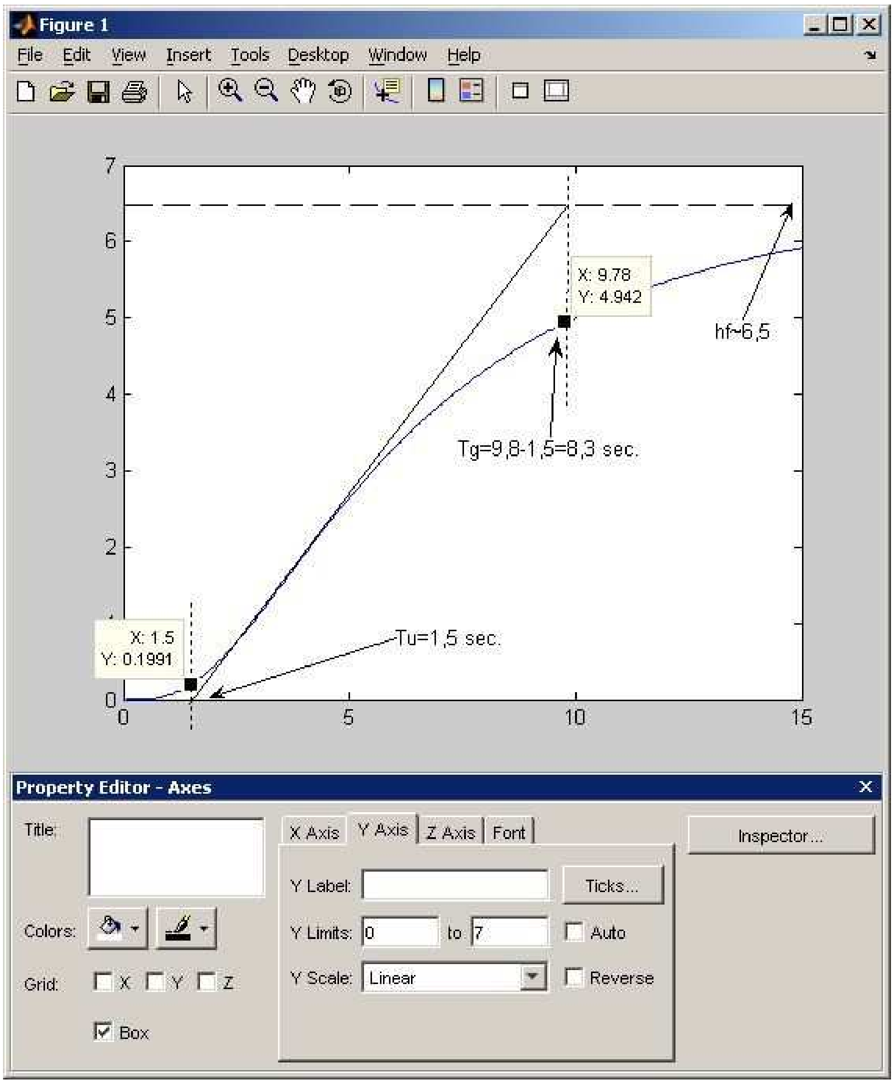
- Második lépésben kinyerjük az ellenőrzőjelet a lementett SD mátrixváltozóból, amelynek 3. oszlopa az ellenőrző jel. A 4.4. kifejezés szerint az SD mátrix 3. oszlopa:

```
>>  hge=SD(:,3);
```

- ♦ Harmadik lépés az eredő szakasz átmeneti függvénye:

```
>>  plot(tout,hge)
```

- ♦ Negyedik lépés a szerkesztés, aminek az eredménye a 7.3. ábra



7.3. ábra Az eredő szakaszt közelítő HPT1 jelátvivő tag paramétereinek meghatározása szerkesztéssel

Emlékeztető: A plot(tout,hge) parancs hatására 30 másodperc az x tengely végértéke.

Az eredő szakaszt közelítő HPT’ jelátvivő tag paramétereinek szerkesztését a kellő pontosság érdekében célszerű az alábbi sorendben végrehajtani.

A **plot** parancsot követően a vízszintes tengely 30 sec-ig van méretezve. Az eredő szakasz beáll a végértékére, ami az ismert módon a „Data Cursor” ikonnal meghatározható. A leolvasott értéket jelölje be az ábrán!

Ezután az „Insert” menüből a „Line” választással megszerkesztjük a végérték egyenest, majd a jobb egérgomb klikkre legördülő menüből a „Line Style” kijelölésével választhat szaggatott „dash”, pontozott „dot”, stb. vonalstílusok között.

A pontosabb szerkesztés érdekében méretezzük át az X tengelyt! Az „Edit Plot” ikon aktiválása mellett kettős bal egérgomb klikk hatására jelenik meg a „Property Editor -Axes” párbeszédablak (7.3. ábra).

Fontos: Az „auto” fülből vegye ki a pipa jelet!

Az átméretezett ábrán az inflexiós pont érintőjének szerkesztése úgy történik, hogy nagyjából letéve a vonalat („Insert” menüből a „Line”) a bal egérgombbal a vonal végei mozgathatók. Az érintő vonal egyik vége az X tengelyre, a másik vége a végérték egyenesre essen!

Ezt követően a „Data Cursor” ikonnal elhelyezhető két sárga információs ablak a görbén, majd a pontos helyet szerkesztő egyenesekkel kijelölve (ehhez az „Edit Plot” ikont kell ismét aktiválni) a sárga információs ablakok a kívánt pozícióba húzhatók (ehhez ismét a „Data Cursor” ikont kell aktiválni).

A sárga információs ablakból leolvasott értékeket a szokásos módon („Insert” menüből a „Text Arrow” választással) írja be az ábrába!

Az M fájl részlete:

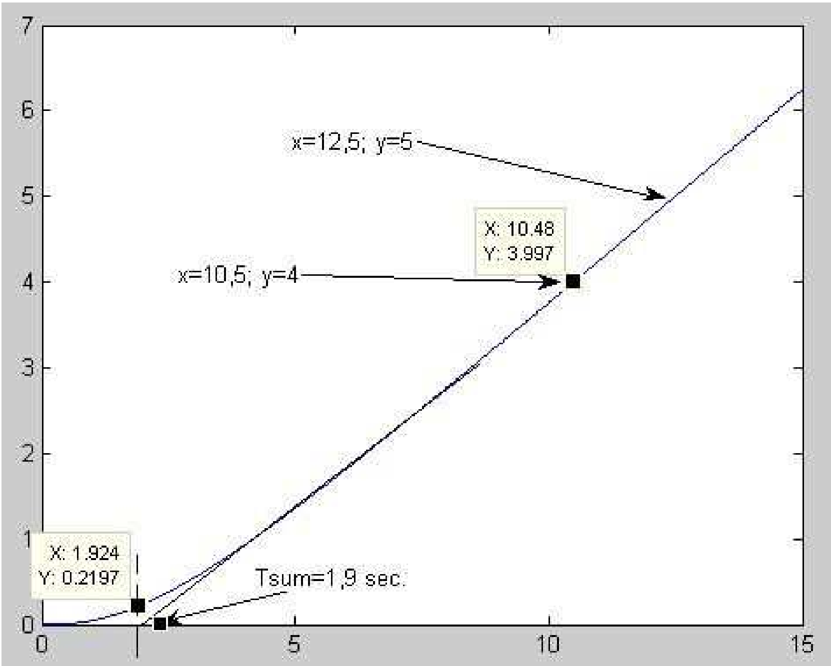
```
% Név Neptunkód Matla_3x
% 1. feladat
% A modell matlab_3x néven lett elmentve.
% A mátrixváltozóneve: SD

% 2. feladat
% szimuláció időtartama Tsum= 30 sec.
% mintavételi idő T0= 0,06 sec.
% Önbeálló szakasz; HPT1 közelítés
hge=SD(:,3);
plot(tout,hge)
% Az ábra: 1kep
% hf=6,5 ez az eredő szakasz Ke=6,5 erősítése.
% látszólagog holtidő Tu=1,5 sec
% látszólagos egy tároló Tg=8,3 sec
```

Kiegészítés

Integráló jellegű eredő szakasz esetén kismértékben eltér a szerkesztés menete. Példaképpen identifikáljuk a következő eredő szakaszt:

$$G_E(s) = \frac{1}{s^3 + 2.5s^2 + s}$$

Végrehajtva a 2. feladat első 3 lépését, szerkeszthetjük a **plot** paranccsal előállított ábrát, aminek az eredménye a 7.4. ábra



7.4. ábra Az eredő szakaszt közelítő IT1 jelátvivő tag paramétereinek meghatározása szerkesztéssel

A szerkesztés javasolt sorrendje:

A TI integrálási idő meghatározása az eredő szakasz átmeneti függvényének az állandó meredekségű szakaszára helyezett két pont (7.4. ábra) koordinátáiból lehetséges. A számítás menete azonos az 5. fejezet 1. mintafeladatában leírttal.

Ezután az „Insert” menüből a „Line” választással megszerkesztjük az átmeneti függvény egyenes szakaszának meghosszabbítását. A szerkesztő egyenes és az X tengely metszéspont jelöli ki a T_Σ látszólagos időállandó leolvasási helyét. Az X, Y koordináták értékei az átmeneti függvényre helyezett „Data Cursor” segítségével jeleníthető meg.

Emlékeztető: Először a „Data Cursor" információs ablakokat helyezze a görbe tetszőleges pontjaira, azután húzza be a szaggatott segédegyenest! (Ehhez az „Edit Plot" ikont kell ismét aktiválni.) Ezt követően a „Data Cursor" ikonnal helyezhető a sárga információs ablak a segédegyenes által kijelölt pontra.

3. A kompenzáló struktúra megválasztását a Melléklet M1 táblázata segíti. A 7.3. ábráról leolvasott értékekből az arány:

   $\frac{T_g}{T_u} = \frac{8,3}{1,5} \approx 5,5$, amihez az M1. táblázat PID struktúrát javasol.

   Az M fájl részlete:

```
% 3. feladat
% Tg/Tu=5,5. PID struktúrát választok
```

Kiegészítés

Integráló jellegű eredő szakasz esetén a P és a PDT kompenzáló struktúra felváltva vizsgálható a „Scope" képernyőjén!

Megjegyzés: Az M3. tábázat alapján az arányos tag azonos értékű, ezért ez nem jelent többletmunkát.

4. A kompenzáló paraméterek megválasztását a Melléklet M2. táblázatának az aperiodikus beállásra javasolt oszlopa segíti.

   $K_C = 0,6 \frac{1}{K_E} \frac{T_g}{T_u}$, számokkal: $K_C = 0,6 \frac{1}{6,5} \frac{8,3}{1,5} \approx 0,5$

   $T_I = T_g = 8,3$ sec.

   $T_D = T_u; = 1,5$ sec.

Mi nem PID, hanem PIDT struktúrát alkalmazunk, ezért legyen a DT ág egy tárolós összetevőjének időállandója.

$T = 0,1T_D = 0,15$ sec.

Ezeket az értékeket be kell írni a SIMULINK modellbe és a „Manual Switch" kapcsolókat a megfelelő állásba kell állítani.

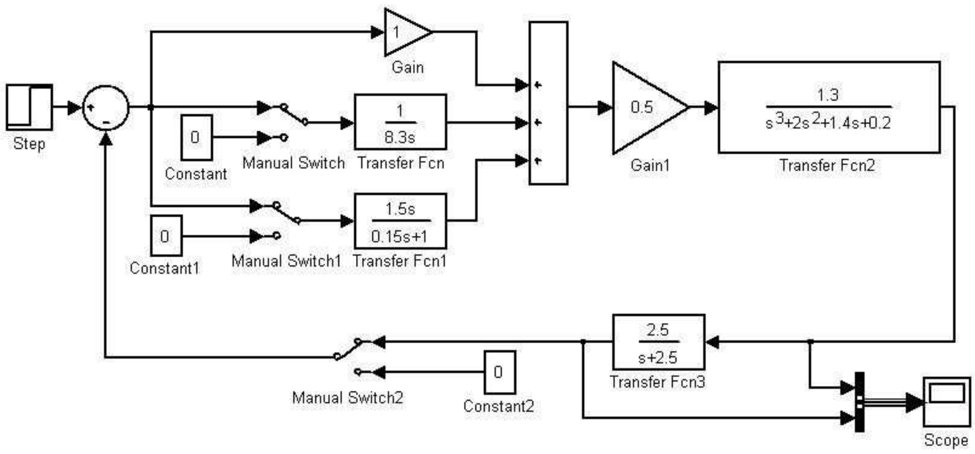
Az M fájl részlete:

```
% 4. feladat
% Kompenzáló paraméterek
% Kc=0,5; Ti=8,3 sec.; Td=1,5 sec.; T=0,15 sec.
```

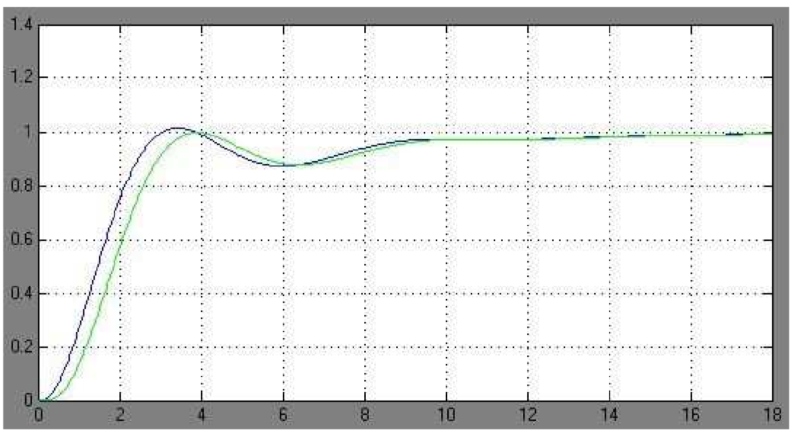
Kiegészítés

Integráló jellegű eredő szakasz esetén az M3. táblázatot használja! Ne válassza a PID kompenzáló struktúrát!

5. Írja be a PIDT kompenzáló tag kiszámolt értékeit a SIMULINK modellbe és állítsa a kapcsolókat a megfelelő állásba! Feltételezve, hogy a zárt szabályozási kör szabályozási ideje kevesebb, mint az eredő szakasz beállási ideje, a szimulációs időtartam legyen 18 sec, a mintavételi idő 0,03 sec! A modell a beállított értékekkel a 7.5. ábrán, az oszcilloszkóp képernyője látható a 7.6. ábrán láthatók.



7.5. ábra PIDT kompenzáló taggal a zárt szabályozási kör modellje



7.6. ábra PIDT kompenzáló taggal a zárt szabályozási kör átmeneti függvénye és ellenőrző jelének időbeli lefolyása

A 7.6 ábrán látott szimuláció eredmény alkalmas az időtartománybeli minőségi jellemzők meghatározására. A 7.6. ábrán az ellenőrző jel késleltetve követi az szabályozott jellemzőt, mert az eredő szakasz tartalmazza az ellenőrző jel késleltetését.

A 7.7. ábra kirajzolásához némi módosítással meg kellett ismételni a 2. feladat második és harmadik lépését:

- ♦ A második lépés a szabályozott jellemző mintasorozatát kinyerjük a lementett SD mátrixváltozóból.

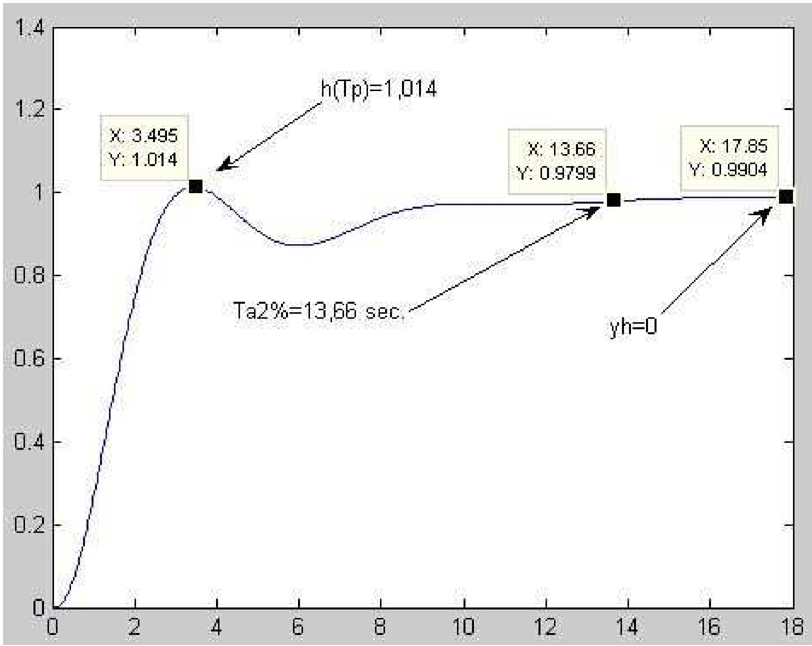
Fontos: A gyakorlatban csak az ellenőrző jel mérhető, de valójában a szabályozott jellemző viselkedése a fontos. Ezért a HPT1 közelítő modellt az ellenőrző jel, de a minőségi jellemzőket a szabályozott jellemző mintavételi értékeiből határoztuk meg!

Az SD mátrix 2. oszlopa az eredő szakasz átmeneti függvénye. A 4.4. kifejezés szerint:

```
>>  hgyr=SD(:,2)
```

- ♦ Harmadik lépésben a **plot** paranccsal kirajzoltatjuk a zárt szabályozási kör átmeneti függvényét:

```
>>  plot(tout,hgyr)
```



7.7. ábra A lementett 2kep

A **plot** parancs eredménye, a minőségi jellemzők a bejelölt leolvasási helyével, a 7.7. ábrán látható.

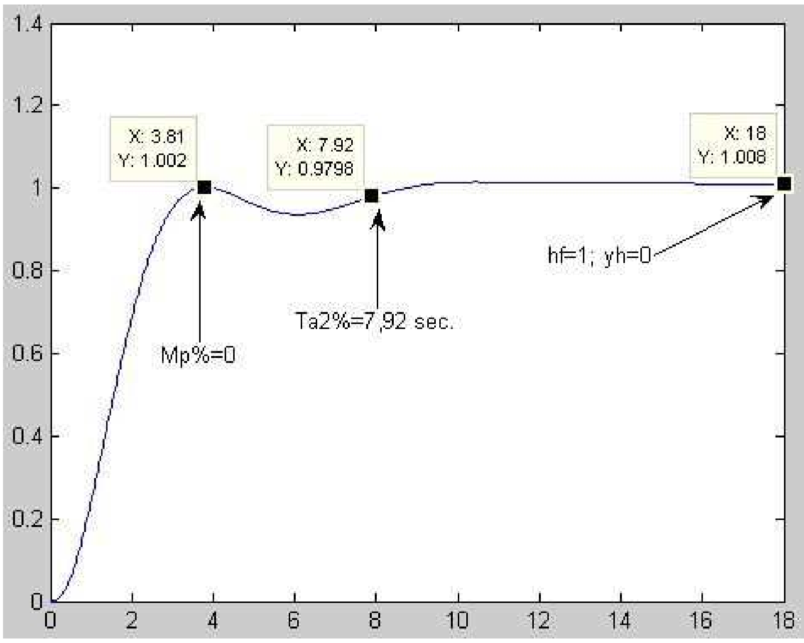
Az M fájl részlet

```
% 5. feladat
% szimulációs időtartam 18 sec.
% mintavételi idő 0,03 sec.
% a modell a beállitott értékekkel elmentve: matlab_3x2
hgyr=SD(:,2)
plot(tout,hgyr)
% Az ábra: 2kep
% yh=0; Ta2%= 13,7 sec,; Mp% = 1,5%
```

Kiegészítés

A zárt szabályozási kör - integráló jellegű eredő szakasz esetén is - önbeálló jellegű, ezért a minőségi jellemzők meghatározása az 5. feladatban bemutatottal teljesen azonos módon történik.

6. A hangolás eredménye a 7.8. ábrán látható.



7.8. ábra A lementett 3kep

Hangolási szempontok:

- A kompenzáló tag K_C erősítésének növelése növeli a túllendülés mértékét, csökkentése csökkenti.
- A kompenzáló tag T_I integrálási idejének csökkentése gyorsítja a szabályozási kört, de növelheti a lengéshajlamot. A differenciálási idő négyszeresénél kisebbre nem ajánlott csökkenteni.
- A kompenzáló tag T_D differenciálási idejének növelése gyorsítja a szabályozási kört, de növelheti a lengéshajlamot. Az integrálási idő negyedénél nagyobbra nem ajánlott növelni.

A gyakorlott mérnöki szem észreveszi a 7.7. ábrán, hogy az integráló hatás viszonylag lassan reagál, ezért a differenciálási idő négyszeresére csökkenthető az értéke. Legyen $T_I = 6$ sec!

A 7.7. ábrán nincs túllendülés, ez K_C érték megtartása mellett érv. Az integrálási idő csökkentése, viszont növeli a lengéshajlamot, ezért kis mértékben (K_C=0,45) csökkentsük a K_C értékét is.

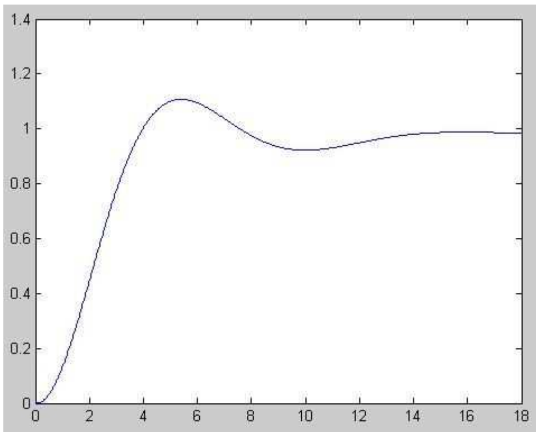
A hangolás eredményeképp mindegyik minőségi jellemző értéke jobb, mint az eredeti beállításokkal.

M fájl részlet:

```
% 6. feladat
% Alulról lassan áll be. Nem lengő: Ti csökkentése: 6 sec.
% Ti csökkentése miatt Kc kismértékű csökkentése: 0,45
% Az ábra: 3kep
% yh=0; Ta2%=7,9 sec,; Mp%=0
% Mindegyik paraméter jobb
```

Kiegészítés

Ha PIDT kompenzáló taggal a zárt szabályozási kör átmeneti függvénye a 7.9. ábrához hasonló, akkor a T_D differenciálási idő növelésével érdemes próbálkozni. (Gépkocsi hasonlattal: kevésbé vadul, de hosszabban nyomjuk kezdetkor a gázt.)



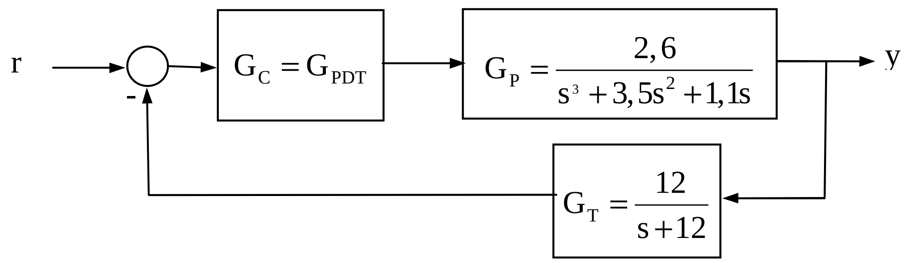
7.9. ábra PIDT kompenzálás túlzott DT hatással

Fontos: A K_C erősítés mellett vagy a T_D differenciálási időt vagy a T_I integrálási időt változtassa! Mind a kettő változtatása nem célravezető.

A „Melléklet 3. Kompenzálás pólus áthelyezéssel” fejezetben van példa a komplex gyökök kezelésére!

7.4. A második minta feladatsor: A G_E(s) eredő szakasz átviteli függvénye alapján történő kompenzálás

1. A jegyzőkönyv részére hozzon létre M fájlt és matlab_3Fx néven mentse! Az x a megoldandó feladata betűkódja.



7.10. ábra. A kompenzálni kívánt szabályozási kör

Hozza létre a 7.8. ábrán látható szabályozási kör G_E(s) eredő szakasz átviteli függvény Ge matlab változóját!

2. Ábrázolja a $G_{PDT}(s) = \frac{s+1}{0.1s+1}$ párhuzamos PDT kompenzáló tag Bode diagramját. Jelölje meg és írja be az M fájlba a legnagyobb pozitív fázistolás (φ_{max}) értékét! Ellenőrizze, hogy a legnagyobb pozitív fázistoláshoz tartozó $\omega(\varphi_{max})$ körfrekvencia értéket $\sqrt{A_D}$ -val elosztva az $\omega_D = \frac{1}{T_D}$ értéket és megszorozva az $\omega_T = \frac{1}{T}$ értéket kapja!

3. Ábrázolja a $G_E(j\omega)$ eredő szakasz körfrekvencia átviteli függvény Bode diagramját! Feltételezve, hogy a $G_0(j\omega)$ hurokátviteli függvénybe $\varphi_t = 60^o$ fázistartalékot akarunk, keresse meg az eredő szakasz Bode diagramján a leendő ω_C körfrekvencia értéket! Számítsa ki a T_D differenciálási idő és a T időállandó értékeit!

4. Hozza létre a $G_{PDT}(s) = \frac{sT_D + 1}{sT + 1}$ kompenzáló tag átviteli függvényét! (K_C továbbra is egy)

5. A $G_{PDT}(s) = \frac{sT_D + 1}{sT + 1}$ kompenzáló taggal hozza létre a $G_0(s)$ hurokátviteli függvény, és határozza meg a szükséges K_C erősítést!
6. Határozza meg a zárt szabályozási kör minőségi jellemzőit!

7.5. A második minta feladatsor megoldása

A feladatsor egy PD kompenzálás egymásra épülő feladataiból áll, Önbeálló eredő szakasz esetén PI kompenzálás lépéseiből áll a feladatsor.

1. A már ismert módon definiálhatók az átviteli függvények:

```
>> Gp=tf(2.6,[1 3.5 1.1 0])
```

Eredmény: Transfer function:

```
           2.6
-------------------------
s^3 + 3.5 s^2 + 1.1 s
```

```
>> Gt=tf(12,[1 12])
```

Eredmény: Transfer function

```
  12
-------
s + 12
```

A szabályozó (kompenzáló tag és a különbségképző) felöl látott eredő szakasz a fenti két árviteli függvény szorzata.

```
>> Ge=Gp*Gt
```

Eredmény: Transfer function:

```
                  31.2
----------------------------------------
s^4 + 15.5 s^3 + 43.1 s^2 + 13.2 s
```

2. A PDT kompenzáló tag paraméterei: $K_C = 1$, $T_D = 0{,}9$ sec., $T = 0{,}1$ sec.

Megjegyzés: Ezzel az értékválasztással lesz az $\omega_D = \frac{1}{\tau_D} = 1\,rad/\sec.$

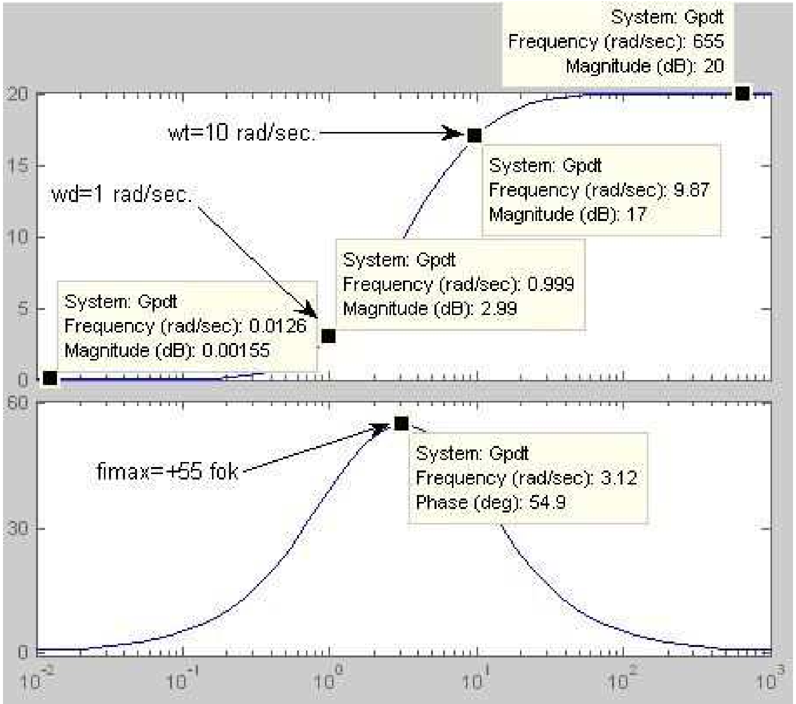
A PDT átviteli függvénye:

```
>> Gpdt1=tf([1 1],[0.1 1])
```

Eredmény: Transfer function

```
  s+ 1
---------
0.1s + 1
```

A Bode diagramja:



7.11. ábra A PDT kompenzáló tag Bode diagramja

Az 5.7. ábra magyarázata mutatta meg, hogy hogyan kell az értékeket megkeresni a Bode diagramon. A 7.11. ábra fázismenetén a legnagyobb pozitív fázistolás: $\varphi_{max} = 55^o$ és az ehhez tartozó körfrekvencia: $\omega(\varphi_{max}) = 3{,}12\,\mathrm{rad/sec}$.

Az ω_D, ω_T, és a $\omega(\varphi_{max})$ körfrekvenciák közötti összefüggés:

$$\omega(\varphi_{max}) = \sqrt{\omega_D \cdot \omega_T} = \frac{\omega_T}{\sqrt{A_D}} = \omega_D \sqrt{A_D} \tag{7.1}$$

ahol az $A_D = \dfrac{\omega_T}{\omega_D} = \dfrac{\tau_D}{T}$ a differenciális erősítés, ami a 7.11. ábrán 10 értékű. A $\tau_D = T_D + T$.

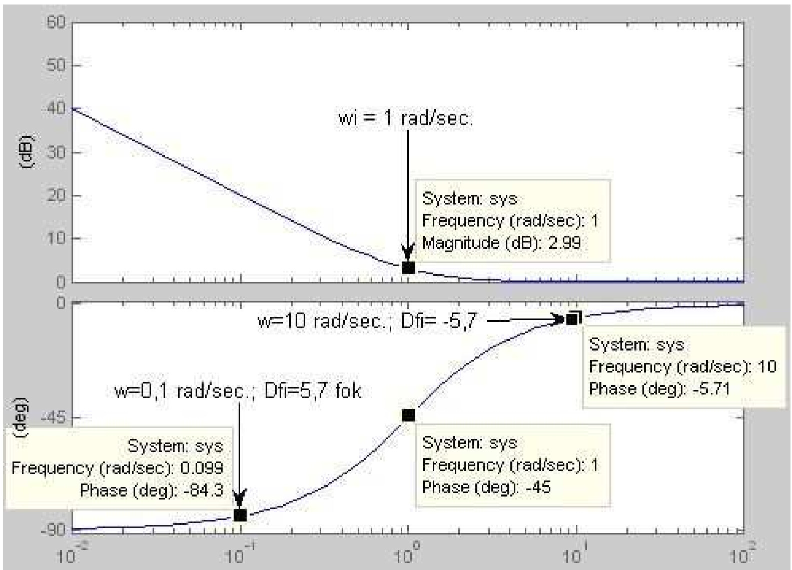
Megjegyzés: Az ipari kivitelű szabályozókban az A_D értéket 5 – 20 között lehet állítani, az áramkörtechnikában szokásos ennél nagyobb A_D értéket alkalmazni.

M fájl részlet:

```
% 2. feladat
Gpdt1=tf([1 1],[0.1 1])
bode(Gpdt)
% Az ábra: 1abra
% fimax=+55 fok; w(fimax)=3,16 rad/sec.; Ad=10
% w(fimax)*sqrt(Ad)=1 rad/sec.; wd=1 rsd/sec;
```

Kiegészítés

Önbeálló szakasz esetén PI kompenzálást valósítunk meg a feladatsorban, ezért a 2. feladat a $G_{PI}(s) = \dfrac{1 + j\omega}{j\omega}$ körfrekvencia átviteli függvény Bode diagramjának ábrázolása. (7.12. ábra).



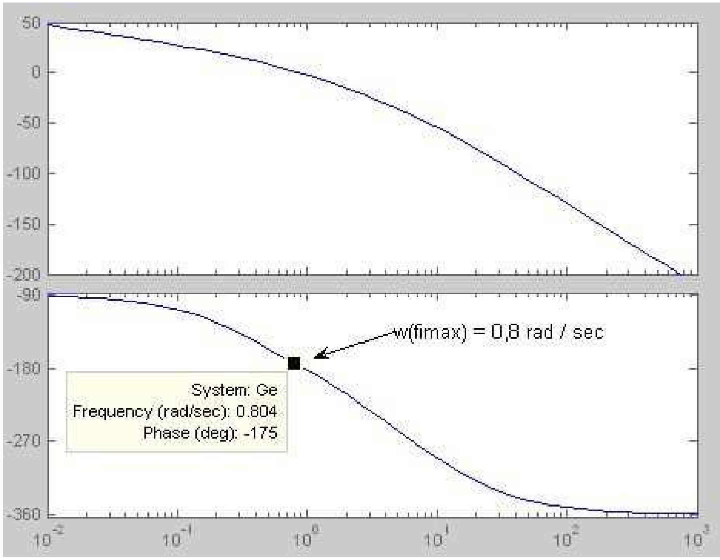
7.12. ábra A PI kompenzáló tag Bode diagramja

A 2. feladat kérdése az ω_I bejelölése, valamint az egy dekáddal nagyobb és az egy dekáddal kisebb körfrekvencia érékeknél a fázistolás eltérése ($\Delta\varphi$) a 0° illetve a -90° tól.

Emlékeztető: Az ω_I bejelölésének a helyét az 5.9. ábra magyarázatban találja.

3. A $\varphi_t = 60^o$ fázistartalék azt jelenti, hogy a leendő ω_C vágási körfrekvencián -120° lesz a tényleges fázistolás.

Ha az $\omega(\varphi_{max})$ körfrekvencia és a leendő ω_C vágási körfrekvencia egybeesik, akkor a $G_E(j\omega)$ eredő szakasz Bode diagramján (10-es differenciálási erősítésű PDT kompenzálás esetén) ezen a körfrekvencián -175° lehet a tényleges fázistolás. Ez az érték van bejelölve a 7.13. ábrán, ami a $G_E(j\omega)$ eredő szakasz Bode diagramja:



7.13. ábra Az eredő szakasz Bode diagramja

A 7.1. kifejezést felhasználva az $\omega_D = \dfrac{\omega(\varphi_{max})}{\sqrt{A_D}}$ és $\omega_T = \omega(\varphi_{max})\sqrt{A_D}$,

behelyettesítve az $\omega(\varphi_{max}) = 0,8$ értéket, a keresett körfrekvencia értékek:

$$\omega_D = \frac{0,8}{\sqrt{3,16}} = 0,25\,\mathrm{rad/sec}. \qquad \text{és } T_D = \frac{1}{0,25} = 4\,\mathrm{sec}. \tag{7.2}$$

$$\omega_T = 0,8 \cdot 3,16 = 2,52\,\mathrm{rad/sec}. \qquad \text{és } T = \frac{1}{2,52} = 0,4\,\mathrm{sec}. \tag{7.3}$$

Megjegyzés: Ha a két időállandó aránya (a kerekítés pontatlanságán belül) nem A_D értékű, akkor valamit elszámolt. Keresse meg a hibát!

M fájl részlet:

```
% 3. feladat
bode(Ge)
% Az ábra: 2abra
% w(fimax)=0,8 rad/sec; wd=0,25 rad/sec.; wt=2,52 rad/sec.
% Td=4 sec.; T=0,4 sec.
```

Kiegészítés

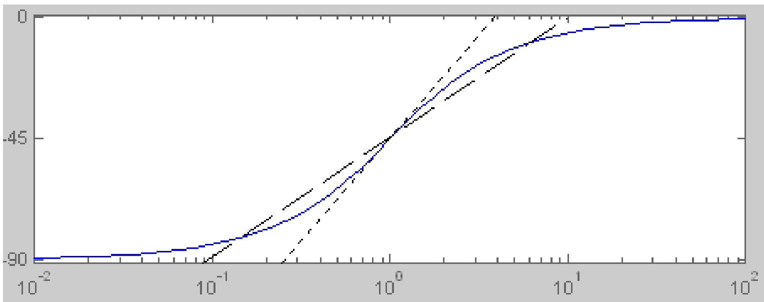
PI kompenzálás esetén a leendő ω_C vágási körfrekvencia a megadott $\varphi_t = 60^o$ fázistartalékhoz tartozó körfrekvencián lesz. Ezért a $G_E(j\omega)$ eredő szakasz Bode diagramján a $\varphi_t - 180^o = 60^o - 180^o = -120^o$ tényleges fázistoláshoz tartozó körfrekvenciát kell megkeresni. Ez a körfrekvencia érték lesz a leendő ω_C vágási körfrekvencia!

Az ω_I körfrekvencia értéknél 1 dekáddal nagyobb körfrekvencia legyen a leendő ω_C vágási körfrekvencia érték! Ezen a körfrekvencián a PI kompenzáló tag fázistolása már közel 0.

Megjegyzés: Ha a leendő ω_C vágási körfrekvencia keresésekor figyelembe veszi, hogy a PI kompenzáló tagnak a +1 D távolságra levő töréspontnál -5,7° fázistolása van, akkor még pontosabbá teheti a méretezést.

Ezzel a választással használhatjuk az $\omega_I = \dfrac{\omega_C}{10}$ közelítést. (lásd 7.12. ábra!) Így $T_I = \dfrac{1}{\omega_I}$.

Magyarázat. A Matlab III. mérés második feladataként végrehajtott PI kompenzálás a papír-ceruza kompenzálási módszer lépéseit követi.



7.14. ábra A regressziós és az inflexiós pontra illesztett érintő egyenes

A 7.14. ábrán a szaggatott vonal a fázismenetre illesztett regressziós egyenes, és a pontozott vonal a fázismenet inflexiós pontjára illesztett érintő egyenes. Papír-ceruza szerkesztéskor az egyenesek töréspontjai a 0°, és a -90° fázistolás tengelyein vannak.

A regressziós egyenes töréspontjai ± 1 D, az érintő egyenes töréspontjai ± 2/3 D távolságra vannak. Papír-ceruza szerkesztéskor a regressziós egyenes a jobb. Analitikus számításokhoz alkalmasabb az érintő egyenes.

4. A PDT kompenzáló tag paraméterei: $K_C = 1$, $T_D = 3{,}6$ sec., $T = 0{,}4$ sec.
   A PDT átviteli függvénye:

```
>> Gpdt=tf([4 1],[0.4 1])
```

Eredmény: Transfer function

```
 4s+ 1
---------
0.4s + 1
```

5. A $G_0(s)$ felnyitott hurokátviteli függvény a $G_E(s)$ eredő szakasz és a $G_{PDT}(s)$ kompenzáló tag szorzata

```
>> G0=Ge*Gpdt
```

Eredmény: Transfer function

```
                        124.8 s + 31.2
----------------------------------------------------------
0.4 s^5 + 7.2 s^4 + 32.74 s^3 + 48.38 s^2 + 13.2 s
```

Majd a Bode diagram:

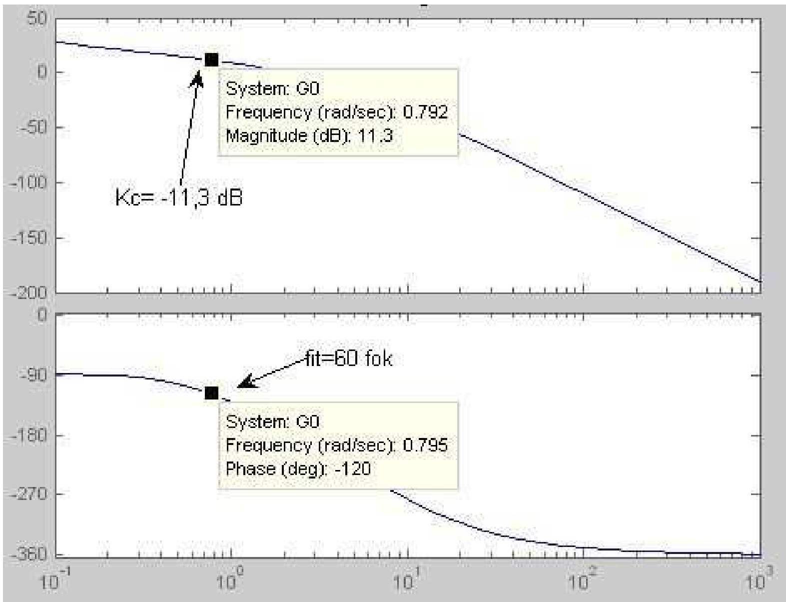
```
>> bode(G0)
```

Eredmény: A 7.15. ábra.

Miután a $G_0(s)$ felnyitott hurokátviteli függvény magába foglalja a $G_{PDT}(s)$ kompenzáló tagot, a Bode diagram fázismenetén a -120° tényleges fázistoláshoz tartozó körfrekvenciát kell megkeresni. Ez a ω_C vágási körfrekvencia, ahol $\varphi_t = 60^o$ a fázistartalék.

A 7.15. ábrán a leendő vágási körfrekvencia $\omega_C = 0{,}8$ rad/sec. Ezen a körfrekvencián az amplitúdó árvitel 11,3 dB. Ahhoz, hogy ez legyen a vágási körfrekvencia, a K_C értékének -11,3 dB-nek kell lennie.

$$K_C = 10^{\frac{-11,3}{20}} = 0,27$$



7.15. ábra A $G_0(s)$ felnyitott hurok Bode diagramja

A 4. és az 5. feladat M fájl részlete:

```
% 4. feladat
Gpdt=tf([4 1],[0.4 1])

% 5. feladat
G0=Gpdt*Ge
bode(G0)
% Az ábra: 3abra
% Az előírt fázistartaléknál w=0,8 rad/sec
% Az amplitúdó átvitel: 11,3 dB
% Kc=0,27
```

Kiegészítés

PI kompenzálás esetén a 4. feladatban természetesen a kiszámolt T_I integrálási idővel a $G_{PI}(s)$ kompenzáló tagot kell deklarálni.

Az 5. feladat szerkesztésének és a számításának menete ugyanez.

6. Először a végleges $G_C(s)$ kompenzáló tagot kell előállítani.

```
>> Gc=0.27*Gpdt
```

Eredmény: Transfer function

```
1.08 s + 0.27
-----------------
  0.4 s + 1
```

*Megjegyzés: A >>bode(Gc*Ge) parancsot begépelve a parancsablakba, a feljövő Bode diagramon gyorsan ellenőrizheti jobb egérgomb → „Characteristics” szolgáltatásaival, hogy az ω_C vágási körfrekvencián megvan-e a megadott $\varphi_t = 60^{\circ}$ fázistartalék. A kerekítések miatt biztosan lesz kb. 5° eltérés. Ha az eltérés meghaladja a 10°-ot, akkor valami hibát követett el.*

Ezt követően a zárt szabályozási kör $G_{yr}(s)$ alapjel átviteli függvényét

```
>> Gyr=feedback(Gc*Gp,Gt)
```

Eredmény: Transfer function

```
                    2.808 s^2 + 34.4 s + 8.424
-------------------------------------------------------------------
0.4 s^5 + 7.2 s^4 + 32.74 s^3 + 48.38 s^2 + 46.9 s + 8.424
```

Ezután a **step** paranccsal kirajzoltatjuk a zárt szabályozási kör átmeneti függvényét. Használva a **step** parancs jobb egérgomb → „Characteristics” szolgáltatásait, meghatározzuk a minőségi jellemzőket.

A zárt szabályozási kör (7.16. ábra) minőségi jellemzői:

Maradó szabályozási eltérés: $y_h = 1 - h(\infty) = 0$

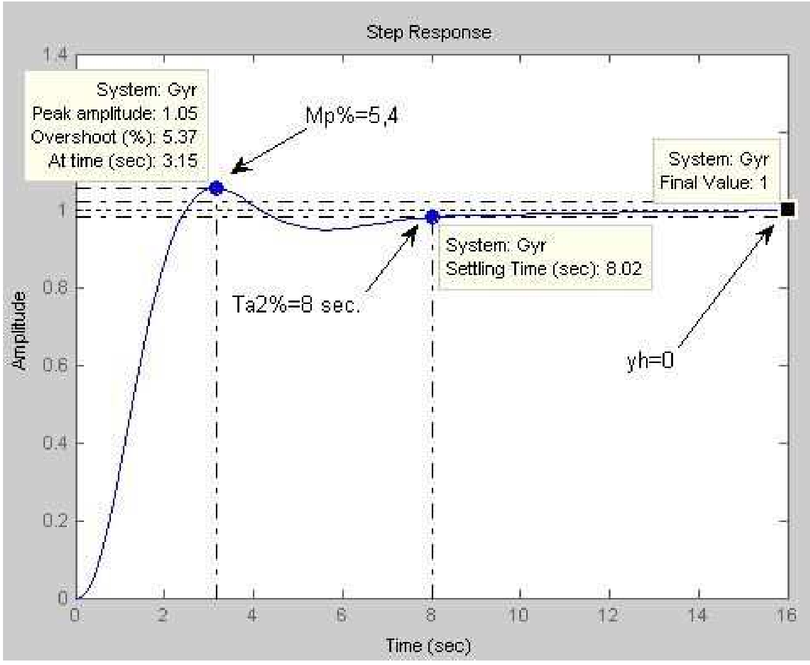
Szabályozási idő: $T_{a2\%} = 8$ sec.

Túllendülés: $M_{P\%} = 5{,}4$.

Ne feledje az értékek helyét az ábrába (7.16. ábra) beírni és az M fájlban is megadni!

M fájl részlet:

```
% 6. feladat
Gc=0.27*Gpdt
Gyr=feedback(Gc*Gp,Gt)
step(Gyr)
% Az ábra: 4abra
% yh=0; Ta2%=8 sec., Mp%=5,4
```



7.16. ábra A $G_0(s)$ felnyitott hurok Bode diagramja

Kiegészítés

A 6. feladat ugyanezen lépésekből áll PI kompenzálás esetén.

Hangolás

A $G_0(j\omega)$ felnyitott hurok körfrekvencia átviteli függvénnyel történő kompenzálásnak az a problematikája, hogy a zárt szabályozási kör időtartománybeli minőségi jellemzői a $G_0(j\omega)$ felnyitott hurok körfrekvencia átviteli függvény ω_C vágási körfrekvenciája és φ_t fázistartalék értékéből csak hozzávetőlegesen adható meg.

A $T_{a5\%}$ szabályozási idő az ω_C vágási körfrekvencia értékéből becsülhető:

$$\frac{\pi}{\omega_C} \le T_{a5\%} \le \frac{3\pi}{\omega_C}$$

A $M_{P\%}$ túllendülés a φ_t fázistartalék értékéből becsülhető:

- Ha a $G_0(j\omega)$ felnyitott hurok körfrekvencia átviteli függvénynek van domináns pólusa vagy póluspárja, akkor:

  $\varphi_t \approx 60^o$ esetén közel aperiodikus a zárt szabályozási kör.

$\varphi_t \leq 30^o$ esetén várhatóan 20%-nál nagyobb a zárt szabályozási kör túllendülése..

Megjegyzés: A domináns pólushoz vagy póluspárhoz tartozó időállandó legalább egy nagyságrenddel nagyobb a többi időállandónál.

- Ha a $G_0(j\omega)$ felnyitott hurok körfrekvencia átviteli függvény három legnagyobb időállandója egy dekádon belül van, akkor:

  $\varphi_t \approx 90^o$ esetén közel aperiodikus a zárt szabályozási kör.

  $\varphi_t \leq 45^o$ esetén várhatóan 20%-nál nagyobb a zárt szabályozási kör túllendülése.

  Megjegyzés: A mintafeladatban a $T_{a2\%}$ szabályozási időt határoztuk meg.

Mellékletek

Melléklet 1. Kompenzáló struktúra ajánlások az eredő szakasz átmeneti függvénye alapján

Az önbeálló szakaszok szabályozásakor a szabályozó jellegének megválasztására F. Piwinger adott ajánlást. Az ajánlás szakasz modelleken végzett nagyszámú empirikus vizsgálattal készült.

Az ajánlás a látszólagos egy tároló T_g és a látszólagos holtidő T_u arányában javasol kompenzálási struktúrát.

M1. táblázat: F. Piwinger ajánlás

P	PI	PID	I
$\frac{T_g}{T_u} \geq 50$	$50 \geq \frac{T_g}{T_u} \geq 7.4$	$7,4 \geq \frac{T_g}{T_u} \geq 3.3$	$3.3 \geq \frac{T_g}{T_u}$

A határok közelében próbálkozás kérdése, hogy melyik struktúra a megfelelőbb. Nincs jelentős eltérés a kétféle (pl.: 7,5 az arány PI vagy PID) struktúra minőségi jellemzői között. A D hatás gyorsítja a rendszert, de zavarérzékeny. A szabályozási kör üzemeltetésének körülményei döntenek az alkalmazhatóságáról.

Melléklet 2. Kompenzáló tag paraméterezése az eredő szakasz átmeneti függvénye alapján

Az M2. táblázat az **önbeálló szakasz**ok értékkövetésére optimalizált paramétereket ajánl. A további megkötések:

- Leggyorsabb (legkisebb T_a szabályozási idő) és aperiodikus (túllendülés nélküli) beállás.
- Leggyorsabb (legkisebb T_a szabályozási idő) és a túllendülés kevesebb, mint 20% ($M_{P\%} < 20$).

Az optimalizálás $K_P = 1$ amplitúdó átvitelű HPT1 szakaszközelítést feltételez. Ezért van az eredő szakasz tényleges K_P erősítése figyelembe véve az M2. táblázatban.

M2. táblázat: Chien-Hrones-Reswick ajánlása

	Leggyorsabb aperiodikus			Leggyorsabb lengő $M_p \le 20\%$		
		T_I	T_D		T_I	T_D
P	$K_C = 0,3 \frac{1}{K_P} \frac{T_g}{T_u}$	-	-	$K_C = 0,7 \frac{1}{K_P} \frac{T_g}{T_u}$	-	-
PI	$K_C = 0,35 \frac{1}{K_P} \frac{T_g}{T_u}$	1.2 T_g	-	$K_C = 0,7 \frac{1}{K_P} \frac{T_g}{T_u}$	-	-
PID	$K_C = 0,6 \frac{1}{K_P} \frac{T_g}{T_u}$	T_g	$0.5T_u$	$K_C = 0,95 \frac{1}{K_P} \frac{T_g}{T_u}$	$1.35T_g$	$0.47T_u$

Az **integráló jellegű szakasz**okra a Friedrich publikált ajánlást. Az optimalizálás IT1 szakaszközelítést feltételez. A vizsgált szakasz T_Σ látszólagos időállandóját és T_I integrálási idejét használja. További megkötés, hogy a zavargerjesztés az integráló hatás előtt éri a zárt szabályozási kört.

Az M3. táblázat az integráló szakaszok értéktartására optimalizált.

M3. táblázat: Friedrich ajánlása nem önbeálló szakaszokra

	Leggyorsabb lengő $M_p \le 20\%$		
		T_I	T_D
P	$K_C = 0,5 \frac{T_I}{T_\Sigma}$	-	-
PD	$K_C = 0,5 \frac{T_I}{T_\Sigma}$	-	$0.5T_\Sigma$
PID	$K_C = 0,4 \frac{T_I}{T_\Sigma}$	$3.2\ T_\Sigma$	$0.8T_\Sigma$

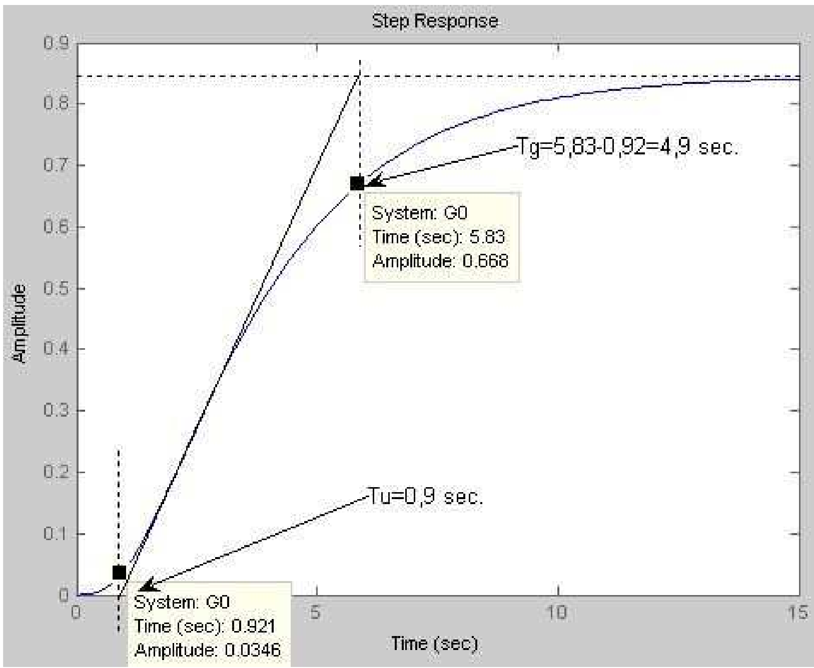
Fontos: Miután nem értékkövetésre optimalizált, értékkövetés (az alapjel változtatása) esetén ne válassza a PIDT kompenzáló struktúrát!

Melléklet 3. Kompenzálás pólusáthelyezéssel

A MATLAB II mérés mintafeladatában kimondatlanul már szerepelt a pólusáthelyezéses kompenzálási technika. A pólusáthelyezéses kompenzálási technika PIDT kompenzáló tagot feltételez és a kompenzáló tag átviteli függvényének zérusait tesz egyenlővé az eredő szakasz két legkisebb pólusával.

Miután bármely árviteli függvény gyökei abszolút értékének reciprokai az árviteli függvény időállandói, a lecserélés szabályai:

- A PIDT kompenzáló tag integrálási ideje legyen $T_I = T_{max}$, ahol T_{max} az eredő szakasz legnagyobb időállandója!
- A PIDT kompenzáló tag differenciálási ideje legyen $T_D = T_{max\,2}$, ahol T_{max2} az eredő szakasz második legnagyobb időállandója, és a tárolós tag időállandója, T legyen a differenciálási idő tizede!
- A PIDT kompenzáló tag erősítése úgy legyen meghatározva, hogy a hurokátviteli függvény $(G_0(s) = G_{PID}(s) \cdot G_E(s))$ rendelkezzen elegendő fázistartalékkal.



8.1. ábra A MATLAB II. mintapélda
eredő szakaszának átmeneti függvénye

Az eljárás csak akkor ad jó minőségi jellemzőket, ha az adott szakaszhoz az M1 táblázat PID kompenzáló tagot javasol.

A MATLAB II. mérés mintapélda eredő szakaszának átmeneti függvénye a szerkesztő vonalakkal a 8.1. ábrán látható.

A 8.1. ábráról leolvasott értékek alapján $\frac{T_g}{T_u} = \frac{4,9}{0,9} = 5,4$, amihez az M1 táblázat a PID struktúrát javasolja.

A mintapélda is igazolta, hogy a pólusáthelyezés még ez esetben sem szolgáltat közel optimális eredményt.

Nézzünk most egy másik mintafeladatot annál az esetnél, amikor komplex póluspár is van a gyökök között.

Legyen:

$$G_E(s) = \frac{28}{s^4 + 11s^3 + 41s^2 + 61s + 28} \tag{8.1}$$

A **roots** paranccsal határozzuk meg a gyököket!

```
>>   roots([1 11 40 58 25])
```

Eredmény:
```
-5.0768
-2.5539 + 0.4918i
-2.5539  - 0.4918i
-0.8153
```

Az időállandók meghatározásához a komplex gyökpár abszolút értékét is meg kell határozni.

```
>>   abs(-2.3953+0.7223i)
```

Eredmény: 2.6008

Az időállandók T_1 = 0,2 sec.; $T_2 = T_3$ = 0,38 sec.; T_4 = 1,23 sec.

A pólusáthelyezéses kompenzálás szabályai szerint:

T_I = 1,23 sec. és T_D = 0,38 sec.; valamint legyen T = 0,04 sec.

A kompenzáló tag K_C erősítése még nem ismert, ezért legyen 1. A három kompenzáló csatornaág összege szolgáltatja a PIDT kompenzáló tag $G_{PIDT}(s)$ átviteli függvényét:

```
>>   Gpidt=1+tf([1.23 1],[1.23 0])+tf([0.38 0],[0.04 1])
```

Eredmény:
```
Transfer function:
0.5658 s^2 + 2.5 s + 1
---------------------------
  0.0492 s^2 + 1.23 s
```

A $G_{01}(s) = G_{PIDT}(s) \cdot G_E(s)$ hurokátviteli függvény meghatározása érdekében először a Ge matlab változót kell definiálni:

```
>>   Ge=tf(28,[1 11 41 61 28])
```

Eredmény: Transfer function:

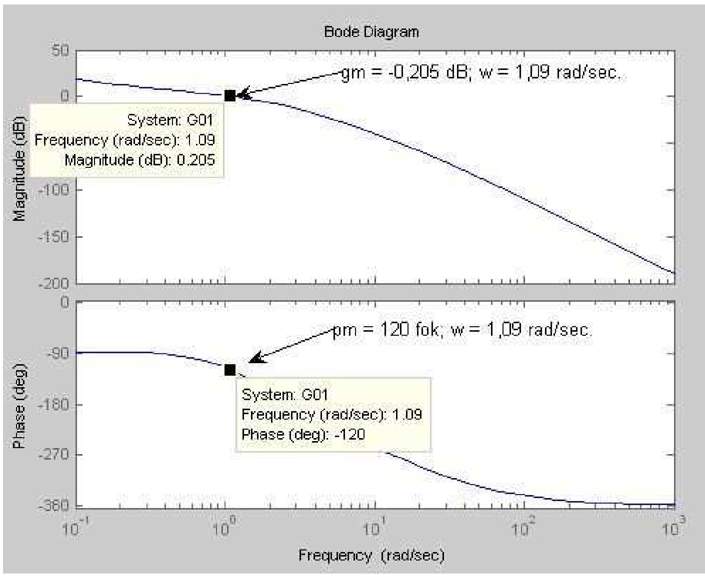
```
                28
-----------------------------------------
s^4 + 11 s^3 + 41 s^2 + 61 s + 28
```

```
>>   G01=Gpidt*Ge
                          15.84 s^2 + 70 s + 28
-----------------------------------------------------------------------------------
0.0492 s^6 + 1.771 s^5 + 15.55 s^4 + 53.43 s^3 + 76.41 s^2 + 34.44 s
```

A kompenzáló tag K_C erősítését $\varphi_t = 60^o$ fázistartalékhoz állítsuk be! A K_C erősítést a 8.2. ábrán látható $G_{01}(s)$ hurokátviteli függvény Bode diagramjáról leolvasott értékből számíthatjuk ki.

```
>>   bode(G01)
```

Eredmény:

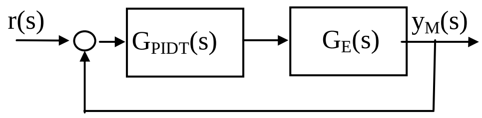


8.2. ábra A hurokátviteli függvény Bode diagramja

Átszámolva a dB értékből a K_C = 0,98, aminek az eltérése az egytől oly csekély a kompenzálási technika egyéb pontatlanságai miatt eltekinthetnénk a figyelembe vételétől, de a megoldás gondolatmenetének teljességéért vegyük most figyelembe.

Miután az eredő szakaszt ismerjük, ezért merev visszacsatolású a szabályozási kör (8.3. ábra).

Megjegyzés*: Ha a távadó nagyon gyors, az időkésleltetése elhanyagolható akkor igaz, hogy a $G_{ymr}(s)$ ~ $G_{yr}(s)$, vagyis az alapjel átviteli függvény jó közelítése a merev visszacsatolással számolt érték.*



8.3. ábra A szabályozási kör

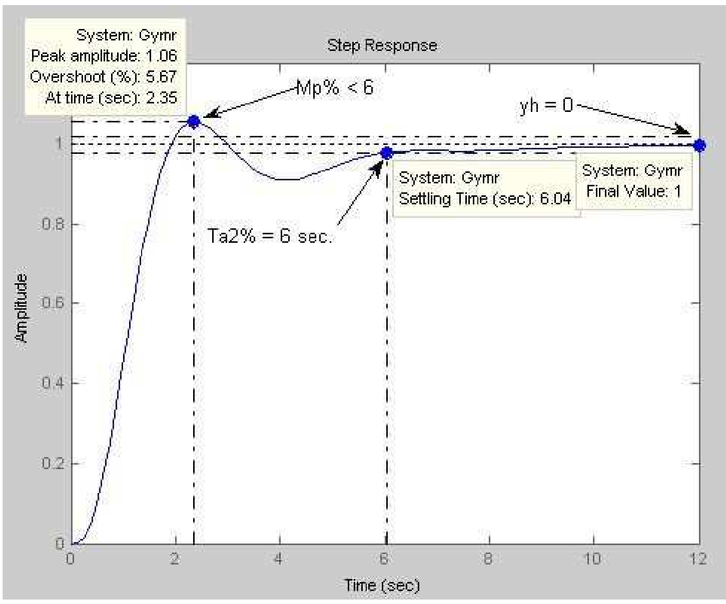
Merev visszacsatolás esetén a visszacsatoló ág átviteli függvénye 1. Az ismert **feedback** parancs szolgáltatja zárt szabályozási kör átviteli függvényét:

```
>> Kc=0.98;
>> Gymr=feedback(Kc*G01,1)
```

Transfer function:

```
                      15.37 s^2 + 67.9 s + 27.16
-------------------------------------------------------------------------
0.0492 s^6 + 1.771 s^5 + 15.55 s^4 + 53.43 s^3 + 91.77 s^2 + 102.3 s + 27.16
```

Az átmeneti függvény kirajzolása a **step** paranccsal történik.



8.4. ábra A mintapélda minőségi jellemzői

A 8.4. ábra jellege a 7.9. ábráéhoz hasonlatos, vagyis hangolási szempontból a T_D növelése lenne kívánatos. Azonban a $T_I/T_D < 4$, ami ennek erős korlátot szab. Összességében a $G_{PIDT}(s)$ kompenzáló tag paramétereinek hangolása már nem eredményez érdemi javulást.

Ha az eredő szakasz gyökei között van komplex póluspár, és a komplex póluspár abszolút értékének reciproka a legnagyobb és egyben a második legnagyobb időállandó, akkor rögtön hangolással kell kezdeni.

Ilyenkor a T_I integráló hatás időállandóját is és a T_D differenciáló hatás időállandóját is módosítani kell. Az időállandók aránya legfeljebb 4 legyen ($T_I/T_D \leq 4$), mert nagyon messze nem kerülhetnek az eredő szakasz legnagyobb és második legnagyobb időállandójától!

Megjegyzés: Ha a T_I integráló hatás időállandója és a T_D differenciáló hatás időállandója azonos értékű, akkor a hatásuk kioltja egymást, ezért nehéz jó kompromisszumot találni.