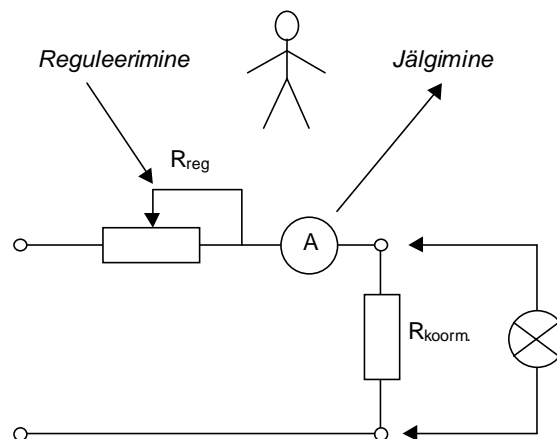


5. AUTOMAATJUHTIMISSÜSTEEMI PID-REGULEERIMINE

5.1. Reguleerimis põhimõtted

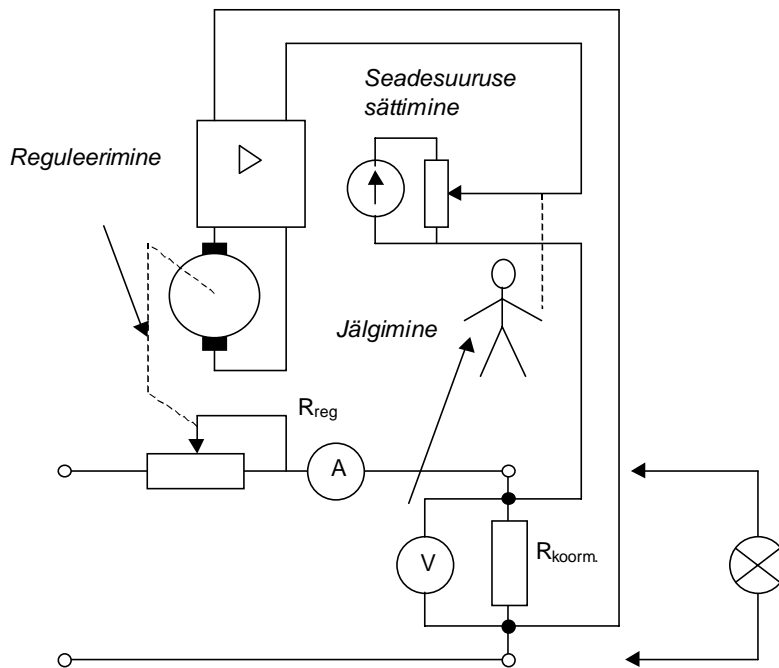
Reguleerimiseks nimetatakse süsteemi väljundsuuruse sihipärasest muutmist vastavalt etteantud kriteeriumile. Reguleerimine võib toimuda kas **käsitsi või automaatselt**. Esimesel juhul muudab reguleeritavat suurust inimesse, teisel juhul teeb seda etteantud kriteeriumile häälestatud **automaatregulaator**. Sõltuvalt väljundsuuruse mõjutamise iseloomust jaotatakse regulaatorid kas pidevatoimelisteks või diskreetse ehk katkelise toimega regulaatoriteks. Pidevatoimelise regulaatori näiteks on potentsiomeeter (vt. joonis). Liuguri asendi sujuval muutmisel muutub sujuvalt ka ahela vool ning koormustakistis eraldunud võimsus. Koormustakistiks võib olla ka näiteks hõõglamp. Sel juhul reguleeritakse potentsiomeetriga hõõglampi voolu ja järelkult ka heledust.

Käsitsi reguleerimise puhul jälgib inimene ahela tegelikku voolu ampermeetriga. Lambi heledust saab jälgida kas nõ. palja silmaga või samuti vastava mõõteriistaga. Seega toimib käsitsi reguleerimisel tagasiside inimese kaudu. Ning regulaatori osa etendab inimene.



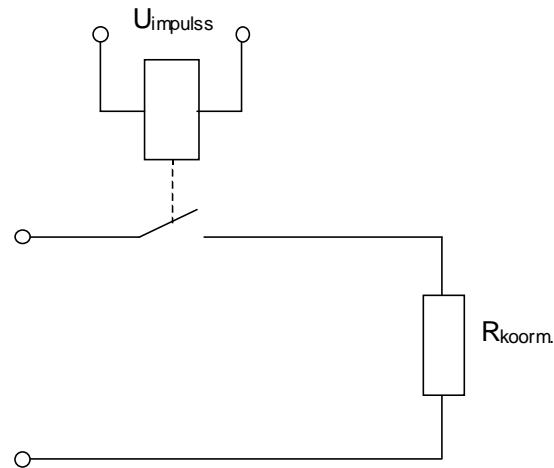
Joonis 5.1. Käsitsi reguleerimine

Automaatreguleerimisel võrreldakse väljundsignaali (pinget) sätitud seadesuurusega (etteandepingega). Kuna signaalide märgid on erinevad, sii lahutatakse seadesuurusest tegelik väljundsuurus. Signaalide vahet võib lugeda reguleerimisveaks. Veasignaal võimendatakse ning antakse täiturmootorile. Sõltuvalt vea märgist (kas hetkel on suurem seadesignaal või tegeli väljundsignaal) hakkab täiturmootor pöörlema ühes või teises suunas. Vastavalt sellele liigutatakse potentsiomeetri liugurit ning pinget suurendatakse või vähendatakse. Kui väljundpinge vähenemisel (näiteks koormuse suurenemise või toiteallika pinge vähenemise tõttu) toimib regulaator häiringutele vastupidiselt ja püüab pinget suurendades taastada endise pinge, on tegemist **negatiivse tagasisidega**

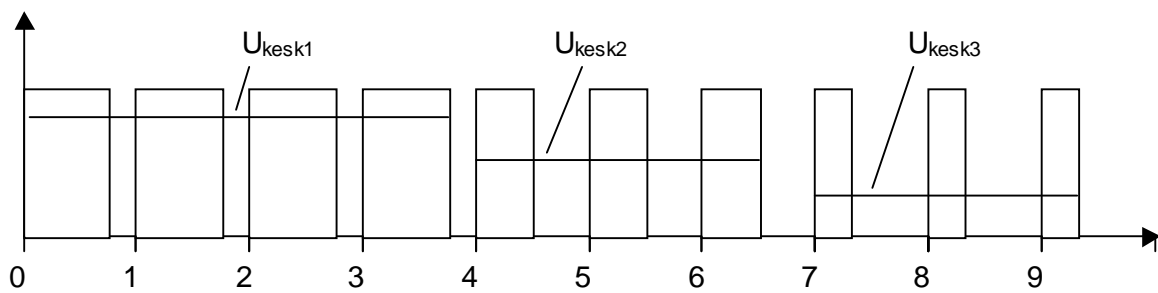


Joonis 5.2. Automaatreguleerimine

Katkelise toimega regulaatori töö põhineb ahela pideval kommuteerimisel, kusjuures reguleerimisel muudetakse ahela sisselülitatud ja väljalülitatud olekute suhtelist kestust. Koos sellega muutub ka koormusele rakendatud pinge ja ahela voolu keskväärtused (vt joonis.). Katkelist reguleerimist võib samuti teostada käsitsi, kuid nt. lambi heleduse reguleerimisel ei suuda inimene ahelat piisavalt kiiresti sisse ja välja lülitada ning lamp hakkab tugevasti vilkuma. Palju edukamalt võiks inimene käsitsi katkeliselt reguleerida nt. kütteradiaatori toitepinget ning selle kaudu toa temperatuuri. Kuna soojuslikud protsessid toimuvad aeglaselt ja toa temperatuur muutub samuti aeglaselt, siis võib lülitussagedus olla samuti väike. Tegelikus elus kasutatakse katkelise toimega regulaatoreid aga üha enam. Põhjuseks on asjaolu, et katkelisel reguleerimisel esinevad energiakaod on palju väiksemad kui pidevreguleerimisel. Kui oletada, et potentsiomeetriga reguleerimisel on potentsiomeetri takistus sama suur kui koormustakistus, siis eraldub potentsiomeetris kadudena sama suur võimsus kui läheb kasulikuks tööks koormustakistis. Katkelisel reguleerimiselt tekivad kaod ainult kommutatsiooniprotsessis. Kuna ahela lülitamine ehk kommutatsioon toimub kiiresti, siis on ka kaod väikesed. Eriti väikesed on kaod juhul kui ahela lülitamiseks kasutatakse kontaktivaabu pooljuhtelemente (transistore või türistore). Nüüdisaegsete jõupooljuhtide lülitusaeg on alla 1 mikrosekundi. Mõnel juhul võib seda aega mõõta isegi nanosekundites ($1 \text{ ns} = 10^{-9} \text{ s}$). Seega on kommutatsioonikadude puhul määravaks suuruseks kommutatsioonisagedus ehk mitu lülitust toimub ühe sekundi jooksul. Jõupooljuhtide lülitussagedused on tänapäeval vahemikus 1...100 kHz.



Joonis 5.3. Katkelise toimega reguleerimine



Joonis 5.4. Pulsilaiusreguleerimise põhimõte

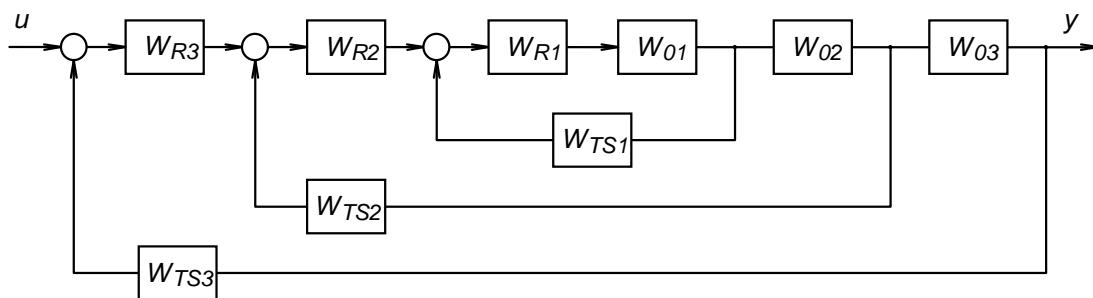
5.2. Alluvkontuuridega struktuurid

Automaatikaobjektide juhtimiseks on mõnikord otstarbekas rakendada alluvkontuuridega struktuure. Praktikas on niisugused struktuurid suure tähtsusega elektriainetes, kus mootori voolu, pöörlemiskiiruse ning võlli pöördenurga juhtimine on omavahel tihedalt seotud. Kõiki loetletud suurusi juhitakse omaette tagasisideahelate kaudu, mis moodustavad eraldi juhtimiskontuurid. Kontuurid paigutuvad kindlas järjekorras üksteise sisse ning seepärast on tegemist sisemiste ja välimiste kontuuridega. Minimaalselt kuulub alluvkontuuridega süsteemi kaks kontuuri, s. o. üks sisemine ja üks välimine kontuur. Kiiremate protsesside juhtimine toimub sisemistes kontuurides, aeglasemaid protsesse juhitakse aga välimiste kontuuridega. Alluvkontuuridega süsteemi struktuur on näidatud joonisel 3.12. Iga välimise kontuuri suhtes võib sisemise kontuuriga haaratavat süsteemiosa vaadelda kui juhtimisobjekti. Ajami korral on kiiruse juhtimisel objektiks voolukontuur, võlli pöördenurga asendi positsioonjuhtimisel aga kiiruskontuur. Seega allub voolukontuur kiiruskontuurile ning viimane omakorda asendi juhtimiskontuurile.

Alluvkontuuridega süsteeme on põhjalikult käsitletud elektriagamite alal. Sisuliselt võib alluvkontuuridega süsteeme vaadelda teatud üleminekuetapina klassikaliselt vea järgi juhtimiselt väljundiskaalal objekti mudeli järgi juhtimisele olekuruumis. Üksikute kontuuride väljundeid võib vaadelda olekumutujatena, nende kontuuride regulaatorid kajastavad aga vastavate juhtimisobjektide mudeleid.

Alluvkontuuridega struktuuri rakendamine elektriagami juhtimiseks võimaldas oluliselt lihtsustada juhtimissüsteemide projekteerimist ning häälestamist. Sõltumatult mootori ja töomasina parameetritest toimub kõigi alluvkontuuridega ajamite projekteerimine ja häälestamine ühiste reeglite järgi. Süsteemi sünteesil lähtutakse eeldusest, et struktuuri kõikide kontuuride ning terve süsteemi staatilised ja dünaamilised omadused on optimeeritud. Seepärast antakse alluvkontuuridega süsteemide korral ette nii üksikute kontuuride kui terve süsteemi optimeeritud ülekandefunktsioonid. Süsteemi kasutamisel eri objektide juhtimiseks, tagatakse süsteemi optimaalsus regulaatori ülekandefunktsiooni ja parameetrite valikuga. Kuna juhtimisobjekti ülekandefunktsioon ning parameetrid võivad muutuda suurtes piirides, peab ka regulaator võimaldama saavutada väga erinevaid dünaamilisi omadusi, s. t. regulaator kajastab (modelleerib) objekti dünaamilisi omadusi. Elektriagamites, kus juhtimisobjektiks on elektrimootor, kasutatakse selleks otstarbeks proportsionaalse, integraalse ja diferentseeriva ehk PID toimega regulaatoreid, mille parameetrid valitakse vastavalt konkreetsele mootorile ja töomasinale.

Ülekandefunktsiooni saab määrata alluvreguleerimisega AJS-is samade põhimõtete järgi, mida käsitleti eelmistes jaotistes. Konkreetse ülesande lahendamisel pole alluvreguleerimisega AJS-i ülekandefunktsiooni vaja alati määrata, sest optimeeritud kontuuride ülekandefunktsioon võib leida ka käsiraamatutest. Joonisel 3.12 on kasutatud järgmisi tähisteid: indeksid 1, 2 ja 3 tähistavad AJS-i kontuure; W_O on reguleerimisobjekti ülekandefunktsioon; W_R - regulaatori ülekandefunktsioon; W_{TS} - tagasisideahela ülekandefunktsioon. AJS-i esimese kontuuri (kinnise ahela) ülekandefunktsioon on [3]:



Joonis 5.5. Kolme alluvkontuuriga juhtimissüsteem

5.3. Reguleerimisprotsesside kvaliteedinäitajad

Automaatjuhtimissüsteemi analüüsil on oluline määrata süsteemi põhiomadused. Nendeks on süsteemi juhitavus ja jälgitavus. Seejärel uuritakse kas süsteem on stabiilne või ei ole.

Süsteem on **juhitav**, kui lõplikus ajavahemikus $t_a \leq t \leq t_l$ on olemas niisugune juhtsignaal $u(t)$, mis viib süsteemi kujutispunkti olekuruumi etteantud algolekust lõppolekusse või sellele piisavalt lähedale. Süsteem on **täielikult juhitav** kui on olemas juhtsignaal, millega saab viia süsteemi kujutispunkti suvalisest olekuruumi punktis koordinaatide alguspunkti. Süsteem on **osaliselt ehk mittetäielikult juhitav** kui juhtsignaal ei mõjuta kõiki süsteemi olekumuutujaid või kui osa olekumuutujaid ei mõjuta süsteemi väljundeid.

R. Kalman formuleeris süsteemi **juhitavuse kriteeriumi** sõltuvalt süsteemi juhitava osa diferentsiaalvõrrandi järgust. Süsteem on täielikult juhitav, kui süsteemi juhitava osa diferentsiaalvõrrandi järk ν võrdub tema olekuvõrrandi järguga n , s. t. $\nu = n$. Süsteem on osaliselt ehk mittetäielikult juhitav kui $0 \leq \nu < n$ ning süsteem on täielikult mittejuhitav kui $\nu = 0$.

Süsteemi jälgitavust iseloomustab tema väljundite ja olekumuutujate vaheline seos, kusjuures süsteem võib olla kas täielikult või osaliselt jälgitav. R. Kalmani järgi on süsteem täielikult jälgitav kui tema jälgitava osa diferentsiaalvõrrandi järk võrdub olekuvõrrandi (maatriksi A) järguga $\nu = n$. Süsteem on osaliselt ehk mittetäielikult jälgitav, kui $0 \leq \nu < n$ ning süsteem pole jälgitav kui $\nu = 0$.

Süsteemi stabiilsust käsitlev teooria põhineb matemaatikast tuntud diferentsiaalvõrrandite lahendite stabiilsusprobleemi uurimisel. Diskreetsete süsteemide korral on lahendid esitatavad arvujadadena ning süsteemi stabiilsust iseloomustab nende jadade koonduvus. Automaatjuhtimise klassikaliste meetoditega kontrollitakse süsteemi stabiilsust ülekandefunktsiooni ning selle tunnusvõrrandi järgi.

Kui süsteem osutub stabiilseks, siis järgneb süsteemi põhjalikum uurimine, mis seisneb süsteemi staatiliste ja dünaamiliste omaduste hindamises.

Süsteemi staatilise täpsuse all mõistetakse süsteemi väljundi vastavust sisendile püsitalitluses. Väljundi y erinevust sisendiga x määratud väärtusest nimetatakse süsteemi **püsitalitlusveaks**. Süsteemi dünaamilisi omadusi hinnatakse siirde- ja sagedustunnusjoonte järgi. Kuna siirdeprotsessi põhjal saab hinnata ka püsitalitlusviga, siis vaadeldakse järgnevalt peamiselt siirdeprotsesside arvutamise ja hindamisega seotud probleeme. Selleks tuleb arvutustega või katseliselt määrata AJS-i **siirdetunnusjoon**, mille kvaliteeti võrreldakse protsessi **soovitud kvaliteedinäitajatega**.

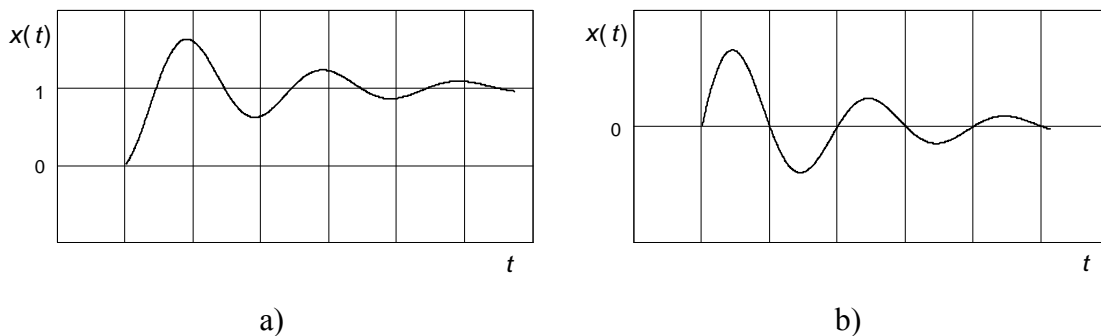
Lineaarsete süsteemide korral võib stabiilsuskontrolli ühitada siirdeprotsessi arvutusega ning hinnata süsteemi stabiilsust siirdetunnusjoone järgi või selle arvutustulemuste alusel.

5.3.1. Süsteemi stabiilsus

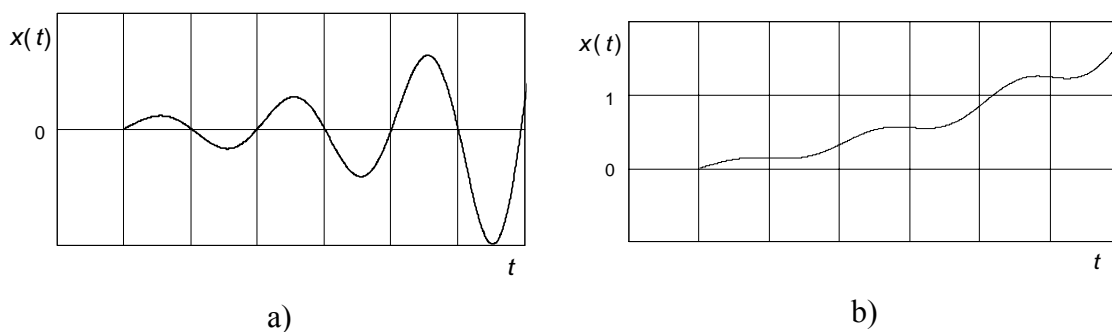
Automaatjuhtimissüsteemi dünaamikat iseloomustab tema tasakaalu- ehk püsioleku rikkumisele järgnev siirdeprotsess. Tasakaaluolekust kõrvalekaldumise põhjuseks võib olla sisendsignaalide, s. o. juhttoime või häiringute muutumine või muud põhjused (nt. süsteemi mõne parameetri muutus). Lineaarne AJS-i võib olla stabiilne või mittestabiilne.

1. Süsteem on **stabiilne** kui siirdeprotsessi järel teatud lõpliku ajavahemiku jooksul taastub AJS-i tasakaaluolek. Seejuures võib stabiilne süsteem tasakaalustuda kas uues olekus (olek 1 joonisel 5.6, a), kui tasakaaluolekust kõrvalekalde põhjus säilib, või esialgses olekus (olek 0 joonisel 5.6, b), kui tasakaaluolekust kõrvalekaldumise põhjus möödub või kui süsteem on antud sisendtoime suhtes staatiliselt invariantne.
2. Süsteem on **mittestabiilne** kui siirdeprotsessi jooksul tasakaaluolek ei taastu. Tasakaalu- ehk püsiolekust kõrvalekaldumisel võib tekkida olekumuutujate kasvava amplituudiga võnkumine (joonis 5.7, a) või kaugevad süsteemi olekumuutujad monotoonselt tasakaaluasendist (joonis 5.7, b).

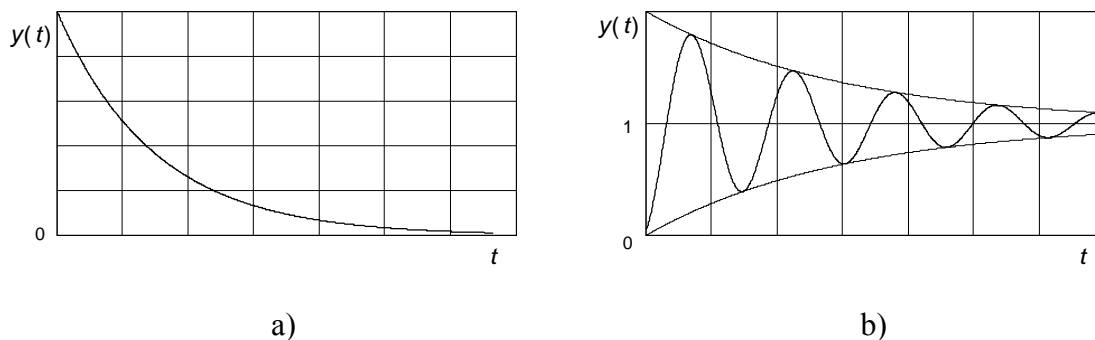
AJS-i stabiilsuse all mõistetakse süsteemi omadust automaatselt taastada tasakaaluolek pärast seda, kui väliste mõjutustega on süsteem tasakaalust välja viidud.



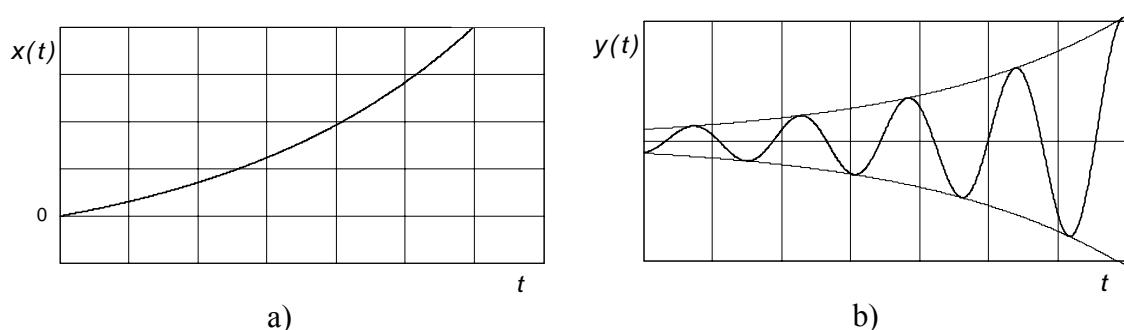
Joonis 5.6. Stabiilse süsteemi siirdeprotsessid tasakaaluolekust kõrvalekaldumisel. a) taastub uus tasakaaluolek, b) taastub endine tasakaaluolek



Joonis 5.7. Mittestabiilse süsteemi siirdeprotsessid
a) olekumuutuja hakkab kasvava amplituudiga võnkuma,
b) olekumuutuja kaugeneb (monotoonselt) tasakaaluasendist



Joonis 5.8. a) Aperiodiline siirdeprotsess, b) võnkuv siirdeprotsess



Joonis 5.9. a) Monotoonne mittestabiilne siirdeprotsess, b) võnkuv mittestabiilne siirdeprotsess

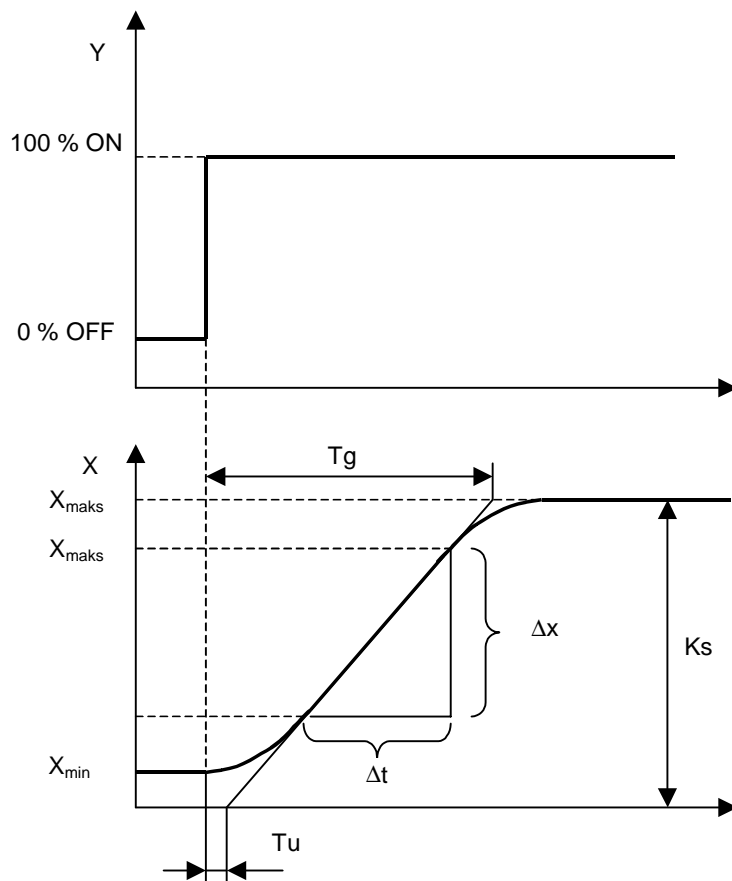
5.3.2. Siirdeprotsessi kvaliteedinäitajad

Enamik protsesse on nn kompenseeritud protsessid (vt. joonis. 5.10) Protsessi hüppekaja iseloomustavad hilistumine (dead time) T_u , siirdeprotsessi kestus (delay time) T_g ning muutuja maksimaalväärtus X_{max} . Neid muutujaid saab määrata geomeetrilise konstruktsiooni abil, mille käigus tõmmatakse siirdekõvra maksimaalse tõusuga osale puutuja alates maksimaalväärtusest kuni (point of inflection of step response). Paljudel juhtudel pole võimalik tõmmata puutujat kuni maksimaalväärtuseni, sest protsessimuutuja tegelik väärtus ei tohi pületada teatud piirväärtusi. Protsessi iseloomustab seepärast puutuja maksimaalne tõus

Protsessi juhitavust iseloomustab suhe $\frac{T_u}{T_g}$

Tabel 5.1

$\frac{T_u}{T_g}$	Protsessi juhitavus
$< 0,1$	Protsess on hästi juhitav
$0,1 \dots 0,3$	Protsess on juhitav
$> 0,3$	Protsess on halvasti juhitav



Joonis 5.10. Reguleerimisprotsessi siirdekaja ja selle kvaliteedinäitajad

Sõltuvalt nende muutujate väärtusest jaotatakse protsessid

- Kiireteks protsessideks, kui $T_u < 0,5$ min ja $T_g < 5$ min ning
- Aeglasteks protsessideks, kui $T_u > 0,5$ min ja $T_g > 5$ min

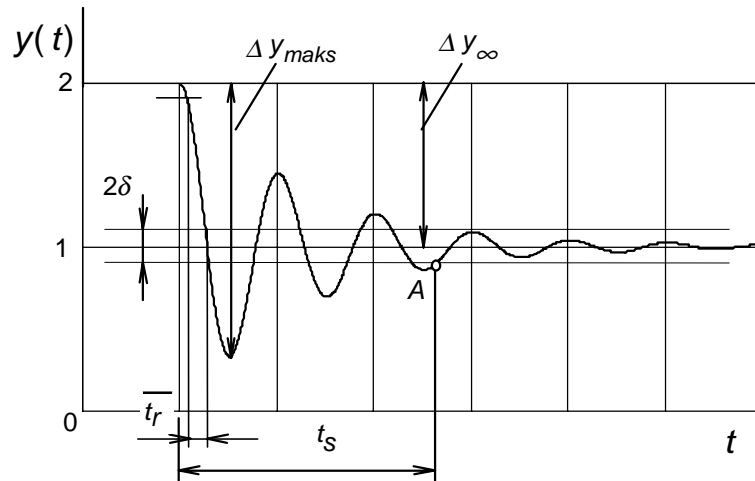
Süsteem on matemaatiliselt ehk formaalselt stabiilne, kui siirdeprotsess sumbub lõpliku ajavahemiku jooksul. Sellest aga ei järeldu, et iga stabiilne süsteem on sobiv praktiliseks kasutamiseks. Praktiliselt mittesobivad on nt. süsteemid, kus tasakaaluoleku taastamise kestus on suurem lubatavast või on siirdeprotsessi võnkumiste amplituud liiga suur. Üldjuhul on AJS kasutuskõlbulik, kui tema **siirdeprotsess on nõutava kvaliteediga**.

Siirdeprotsessi kvaliteedi iseloomustamiseks kasutatakse järgmisi põhinäitajaid (joonis 5.11):

- staatiline viga (*steady-state error*) ε ,
- esifrondi kestus (*rise time*) t_r ,
- siirdeprotsessi kestus (*settling time*) t_s ,
- maksimaalne ülereguleerimine (*peak overshoot*) ja sellele vastav aeg (*time-to-peak overshoot*) t_p ,
- võnkuvus ehk poollainete arv võnkuva siirdeprotsessi kestel.

Joonisel 5.11 toimub siire tasakaaluolekust (püsiolekust) 2 tasakaaluolekusse 1. Tekkiv siirdeprotsess on võnkeline ning selle kestus on t_s .

Staatiline viga (*steady-state error*) ε iseloomustab süsteemi täpsust ja on siirdeprotsessi väljakujunenud püsioleku ning juhttoimega määratud püsiolekute vahe. Süsteeme, mille staatiline viga on võrdne nulliga nimetatakse **astatilisesteks süsteemideks**.



Joonis 5.11. Siirdeprotsessi (hüppekaja) kvaliteeti iseloomustavad näitajad

Esifrondi kestuseks (*rise time*) t_r loetakse tavaliselt ajavahemikku, mis kulub hüppekaja muutumiseks vahemikus (0,1...0,9) Δy_∞ . Peale selle määratluse kasutatakse ka teisi viimasega sarnaseid määratlusi.

Siirdeprotsessi kestus (*settling time*) t_s on ajavahemik, mille kestel

$$\frac{\Delta y(t) - \Delta y_\infty}{\Delta y_\infty} \cdot 100\% > \delta, \quad (5.1)$$

kus $\Delta y(t)$ on hüppekaja hetkväärtus (kõrvalekalle esialgse tasakaaluoleku suhtes); Δy_∞ - hüppekaja püsiväärtus (uue ja esialgses püsioleku erinevus); δ - etteantud suurus, mille abil määratakse siirdeprotsessi kestus. Joonisel 5.10 on näidatud punkt A, kust edasi siirdeprotsess kulgeb etteantud vahemikus $\pm\delta$. Enamasti võetakse $\delta = 5\%$ hüppekaja püsiväärtusest. Aperiodilise protsessi korral on siirdeprotsessi kestus sel juhul $t_s = 3\tau$, kus τ on protsessi ajakonstant.

Siirdeprotsessi esifrondi- ja kogukestus iseloomustavad süsteemi toimekiirust. Viimane on suure tähtsusega neis süsteemides, mille töös etendavad suurt osa dünaamilised protsessid, nt. käivitus- ja pidurdusprotsessid. Niisugusteks süsteemideks on näiteks pikihöövelpingid, kus siirdeprotsess moodustab olulise osa kogu töösüklist. Süsteemide toimekiiruse suurendamine ja ühtlasi siirdeprotsessi kestuse lühendamine aitab oluliselt suurendada masinate tootlikkust.

Maksimaalne ülereguleerimine (*peak overshoot*) σ on määratav seosega

$$\sigma = \frac{\Delta y_{maks} - \Delta y_{\infty}}{\Delta y_{\infty}} \cdot 100\%, \quad (5.2)$$

kus Δy_{maks} on vaadeldava suuruse maksimaalne kõrvalekalle.

Lubatav ülereguleerimine määratakse konkreetseid tootmisprotsessi tingimusi arvestades. Mitmete raamatute andmeil on ülereguleerimine erinevates süsteemides vahemikus $\sigma = 4,3...53$ %. Näiteks, maksimaalne ülereguleerimine $\sigma = 4,3$ % on kehtestatud mooduloptimumile häälestatud alluvkontuuridega süsteemi esimese kontuuri kohta, $\sigma = 53$ % vastab aga süsteemi kontuuri sümmeetrilisele optimumile.

Võnkuvus ehk poollainete arv siirdeprotsessi kestel (ajavahemiku t_s kestel) näitab, mitu korda protsessi kestel hüppekaja hetkväärtus ületab etteantud püsiväärtuse. Joonisel 5.10 näidatud hüppekaja puhul on poollainete arv 5. Nagu lubatav ülereguleerimine, määratakse ka soovitatav poollainete arv siirdeprotsessi kestel, konkreetseid tootmisprotsessi tingimusi arvestades.

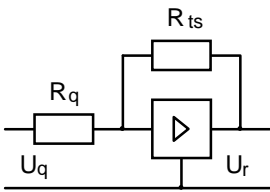
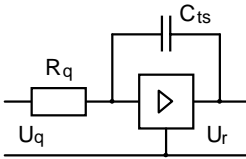
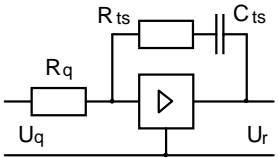
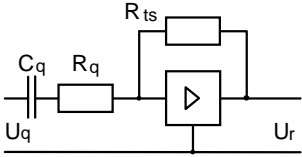
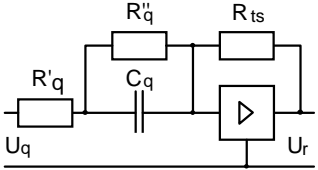
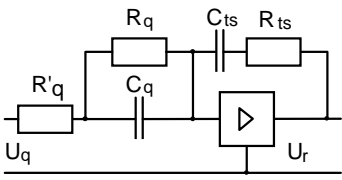
Siirdeprotsessi kvaliteeti saab hinnata kas eksperimentaalselt määratud või arvutatud hüppekaja järgi, kusjuures hüppekaja arvutamiseks kasutatakse nii analüütilisi, numbrilisi kui ka kaudseid lihtsustatud meetodeid.

Analüütiliseks meetodiks on Laplace teisendusel põhinev hüppekaja arvutamise operaatormeetod. Meetod on rakendatav kõigi lineaarsete süsteemide korral, mille diferentsiaalvõrranditel on olemas üldlahend. Analüütiline meetod võimaldab kirjeldada hüppekaja analüütilise avaldisega.

Numbrilised meetodid põhinevad diferentsiaalvõrrandite asendamises diferentsvõrranditega ning nende lahendamises mitmete eri meetoditega. Kõige lihtsamaks diferentsiaalvõrrandi numbrilise lahendamise meetodiks on Newton-Euleri meetod, mis seisneb diferentsiaalade asendamises lõplike juurdekasvudega. Seda meetodit saab edukalt rakendada ka programmeeritavatel kalkulaatoritel. Täpsemateks, kuid suuremat arvutusmahtu nõudvateks meetoditeks on Runge-Kutta meetodid. Tänapäeval kuuluvad need meetodid mitmesuguste matemaatikapakettide (MathCAD või MATLAB) koosseisu.

5.4. Automaatjuhtimissüsteemides kasutatavad regulaatorid

Tabel 5.2

Reg. tüüp	Operatsioonivõimendiga regulaatori skeem	Ülekandefunktsioon $W(p)$	Parameetrid
P		$-k_{rq}$	$k_{rq} = \frac{R_{ts}}{R_q}$
I		$-\frac{1}{\tau_r p}$	$\tau_r = C_{ts} R_q$
PI		$-\beta_{rq} \frac{\tau_r p + 1}{\tau_r p}$	$\beta_{rq} = \frac{R_{ts}}{R_q}$ $\tau_r = R_{ts} C_{ts}$
D		$-\frac{\tau_r p}{\tau'_r p + 1}$	$\tau_r = C_q R_{ts}$ $\tau'_r = C_q R_q$ $R_q \ll R_{ts}$
PD		$-k_{rq} \frac{\tau_r p + 1}{\tau'_r p + 1}$	$k_{rq} = \frac{R_{ts}}{R'_q + R''_q}$ $\tau_r = C_q R''_q$ $\tau' = C_q \frac{R'_q R''_q}{(R'_q + R''_q)}$
PID		$-\beta_{rq} \frac{(\tau_{r1} p + 1)(\tau_{r2} p + 1)}{\tau_{r1} p (\tau'_{r2} p + 1)}$	$\beta_{rq} = \frac{R_{ts}}{(R'_q + R''_q)}$ $\tau_{r1} = C_{ts} R_{ts}$ $\tau_{r2} = C_q R''_q$ $\tau'_{r2} = C_q \frac{R'_q R''_q}{(R'_q + R''_q)}$

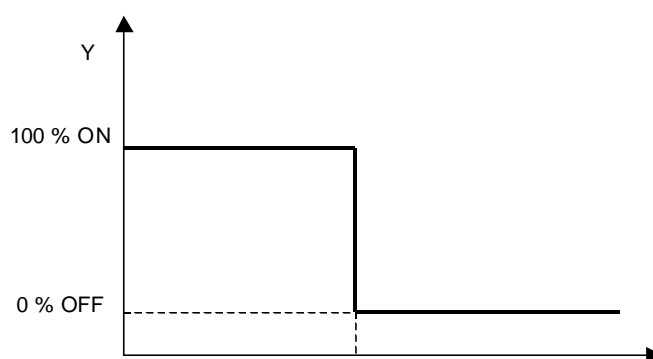
Regulaatori struktuuri valik sõltuvalt juhitavast suurusest

Tabel 5.3

Protsessimuutuja	Regulaator			
	P	PD	PI	PID
	Staatiline süsteem (püsitalitlusviga)		Astaatiline süsteem (püsitalitlusviga puudub)	
Temperatuur	Üksikjuhtimisel sobiv kui $T_u/T_g < 0,1$	Sobib hästi	Kõige paremad regulaatorid nõudlike rakenduste puhul (v. a. erijuhtumid)	
Rõhk	Hilistumise puudumisel sobiv	Mittesobiv	Kõige paremad regulaatorid nõudlike rakenduste puhul (v. a. erijuhtumid)	
Vooluhulk	Vähesobiv	Mittesobiv	Sobiv, kuid I-regulaator võib olla parem	

Releetoimeline binaarregulaator (*two step controller*)

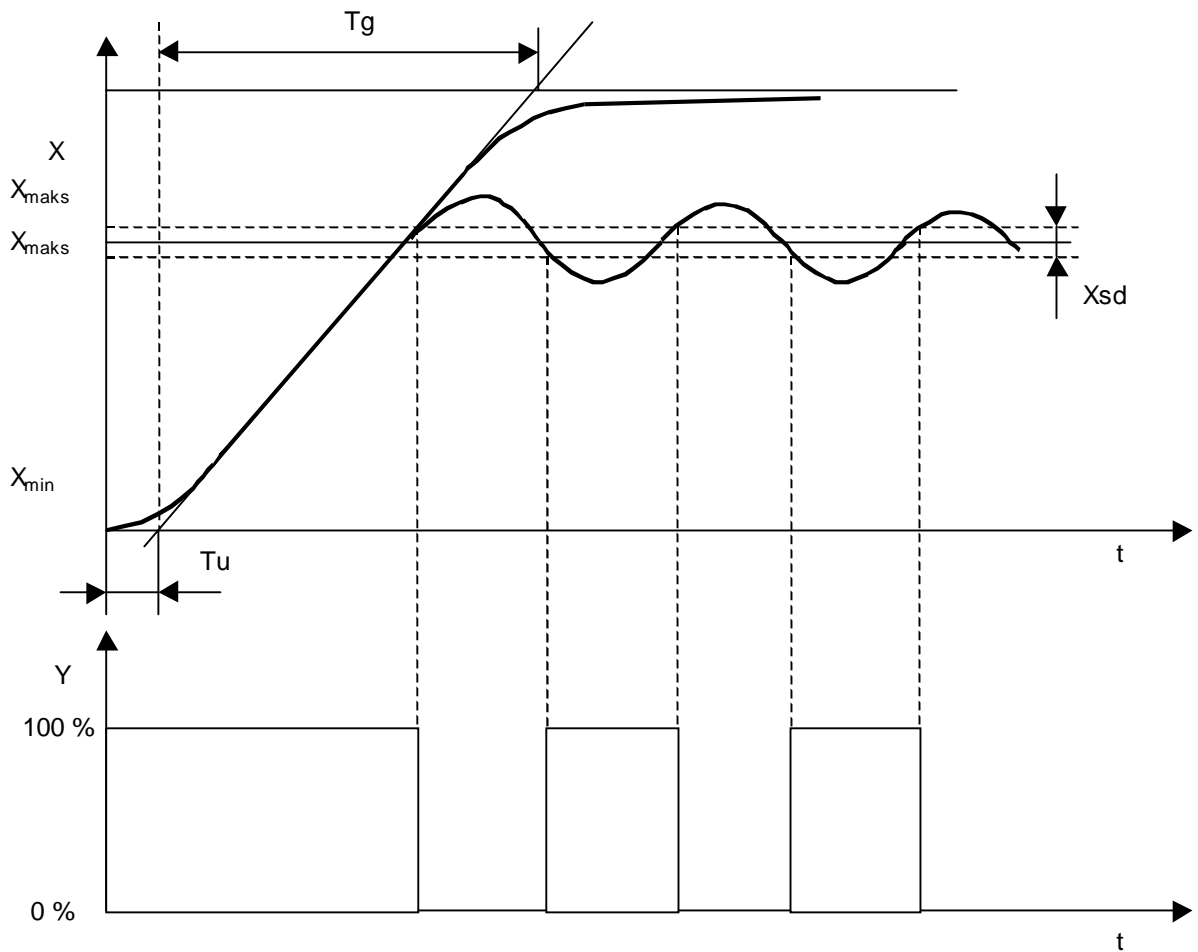
Binaarregulaatoril on kaks olekut (ON ja OFF), mis vastavad 100 % või 0 % signaalile. Reguleerimisel toimub reguleeritava suuruse X võnkumine sätteväärtuse W ümber. Võnkumiste amplituud ja periood on seda suuremad, mida suurem on süsteemis hilistumisaja T_u ja ajakonstandi T_g suhe. Niisuguseid regulaatoreid kasutatakse peamiselt lihtsates temperatuuri reguleerimise süsteemides (nt. elektriküttekehad, -ahjud, -radiaatorid).



Joonis 5.11. Releetoimeline kaheastmeline binaarregulaator

Tagasisidega binaarregulaatorid

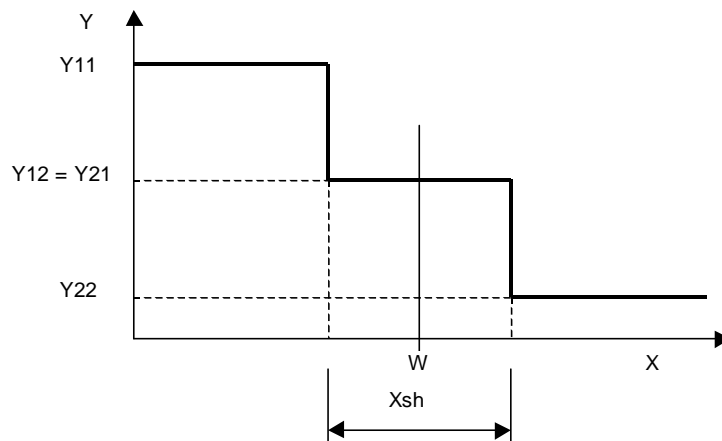
Binaarregulaatori omadusi saab parandada (nt. suure hilistumisega ahjude puhul) elektrooniliste tagasiside abil. Tagasisidet kasutatakse lülitussageduse suurendamiseks, mis aitab vähendada ka võnkumiste amplituudi. Lisaks sellele paranevad juhtimisomadused dünaamilises talitluses. Lülitussagedus võib olla piiratud väljundmooduliga. Mehhaanilise täiturite puhul (nt. reelede ja kontaktorite puhul) ei tohiks lülitussagedus olla suurem kui 1...% lülitust minutis. Suuremaid lülitussagedusi võimaldavad pooljuhtkommutaatorid (nt. IGBT transistoridel või türistoridel lülitid). Suure lülitussageduse tõttu pole väljundsuuruse võnkumine enam märgatav ning reguleerimisel saadakse pidevreguleerimisega võrreldavad tulemused.



Joonis 5.12. Tgagasisideta binaarregulaatori talitus.

Kui pidevreguleerimise korral on regulaatori väljundis pidevsignaali, siis tagasisidega binaarregulaatori puhul on regulaatori juhttoimeks pulsilaiusmoduleeritud signaal.

Kolmeastmelised regulaatorid. Kolmeastmelisi regulaatoreid kasutatakse kütte- ja jahutus- ja kliimasüsteemide juhtimisel. Samuti ka plastipresside temperatuuri reguleerimisel. Nendel regulaatoritel on väljundis kaks lülitit. Regulaatori talitlust saab optimeerida elektrontagasiside abil.



Joonis 5.13. Kolmeastmeline binaarregulaator

5.5. Arvregulaatorite numbriline modelleerimine

Traditsiooniliselt kasutatakse ajamites operatsioonivõimendite baasil ehitatud proportsionaalseid, integraalseid ja diferentsiaalseid ning mitmesuguseid kombineeritud regulaatoreid ehk lühendatud P, I, D, PI, PD ja PID regulaatoreid. Raaljuhtimisekorral on nende analoogideks vastavad arvregulaatorid. Arvregulaatorite sisend- ja väljundfunktsioone esitatakse diskreetsete väärtuste jadana, kus muutujate hetkväärtused on fikseeritud ajaintervalli järel. Funktsiooni tuletisteks aja järgi on vastavat järku diferentsfunktsioonid, integraalideks aga summafunktsioonid. Regulaatori väljundfunktsioon iseloomustab väljundsuuruse sõltuvust sisendsuurusest.

Näiteks, pidevatoimelise proportsionaalregulaatori ehk regulaatori tööd kirjeldab väljundfunktsioon

$$U_{reg} = K_p \cdot \Delta U(t), \quad (5.3)$$

kus K_p on regulaatori ülekandetegur, $\Delta U(t)$ sisendsignaali. Diskreetse regulaatori korral tuleb pidev aeg asendada diskreetse ajaga n , mis kujutab endast järjestikulise ajaintervallide järjekorranumbrit.

$$U_{reg} = K_p \cdot \Delta U(n) \quad (5.4)$$

Negatiivse tagasisidega süsteemis kujutab regulaatori sisendsuurus $\Delta U(n)$ endast seadesignaali ja tagasisidesignaali vahet.

$$\Delta U(n) = U_s(n) - U_{ts}(n) \quad (5.5)$$

Diskreetse regulaatori väljundfunktsioon

$$U_{reg}(n) = K_p \cdot [U_s(n) - U_{ts}(n)] \quad (5.6)$$

Analoogiliselt on avaldatav ka integraal- ehk I regulaatori väljundfunktsioon

$$U_{reg}(n) = K_i \cdot \sum_{j=0}^n [U_s(j) - U_{ts}(j)] \quad (5.7)$$

Ajamite juhtimisel leiavad kõige enam kasutamist PI regulaatorid, millel on nii kui ka regulaatori omadused. Diskreetse PI regulaatori väljund funktsioon

$$U_{reg}(n) = K_p \cdot [U_s(n) - U_{ts}(n)] + K_i \cdot \sum_{j=0}^n [U_s(j) - U_{ts}(j)], \quad (5.8)$$

kus võrrandi parempoolse avaldise esimene liige vastab P regulaatorile, teine aga I regulaatorile.

Diskreetse diferentsiaalregulaatori ehk D regulaatori tööd kirjeldab väljundfunktsioon

$$U_{reg}(n) = K_d \cdot [[U_s(n) - U_{ts}(n)] - [U_s(n-1) - U_{ts}(n-1)]]. \quad (5.9)$$

Kui regulaatori sisendis toimiv seadesignaal ei muutu, siis avaldub regulaatori väljundfunktsioon järgmiselt:

$$U_{reg}(n) = -K_d \cdot [U_{ts}(n) - U_{ts}(n-1)]. \quad (5.10)$$

Viimasest võrrandist järeldub, et regulaatori väljundsignaal ei sõltu seadesignaalist ning on määratud ainult tagasisidesignaali muutumise kiirusega.

P, I ja D regulaatorite ühendamisel saadakse PID regulaator, mille väljundfunktsiooni võib esitada kolme liidetava summana.

$$U_{reg}(n) = K_p \cdot [U_s(n) - U_{ts}(n)] + K_i \cdot \sum_{j=0}^n [U_s(j) - U_{ts}(j)] + K_d \cdot [U_{ts}(n) - U_{ts}(n-1)] \quad (5.11)$$

PID regulaator häälestatakse võimendustegurite K_p , K_i ja K_d valikuga. On võimalik luua regulaator, mille väljundfunktsioonis on ülekaalus üks selle komponentidest, kas P, või regulaator, kuid võib ka erinevate regulaatorite toimed omavahel tasakaalustada. Siirdeprotsesside kiirendamiseks (forsseerimiseks) tuleb suurendada K_p või K_d . Süsteemi staatiline viga väheneb aga K_i ja K_p suurendamisel. Teiselt poolt on võimendustegurite suurendamisel oht, et süsteem muutub mittestabiilseks ning seal tekivad sumbumatud isevõnkumised.

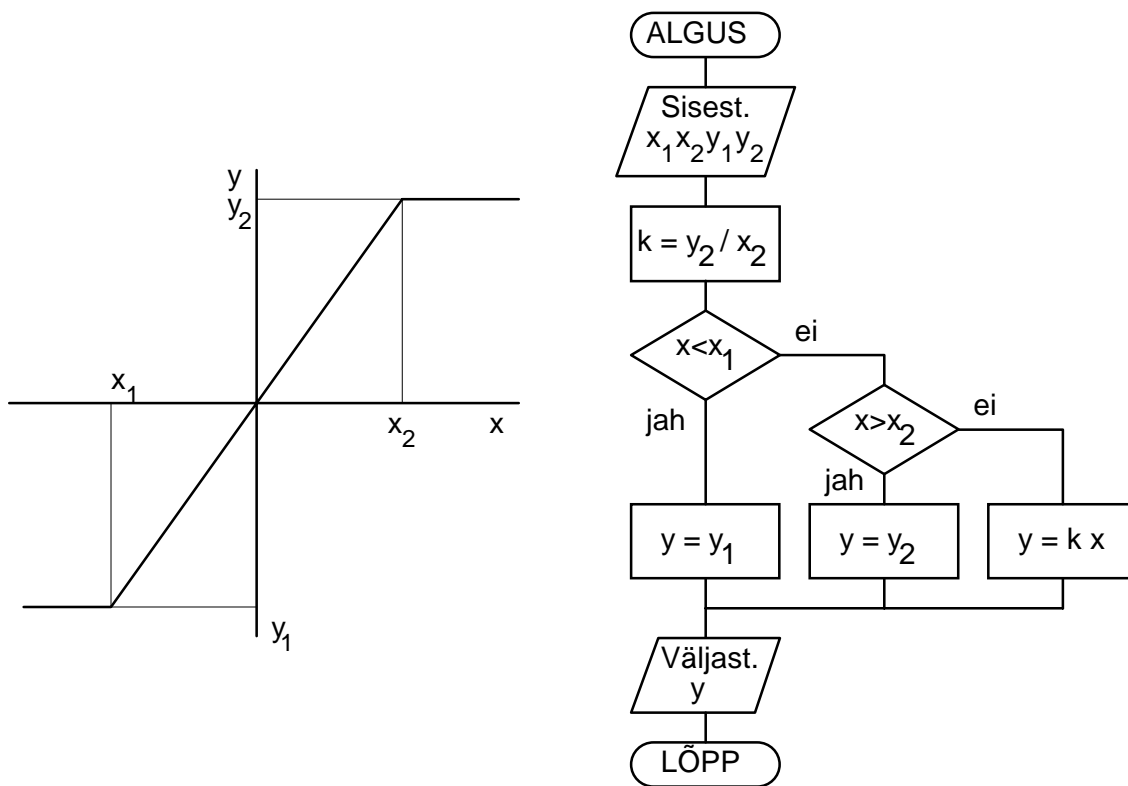
Kuna arvregulaator toimib diskreetsete ajaintervallide järel, siis tekib süsteemis paratamatult hilistumine. Suletud diskreetses juhtimissüsteemis põhjustab hilistumine väljundsuuruse võnkumist ning järelkult halvendab süsteemi töö kvaliteeti. Hilistumisprobleemi võib vaadelda kahest seisukohast: esiteks selle modelleerimise ja teiseks hilistumise korrigeeriva regulaatori seisukohast.

Hilistumise modelleerimisel tuleb arvestada asjaolu, et ajami olekut iseloomustavate suuruste mõõtmise ning regulaatori poolt väljastatava juhttoime vahel tekib viivitus, mille jooksul ajam töötab edasi ja tema tegelik olek ei vasta enam mõõdetule. Kõigepealt tuleb kindlaks määrata hilistumisaja t_h ja integreerimisel kasutatava ajaintervallivaheline seos. Arvutustäpsuse huvides on kasulik valida ajaintervall nii, et see oleks tunduvalt väiksem kui t_h . Lisaks peaks t_h olema ajaintervallist täisarv kordi suurem. Näiteks $t_h = L \cdot t$, kus L on täisarv. Juhttoime hilistumisega ajami tööd modelleerivas algoritmis on sel juhul mitu tsükli. Ajami liikumisvõrrandeid integreeritakse väikese perioodiga t , juhttoimet muudetakse pärast integreerimistüklimöödumist perioodiga t_h . Seejuures tuleb arvestada, et regulaatori väljundsuurus on arvatud andmete põhjal, mis iseloomustasid ajami olekut eelmise hilistumistsükli lõpul. Juhttoime hilistumist modelleeriva algoritmi plokk skeem on näidatud joonisel. Hilistumisest tingitud võnkumisi saab vähendada impulsskorrektsiooni abil. Niisuguse regulaatori väljundfunktsioon on järgmine:

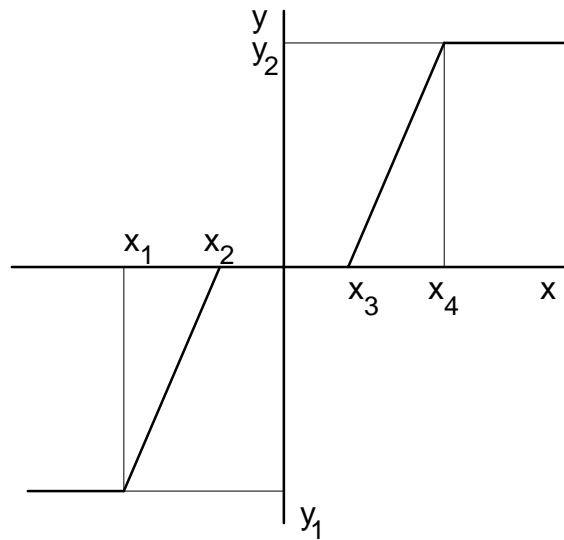
Tavaliselt on hilistumist korrigeeriv regulaator kasutusel koos teiste regulaatoritega ning seda tuleb vaadelda kui täiendavat vahendit ajami dünaamika parandamiseks.

Mittelineaarsete karakteristikute modelleerimine

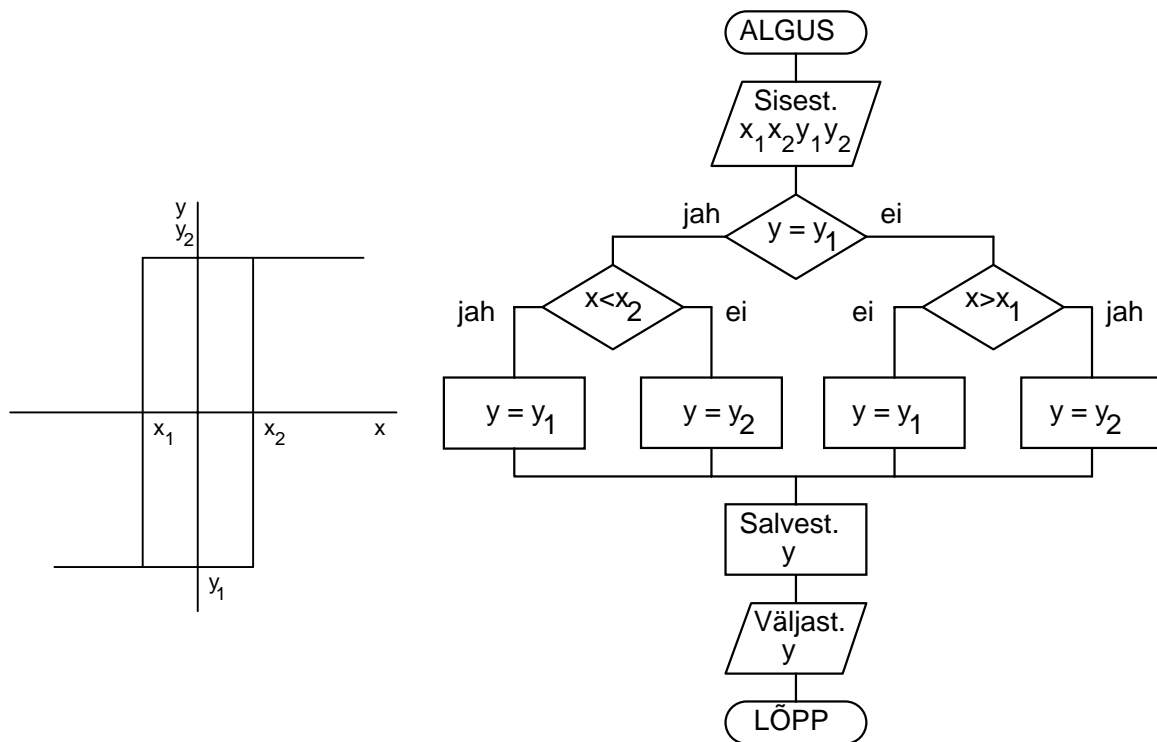
Analoogsüsteemides kasutatakse süsteemi muutujate piiramiseks mittelineaarse karakteristikuga regulaatoreid. Tüüpilisteks mittelineaarsuseks on "tundetustsoon" ja "küllastus". Analoogiliste piirangute loomine on võimalik ka arvregulaatorites. Joonisel 4. on toodud tundetustsooni ja küllastusega funktsiooni algoritm. Kuna tegemist on tükiti lineaarse funktsiooniga, siis tuleb programmeerimisel ette anda selle üksikute lõikude võrrandid ning käänupunktide koordinaadid. Arvregulaatoris realiseeritakse piirangute algoritm alamprogrammina, mis täidetakse pärast regulaatori väljundfunktsiooni programmi. Sarnaste piirangute korral võib piirangute alamprogrammi kasutada korduvalt erinevate regulaatorite juures, muutes seejuures vaid piirangufunktsiooni parameetrite arvvaartusi.



Joonis 5.14 Küllastustsooniga mittelineaarne karakteristik ja sellele vastav modelleerimise algoritm



Joonis 5.15 Tundetuse- ja küllastustsooniga mittelineaarne karakteristik

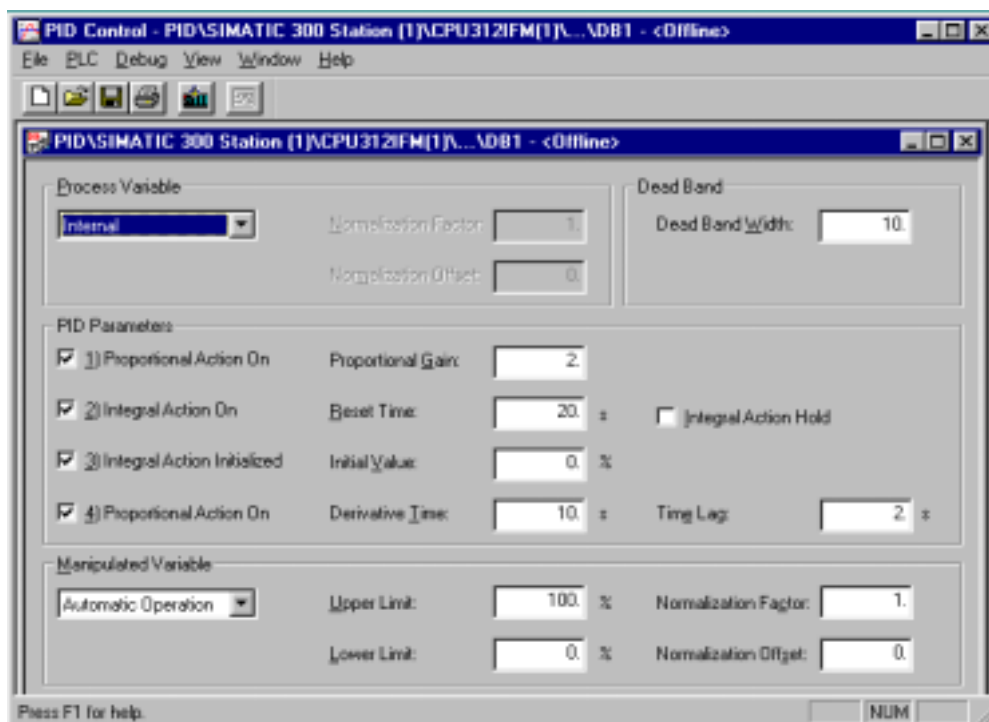


Joonis 5.16. Tükiti lineaarne hüsteeresifunktsioon ning algoritmi plokkskeem selle modelleerimise

5.6. Reguleerimistarkvara SIMATIC S7/M7/C7 kontrollritele

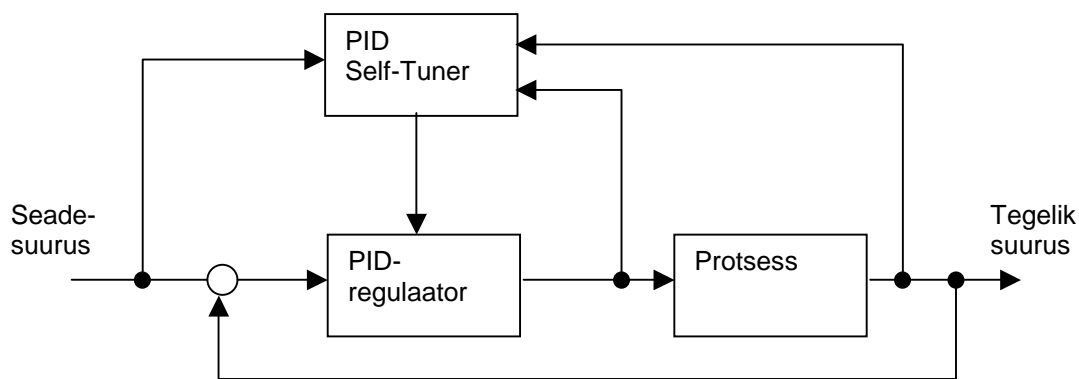
Reguleerimise ja automaatjuhtimise ülesannete lahendamiseks programmeeritavatel kontrollritel (nt. SIMATIC S7/M7/C7) saab kasutada mitmeid reguleerimisviise. Selle jaoks on firmal Siemens välja töötatud mitu tarkvarapaketti.

- **PID Control in STEP 7** sisaldab lihtsaid reguleerimisalgoritme, standardfunktsioonplokke ning PID-parameetrite määramiseks kasutajaliidest *PID Control Parameter Assignment* (joonis 5.17). Kasutatakse enamasti lihtsate temperatuurireguleerimisülesannete lahendamiseks. PID-juhtimine on integreeritud kontrollrite S7-200 jaoks tarkvarapaketti *STEP 7-MicroWin* ja on lihtsalt määratletav *Instruction Wizard* abil; samuti on PID-juhtimine kontrollrite S7-300/400 jaoks integreeritud tarkvarapaketti *SIMATIC Manager (STEP7)* ja *CFC* ning alates protsessoritest CPU 314 IFM riistvarse süsteemsete funktsioonplokkidena. Kontrollrite S7-200 iga protsessor CPU võimaldab reguleerida kuni 8 üksteisest sõltumatut ahelat. S7-300/400 korral sõltub reguleeritavate ahelate arv juhtseadme mälu mahust. *PID Control* pakett sisaldab järgmisi funktsioonplokke: pidevjuhtimiseks FB41 “CONT_C”, sammjuhtimiseks FB42 “CONT_S” ja pulsilaiusmodulatsiooniks FB43 “PULSGEN”. Kautajaliides “PID Control Parameter Assignment” võimaldab kõigile eelmainitud funktsioonplokkidele ja kontrolleri protsessorites (alates CPU314IFM) paiknevatele süsteemsetele funktsioonplokkidele vastavad juhtimisparameetrid ette anda. Funktsioonplokke saab ka omavahel kombineerida (nt. FB41 ja FB43) ja luua segajuhtimissüsteeme.



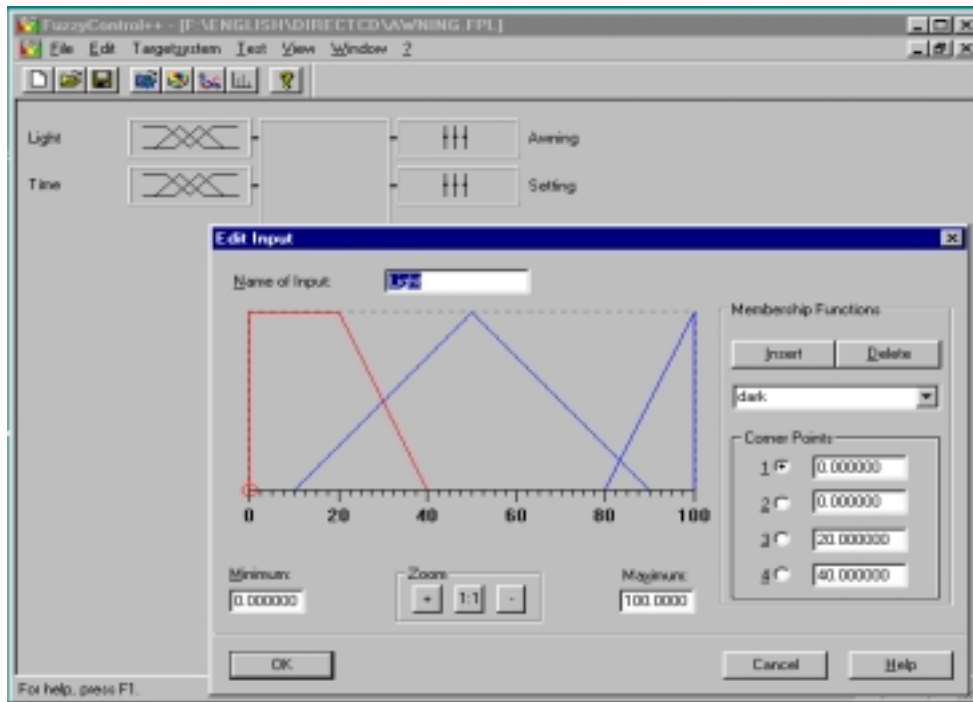
Joonis 5.17. PID-parameetrite määramine programmeerimissüsteemis STEP7

- **Standard PID Control** sisaldab *eelmääratlustega* (sks. *vorkonfektionierte*) reguleerimisstruktuure lihtsate ja keskmise raskusastmega reguleerimisülesannete lahendamiseks nagu temperatuuri, rõhu, voolu jne. reguleerimine. Standard PID Control tarkvarapaketti kasutatakse kontrollritel S7-300/400 baseeruvate reguleerimisülesannete lahendamiseks toiduaine-, keemia-, farmaatsia-, klaasi-, keraamika- ja masinatööstuses ning kliimatehnika valdkonnas. Standard PID Control võimaldab regulaatori graafilist koostamist ja parameetrite määratlemist.
- **Modular PID Control** sisaldab modulaarseid tarkvaraplokke raskete ja mahukate reguleerimisülesannete lahendamiseks, kus nõutakse suurt mälumahtu ja lühikesi lülitusaegu kuid see ei välista tema kasutamist ka väikeste protsesside juhtimiseks. Modular PID Control võimaldab koostada reguleerimisstruktuure SIMATIC-sarja kõikidele kontrolleritele nagu S7, M7 ja C7. Modular PID Control võimaldab regulaatori graafilist koostamist ja parameetrite määratlemist.
- **PID Self-Tuner** sisaldab funktsiooniplokki PID-regulaatorite *online*-eneseoptimeerimiseks. See funktsiooniplokk võimaldab optimeerida PID-regulaatorite parameetreid etteantud jooksva protsessi järgi (joonis 5.18). *PID Self-Tuner* on mõeldud temperatuuri-, vooluhulga- ja nivooregulaatorite optimeerimiseks.



Joonis 5.18. PID-regulaatori optimeerimise põhimõte

- **Fuzzy Control** on Windows 95 keskkonnas töötav tarkvarapakett (joonis 5.19), mis kujutab parametreerimistöõriista, et mugavamalt sätestada protsessi sisend- ja väljundparameetreid ja sätestatud (koostatud programmi) protsessi juhtimiseks tööle rakendada. Peale selle sisaldab ta standardseid funktsioone, mis lihtsustavad programmeerimist ja sätestamist. Spetsiaalset lisariistvara antud tarkvarale ei ole vaja. Seda paketti kasutatakse keeruliste protsesside juhtimiseks, kus on raske või võimatu protsessi matemaatiliselt kirjeldada; saab juhtida ka neid protsesse, mida varem sai juhtida ainult käsitsi. Protsessi matemaatilisi seoseid pole vaja tunda; piisab teadmistest, kuidas protsess peaks toimima, kui kasutada KUI-SIIS-sõnade kombinatsiooni.



Joonis 5.19. Fuzzy Control ++ tööaken

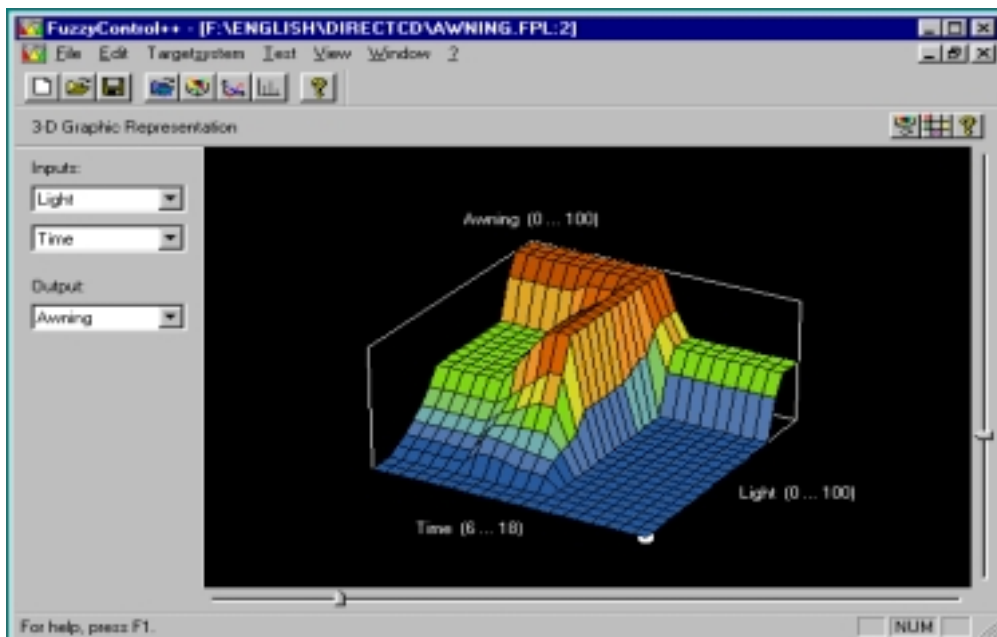
Kasutusalaudeks on tehnilist laadi rakendused kõrge juhtimistehnilise osaga. Keskkonnakaitsealal on palju mittelinearseid protsesse, mida sageli veel käsitsi juhitakse. Just need kuuluvad *Fuzzy Control* paketi automaatseerimisele, nt.

- vingugaasi eraldamine puhastites,
- veemajandus jne.

Fuzzy Control on mõeldud ka juhtimisülesannete optimeerimiseks. Näiteks saab võimalikult võrdselt koormata seadmete eri osi; mille tulemusel mahtuvuste kasutamine paraneb, protsessi seisakud ja vead vähenevad, voolutipud alanevad.

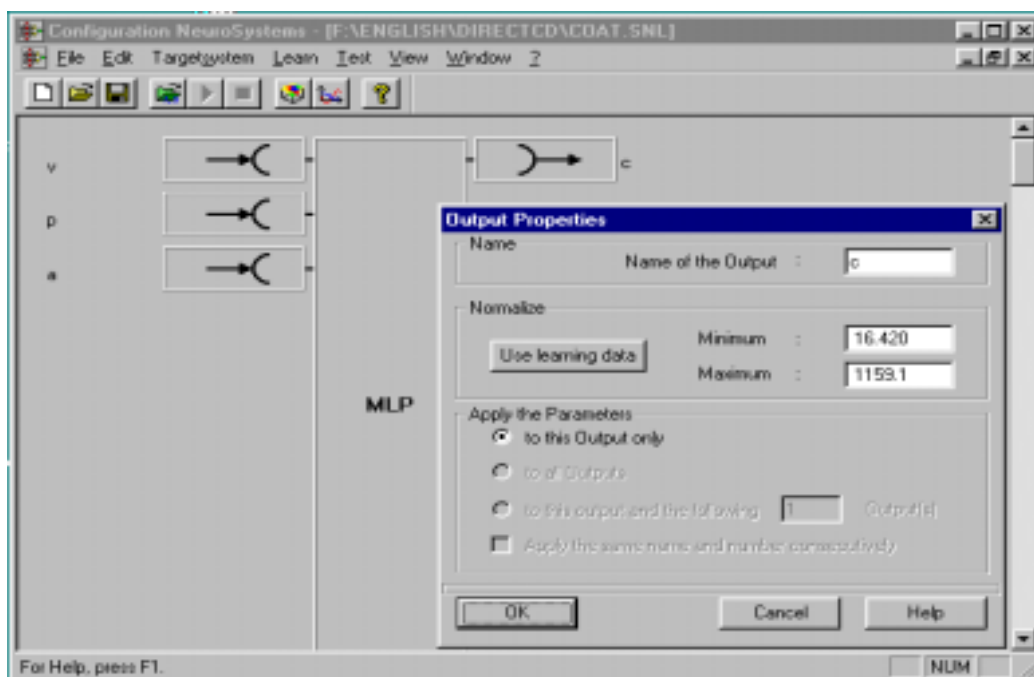
Tarkvarapakett *Fuzzy Control* teeb hägusjuhtimissüsteemi põhimõtted kasutatavaks automatiseerimissüsteemidele SIMATIC S7-300 (alates CPU 313) ja SIMATIC S7-400 ja samuti ka SIMATIC C7. *Fuzzy Control* kujutab ka toetust tavapärasele PID-regulaatorile nagu näiteks “*Standard PID Control*” või “*Modular PID Control*” eriti mittelineaarsuste korral.

Kui kasutaja on defineerinud protsessi sisend- ja väljundparameetrid, neid siduva funktsiooni ja loonud reeglite (juhiste, põhimõtete) tabeli, saab ta simuleerida süsteemi tööd kahe- ja kolmemõõtmelistel graafikutel (joonis 5.20).



Joonis 5.20. Kolmemõõtmeline esitus tarkvarapaketi Fuzzy Control++

- **NeuroSystems** - on Windows 95 keskkonnas töötav tarkvarapakett (joonis 5.21), mida kasutatakse neuronvõrkude arendamiseks ja konfigureerimiseks SIMATIC S7 seadmete ja protsessivisualiseerimissüsteemi WinCC juures. NeuroSystems tarkvara saab kasutada süsteemide juhtimiseks ja juhtimismudelite koostamiseks, optimeerimiseks ja juhtimismustrite identifitseerimiseks. Samuti võimaldab ta optimeerida MLP (multi layer perceptrons), RBF (radial basis function networks) ja NFN (neuro-fuzzy network) andmesidevõrke.



Joonis 5.21. NeuronSystems tööaken

5.7. Analoogsignaali ja pidevjuhtimine

SIEMENS'i kontrolleri SIMATIC S7 puhul on sõltuvalt protsessori tüübist kesk- ehk protsessoriplokki integreeritud peale digitaalsisendite ja -väljundite ka analoogsisendid ja -väljundid. Kui keskplokis analoogsisendid ja -väljundid puuduvad, saab need lisada vastava moodulina. Info lugemine ja kirjutamine väljunditesse toimub vastavate käskude ja aadresside abil. Olgu teada, et kõikidesse analoogsisenditesse tulevad analoogsignaali muundatakse sõltumata kasutaja toimingutest (analoog-digitaal-muunduriga) digitaalsele kujule ning analoogväljunditesse minev signaal muundatakse digitaalselt kujult (digitaal-analoog-muunduriga) analoogkujule. Programmeerija opereerib suuruste digitaalväärtustega. Nii analoogsisendi kui ka -väljundi jaoks on digitaalsuuruse esitus ühesugune, omades mälu kindlat piirkonda. Analoogsuurust esitatakse binaarsel kujul kahendarvuna (kahendkomplementina). Analoogsuuruse esitus analoogmoodulites on toodud tabelis 5.3.

Tabel 5.3. Analoogsuuruse esitus moodulites

Eraldusvõime	Analoogsuurus															
Bittide arv	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Biti kaal	S	2^{14}	2^{13}	2^{12}	2^{11}	2^{10}	2^9	2^8	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0

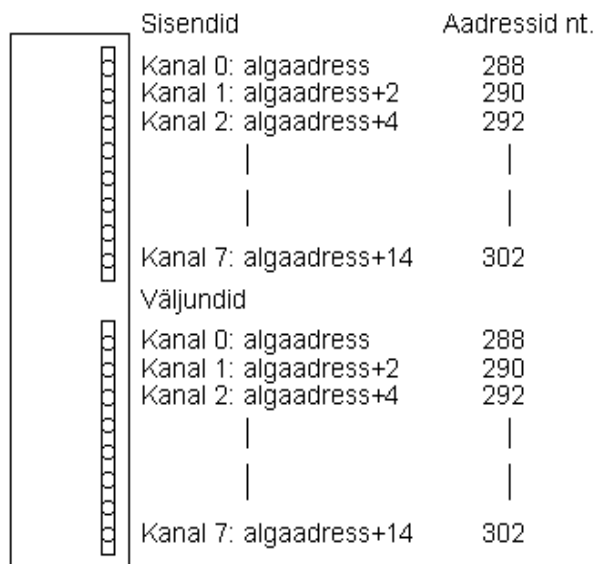
Sümbol S (*sign*) tähistab analoogsuuruse märgi kohta, mis paikneb alati 15-ndas bitis, s. t. kahendarvu vanimas vasakpoolses bitis. Kui selle biti olek on "0", on analoogsuuruse märk "+". Olekul "1" on märk "-". Kui analoogsuuruse esituseks (sõltuvalt analoogmoodulitest) kasutatakse vähem kui 15 bitti, s. t. kui analoogmooduli eraldusvõime on alla 15 biti, sisestatakse märk AKU vasakpoolsesse vanema baidi vanimasse bitti. Noorema baidi vastavate bittide positsioonid täidetakse automaatselt olekuga "0". Tabel 5.4 näitab analoogsuuruste digitaalset esitust eri eraldusvõimega analoogmoodulites ning missugustele bittidele (tähistatud sümboliga x) omistatakse olek "0", kui analoogsuuruse kirjeldamiseks kasutatakse vähem kui 15 bitist esitust. Eraldusvõime sõltub valitud analoogmoodulist (analoog-digitaal-muunduri kohtade arvust) ja selle parameetritest ning määrab analoogsuuruse täpsuse (tabel 5.4). Mida väiksem eraldusvõime, seda ebatäpsem on reaalse analoogväärtuse esitus. See tähendab, et väljundmooduli väljundpinge või -vool muutuvad suuremate astmete kaupa. Vastupidi, mida suurem on eraldusvõime, seda suurem on täpsus ja seda sujuvamalt muutub näiteks väljundsuurus (pinge, vool).

Tabel 5.4. Analoogmooduli täpsus sõltuvalt eraldusvõimest

Eraldusvõime bittides (+S)	Väärtus ehk ühik (täpsus)		Analoogsuuruse digitaalkuju	
	Kümnendarv	Kuueteistkümnendarv	Vanem bait	Noorem bait
15	1	1 _H	S0000000	00000001
14	2	2 _H	S0000000	0000001x
13	4	4 _H	S0000000	000001xx
12	8	8 _H	S0000000	00001xxx
11	16	10 _H	S0000000	0001xxxx
10	32	20 _H	S0000000	001xxxxx
9	64	40 _H	S0000000	01xxxxxx
8	128	80 _H	S0000000	1xxxxxxx

Andurite ning täiturite ühendusviisid ja eri moodulite mõõtepiirkonnad on toodud kataloogides. Analoogsisend- ja -väljundmoodulite tööpõhimõte seisneb selles, et analoog-digitaalmuundurist saadakse digitaalsuurus, mis kantakse mällu ja/või kontrolleri andmesiinile. Kui analoogsisendmoodulil on enam kui üks sisend, siis loetakse ja muundatakse sisendsuursi üksteise järel jadamisi. Seega loetakse analoogsisendite signaale tsükliliselt ja tsükli kestuse määrab sisendite arv. Analoogiliselt sisendmoodulile toimib ka väljundmoodul, s. t. digitaalsuurused laetakse tsükliliselt vastavasse väljundisse ja muundatakse analoogsignaaliiks.

Kasutaja saab analoogsuursuse digitaalsel kujul analoogsisendmoodulist kätte ning saab saata digitaalsuursuse analoogväljundmoodulisse siirdamiskäskudega (*MOVW*). Käsu operandi tunnuseks on vastavalt sisendi korral *AIW* ja väljundi korral *AQW*. Operandi tunnusele järgneb aadress, mis määrab, millise sisend- või väljundsignaali jaoks operatsiooni ehk käsku täidetakse. Seega nt. teades, et analoogsisendi aadress on 0, loetakse tema signaaliväärtust sisendist järgmiselt: *MOVW AIW0*, *VW10*. Selleks et kirjutada või saata signaali väärtus digitaalsel kujul väljundisse aadressiga 2, kirjutatakse järgmine käsuriida: *MOVW VW10*, *AQW2*.



Joonis 5.22. Analoogsisend- ja -väljundmoodulite adresseerimine

Analoogsisend ja väljundid ehk kanalid omavad alati sõnaaadressi ja iga kanali aadress sõltub mooduli algaadressist. Analoogsisend-/-väljundmoodulil sisendite algaadress vastab väljundite algaadressile. Puhtalt analoogsisend- või -väljundmoodulil ei ole eri kanalitel sarnaseid aadresse ja iga järgneva kanali sõnaaadress on 2 võrra suurem eelmisest (joonis 5.22).

SIMATIC S7-200 sarja kontrolleriites teisendatakse analoogsuurus sõnapikkuseks täisarvuliseks väärtuseks vastavalt alljärgnevatele määrangutele:

Tabel 5.5. Analoogsuurused vastavalt esitusviisile täisarvulisel kujul

Suuruse esitusviisi määrand	Täisarvuline suurus vahemikus
Unipolaarne	0...32000
Bipolaarne	-32000...32000

Teades, kuidas lugeda sisendmoodulis analoogväärtust digitaalsel kujul ja kuidas saata analoogväärtust väljundmoodulisse digitaalkujul, on vaja lisaks teada ka seda, kuidas on ette antud juhitava väärtuse muutumise ülem- ja alampiir. Selleks on vaja teada missuguse voolu-, pinge- või temperatuuripiirkonna mõõtmiseks või signaalinivoo väljastamiseks on vastav sisend- ja väljundmoodul ette nähtud. Tuleks lisada, et eri piirkondade ja parameetrite määratlemiseks saab kasutada vastavaid mõõtemoduleid, STEP 7 keele käsked ja süsteemseid või standardfunktsioone ning funktsioonplokke. Andmed moodulite ja mõõtepiirkondade kohta on toodud käsiraamatutes. Iga mooduli mõõtepiirkond jaotatakse 5 osaks, milleks on ületäitumis-, ülejuhtimis-, nimi-, alajuhtimis- ja alatäitumispiirkond. Tabelis 5.6 on toodud nt. SIMATIC S7-300/400 sarja kontrolleri sisendmooduli 1...5 V ja 4...20 mA mõõtepiirkondade esitus.

Tabel 5.6. Analoogväärtuse esitus digitaalkujul S7-300/400 sarja kontrolleri sisendmoodulil

Mõõtepiirkond 1...5 V	Mõõtepiirkond 4...20 mA	Ühikud		Piirkond
		Kümnendarv	Kuuteistkümnendarv	
>5.7036	>22.810	32767	7FFF _H	Ületäitumine
5.7036	22.810	32511	7EFF _H	Ülejuhtimine
:	:	:	:	
5.001	20.0005	27649	6C01 _H	Nimipiirkond
5.000	20.000	27648	6C00 _H	
4.000	16.000	20736	5100 _H	
:	:	:	:	
1.000	4.000	0	0 _H	Alajuhtimine
0.9999	3.9995	-1	FFFF _H	
:	:	:	:	
0.2963	1.1852	-4864	ED00 _H	Alatäitumine
<0.2963	<1.1852	-32768	8000 _H	

Olgu lisatud, et üle- või alatäitumise korral on sisendmoodulis alati analoogsuuruse digitaalväärtus vastavalt 7FFF_H või 8000_H. Analoogväljundmoodulis on üle- või alatäitumise korral analoogsuuruse väärtus 0, et kaitsta ülepinge eest sinna ühendatud täitureid. Täpsemaks selgituseks on tabelis 5.7 toodud väljundsuuruse esitus analoogväljundmoodulis.

Käsiraamatutes esitatud mõõtepiirkondade tabelid (nt. tabelid 5.6 ja 5.7) võimaldavad leida, millisele analoogväärtustele vastab teatud kümnend- või kuuteistkümnendarv. Teades seoseid analoog- ja digitaalsuuruste vahel, on lihtne programmiselt määratleda juhtimistingimusi.

Tabel 5.7. Analooqväärtuse esitus digitaalkujul S7-300/400 sarja kontrolleri väljundmoodulil

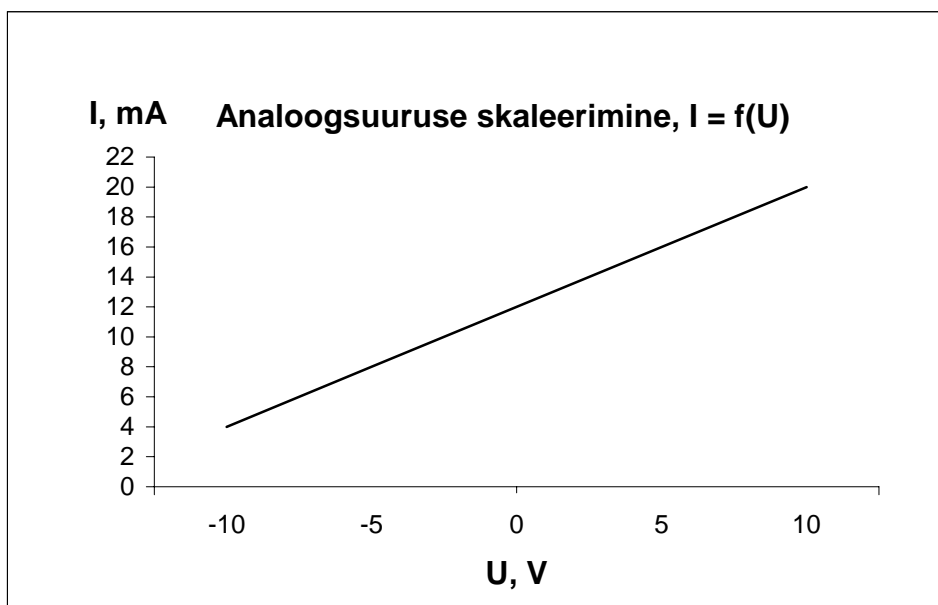
Mõõtepiirkond ±10 V	Ühikud		Piirkond
	Kümnendarv	Kuueteist- kümnendarv	
0	>32511	>7EFF _H	Ületäitumine
11.7589 : 10.0004	32511 : 27649	7EFF _H : 6C01 _H	Ülejuhtimine
10.0000 : 0 : -10.0000	27648 : 0 : -27648	6C00 _H : 0 _H : 9400 _H	Nimipiirkond
-10.0004 : -11.7589	-27649 : -32512	93FF _H : 8100 _H	Alajuhtimine
0	<-32512	<8100 _H	Alatäitumine

5.8. Analooosuuste normeerimine

Et teostada ükskõik millist reguleerimis- jm. operatsiooni tuleb analooosuurus sageli arvutuste jaoks viia teisele kujule ehk teisendada. Seda nimetataksegi skaleerimiseks.

Mõned näited teisendamise kohta:

1. Pinge pingeks ($0...10V \rightarrow \pm 10V$, $\pm 10V \rightarrow 1...5V$, $0...10V \rightarrow 1...5V$ jne.)
2. Pinge vooluks ($0...10V \rightarrow 0...20mA$, $\pm 10V \rightarrow 0-20mA$, $0...10V \rightarrow 4...20mA$ jne.)
3. Pinge rõhuks ($0...10V \rightarrow 0...10bar$, $1...5V \rightarrow 0...1bar$, $\pm 10V \rightarrow 0...2bar$ jne.)
4. Pinge vedeliku vooluhulgaks ($0...10V \rightarrow 0...1000 m^3$, $1...5V \rightarrow 0...100 l$ jne.)
5. Pinge vedeliku kuluks ($0...10V \rightarrow 0...1 m^3/s$, $1...5V \rightarrow 0...100 l/s$ jne.)
6. Pinge temperatuuriks ($0...10V \rightarrow 0...100 ^\circ C$, $\pm 10V \rightarrow -100...100 ^\circ C$ jne.)
7. Vool vooluks ($0-20mA \rightarrow 4...20mA$, $\pm 20mA \rightarrow 4...20mA$ jne.)
8. Vool pingeks ($0...20mA \rightarrow 0...10V$, $0-20mA \rightarrow \pm 10V$, $4...20mA \rightarrow 0...10V$ jne.)
9. Vool rõhuks ($4...20mA \rightarrow 0...10bar$, $\pm 20mA \rightarrow 0...1bar$, $0-20mA \rightarrow 0...2bar$ jne.)
10. Vool temperatuuriks ($4...20mA \rightarrow 0...100 ^\circ C$, $\pm 20mA \rightarrow -100...100 ^\circ C$ jne.)
11. jne.



Joonis 5.23. Pingeväärtuse skaleerimine vooluväärtuseks

Selleks, et teostada eeltoodud suuruste teisendamist on vaja teada teisendatava suuruse (edaspidi sisendsuurus) ja teisenduse tulemi (edaspidi väljundsuurus) maksimum- ja miinimumväärtusi. Kui eeltoodud andmed on teada, tuleb nad asetada alltoodud valemisse ja teostada skaleerimine ehk teisendus.

$$\frac{(S_{teg} - S_{min}) \cdot (V_{maks} - V_{min})}{(S_{maks} - S_{min})} + V_{min} = V_{teg}, \text{ kus}$$

S_{teg} - tegelik sisendsuuruse väärtus

S_{maks} - sisendsuuruse maksimaalne väärtus

S_{min} - sisendsuuruse minimaalne väärtus

V_{teg} - arvutatud tegelik väljundsuuruse väärtus

V_{maks} - väljundsuuruse maksimaalne väärtus

V_{min} - väljundsuuruse minimaalne väärtus

Sama valemi abil on võimalik ka täisarvuks või reaalarvuks teisendatud analoogsuurusi skaleerida.

Analoogsuuruste teisendamist suhte kujule, s. t. kui suure osa moodustab (mitu protsenti) tegelik sisend- või väljundsuurus tema lubatud maksimaalväärtusest, nimetatakse normeerimiseks. Juhul kui tegelik suurus on üle lubatud maksimaalväärtuse, siis korrigeeritakse tegeliku suuruse väärtust nii, et tema väärtus võrdustatakse maksimaalväärtusega. Juhul kui tegelik suurus on alla lubatud miinimumväärtuse, siis korrigeeritakse tegeliku suuruse väärtust nii, et tema väärtus võrdustatakse miinimumväärtusega.

$$\frac{S_{teg}}{S_{maks} - S_{min}} + x = M_{norm}, \text{ kus}$$

- S_{teg} - tegelik sisendsuuruse väärtus
 S_{maks} - sisendsuuruse maksimumväärtus
 S_{min} - sisendsuuruse miinimumväärtus
 x - unipolaarse suuruse korral 0 ja bipolaarse suuruse korral 0,5
 M_{norm} - normeeritud suurus

Normeeritud kujult suuruse teisendamine väljundsuuruse kujule ehk täisarvu kujule

$$(N_{norm} - x) \cdot (V_{maks} - V_{min}) = V_{teg}, \text{ kus}$$

- V_{teg} - tegelik väljundsuuruse väärtus
 V_{maks} - väljundsuuruse maksimumväärtus
 V_{min} - väljundsuuruse miinimumväärtus
 x - unipolaarse suuruse korral 0 ja bipolaarse suuruse korral 0,5
 N_{norm} - normeeritud suurus

5.9. Signaali tundetustsooni ja piirangu määramine

Protsesside reguleerimisel kasutatakse signaali tundetustsooni, et vältida juhitava protsessi võnkumist, kui protsessisuuruse ja etteandesuuruse erinevus on suhteliselt väike, nt. 5 %.

Tundetustsooni määramisel tuleb kasutada järgmist seaduspärasust, et juhtimist ei toimu kui $x - \Delta x < w < x + \Delta x$, kus $\Delta x = k \cdot w$ ja $k = 0 \dots 1$ (0...100 %). Sellise seaduspärasuse järgi võib terve reguleeritava piirkonna reguleerimisvabaks muuta kui $k = 1$.

Signaali väärtuse piiramist kasutatakse sageli koos regulaatoritega, et tagada seade-, juht- või tagasisidesignaali jäämine lubatud piiridesse.

Seega saab signaali piiramiseks kasutada järgmisi võrdlusi ja omistamistehteid sisendsignaali piiramisel

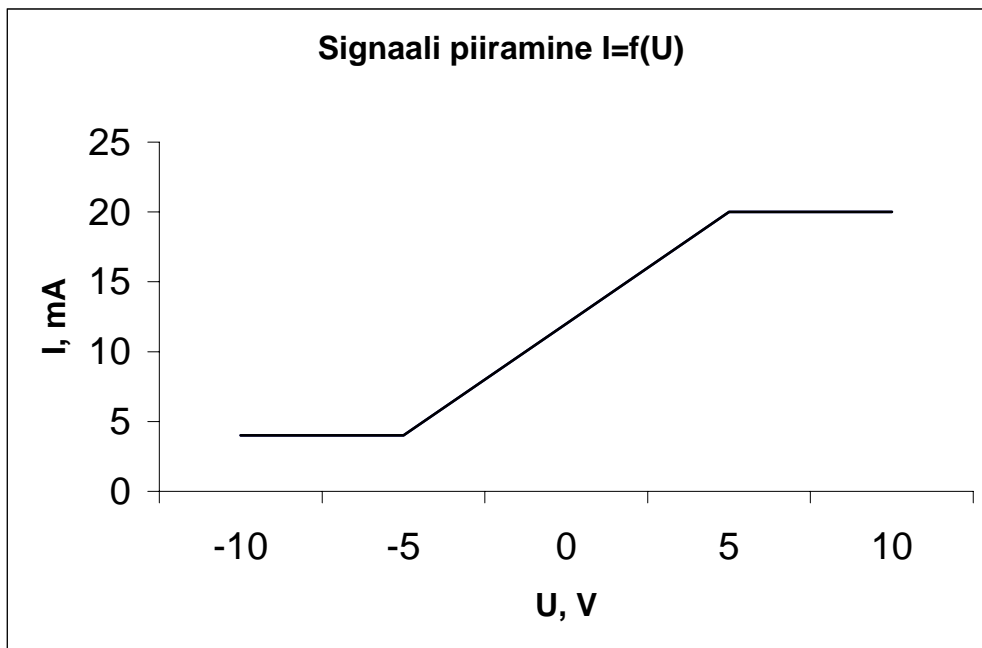
Kui $S_{teg} > S_{max,piir}$, siis $S_{teg} = S_{max,piir}$

Kui $S_{teg} < S_{min,piir}$, siis $S_{teg} = S_{min,piir}$

või väljundsignaali piiramisel

Kui $V_{\text{teg}} > V_{\text{max.piiir}}$, siis $V_{\text{teg}} = V_{\text{max.piiir}}$

Kui $V_{\text{teg}} < V_{\text{min.piiir}}$, siis $V_{\text{teg}} = V_{\text{min.piiir}}$



Joonis 5.24. Vooluväärtuse piiramine pingeväärtuse järgi

5.10. Kahe- ja kolmeolekuline binaarjuhtimine

Binaarjuhtimine erineb pidevjuhtimisest selle poolest, et juhttoime on diskreetne. Eristatakse kahe- ja kolmeolekulist binaarjuhtimist.

Kahe olekuga binaarjuhtimise tüüpnäiteks on küttesüsteemi juhtimine, kus kütteelementi lülitatakse sisse või välja sõltuvalt sellest kas temperatuur on seadesuurusest väiksem või suurem.

Küttekeha väljalülitamine toimub tingimusel kui $x > w + \Delta x / 2$

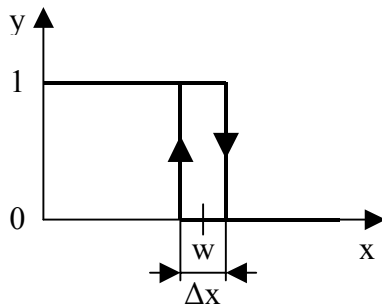
ja sisselülitamine toimub tingimusel kui

$x < w - \Delta x / 2$

Hüsterees on vajalik selleks, kui tagasiside signaal saab võrdseks etteandesignaaliga, et siis ei toimuks juhitava objekti pidevat sisse- ja väljalülitamist, mis võib põhjustada juhitava objekti riknemise.

Kahe olekuga binaarjuhtimine on näidatud joonisel 5.25, kus kasutatakse alljärgnevalt loetletud tähiseid.

- w - etteandesuurus
- x - tagasiside ehk protsessi suurus
- y - juhttoime ehk juhtsuurus
- Δx - hüsterees ehk sisse- ja väljalülitamise vahemik



Joonis 5.25. Kahe olekuga binaarjuhtimine

Selleks, et arvutada välja kui pika aja tagant protsessi juhtimise käigus vastavalt etteantud hüstereesile tuleb objekti sisse või välja lülitada, tuleb teada objekti ajakonstanti T_s . Näiteks küttekeha puhul aega, mis kulub küttekehal maksimaalse temperatuuri saavutamiseks. Kui eelnimetatud ajakonstandi ja etteande suuruse väärtus on teada, saab järgmise valemi abil arvutada küttekeha sisse-väljalülitamisperioodi T , kompenseerimaks küttekeha soojenemiseks ja jahtumiseks kuluvat aega.

$$T = 2 \cdot \Delta x \cdot T_s / w$$

Jagades saadud tulemuse kahega saame poolperioodi aja. Eeltoodud valemit kasutatakse juhul kui reguleeritav objekti iseloomustab pikk sisse- ja/või väljalülitamisviivitus. Näiteks magnetklappide juhtimiseks ei kasutata eeltoodud valemit.

Kolme sammuga juhtimise tüüpiline näide on näiteks veekulu reguleerimine alalisvoolumootoriga varustatud ventiili abil, kus väljundist y_1 antav impulss avab ventiili ja väljundist y_2 antav impulss suleb ventiili. Väljundite juhtimist iseloomustab alljärgnev joonis XX.

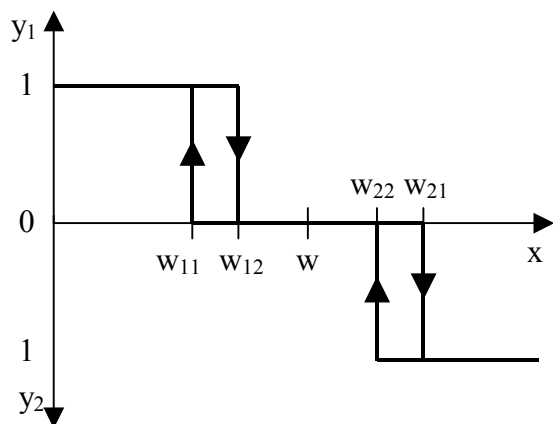
Kui $x < w - w_{11}$, siis väljund y_1 lülitatakse sisse

Kui $x > w_{12}$, siis väljund y_1 lülitatakse välja

Kui $x > w + w_{21}$, siis väljund y_2 lülitatakse sisse

Kui $x < w_{22}$, siis väljund y_2 lülitatakse välja

Eeltoodud võrdluste jaoks võib w_{11} , w_{12} , w_{21} , w_{22} välja arvutada w suurusest protsentuaalselt s.t. et ette on antud kui mitme protsendi võrra w_{11} või w_{12} on väiksem ja kui mitme protsendi võrra w_{21} või w_{22} on väiksem etteandesuurusest w . $w_{11} = w_{21} = k_1 \cdot w$, $k_1 = 0 \dots 1$ (0...100%) kus k_1 on sisselülitamise tingimuse suhteline suurus. $w_{12} = w_{22} = k_2 \cdot w$, $k_2 = 0 \dots 1$ (0...100%) kus k_2 on väljalülitamise tingimuse suhteline suurus.

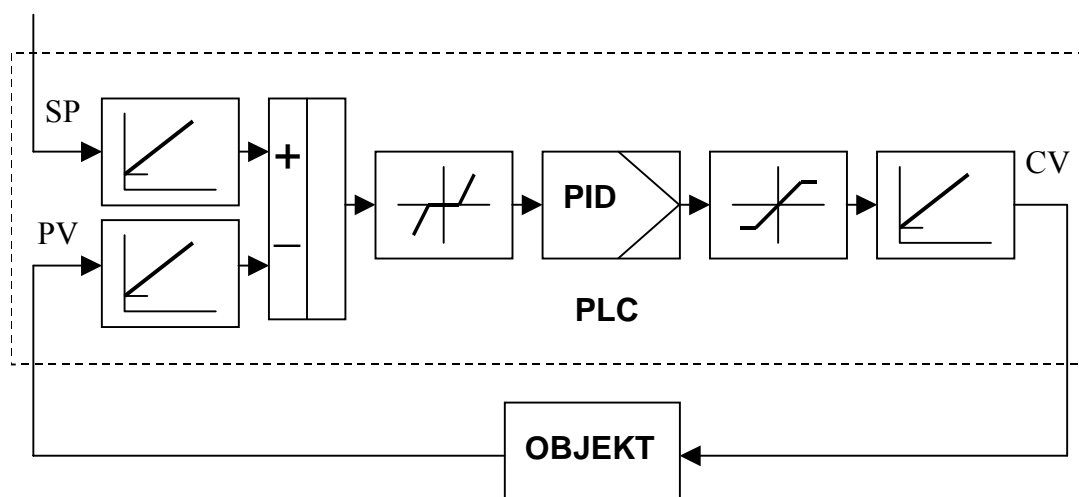


Joonis 5.26. Kolme olekuga binaarjuhtimine

5.11. PID- juhtimine

PID-juhtimist saab kontrollerial realiseerida süsteemsete funktsioonplokkidega; nende puudumisel võib kirjutada vastava reguleerimisprogrammi. Joonisel 5.27 on toodud PID-juhtimise plokk skeem tehnoloogilises protsessis, kus

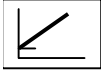

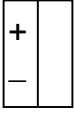

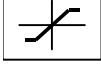
SP on etteandesuurus (*setpoint*),
 CV - juhtsuurus (*control variable*) ja
 PV - protsessi suurus (*process variable*)



Joonis 5.27. PID-juhtimise põhimõtteskeem

Plokkskeemil kasutatakse süsteemi elementide tähistamiseks alljärgnevat tingmärke.

Tabel 5.8.

	Signaali normeerimine, skaleerimine ja teisendamine (<i>Signal Normalizing</i>)
	Reguleerimisvaba piirkonna määramine (<i>Dead Band Width</i>)
	Kahe signaali vahe leidmine (<i>Difference</i>)
	PID-reguleerimine (<i>PID-loop</i>)
	Signaali piiramine (<i>Signal Limit</i>)

PID-regulaatori toime koosneb proportsionaalse, integreeriva ja diferentseeriva toime summast

$$CV(t) = K_P \cdot e + K_I \int_0^t e dt + CV_{initial} + K_D \cdot \frac{de}{dt},$$

kus

e - juhtimisviga ehk etteande- ja protsessisuuruse erinevus (vahe)

K_P - proportsionaalse toime võimendustegur

K_I - integreeriva toime võimendustegur

K_D - diferentseeriva toime võimendustegur

Numbrilisel kujul leitakse integraal summa funktsioonina, diferentsiaal aga vahefunktsioonina teatud ajahetkel t .

$$PID_CV_N = K_P \cdot e_N + K_I \cdot \sum_1^N e_i + CV_{initial} + K_D \cdot (e_N - e_{N-1}), i = 1 \dots N$$

Väljund = P-lüli + I-lüli + D-lüli

kus

- $CV_{initial}$ - juhtseadme väljundi algväärtus
 PID_CV_N - juhtseadme tulem diskreetsel ajahetkel N
 e_N - juhtseadme vea väärtus diskreetsel ajahetkel N
 e_{N-1} - juhtimisvea väärtus diskreetsel ajahetkel $N - 1$

Proportsionaalne lüli arvutab juhttoime sõltuvalt võimendustegurist ning etteande ja protsessisuuruse vahelisest veast (edaspidi lihtsalt **viga**). Integreeriv lüli arvutab juhttoime proportsionaalselt ajas summeeritud vea järgi. Diferentseeriva lüli korral arvutatakse juhttoime proportsionaalselt ajas muutuva vea järgi. P, I ja D-lülid on matemaatiliselt kirjeldatavad kontrolleri programmis järgmisel kujul:

P $P_CV_N = K_P * C * (SP_N - PV_N)$

I $I_CV_N = K_P * T_S / T_I * (SP_N - PV_N) + I_CV_{N-1}$

D $D_CV_N = K_P * T_D / T_S * ((SP_N - PV_N) - (SP_{N-1} - PV_{N-1}))$, kui $SP_N = SP_{N-1}$, siis
 $D_CV_N = K_P * T_D / T_S * (PV_{N-1} - PV_N)$

kus

P_CV_N - proportsionaalse lüli juhttoime diskreetsel ajahetkel N

I_CV_N - integreeriva lüli juhttoime diskreetsel ajahetkel N

I_CV_{N-1} - integreeriva lüli juhttoime diskreetsel ajahetkel $N-1$

D_CV_N - diferentseeriva lüli juhttoime diskreetsel ajahetkel N

SP_N - etteandesuurus diskreetsel ajahetkel N

PV_N - protsessisuurus diskreetsel ajahetkel N

SP_{N-1} - etteandesuurus diskreetsel ajahetkel $N-1$

PV_{N-1} - protsessisuurus diskreetsel ajahetkel $N-1$

K_P - võimendustegur

C - konstant

T_S - kvantimis- ehk programmiitsükli aeg

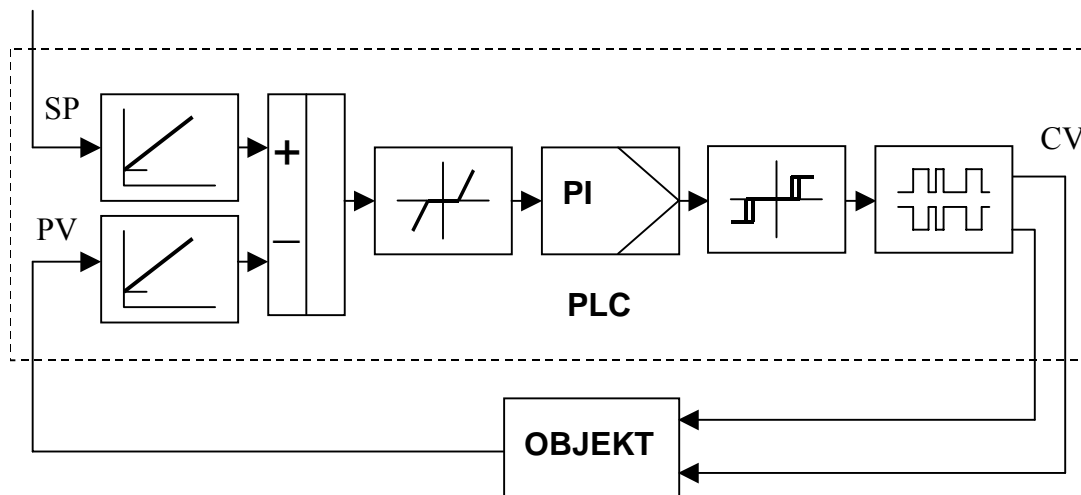
T_I - integreerimisaeg

T_D - diferentseerimisaeg

Seega kujutab PID-juhtimise kohta käiv valem endast eeltoodud osalahendite summat:

$$PID_CV_N = P_CV_N + I_CV_N + D_CV_N .$$

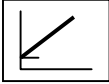
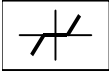
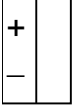
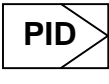
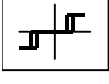
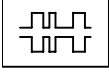
Eeltoodud valem sobib pidevjuhtimisega reguleerimise teostamiseks. Sageli aga kasutatakse ventiilide, klappide jne. Juhtimiseks kombineeritud impulss-sammjuhtimist.



Joonis 5.28 Impulss-binaarjuhtimine

Plokkskeemil kasutatakse süsteemi elementide tähistamiseks alljärgnevaid tingmärke.

Tabel 5.9.

	Signaali normeerimine, skaleerimine ja teisendamine (<i>Signal Normalizing</i>)
	Reguleerimisvaba piirkonna määramine (<i>Dead Band Width</i>)
	Kahe signaali vahe leidmine (<i>Difference</i>)
	PID-reguleerimine (<i>PID-loop</i>)
	Kolme olekuga juhtimine (<i>Three step control</i>)
	Kaheväljundiline pulsilaiusmodulaator (<i>pulse width modulation</i>)

Kombineeritud impulss-sammjuhtimise realiseerimiseks tuleb kasutada eeltoodud valemi edasiarendust, kuna väljundsuuruseks on impulsside arv.

$$\Delta \text{PID_CV} = \text{PID_CV}_N - \text{PID_CV}_{N-1},$$

$$T_Y = T_M / T_S,$$

$$T_{CV} = T_Y * \Delta \text{PID_CV} / \Delta \text{PID_CV}_{MAX}$$

T_M - täiturseadme 0...100% ümberlülamise aeg

T_Y - maksimaalne impulsside arv täiturseadme 0...100% ümberlülamise jooksul

$\Delta \text{PID_CV}$ - ajahtkel N ja N-1 arvutatud toimesuuruste vahe

$\Delta \text{PID_CV}_{MAX}$ - maksimaalne võimalik toimesuuruste väärtuste vahe

T_{CV} - toimesuuruste väärtuste vahe teisendatud impulssväärtus

Sõltuvalt juhitavate protsesside erisustest, ei ole meil sageli vaja PID regulaatorit, vaid vajame hoopis PI, PD, ID, P, I või D regulaatorit. Selleks tuleb omistada PID-regulaatori vastavatele ajakonstantidele kindlad väärtused, et tagada arvutustulemuse õigsus:

Tabel 5.10. Ajakonstantide väärtused regulaatorite arvutamisel

Regulaatori tüüp	Ajakonstantide väärtused
P	P_CV_N arvutamisel $T_I = \infty$, $T_D = 0$
I	P_CV_N arvutamisel $K_P = 0$, I_CV_N arvutamisel $K_P = 1$ D_CV_N arvutamisel $T_D = 0$
D	P_CV_N arvutamisel $K_P = 0$, I_CV_N arvutamisel $T_I = \infty$ D_CV_N arvutamisel $K_P = 1$
PI	D_CV_N arvutamisel $T_D = 0$
PD	I_CV_N arvutamisel $T_I = \infty$
ID	P_CV_N arvutamisel $K_P = 0$, I_CV_N arvutamisel $K_P = 1$ D_CV_N arvutamisel $K_P = 1$

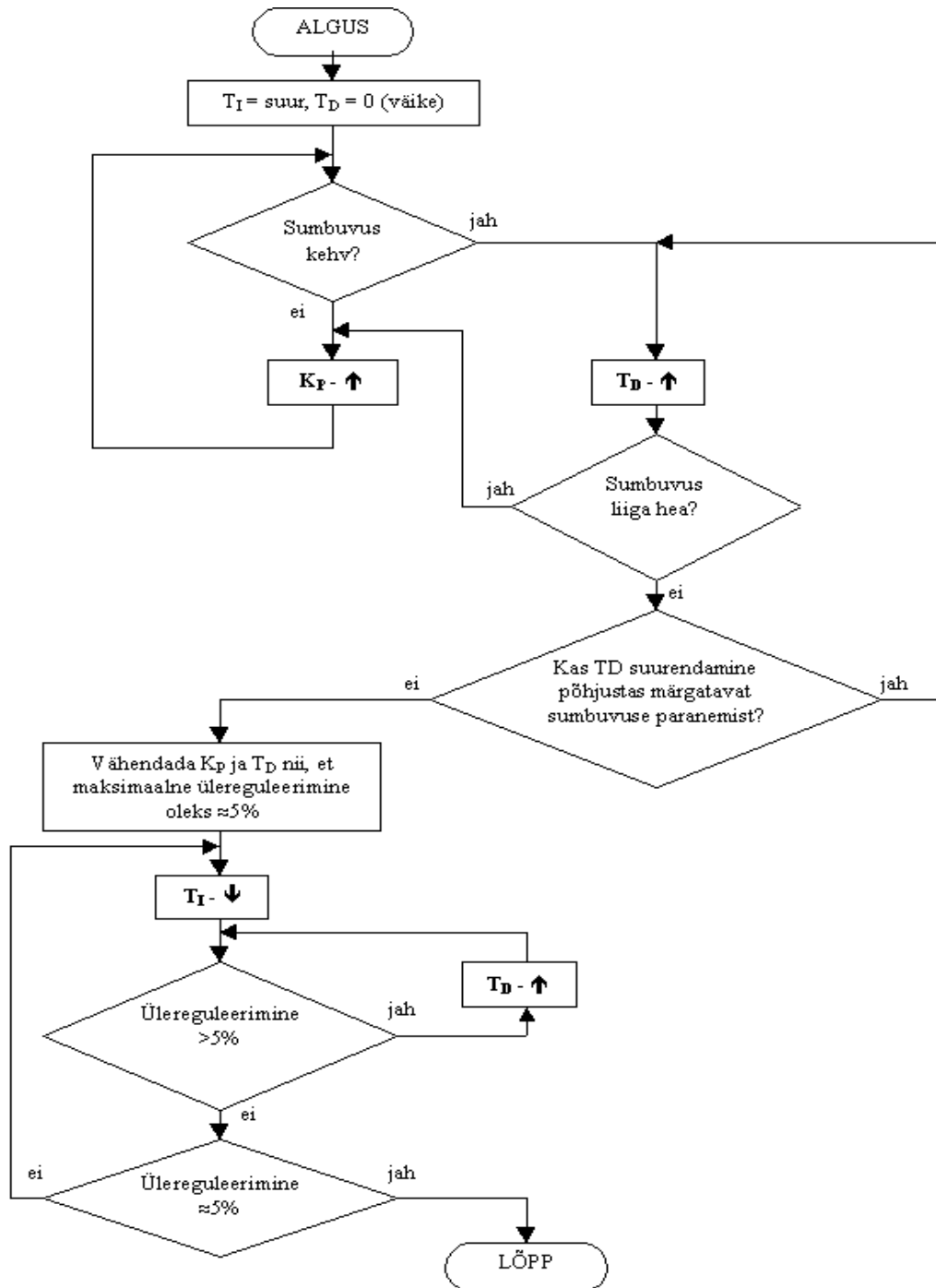
Üks keerulisemaid etappe on PID-regulaatorite häälestamine. Häälestamiseks on kaks põhimeetodit, milledeks on:

1. Juhitava protsessi matemaatilise mudeli koostamine ja regulaatori parameetrite välja arvutamine kasutades selleks erinevaid arvutusmeetodeid.
2. Katse-eksitus meetodil regulaatori parameetrite määramine.

Esimene meetod eeldab küllaltki täpset protsessi tundmist ja selle matemaatilise kirjeldamise oskust. Kuigi tänapäeva arvutustehnika võimaldab modelleerida ka väga keerukaid protsesse

ja väga lühikese ajaga välja arvutada nende juhtimiseks kasutatavate regulaatorite parameetreid, erinevad nad sageli teatud määral reaalsest ja vajavad järelhäälestust. Esimese meetodi kohta leiab täpsemat selgitust automaatjuhtimise teooriat käsitlevates õpikutes ja raamatutes.

Teine meetod tähendab seda, et regulaatori parameetrid määratakse katseliselt reaalses protsessis, muutes regulaatori parameetreid kindla algoritmi järgi ja jälgides protsessi juhtimise kvaliteedi muutust.



Joonis 5.29. PID-regulaatori häälestamise algoritm katse-eksitusmeetodil

Tabel 5.11. Sobiva regulaatori valik objekti tüübi järgi

Protsess ehk objekt	Regulaator					
	P	I	PI	PD	PID	Kaheolekuline juhtimine
Inertsivaba aperioidiline lüli (P_0)	-	X	XX	-	-	-
Aperioidiline lüli (PT_1)	X	X	X	-	-	X
Võnkelüli (PT_2)	-	-	X	-	XX	X
Hilistuslüli (PT_t)	-	X	XX	-	-	-
Hilistusega aperioidiline lüli (PT_tT_1)	X	-	X	X	X	X
Ideaalne integreeriv lüli (I_0)	X	-	X	-	-	X
Reaalne integreeriv lüli (IT_1)	-	-	X	X	XX	X
Hilistusega integreeriv lüli (IT_t)	-	-	-	X	X	-

- ebasobiv; X sobiv; XX eriti sobiv

Tabel 5.12. Sobiva regulaatori valik enim kasutatud juhtsuurustele

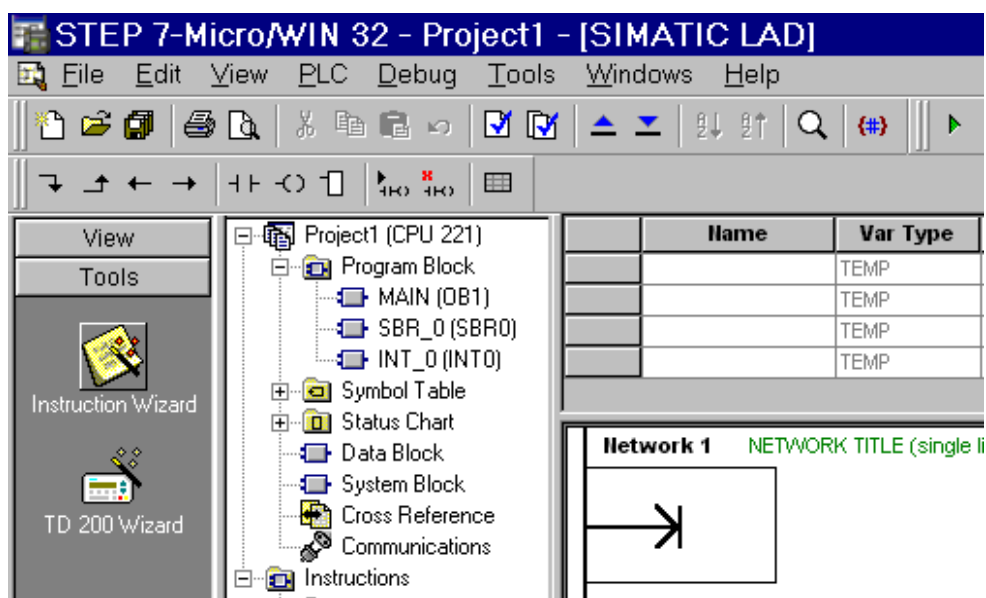
Regulaator	P	PD	PI	PID
Juhtsuurus	Püsihälbega		Juhusliku hälbega	
Temperatuur	Sobib, kui pole kõrged nõudmised ja $T_u/T_g < 0,1$	Väga sobiv	Kõige paremini sobiv keeruliste reguleerimis-ülesannete täitmiseks	
Rõhk	Sobib kui pole märkimisväärne viivitus	Ebasobiv	Kõige paremini sobiv keeruliste reguleerimis-ülesannete täitmiseks	
Vool, kulu	Vähem sobiv, kuna võimendutegur sageli liiga suur	Ebasobiv	Sobiv, kuid I-regulaator sageli parem	Eriti sobiv

5.12. PID-juhtimine tööstuskontrolleri SIMATIC S7-200 abil

PID-juhtimise realiseerimiseks tööstuskontrolleeril on mitmeid võimalusi, mis on toodud alljärgnevalt:

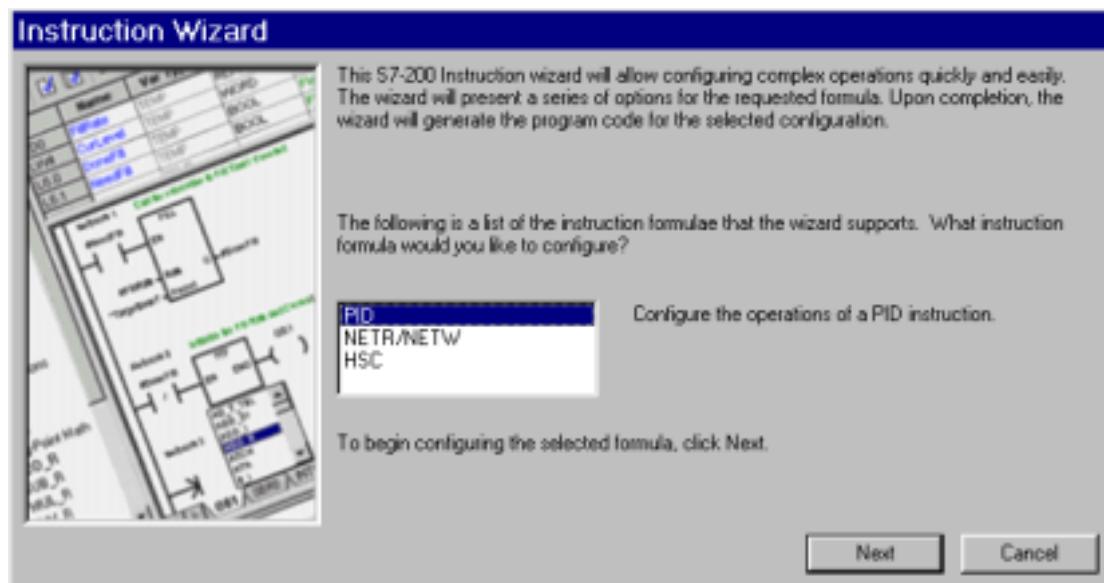
1. kasutada tarkvarapakettiga integreeritud või tarkvarapaketile lisatavat rutiini, mis, kirjutab automaatselt vastavalt teie poolt etteantud lähteandmetele programmi.
2. kasutada oma programmis tootja poolt valmistatud programm- ja/või funktsioonplokke, millel tuleb määratleda reguleerimisparameetrid
3. kirjutada PID-juhtimise programm koos vastavate arvutustoimingutega otsast lõpuni ise.

Alljärgnevalt on toodud SIMATIC S7-200 sarja tööstuskontrolleeri tarkvaraga MicroWIN integreeritud automaatse PID-juhtimise programmi kirjutamise rutiin, mida saab valida ikooni Instruction Wizard alt (joonis 5.30).



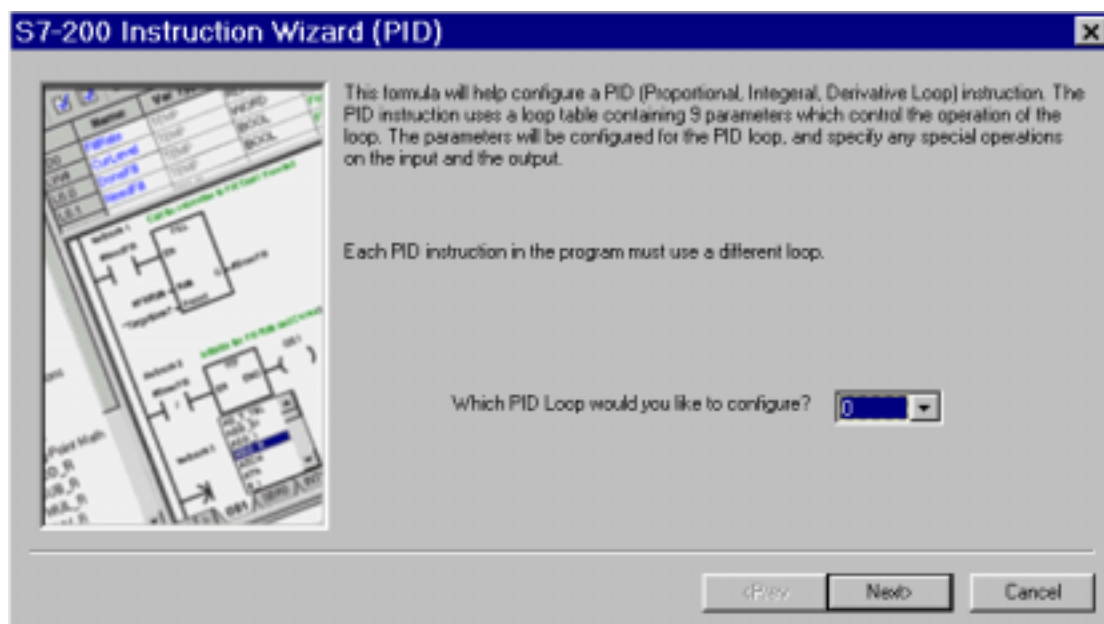
Joonis 5.30. Instruction Wizard rutiini käivitamine

S7-200 sarja tööstuskontrolleerite Instruction Wizard võimaldab määratleda ja sätestada komplekseid operatsioone ja genereerib automaatselt. Instruction Wizard rutiin võimaldab lisaks PID-juhtimisprogrammide genereerida ka andmevahetuseks ja kõrgsagedusloenduseks programme.



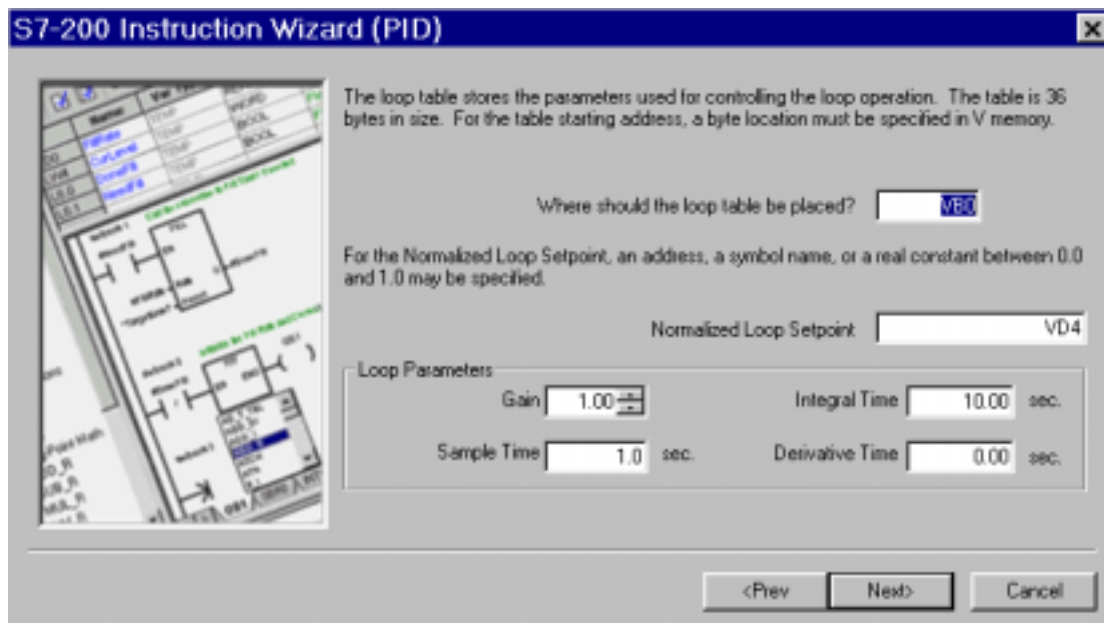
Joonis 5.31. PID-juhtimise rutiini valik

Järgneval joonisel 5.32 PID-regulaatori number mida kasutatakse. Kokku on võimalik kasutada kuni 8 PID regulaatorit eri juhtimistoiminguteks.



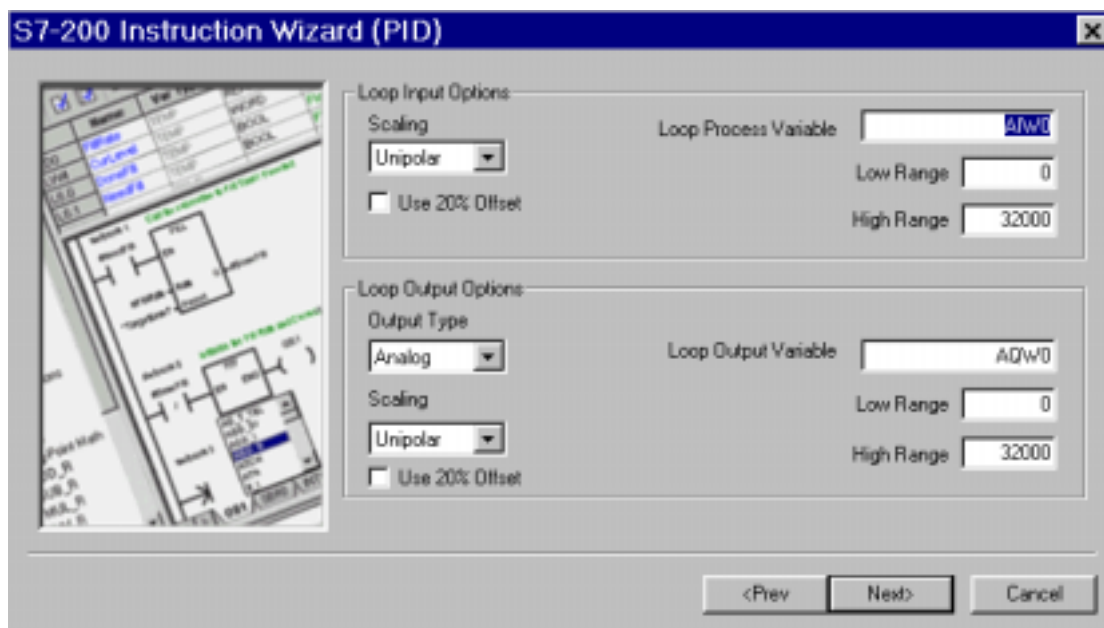
Joonis 5.32. PID-regulaatori valik

Kui PID-regulaatori number on valitud, määratakse mäluaadress kust alates salvestatakse juhtimisparameetrid. Teisena määratakse mäluaadress kus paikneb normaliseeritud etteandesuurus. Kolmandana kirjeldatakse juhtimisparameetrid nagu võimendustegur, integreerimisaeg, kvantimisaeg, differentseerimisaeg (joonis 5.33).



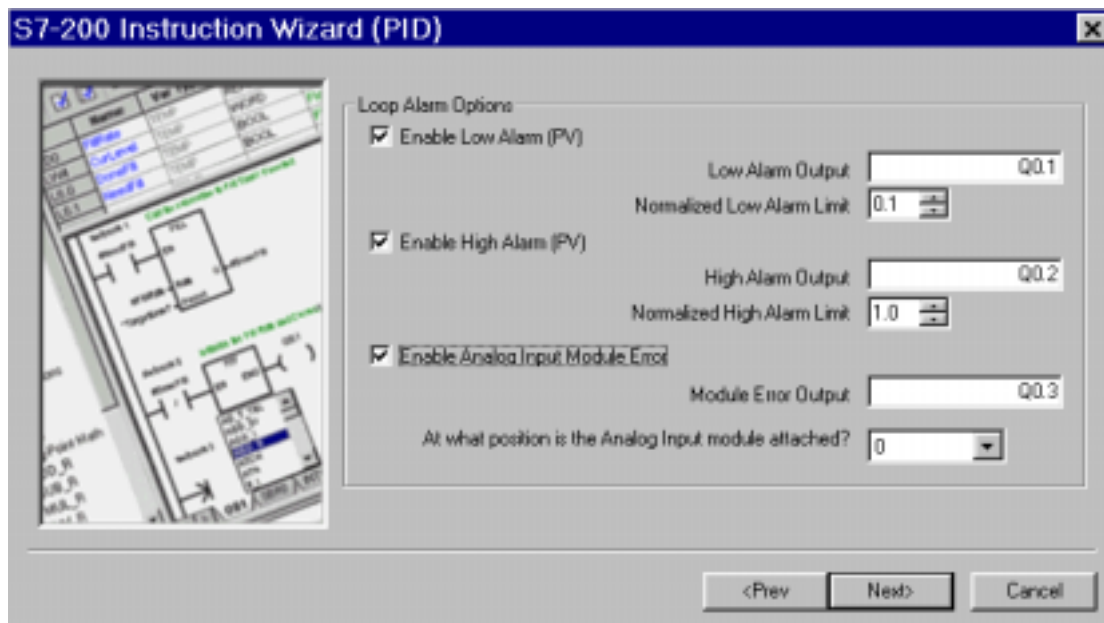
Joonis 5.33. PID-regulaatori parameetrite valik

Joonisel 5.34 toodu Wizard'i aknas määratakse analoogsisendite ja –väljundite parameetrid, kas on tegemist unipolaarse või bipolaarse sisend- või väljundsuurusega. Samuti on võimalik juhtsignaal määrata (väljund) analoog- või digitaalsignaalina.



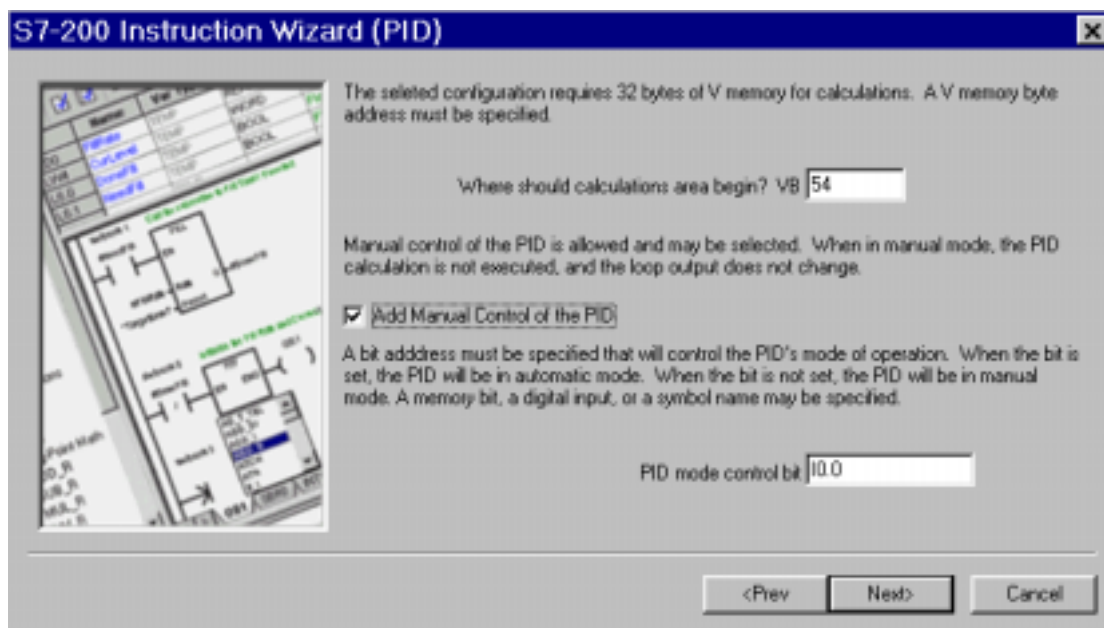
Joonis 5.34. PID regulaatoriga seotud analoogsisendite ja –väljundite määrangud

Joonisel 5.35 toodu aknas määratakse häirete esitamine. Eristatakse kolme häiret: protsessist tulev signaal alla või üle lubatud väärtuse ning analoogsisendmooduli viga. Häiretest signaliseerimiseks määratakse neile vastavad kontrolleri digitaalväljundid.



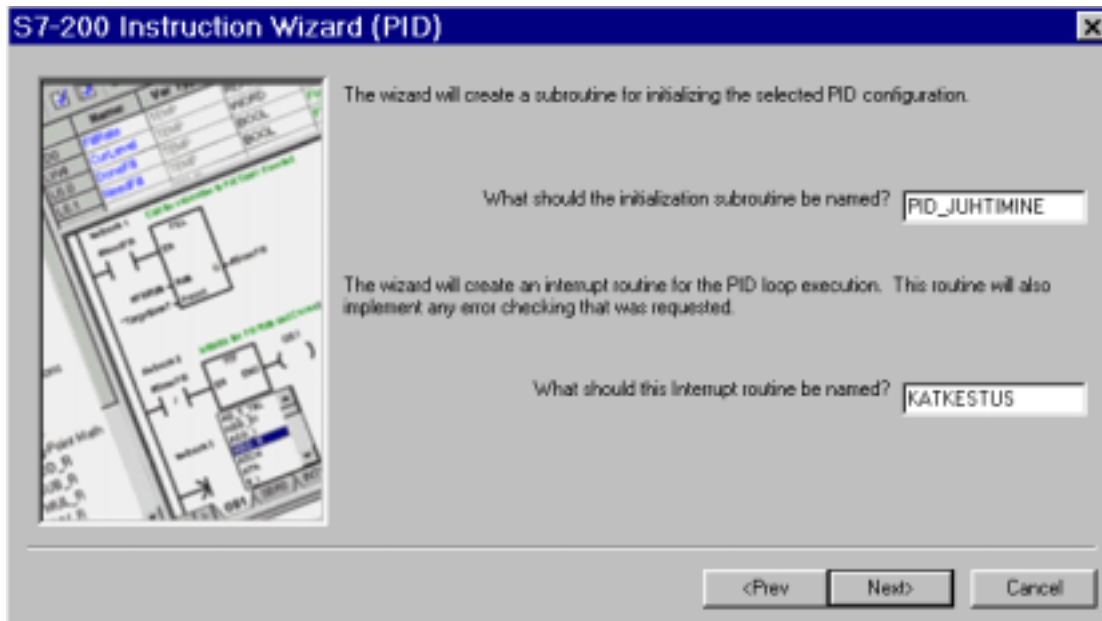
Joonis 5.35. PID-reguleerimisega seotud häirete määrangud

Järgmisena määratakse mälu piirkonna algusaadress millest alates reserveeritakse PID operatsiooni arvutustoiminguteks. Samuti on võimalik sätestada käsi- ja automaatjuhtimise ümberlülituse võimalus PID-regulaatorile.



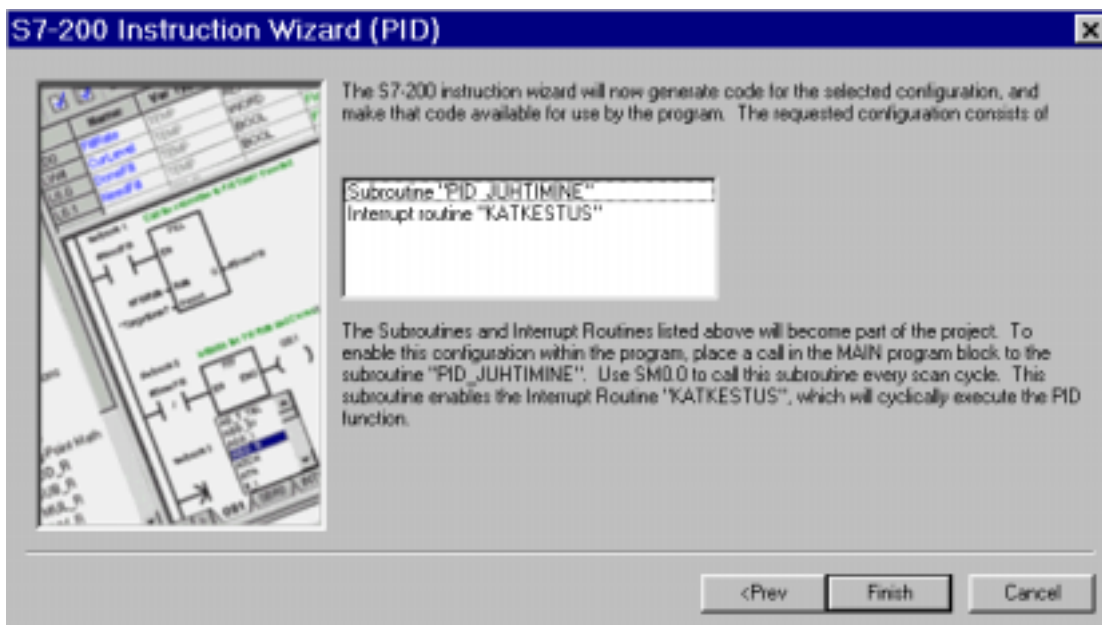
Joonis 5.36. PID-reguleerimise arvutusteks mälu piirkonna reserveerimine

Eelviimases määrangute aknas küsitakse alamprogrammiplokkide nimesid kuhu genereeritakse juhtprogramm.



Joonis 5.37. PID-reguleerimisega seotud alaprogrammidele nimetuste andmine

Viimases määrangute aknas teavitatakse veelkord sellest millised alamprogrammiplokid genereeritakse ja mis moodi tuleb edasi toimida, et need ülejäänud kontrolleri programmiga siduda.



Joonis 5.38. Ülevaade, millistest alamprogrammidest PID-reguleerimine koosneb

Vajutades nupule Finish genereeritakse vastavalt teie määrangule reguleerimiseks alamprogrammiplokid koos inglise või saksa keelsete kommentaaridega. Järgnevalt on toodud genereeritud alamprogrammiplokid koos eestikeelsete kommentaaridega.

OB1

```
NETWORK 1      //OB1-s paiknev programmi lõik
//
//NETWORK COMMENTS
//
LD      SMO.0           //alati
CALL   PID_JUHTIMINE   //kutsutakse välja alamprogramm PID_JUHTIMINE
```

SBR1

```
NETWORK 1      //SBR1-s paiknev programmi lõik
//PID initsialiseerimine ja juhtimine
//See alamprogramm initsialiseerib PID juhtimiseks kasutatavad muutujad ja
//PID-katkestusprogrammi "Katkestus".
//PID katkestusprogramm käivitatakse tsüklliselt vastavalt PID-tsükli ajale
LDN    SMO.1           //esimese kontrolleri tsükliga
JMP    0               //toimub initsialiseerimine
PID    VBO, 0
NOT
MOVW   +0, VW54        //initsialiseeri PID katkestuse loendur
MOVB   100, SMB34      //salvesta PID katkestuse intervall
ATCH   KATKESTUS, 10   //määratle katkestus PID-juhtimiseks
ENI

NETWORK 2
LBL    0
```

INT1

```
NETWORK 1      //katkestuste loenduri sisu suurendamine ühe vorra
LD      SMO.0
INCU   VW54

NETWORK 2
LDW>=  VW54, +10      //Kas on aeg PID operatsiooni teostamiseks
MOVW   +0, VW54       //kui nii, nulli loendur ja jätk
NOT
JMP    0               //kui ei, hüppa katkestusprogrammi lõppu

NETWORK 3      //Kontrollitakse kas on lubatud analoogsuuruse väärtus
LDB=   SMB9, 0        //analoogmooduli korrasoleku kontroll
NOT
O      SMO.7           //analoogsisendmooduli olemasolu kontroll
ON     SMO.4
LDN    SMO.2
AN     SMO.3
OLD
=      QO.3            //Vea esinemisel lülitatakse väljund sisse
```

```

NETWORK 4      //Arvutatakse normaliseeritud suurus ja salvestatakse PID
//jaoks reserveeritud mäliu PV asukohale
//Kui analoogmoodul on töökorras siis arvutatakse normaliseeritud väärtus ja
//salvestatakse PID jaoks reserveeritud mäliu PV asukohale
//Kui mooduli viga, siis kasutatakse eelmist kehtivat PV väärtust.
LDN   Q0.3          //Vea puudumisel
LPS           //Kustutakse tööpiirkond.
XORV   VV68, VV68   //Loetakse analoogsisendsuurus
MOVW   AIW0, VW70
&      V70.7        //kontrollitakse märgi bitti.
MOVW   16#FFFF, VV68 //Laiendatakse märgi bitti.
LRD
DTR    VD68, VDO    //teisendatakse reaalarvuks ja salvestatakse mäliu
LPP           //normaliseeritakse väärtus vahemikus 0.0 kuni 1.0
/R      32000.0, VDO

NETWORK 5
LD      SNO.0        //Alati
MOVR   VD4, VD4

NETWORK 6      //Kui protsessi suuruse väärtus on üle maksimum suuruse,
//siis omistatakse tema väärtuseks maksimumväärtus
LDR>=  1.0, VDO
NOT
MOVR   1.0, VDO

NETWORK 7      //Kui protsessi suuruse väärtus on alla miinimum suurust,
//siis omistatakse tema väärtuseks miinimumväärtus
LDR<=  0.0, VDO
NOT
MOVR   0.0, VDO

NETWORK 8      //Kontrollitakse kas on alumine häire
LDR<=  VDO, 0.1     //Kui PV is alla alumise häire piiri
=      Q0.1         //siis lülitatakse alumise häire indikaator sisse

NETWORK 9      //Kontrollitakse kas on ülemine häire
LDR>=  VDO, 1.0     //Kui PV is üle ülemise häire piiri
=      Q0.2         //siis lülitatakse ülemise häire indikaator sisse

NETWORK 10     //Automaatjuhtimise korral käivitatakse PID arvutus
LD      IO.0        //Kui on automaatjuhtimine
PID    VBO, 0       //teostatakse PID arvutus

NETWORK 11     //Skaleeritakse PID väljund ja kirjutatakse ta analoogväljundisse
LD      SNO.0        //Alati
MOVR   VDS, VD68    //salvestatakse PID-väljund tööväljale ehk mäliu
*R     32000.0, VD68 //skaleeritakse suurus
TRUNC  VD68, VD68   //teisendatakse suurus täisarvuks
MOVW   VW70, AQW0   //ja kirjutatakse analoogväljundisse

NETWORK 12
LBL    0

```

DB

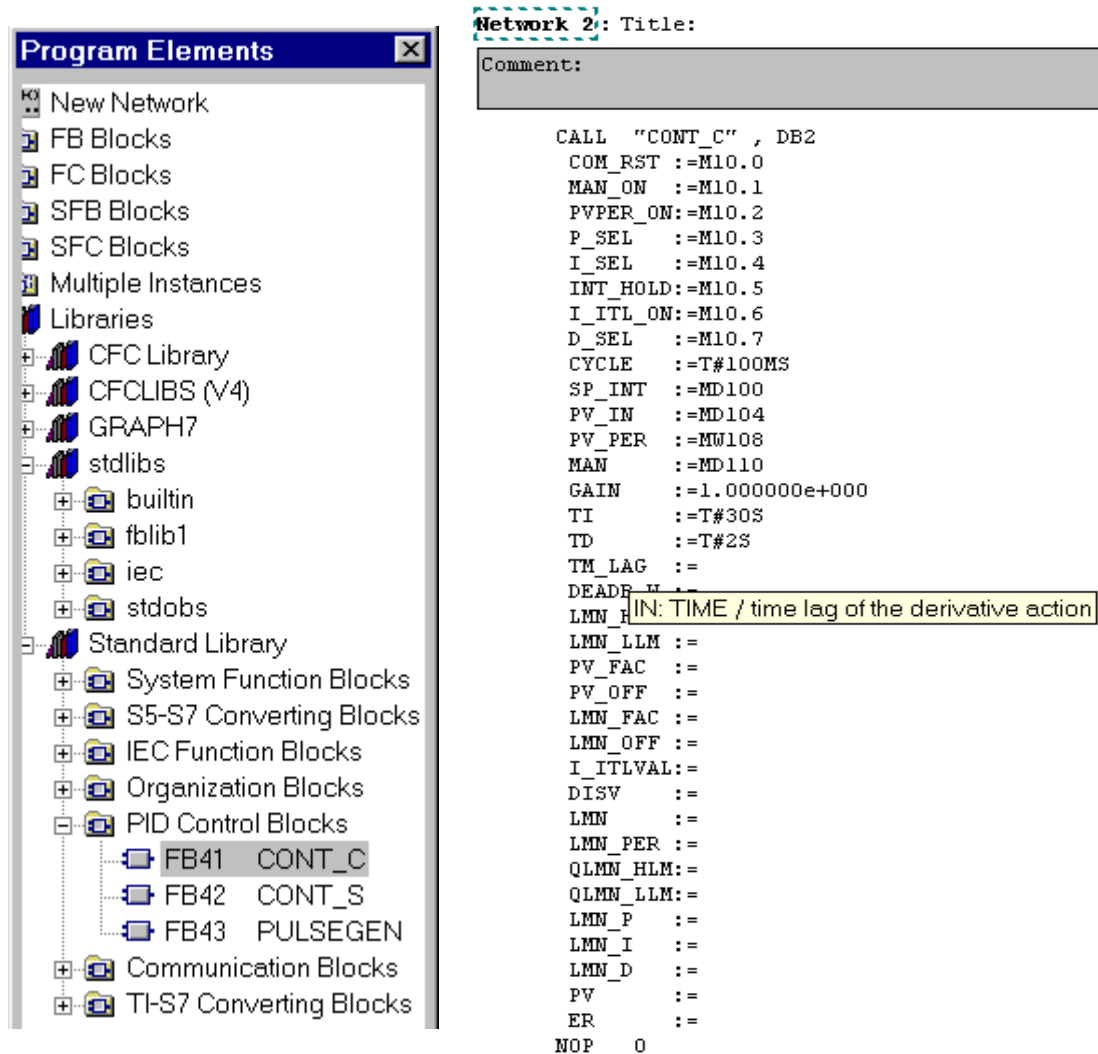
```

//
//DATA BLOCK COMMENTS
//
//Press F1 for help and example data block
//
//
//The following were generated by the S7-200 Instruction Wizard, PID Formula.
//Parameter Table for PID 0.
//
VDO    0.0          //Process Variable
VD4    0.0          //Loop Setpoint
VDS    0.0          //Calculated Loop Output
VD12   1.0          //Loop Gain
VD16   1.0          //Sample Time
VD20   10.0         //Integral Time
VD24   0.0          //Derivative Time
VD28   0.0          //Integral Sum or Bias
VD32   0.0          //Value of Process Variable stored from last execution.

```

5.13. PID-juhtimine tööstuskontrolleri SIMATIC S7-300/400 abil

S7-300/400 sarja kontrollerite tarkvara SIMATIC Manager võimaldab PID reguleerimiseks kasutada valmis kujul funktsioonplokke, kus vaid sisend- ja väljundparameetrid tuleb programmeerijal sätestada.



Program Elements

- New Network
- FB Blocks
- FC Blocks
- SFB Blocks
- SFC Blocks
- Multiple Instances
- Libraries
 - CFC Library
 - CFCLIBS (V4)
 - GRAPH7
 - stdlibs
 - builtin
 - fblib1
 - iec
 - stdobs
 - Standard Library
 - System Function Blocks
 - S5-S7 Converting Blocks
 - IEC Function Blocks
 - Organization Blocks
 - PID Control Blocks
 - FB41 CONT_C
 - FB42 CONT_S
 - FB43 PULSEGEN
 - Communication Blocks
 - TI-S7 Converting Blocks

Network 2: Title:

Comment:

```
CALL "CONT_C" , DB2
COM_RST :=M10.0
MAN_ON :=M10.1
PVPER_ON:=M10.2
P_SEL :=M10.3
I_SEL :=M10.4
INT_HOLD:=M10.5
I_ITL_ON:=M10.6
D_SEL :=M10.7
CYCLE :=T#100MS
SP_INT :=MD100
PV_IN :=MD104
PV_PER :=MW108
MAN :=MD110
GAIN :=1.000000e+000
TI :=T#30S
TD :=T#2S
TM_LAG :=
DEADR_H :=
LMN_HIN: TIME / time lag of the derivative action
LMN_LLM :=
PV_FAC :=
PV_OFF :=
LMN_FAC :=
LMN_OFF :=
I_ITLVAL:=
DISV :=
LMN :=
LMN_PER :=
QLMN_HLM:=
QLMN_LLM:=
LMN_P :=
LMN_I :=
LMN_D :=
PV :=
ER :=
NOP 0
```

Joonis 5.39. Tarkvarasse SIMATIC Manager reguleerimiseks integreeritud funktsioonplokid