

SIMATIC

Modular PID Control

Handbuch

Vorwort, Inhaltsverzeichnis

Produktübersicht
Modular PID Control **1**

Beschreibung der einzelnen
Funktionen **2**

Beispiele **3**

Technische Daten **4**

Projektiersoftware für
Modular PID Control **5**

Literaturverzeichnis **A**

Index

Sicherheitstechnische Hinweise

Dieses Handbuch enthält Hinweise, die Sie zu Ihrer persönlichen Sicherheit sowie zur Vermeidung von Sachschäden beachten müssen. Die Hinweise sind durch ein Warndreieck hervorgehoben und je nach Gefährungsgrad folgendermaßen dargestellt:



Gefahr

bedeutet, dass Tod, schwere Körperverletzung oder erheblicher Sachschaden eintreten **werden**, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.



Warnung

bedeutet, dass Tod, schwere Körperverletzung oder erheblicher Sachschaden eintreten **können**, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.



Vorsicht

bedeutet, dass eine leichte Körperverletzung oder ein Sachschaden eintreten können, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

Vorsicht

bedeutet, dass ein Sachschaden eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

Achtung

ist eine wichtige Information über das Produkt, die Handhabung des Produktes oder den jeweiligen Teil der Dokumentation, auf den besonders aufmerksam gemacht werden soll.

Qualifiziertes Personal

Inbetriebsetzung und Betrieb eines Gerätes dürfen nur von **qualifiziertem Personal** vorgenommen werden. Qualifiziertes Personal im Sinne der sicherheitstechnischen Hinweise dieses Handbuchs sind Personen, die die Berechtigung haben, Geräte, Systeme und Stromkreise gemäß den Standards der Sicherheitstechnik in Betrieb zu nehmen, zu erden und zu kennzeichnen.

Bestimmungsgemäßer Gebrauch

Beachten Sie folgendes:



Warnung

Das Gerät darf nur für die im Katalog und in der technischen Beschreibung vorgesehenen Einsatzfälle und nur in Verbindung mit von Siemens empfohlenen bzw. zugelassenen Fremdgeräten und -komponenten verwendet werden.

Der einwandfreie und sichere Betrieb des Produktes setzt sachgemäßen Transport, sachgemäße Lagerung, Aufstellung und Montage sowie sorgfältige Bedienung und Instandhaltung voraus.

Marken

SIMATIC®, SIMATIC HMI® und SIMATIC NET® sind Marken der SIEMENS AG.

Die übrigen Bezeichnungen in dieser Schrift können Marken sein, deren Benutzung durch Dritte für deren Zwecke die Rechte der Inhaber verletzen kann.

Copyright © Siemens AG 1999 - 2003 All rights reserved

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts ist nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte vorbehalten, insbesondere für den Fall der Patenterteilung oder GM-Eintragung

Siemens AG
Bereich Automation and Drives
Geschäftsgebiet Industrial Automation Systems
Postfach 4848, D- 90327 Nürnberg

Siemens Aktiengesellschaft

Haftungsausschluss

Wir haben den Inhalt der Druckschrift auf Übereinstimmung mit der beschriebenen Hard- und Software geprüft. Dennoch können Abweichungen nicht ausgeschlossen werden, so dass wir für die vollständige Übereinstimmung keine Gewähr übernehmen. Die Angaben in dieser Druckschrift werden regelmäßig überprüft, und notwendige Korrekturen sind in den nachfolgenden Auflagen enthalten. Für Verbesserungsvorschläge sind wir dankbar.

© Siemens AG 2003
Technische Änderungen bleiben vorbehalten.

A5E00xxxxx-01

Vorwort

Zweck des Handbuchs

Dieses Handbuch unterstützt Sie bei der Auswahl, Konfiguration, Parametrierung eines für Ihre Regelungsaufgabe optimalen Reglerbausteins.

Sie werden mit der Funktionsweise des Reglerbausteins sowie der Hantierung der Projektiersoftware vertraut gemacht.

Erforderliche Grundkenntnisse

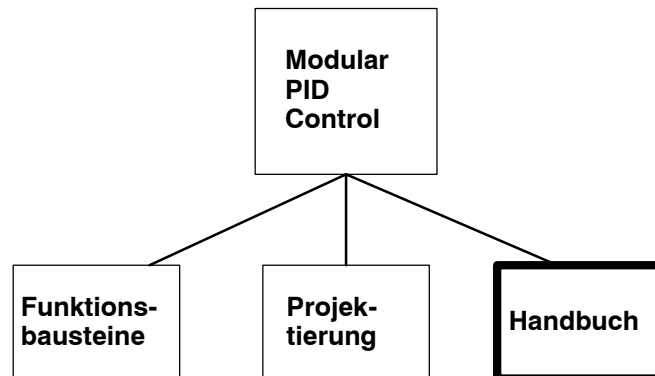
Zum Verständnis des Handbuchs sind allgemeine Kenntnisse auf dem Gebiet der Automatisierungstechnik und der Regelungstechnik erforderlich.

Außerdem werden Kenntnisse über die Verwendung von Computern oder PC-ähnlichen Arbeitsmitteln (z. B. Programmiergeräten) unter dem Betriebssystem Windows vorausgesetzt. Da Modular PID Control auf der Basissoftware STEP 7 aufsetzt, sollten Sie auch Kenntnisse im Umgang mit der Basissoftware haben. Diese werden im Handbuch *"Programmieren mit STEP 7 V5.1"* vermittelt.

Gültigkeitsbereich des Handbuchs

Das Handbuch ist gültig für die Pakete Modular PID Control FB V5.0 und Modular PID Control Tool V5.0.

Einordnung in die Informationslandschaft



Das Softwarepaket Modular PID Control umfaßt drei Einzelprodukte:

- Das Produkt “Modular PID Control FB” umfaßt Funktionsbausteine und Beispiele
- Das Produkt “Modular PID Control Tool” enthält hauptsächlich die Werkzeuge zum Konfigurieren der Reglerbausteine.

Das Produkt wird im folgenden “Projektiersoftware” genannt.

Leserkreis

- S7-Programmierer
- Regelungsprogrammierer
- Bedienpersonal
- Servicepersonal

Lesekonventionen

Damit Sie die Informationen in diesem Handbuch leichter finden können, wurden bestimmte Konventionen festgelegt:

- Überfliegen Sie zunächst die Überschriften und die Titel am linken Rand, um schnell Hinweise auf den Inhalt zu erhalten.
- Ein Themenblock beantwortet entweder eine Fragestellung zur Funktionalität oder gibt Auskunft über erforderliche bzw. empfohlene Abläufe.
- Verweise auf weitere Behandlung eines Themas in anderen Kapiteln werden mit (*siehe Kapitel x.y*) dargestellt. Hinweise auf andere Dokumentationen sind mit Hilfe von Literaturnummern in Schrägstrichen /.../ angegeben. Anhand dieser Nummern können Sie dem Literaturverzeichnis am Ende des Handbuchs den genauen Titel der Dokumentation entnehmen.
- Ein Glossar mit den wichtigsten Regelungsbegriffen finden Sie im Handbuch "Standard PID Control"

Weitere Unterstützung

Bei Fragen zur Nutzung der im Handbuch beschriebenen Produkte, die Sie hier nicht beantwortet finden, wenden Sie sich bitte an Ihren Siemens-Ansprechpartner in den für Sie zuständigen Vertretungen und Geschäftsstellen.

Ihren Ansprechpartner finden Sie unter:

<http://www.siemens.com/automation/partner>

Trainingscenter

Um Ihnen den Einstieg in das das Automatisierungssystem SIMATIC S7 zu erleichtern, bieten wir entsprechende Kurse an. Wenden Sie sich bitte an Ihr regionales Trainingscenter oder an das zentrale Trainingscenter in D 90327 Nürnberg.

Telefon: +49 (911) 895-3200.

Internet: <http://www.sitrain.com>

A&D Technical Support

Weltweit erreichbar zu jeder Tageszeit:



| | | |
|---|---|--|
| <p>Weltweit (Nürnberg) Technical Support</p> <p>Ortszeit: 0:00 bis 24:00 / 365 Tage Telefon: +49 (180) 5050-222 Fax: +49 (180) 5050-223 E-Mail: adsupport@siemens.com GMT: +1:00</p> | | |
| <p>Europa / Afrika (Nürnberg) Authorization</p> <p>Ortszeit: Mo.-Fr. 8:00 bis 17:00 Telefon: +49 (180) 5050-222 Fax: +49 (180) 5050-223 E-Mail: adsupport@siemens.com GMT: +1:00</p> | <p>United States (Johnson City) Technical Support and Authorization</p> <p>Ortszeit: Mo.-Fr. 8:00 bis 17:00 Telefon: +1 (423) 262 2522 Fax: +1 (423) 262 2289 E-Mail: simatic.hotline@sea.siemens.com GMT: -5:00</p> | <p>Asien / Australien (Peking) Technical Support and Authorization</p> <p>Ortszeit: Mo.-Fr. 8:00 bis 17:00 Telefon: +86 10 64 75 75 75 Fax: +86 10 64 74 74 74 E-Mail: adsupport.asia@siemens.com GMT: +8:00</p> |
| <p>Technical Support und Authorization sprechen generell Deutsch und Englisch.</p> | | |

Service & Support im Internet

Zusätzlich zu unserem Dokumentations-Angebot bieten wir Ihnen im Internet unser komplettes Wissen online an.

<http://www.siemens.com/automation/service&support>

Dort finden Sie:

- der Newsletter, der Sie ständig mit den aktuellsten Informationen zu Ihren Produkten versorgt.
- die für Sie richtigen Dokumente über unsere Suche in Service & Support.
- ein Forum in welchem Anwender und Spezialisten weltweit Erfahrungen austauschen.
- Ihren Ansprechpartner für Automation & Drives vor Ort.
- Informationen über Vor-Ort Service, Reparaturen, Ersatzteile. Vieles mehr steht für Sie unter dem Begriff "Leistungen" bereit.

Inhaltsverzeichnis

| | | |
|----------|--|------------|
| | Vorwort | iii |
| 1 | Produktübersicht Modular PID Control | 1-1 |
| 1.1 | Das Produkt Modular PID Control | 1-1 |
| 1.2 | Die Komponenten von Modular PID Control | 1-2 |
| 1.3 | Einsatzumgebung und Anwendungsbereich | 1-3 |
| 2 | Beschreibung der einzelnen Funktionen | 2-1 |
| 2.1 | Allgemein | 2-1 |
| 2.1.1 | A_DEAD_B: adaptive dead band (adaptive Totzone) | 2-2 |
| 2.1.2 | CRP_IN: change range periphery input (analogen Eingangswert wandeln) | 2-8 |
| 2.1.3 | CRP_OUT: change range periphery output (analogen Ausgangswert wandeln) | 2-11 |
| 2.1.4 | DEAD_T: dead time (Totzeit) | 2-13 |
| 2.1.5 | DEADBAND: dead band (Totzone) | 2-17 |
| 2.1.6 | DIF: differentiator (Differenzierer) | 2-20 |
| 2.1.7 | ERR_MON: error-signal monitoring (Regeldifferenzüberwachung) | 2-24 |
| 2.1.8 | INTEG: integrator (Integrierer) | 2-28 |
| 2.1.9 | LAG1ST: first-order lag element (Verzögerungsglied 1. Ordnung) | 2-33 |
| 2.1.10 | LAG2ND: second-order lag element (Verzögerungsglied 2. Ordnung) | 2-37 |
| 2.1.11 | LIMALARM: limit alarm (Grenzwertmelder) | 2-41 |
| 2.1.12 | LIMITER: limiter (Begrenzung) | 2-45 |
| 2.1.13 | LMNGEN_C: output continuous PID controller (Ausgabe des kontinuierlichen PID-Reglers) | 2-48 |
| 2.1.14 | LMNGEN_S: output PID step controller (Ausgabe PID-Schrittregler) | 2-55 |
| 2.1.15 | LP_SCHED: loop scheduler (Regleraufrufverteiler) | 2-65 |
| 2.1.16 | NONLIN: nonlinear static function (nichtlineare statische Kennlinie) | 2-72 |
| 2.1.17 | NORM: physical norm (physikalische Normierung) | 2-77 |
| 2.1.18 | OVERRIDE: override controlling (Override Regelung) | 2-79 |
| 2.1.19 | PARA_CTL: parameter controlling (Parametersteuerung) | 2-82 |
| 2.1.20 | PID: PID-algorithm (PID-Algorithmus) | 2-86 |
| 2.1.21 | PULSEGEN: pulse generator (Impulsformer) | 2-97 |
| 2.1.22 | RMP_SOAK: ramp soak (Zeitplangeber) | 2-107 |
| 2.1.23 | ROC_LIM: rate of change limiter (Rampe bzw. Hochlaufgeber) | 2-118 |
| 2.1.24 | SCALE: linear scaling (lineare Skalierung) | 2-127 |
| 2.1.25 | SP_GEN: setpoint generator (Sollwertgeber) | 2-129 |
| 2.1.26 | SPLT_RAN: split-ranging (Bereichsaufteilung) | 2-133 |
| 2.1.27 | SWITCH: switch (Analogwertschalter) | 2-136 |

| | | |
|----------|--|------------|
| 3 | Beispiele | 3-1 |
| 3.1 | Anwendung von Modular PID Control | 3-1 |
| 3.2 | Beispiel 1: Festwertregler mit schaltendem Ausgang für integrierende Stellglieder mit Streckensimulation | 3-4 |
| 3.2.1 | PIDCTR_S: Festwertregler mit schaltendem Ausgang für integrierende Stellglieder | 3-6 |
| 3.2.2 | PROC_S: Regelstrecke für Schrittreger | 3-7 |
| 3.3 | Beispiel 2: Festwertregler mit kontinuierlichem Ausgang mit Streckensimulation | 3-8 |
| 3.3.1 | PIDCTR_C: Festwertregler mit kontinuierlichem Ausgang für integrierende Stellglieder | 3-9 |
| 3.3.2 | PROC_C: Regelstrecke für kontinuierlichen Regler | 3-10 |
| 3.4 | Beispiel 3: Festwertregler mit schaltendem Ausgang für proportionale Stellglieder mit Streckensimulation | 3-11 |
| 3.4.1 | PIDCTR: Führungsregler für kontinuierliche Regler mit Impulsformer ... | 3-13 |
| 3.4.2 | PROC_P: Regelstrecke für kontinuierlichen Regler mit Impulsformer ... | 3-14 |
| 3.5 | Beispiel 4: Einschleifiger Verhältnisregler (RATIOCTR) | 3-15 |
| 3.6 | Beispiel 5: Mehrschleifiger Verhältnisregler | 3-17 |
| 3.7 | Beispiel 6: Mischungsregler | 3-20 |
| 3.8 | Beispiel 7: Kaskadenregler | 3-23 |
| 3.9 | Beispiel 8: Regler mit Vorsteuerung (CTRC_PRE) | 3-25 |
| 3.10 | Beispiel 9: Regler mit Störgrößenaufschaltung (CTR_C_FF) | 3-27 |
| 3.11 | Beispiel 10: Bereichsauswahlregler (SPLITCTR) | 3-29 |
| 3.12 | Beispiel 11: Ablöseregler (OVR_CTR) | 3-32 |
| 3.13 | Beispiel 12: Mehrgrößenregler | 3-35 |
| 4 | Technische Daten | 4-1 |
| 4.1 | Laufzeiten | 4-1 |
| 4.2 | Arbeitsspeicherbelegung | 4-2 |
| 4.3 | Schätzformeln | 4-4 |
| 5 | Projektiersoftware für Modular PID Control | 5-1 |
| A | Literaturverzeichnis | A-1 |
| 6 | Index | |

Produktübersicht Modular PID Control

1

1.1 Das Produkt Modular PID Control

Konzept von Modular PID Control

Das Softwareprodukt "Modular PID Control" besteht aus einem Satz von **Funktionsbausteinen** (FBs) und **Funktionen** (FCs), in denen die Algorithmen zur Bildung von regelungstechnischen Funktionen enthalten sind. Es handelt sich also um eine reine Softwareregung, bei der Sie durch Verschalten einzelner Bausteine Reglerstrukturen realisieren können.

Hinzu kommen vorkonfektionierte Reglerstrukturen (Einschleifiger Festwertregler, Verhältnisregler usw.) in Form von Beispielen, die Sie auf Ihre konkrete Regelungsaufgabe anpassen können.

Bei Bearbeitung von vielen Regelkreisen, die je nach Trägheit der jeweiligen Regelstrecke unterschiedlich oft - dabei aber äquidistant - bearbeitet werden sollen, steht ein **Regleraufrufverteiler** (Loop Scheduler = LP_SCHED) zur Verfügung, mit dessen Hilfe die Konfiguration umfangreicher Anlagenregelungen sehr übersichtlich und deshalb einfach wird. Außerdem wird eine gleichmäßige Auslastung der CPU gewährleistet.

Zur Inbetriebnahme und zum Testen einzelner Regelkreise gibt es die Projektiersoftware "Modular PID Control Tool". Es enthält ein Kreisbild, einen Kurvenschreiber zum Bedienen und Beobachten der Prozeßgrößen und einen Algorithmus zum Optimieren der PID-Parameter.

Übersicht über die Grundfunktionen

Bei vielen Regelungsaufgaben steht nicht allein der klassische PID-Regler als prozeßbeeinflussendes Element im Vordergrund, sondern es werden auch hohe Anforderungen an die Signalverarbeitung gestellt.

Ein mit Hilfe des Softwarepakets "Modular PID Control" gebildeter Regler setzt sich deshalb aus einer Reihe von Teilfunktionen zusammen, die von Ihnen separat parametrisiert und verschaltet werden können. Zusätzlich zum eigentlichen Regler mit dem PID-Algorithmus sind auch Funktionen zur Aufbereitung des Soll- und Istwertes sowie zur Nachbearbeitung der errechneten Stellgröße integriert.

1.2 Die Komponenten von Modular PID Control

Modular PID Control FB

Der Lieferumfang von "Modular PID Control FB" besteht aus einer Bausteinbibliothek mit Funktionsbausteinen sowie 12 vorkonfigurierten Beispielen aus der Regelungstechnik.

Sie können die Software mit dem SETUP-Programm auf PG/PC installieren. Die Online-Hilfe liefert Ihnen Nachschlagewissen zu Teilfunktionen und Einzelparametern während der praktischen Aufgabenbearbeitung.

Modular PID Control Tool

Mit dem Softwarepaket "Modular PID Control Tool" können Sie Ihre Reglerstruktur in Betrieb nehmen und die PID-Parameter optimieren.

Die Projektiersoftware enthält ein Kreisbild, einen Kurvenschreiber und einen Algorithmus zum Einstellen bzw. Optimieren der PID-Reglerparameter. Eine detaillierte Beschreibung der Projektiersoftware finden Sie im Kapitel 5.

Modular PID Control Handbuch

Das Handbuch liegt Ihnen gerade vor. Detailinformationen zur Struktur entnehmen Sie bitte dem Inhaltsverzeichnis.

1.3 Einsatzumgebung und Anwendungsbereich

Hardwareumgebung

Die mit dem Softwarepaket "Modular PID Control" erzeugten Regler sind auf den Zielsystemen (CPU mit Gleitpunkt und Weckalarm) der S7-300- und der S7-400-Familie sowie Win AC ablauffähig.

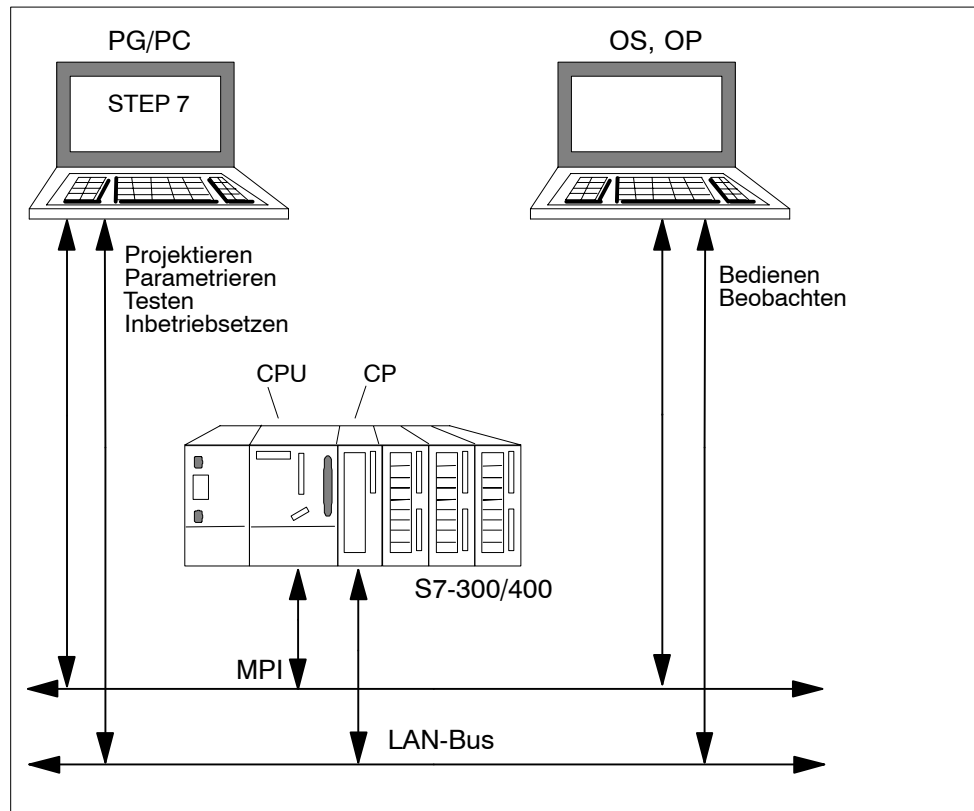


Bild 1-1 Einsatzumgebung von "Modular PID Control"

Softwareumgebung

Modular PID Control ist für den Einsatz in der Programmgruppe STEP 7 konzipiert.

Die Erstellungssoftware für Modular PID Control kann lokal auf einem PG/PC oder aber in einem Netz auf einem zentralen Netzlaufwerk installiert werden.

Funktionsspektrum der Modular PID Control

Es können sowohl träge Strecken (Temperatur, Füllstände usw.) als auch sehr schnelle Strecken (Durchflüsse, Drehzahlen usw.) geregelt werden. Folgende Reglertypen können realisiert werden:

- kontinuierliche PID-Regler
- PID-Schrittregler für integrierende Stellglieder
- Puls-Pause-Regler

Sie können zu einer der folgenden Reglerstrukturen verschaltet werden:

- Festwertregler
- Kaskadenregler
- Verhältnisregler
- Mischungsregler
- Splitrange-Regler
- Override-Regler
- Regler mit Störgrößenverarbeitung
- Mehrgrößenregler

Beschreibung der einzelnen Funktionen

2

2.1 Allgemein

Konventionen zu Parameter- und Blocknamen im Blockschaltbild

Für die Bezeichnungen der Parameter wurden nicht mehr als 8 Zeichen gewählt.

Es gelten die folgenden Konventionen zur Bezeichnung der Parameter:

Erstbuchstabe:

| | |
|------|--|
| Q | allgemeiner Ausgang vom Typ BOOL (boolsche Größe) |
| SP | setpoint (Führungsgröße) |
| PV | process variable (Meßwert oder Regelgröße) |
| LMN | manipulated variable (Stellgröße oder auszugebendes analoges Ausgangssignal) |
| DISV | disturbance variable (Störgröße) |

Folgebuchstaben:

| | |
|-----|---|
| MAN | manual value (Handwert) |
| INT | intern |
| EXT | extern |
| _ON | BOOL-Größe zum Einschalten einer Funktion |

Aufrufdaten

Die meisten Bausteine von Modular PID Control benötigen regelkreisspezifische Aufrufdaten wie Neustartbit und Abtastzeit. An den Eingängen COM_RST und CY-CLE werden diese Werte übertragen.

Bemerkungen zu den Baustein-Parameterleisten (Eingangs-, Ausgangs- und Durchgangparameter)

- **Vorbelegung:** Hier stehen die Vorbelegungswerte bei Neuerstellung einer Instanz.
- **Zulässiger Wertebereich:** Die Werte am Eingangsparameter sollten den zulässigen Wertebereich nicht überschreiten. Eine Überprüfung des Bereiches innerhalb der Bausteinabarbeitung erfolgt **nicht**. Bei Angabe Technischer Wertebereich ist eine physikalische Größe gemeint, deren Wert zwischen ca. $\pm 10^6$ liegen sollte.

2.1.1 A_DEAD_B: adaptive dead band (adaptive Totzone)

Anwendungsbereich

Bei verrauschtem Istwert wird bei optimal eingestelltem Regler der Rauschanteil auch am Reglerausgang wirksam; dies führt durch hohe Schalzhäufigkeit (Schrittregler) zu einem erhöhten Verschleiß des Stellgliedes. Durch Unterdrückung des Rauschanteils wird ein Oszillieren des Reglerausgangs vermieden.

Blockschaltbild

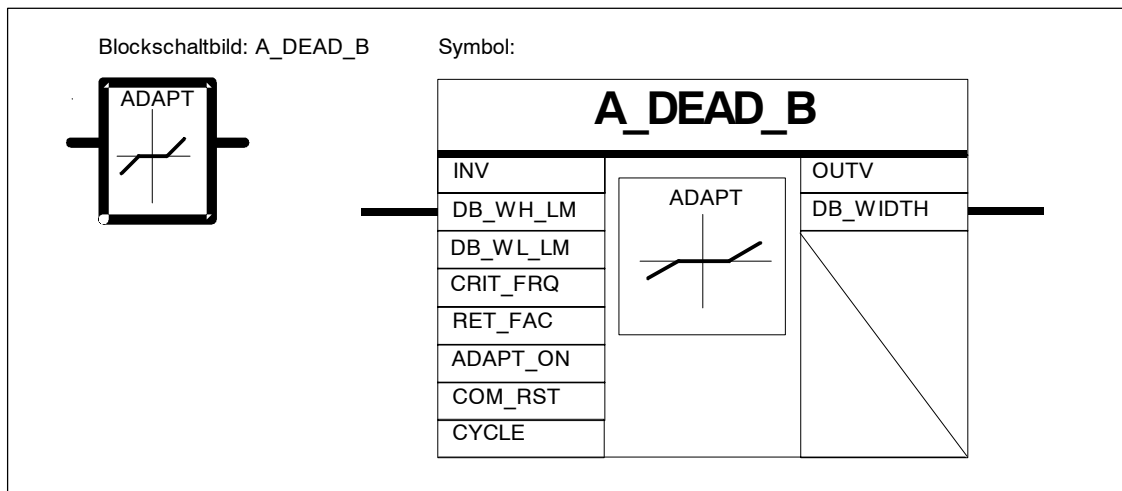


Bild 2-1 A_DEAD_B, Blockschaltbild und Symbol

Funktionsbeschreibung

Der Baustein filtert höherfrequente Störsignale aus der Regeldifferenz. Er bildet einen unempfindlichen Bereich um den Nullpunkt. Liegt die Eingangsgröße innerhalb dieses Bereiches so wird am Ausgang Null ausgegeben. Die Bereichsbreite wird der Amplitude des Störsignals automatisch angepaßt.

Der Baustein arbeitet nach der Funktion:

$$\text{OUTV} = \text{INV} + \text{DB_WIDTH} \quad \text{für} \quad \text{INV} < -\text{DB_WIDTH}$$

$$\text{OUTV} = 0.0 \quad \text{für} \quad -\text{DB_WIDTH} \leq \text{INV} \leq +\text{DB_WIDTH}$$

$$\text{OUTV} = \text{INV} - \text{DB_WIDTH} \quad \text{für} \quad \text{INV} > +\text{DB_WIDTH}$$

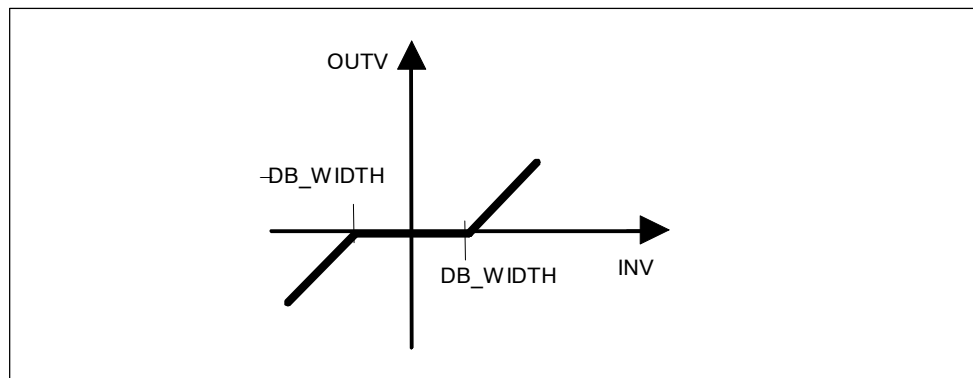


Bild 2-2 $\text{OUTV} = f(\text{INV})$

Adaption der Totzone

Aus Stabilitätsgründen wird die wirksame Totzonenbreite DB_WIDTH nach unten auf den an der Eingangsleiste vorgebbaren Wert DB_WL_LM begrenzt. Überschreitet das vom Meßrauschen überlagerte Eingangssignal INV die augenblicklich eingestellte Totzonenbreite in negativer (1), positiver (2) und wieder negativer Richtung (3) innerhalb der Periodendauer $1/CRIT_FRQ$, wird die wirksame Totzonenbreite um den Betrag 0.1 erhöht (vergl. Bild 2–4). Der Vorgang wird bei positiver oder negativer Überschreitung gestartet. Bei jedem weiteren Überschreiten (3 → 4) der gegenüberliegenden Totzonenbreite innerhalb der halben Periodendauer wird sie erneut um 0.1 vergrößert. Dieser Vorgang wiederholt sich so oft, bis die Totzonenbreite mit der Amplitude des Meßrauschens übereinstimmt. Um nicht beliebig große Eingangssignale zu unterdrücken, wird die wirksame Totzonenbreite durch den Eingang DB_WH_LM nach oben begrenzt. Tritt dagegen innerhalb der Zeit $RET_FAC * 1/CRIT_FRQ$ kein Überschreiten der Totzonenbreite auf, so wird sie um den Betrag 0.1 zurückgenommen.

$CRIT_FRQ$ gibt dabei die Grenzfrequenz an, ab der ein Signalanteil als Störung aufgefaßt wird. Sie wird nach oben und unten wie folgt begrenzt:

$$0.01 \leq CRIT_FRQ \leq 1/(3 * CYCLE) \quad \text{mit } CYCLE: \text{ Abtastzeit in sec}$$

Der Parameter RET_FAC legt das Vielfache von $1/CRIT_FRQ$ fest, nach dem die Totzonenbreite wieder verkleinert wird.

Die Adaptionlogik arbeitet nur, wenn die Eingangsgröße ohne Störanteil nahe bei Null liegt.

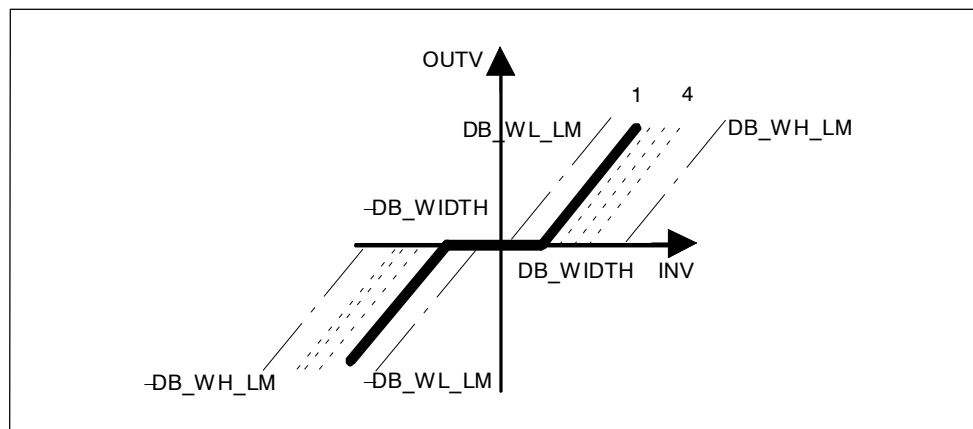


Bild 2-3 Adaption der Totzone

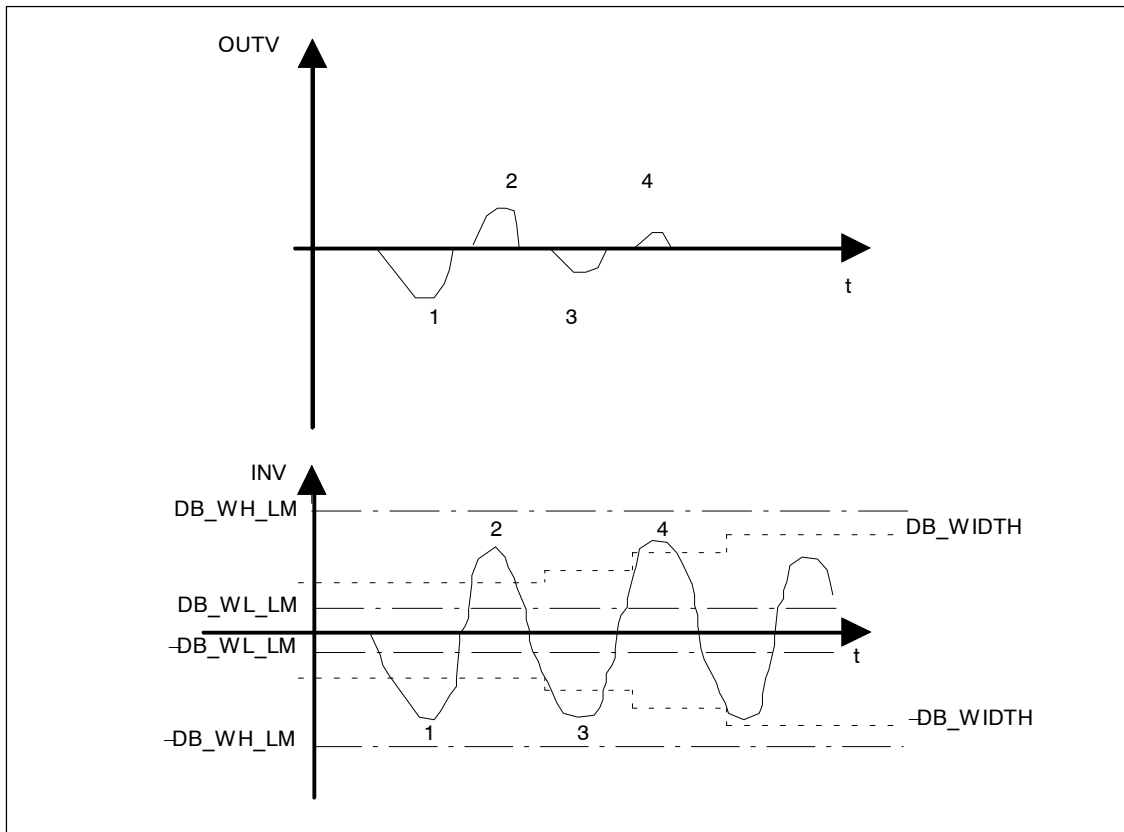


Bild 2-4 Adaption der Totzone

Eingangsparameter

Die folgende Tabelle zeigt Ihnen Datentyp und Struktur der Eingangsparameter von A_DEAD_B.

Tabelle 2-1 Eingangsparameter von A_DEAD_B

| Daten- typ | Parameter | Kommentar englisch deutsch | zulässiger Wertebereich | Vorbelegung |
|---------------|-----------|---|---|-------------|
| REAL | INV | input variable Eingangsgröße | technischer Wertebereich | 0.0 |
| REAL | DB_WH_LM | dead band width high limit obere Begrenzung der Totzonenbreite | tech. Wert. > DB_WL_LM | 5.0 |
| REAL | DB_WL_LM | dead band width low limit untere Begrenzung der Totzonenbreite | tech. Wert. < DB_WH_LM | 1.0 |
| REAL | CRIT_FRQ | critical frequency Grenzfrequenz | ≥ 0.01 und $\leq 1/(3 * CYCLE)$ | 0.1 |
| INT | RET_FAC | return factor Rücknahmefaktor | ≥ 1 | 1 |
| BOOL | ADAPT_ON | adaptive algorithm on Adaptionsalgorithmus einschalten | | FALSE |
| BOOL | COM_RST | complete restart Neustart | | FALSE |
| TIME | CYCLE | sample time Abtastzeit | ≥ 1 ms | T#1s |

Ausgangsparameter

Die folgende Tabelle zeigt Ihnen Datentyp und Struktur der Ausgangsparameter von A_DEAD_B.

Tabelle 2-2 Ausgangsparameter von A_DEAD_B

| Daten- typ | Parameter | Kommentar englisch deutsch | Vorbelegung |
|---------------|-----------|--|-------------|
| REAL | OUTV | output variable Ausgangsgröße | 0.0 |
| REAL | DB_WIDTH | effective dead band width wirksame Totzonenbreite | 0.0 |

Neustart

Bei Neustart wird $OUTV=0.0$ und die wirksame Totzonenbreite $DB_WIDTH = DB_WL_LM$ gesetzt.

Normalbetrieb

Im Normalbetrieb gelten für die Adaption folgende Bedingungen:

- Adaption Aus

Wenn die Adaption ausgeschaltet ist ($ADAPT_ON = FALSE$), wird als wirksame Totzonenbreite DB_WIDTH der letzte DB_WIDTH -Wert eingefroren.

- Adaption Ein

Mit $ADAPT_ON = TRUE$ kann ein Adaptionalgorithmus zugeschaltet werden, der die wirksame Totzonenbreite berechnet. Dadurch wird die Totzonenbreite an die Amplitude des Rauschsignals, das der Eingangsgröße überlagert ist, angepaßt und damit der Rauschanteil auch bei schwankender Rauschamplitude unterdrückt.

Bei azyklischem Bausteinaufruf muß die Adaption abgeschaltet werden ($ADAP_ON = FALSE$).

Bausteininterne Begrenzungen

Die Werte der Eingangsparameter werden im Baustein nicht begrenzt; eine Prüfung der Parameter findet nicht statt.

Beispiel

Wird bei der Inbetriebsetzung aufgrund von Meßrauschen die Adaption eingeschaltet und stellt sich nach einer gewissen Zeit eine stabile Totzonenbreite ein, so kann die Adaption ausgeschaltet werden. Die über die Adaption eingestellte Totzonenbreite bleibt erhalten, solange kein Neustart durchgeführt wird.

2.1.2 CRP_IN: change range periphery input (analogen Eingangswert wandeln)

Anwendungsbereich

Der Baustein paßt den Wertebereich der analogen Peripherie an die interne Darstellung der Modularen Regelung an. Der Baustein kann z.B. im Istwertzweig aufgerufen werden.

Blockschaltbild

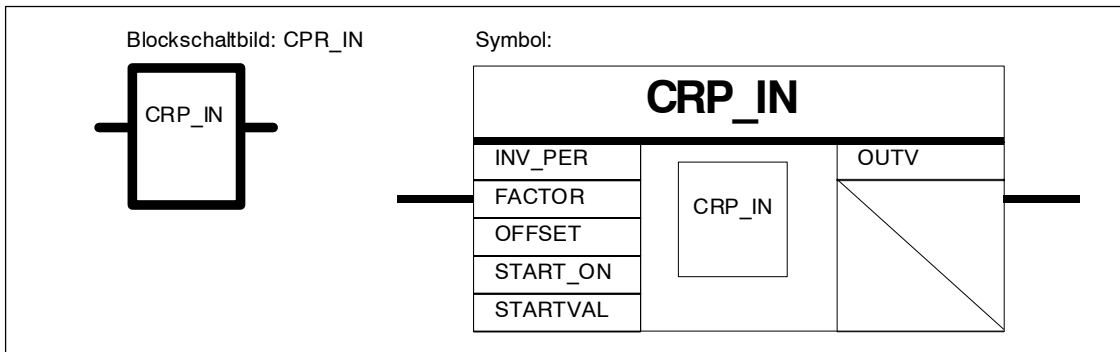


Bild 2-5 CRP_IN, Blockschaltbild und Symbol

Funktionsbeschreibung

CRP_IN wandelt einen Eingangswert im Peripherieformat in einen normierten Gleitpunktwert der Modularen Regelung.

| Peripherie-Wert | Ausgangswert in % |
|-----------------|-------------------|
| 32767 | 118,515 |
| 27648 | 100,000 |
| 1 | 0,003617 |
| 0 | 0,000 |
| -1 | -0,003617 |
| -27648 | -100,000 |
| -32768 | -118,519 |

Der Gleitpunktwert kann über einen Skalierungsfaktor und einen Offset angepaßt werden. Der Ausgang ergibt sich aus:

$$\text{OUTV} = \text{INV_PER} * 100/27648 * \text{FACTOR} + \text{OFFSET}$$

Zum Inbetriebnehmen, Testen oder bei Peripheriestörungen kann auf einen Inbetriebnahmewert umgeschaltet werden. Ist START_ON = TRUE wird am Ausgang OUTV der Wert in STARTVAL ausgegeben.

Hinweis

Eine Prüfung auf positiven/ negativen Überlauf wird nicht durchgeführt.

Eingangsparameter

Die folgende Tabelle zeigt Ihnen Datentyp und Struktur der Eingangsparameter von CRP_IN.

Tabelle 2-3 Eingangsparameter von CRP_IN

| Datentyp | Parameter | Kommentar englisch deutsch | Zulässiger Wertebereich | Vorbelegung |
|----------|-----------|--|--------------------------|-------------|
| WORD | INV_PER | input variable periphery Eingangsgröße für Peripherie | technischer Wertebereich | 0 |
| REAL | FACTOR | scaling factor Skalierungsfaktor | | 1.0 |
| REAL | OFFSET | offset Offset | technischer Wertebereich | 0.0 |
| BOOL | START_ON | startup value on Inbetriebnahmewert einschalten | | TRUE |
| REAL | STARTVAL | startup value Inbetriebnahmewert | technischer Wertebereich | 0.0 |

Ausgangsparameter

Die folgende Tabelle zeigt Datentyp und Struktur der Ausgangsparameter CRP_IN:

Tabelle 2-4 Ausgangsparameter von CRP_IN

| Datentyp | Parameter | Kommentar englisch deutsch | Vorbelegung |
|----------|-----------|----------------------------------|-------------|
| REAL | OUTV | output variable Ausgangsgröße | 0.0 |

Neustart

Der Baustein hat keine Neustarroutine.

Normalbetrieb

Der Baustein enthält keine vom Normalbetrieb abweichende Betriebsarten.

Bausteininterne Begrenzungen

Die Werte der Eingangsparameter werden im Baustein nicht begrenzt; eine Prüfung der Parameter findet nicht statt.

2.1.3 CRP_OUT: change range periphery output (analogen Ausgangswert wandeln)

Anwendungsbereich

Der Baustein paßt einen Gleitpunktwert der Modularen Regelung an das Peripherieformat an.

Blockschaltbild

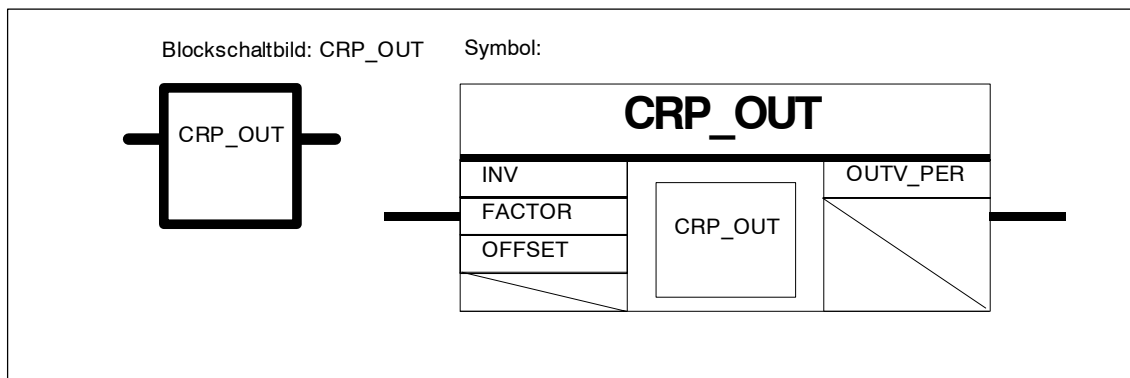


Bild 2-6 CRP_OUT, Blockschaltbild und Symbol

Funktionsbeschreibung

CRP_OUT wandelt einen Eingangswert (normierter Gleitpunktwert der Modularen Regelung) in das Peripherieformat der analogen Peripherie.

Tabelle 2-5 Eingangswert/Peripheriewert

| Eingangswert in % | Peripherie-Wert |
|-------------------|-----------------|
| 118,515 | 32767 |
| 100,000 | 27648 |
| 0,003617 | 1 |
| 0,000 | 0 |
| -0,003617 | -1 |
| -100,000 | -27648 |
| -118,519 | -32768 |

Der Gleitpunktwert kann über einen Skalierungsfaktor und einen Offset angepaßt werden. Der Ausgang ergibt sich aus:

$$\text{OUTV_PER} = (\text{INV} * \text{FACTOR} + \text{OFFSET}) * 27648/100$$

Hinweis

Eine Prüfung auf positiven/negativen Überlauf wird nicht durchgeführt.

Eingangsparameter

Die folgende Tabelle zeigt Ihnen Datentyp und Struktur der Eingangsparameter von CRP_OUT.

Tabelle 2-6 Eingangsparameter von CRP_OUT

| Daten- typ | Parameter | Kommentar | englisch deutsch | zulässiger Wertebereich | Vorbelegung |
|---------------|-----------|-------------------------------------|---------------------|-----------------------------|-------------|
| REAL | INV | input variable Eingangsgröße | | technischer Wertebereich | 0.0 |
| REAL | FACTOR | scaling factor Skalierungsfaktor | | | 1.0 |
| REAL | OFFSET | offset Offset | | technischer Wertebereich | 0.0 |

Ausgangsparameter

Die folgende Tabelle zeigt Ihnen Datentyp und Struktur der Ausgangsparameter von CRP_OUT.

Tabelle 2-7 Ausgangsparameter von CRP_OUT

| Daten- typ | Parameter | Kommentar | englisch deutsch | Vorbelegung |
|---------------|-----------|---|---------------------|-------------|
| WORD | OUTV_PER | output variable periphery Ausgangsgröße für Peripherie | | 0 |

Neustart

Der Baustein hat keine Neustarroutine.

Normalbetrieb

Der Baustein enthält keine vom Normalbetrieb abweichende Betriebsarten.

Bausteininterne Begrenzungen

Die Werte der Eingangsparameter werden im Baustein nicht begrenzt; eine Prüfung der Parameter findet nicht statt.

2.1.4 DEAD_T: dead time (Totzeit)

Anwendungsbereich

Der Baustein findet Anwendungen in Verhältnisregelungen, bei denen die einzelnen Komponenten auf unterschiedlich langen Transportwegen zusammengeführt werden.

Blockschaltbild

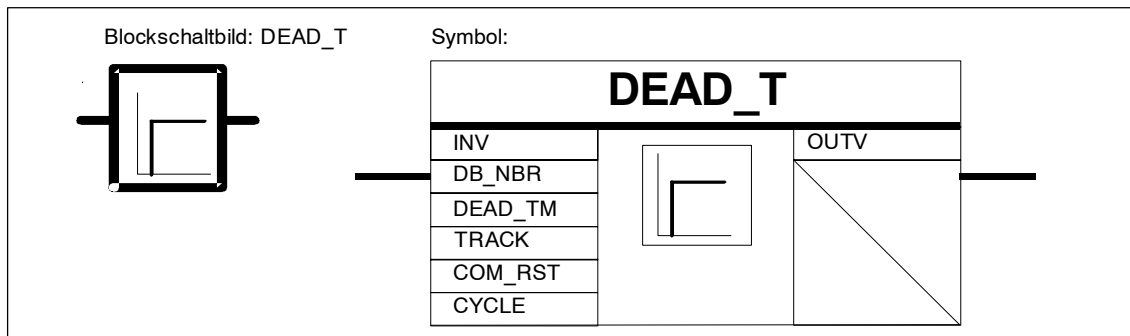


Bild 2-7 DEAD_T, Blockschaltbild und Symbol

Funktionsbeschreibung

Der Baustein verzögert die Ausgabe eines Eingangswertes um eine einstellbare Zeit (Totzeit). Die Eingangswerte werden in einem globalen Datenbaustein zwischengespeichert. Die maximale Totzeit hängt von der Länge dieses Datenbausteins ab. Die Daten im globalen Datenbaustein DB_NBR werden wie in einem Ringpuffer abgearbeitet.

Tabelle 2-8 Eingangswert

| Nr. | Eingangswert | |
|-----|--------------|--------------------------------|
| 0 | INV[0] | □ |
| 1 | INV[1] | ↓ |
| 2 | INV[2] | ⇔ OUTV/INV Lese-/Schreibzeiger |
| ... | ... | ↓ |
| ... | ... | |
| n | INV[n] | □ DEAD_TM = (n+1) • CYCLE |
| ... | ... | |
| m | INV[m] | |

Über den Lese-/Schreibzeiger wird der Speicherplatz gelesen und an OUTV ausgegeben. Danach wird INV in den gleichen Speicherplatz geschrieben. Der Speicherplatzindex für den Lese-/Schreibzeiger wird pro Bausteinbearbeitung um 1 erhöht. Hat er n erreicht, springt er auf 0 zurück.

Bei vorgegebener Totzeit DEAD_TM und fester Abtastzeit CYCLE muß der Datenbaustein

$$\frac{\text{DEAD_TM}}{\text{CYCLE}}$$

Speicherungen ermöglichen. Eine Speicherung (Datentyp: REAL) belegt 4 Byte. DEAD_TM muß ein ganzzahliges Vielfaches von CYCLE sein.

$$\text{DB-Länge (in Byte)} \geq \frac{\text{DEAD_TM}}{\text{CYCLE}} * 4$$

Bei TRACK=TRUE wird der Eingangswert direkt ausgegeben.

Hinweis

Der Baustein überprüft weder, ob ein globaler DB mit der Nummer DB_NBR wirklich vorhanden ist, noch ob die Parameter DEAD_TM (Totzeit) und CYCLE (Abtastzeit) zur Länge des Datenbausteins passen. Bei falscher Parametrierung geht die CPU mit der Meldung "interner Systemfehler" in den Betriebszustand STOP.

Eingangsparameter

Die folgende Tabelle zeigt Ihnen Datentyp und Struktur der Eingangsparameter von DEAD_T.

Tabelle 2-9 Eingangsparameter von DEAD_T

| Datentyp | Parameter | Kommentar | englisch deutsch | zulässiger Wertebereich | Vorbelegung |
|----------|-----------|--|---------------------|-------------------------------|-------------|
| REAL | INV | input variable Eingangsgröße | | technischer Wertebereich | 0.0 |
| BLOCK_DB | DB_NBR | data block number Datenbausteinnummer | | | DB 1 |
| TIME | DEAD_TM | dead time Totzeit | | ≥ CYCLE ≤ DB-Länge/4*CYCLE | 10s |
| BOOL | TRACK | tracking OUTV=INV Nachführen OUTV=INV | | | FALSE |
| BOOL | COM_RST | complete restart Neustart | | | FALSE |
| TIME | CYCLE | sample time Abtastzeit | | ≥ 1ms | 1s |

Ausgangsparameter

Die folgende Tabelle zeigt Ihnen Datentyp und Struktur der Ausgangsparameter von DEAD_T.

Tabelle 2-10 Ausgangsparameter von DEAD_T

| Daten- typ | Parameter | Kommentar englisch deutsch | Vorbelegung |
|---------------|-----------|----------------------------------|-------------|
| REAL | OUTV | output variable Ausgangsgröße | 0.0 |

Globaler Datenbaustein

Die folgende Tabelle zeigt Ihnen Datentyp und Parameter des Globalen Datenbausteins.

Tabelle 2-11 Parameter des Globalen Datenbausteins

| Daten- typ | Parameter | Kommentar englisch deutsch | zulässiger Wertebereich | Vorbelegung |
|---------------|-----------|--|-----------------------------|-------------|
| REAL | INV[0] | input variable [0] Eingangswert [0] | technischer Wertebereich | 0.0 |
| REAL | INV[1] | input variable [1] Eingangswert [1] | technischer Wertebereich | 0.0 |
| REAL | INV[2] | input variable [2] Eingangswert [2] | technischer Wertebereich | 0.0 |
| REAL | INV[3] | input variable [3] Eingangswert [3] | technischer Wertebereich | 0.0 |

Neustart

Bei Neustart werden alle gespeicherten Eingangswerte gelöscht und OUTV=0.0 ausgegeben.

Normalbetrieb

Die Eingangswerte werden, um die Totzeit verzögert, ausgegeben. Bei Online-Änderungen der Totzeit können beim Ausgangswert Sprünge auftreten.

- Nachführen

Ist Nachführen (TRACK=TRUE) eingeschaltet, wird der Eingangswert unverzögert an OUTV übergeben. Die Zwischenspeicherung der Eingangswerte wird nicht unterbrochen, damit nach Abschalten von Nachführen die Eingangswerte nach der eingestellten Totzeit ausgegeben werden können. Bei TRACK=FALSE springt OUTV auf INV[DEAD_TM]!

Bausteininterne Begrenzungen

Die Werte der Eingangsparameter werden im Baustein nicht begrenzt; eine Prüfung der Parameter findet nicht statt.

Beispiel

Bei einer Abtastzeit von $CYCLE=1s$ und einer Totzeit von $DEAD_TM=4s$ müssen 4 Eingangswerte zwischengespeichert werden. Der Datenbereich muß 16 Byte lang sein.

Tabelle 2-12 Doppelwort/Eingangswert

| Datendoppelwort | Eingangswert |
|-----------------|--------------|
| 0 | INV[0] |
| 4 | INV[1] |
| 8 | INV[2] |
| 12 | INV[3] |

2.1.5 DEADBAND: dead band (Totzone)

Anwendungsbereich

Bei verrauschtem Istwert wird bei optimal eingestelltem Regler der Rauschanteil auch am Reglerausgang wirksam; dies führt durch hohe Schalthäufigkeit (Schritttregler) zu einem erhöhten Verschleiß des Stellgliedes. Durch Unterdrückung des Rauschanteils wird ein Oszillieren des Reglerausgangs vermieden. Bei Verwendung zur Regeldifferenzbildung muß der Offset $DEADB_O=0.0$ gesetzt werden.

Blockschaltbild

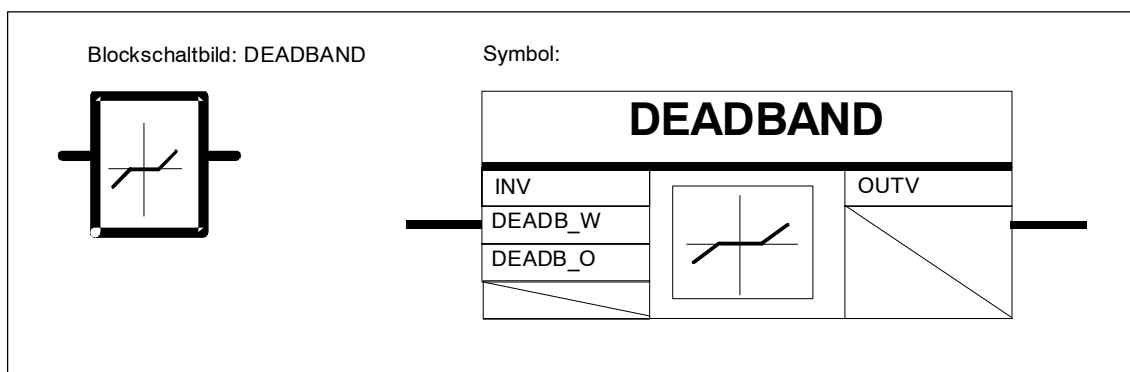


Bild 2-8 DEADBAND, Blockschaltbild und Symbol

Funktionsbeschreibung

Der Baustein DEADBAND (Totzone) unterdrückt bei der Eingangsgröße INV kleine Schwankungen um einen festgelegten Nullpunkt; außerhalb dieser Schwankungen steigt die Ausgangsgröße OUTV proportional zur Eingangsgröße. Er arbeitet nach der Funktion:

$$\begin{aligned} \text{OUTV} &= \text{INV} + \text{DEADB_W} - \text{DEADB_O} && \text{für } \text{INV} < \text{DEADB_O} - \text{DEADB_W} \\ \text{OUTV} &= 0.0 && \text{für } \text{DEADB_O} - \text{DEADB_W} \leq \text{INV} \\ &&& \text{und } \text{INV} \leq \text{DEADB_O} + \text{DEADB_W} \\ \text{OUTV} &= \text{INV} - \text{DEADB_W} - \text{DEADB_O} && \text{für } \text{INV} > \text{DEADB_O} + \text{DEADB_W} \end{aligned}$$

Die Signalverfälschung entspricht dem Wert $DEADB_W$. Der Mittelpunkt der Totzone wird mit $DEADB_O$ festgelegt.

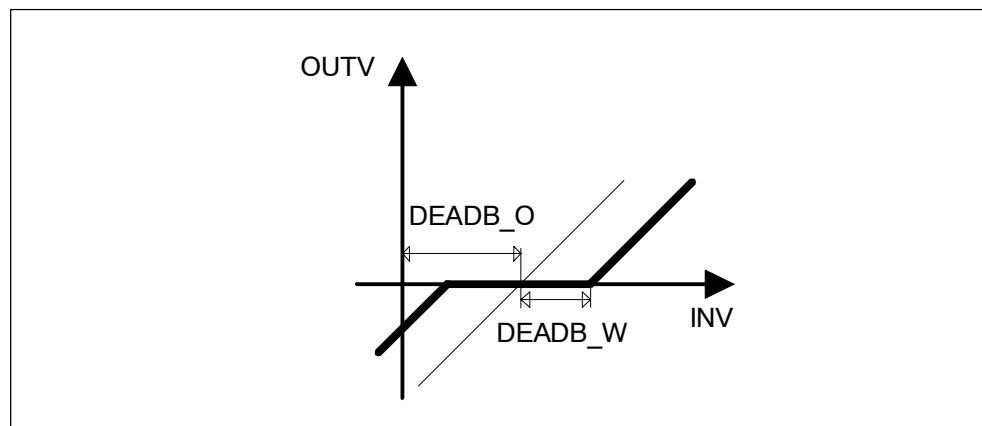


Bild 2-9 $OUTV = f(INV)$

Totzonbreite $DEADB_W$ und Totzonenversatz $DEADB_O$ sind parametrierbar.

Eingangsparameter

Die folgende Tabelle zeigt Ihnen Datentyp und Struktur der Eingangsparameter von DEADBAND.

Tabelle 2-13 Eingangsparameter von DEADBAND

| Daten- typ | Parameter | Kommentar | englisch deutsch | zulässiger Wertebereich | Vorbelegung |
|---------------|-----------|-------------------------------------|---------------------|-----------------------------|-------------|
| REAL | INV | input variable Eingangsgröße | | technischer Wertebereich | 0.0 |
| REAL | DEADB_W | dead band width Totzonenbreite | | tech. Wert. ≥ 0.0 | 1.0 |
| REAL | DEADB_O | dead band offset Totzonenversatz | | technischer Wertebereich | 0.0 |

Ausgangsparameter

Die folgende Tabelle zeigt Ihnen Datentyp und Struktur der Ausgangsparameter von DEADBAND.

Tabelle 2-14 Ausgangsparameter von DEADBAND

| Daten- typ | Parameter | Kommentar | englisch deutsch | Vorbelegung |
|---------------|-----------|----------------------------------|---------------------|-------------|
| REAL | OUTV | output variable Ausgangsgröße | | 0.0 |

Neustart

Der Baustein besitzt keine Neustartroutine.

Normalbetrieb

Der Baustein enthält keine vom Normalbetrieb abweichende Betriebsarten.

Bausteininterne Begrenzungen

Die Werte der Eingangsparameter werden im Baustein nicht begrenzt; eine Prüfung der Parameter findet nicht statt. Die Totzonenbreite darf nur positive Werte annehmen.

Beispiel

Bild 2-10 zeigt Ihnen die Überbrückung eines Rauschanteils mit Offset

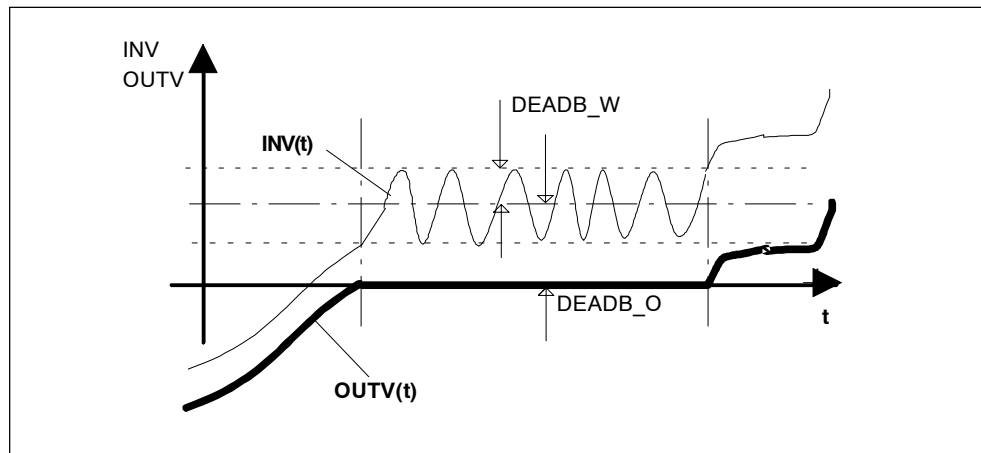


Bild 2-10 Unterdrückung eines Rauschanteils mit Offset

2.1.6 DIF: differentiator (Differenzierer)

Anwendungsbereich

Prozeßgrößen werden dynamisch differenziert. So kann z.B. aus dem zurückgelegten Weg die Geschwindigkeit berechnet werden. Der Differenzierer ist einsetzbar zur Störgrößenaufschaltung, zur Vorsteuerung und zum Aufbau eines Reglers.

Blockschaltbild

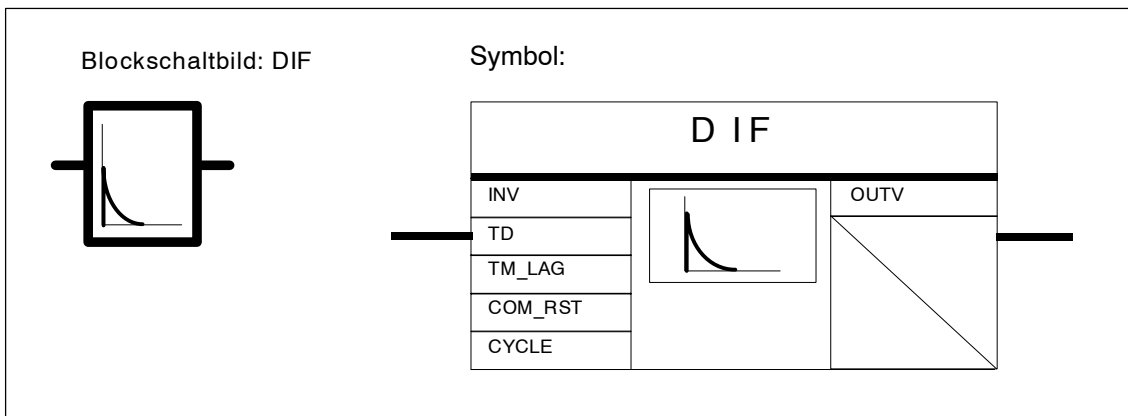


Bild 2-11 DIF, Blockschaltbild und Symbol

Funktionsbeschreibung

Der Baustein differenziert die Eingangsgröße über der Zeit und glättet das Signal mit einem Verzögerungsglied 1. Ordnung.

Eingangsparameter

Die folgende Tabelle zeigt Ihnen Datentyp und Struktur der Eingangsparameter von DIF.

Tabelle 2-15 Eingangsparameter von DIF

| Datentyp | Parameter | Kommentar | englisch deutsch | zulässiger Wertebereich | Vorbelegung |
|----------|-----------|---|---------------------|-----------------------------|-------------|
| REAL | INV | input variable Eingangsgröße | | technischer Wertebereich | 0.0 |
| TIME | TD | derivative time value Differenzierzeit | | \geq CYCLE | T#25s |
| TIME | TM_LAG | time lag Verzögerungszeit | | | T#5s |

Tabelle 2-15 Eingangparameter von DIF, Fortsetzung

| Datentyp | Parameter | Kommentar englisch deutsch | zulässiger Wertebereich | Vorbelegung |
|----------|-----------|----------------------------------|----------------------------|-------------|
| BOOL | COM_RST | complete restart Neustart | | FALSE |
| TIME | CYCLE | sample time Abtastzeit | ≥ 1ms | T#1s |

Ausgangsparameter

Die folgende Tabelle zeigt Ihnen Datentyp und Struktur der Ausgangsparameter von DIF.

Tabelle 2-16 Ausgangsparameter von DIF

| Datentyp | Parameter | Kommentar englisch deutsch | Vorbelegung |
|----------|-----------|----------------------------------|-------------|
| REAL | OUTV | output variable Ausgangsgröße | 0.0 |

Neustart

Bei Neustart werden alle Signalausgänge auf Null gesetzt. Der Differentiator wird intern mit dem momentanen Eingangswert INV vorbelegt. Der Übergang in den Normalbetrieb ist damit bei gleichbleibender Eingangsgröße stoßfrei.

Normalbetrieb

Beim Differenzieren arbeitet der Baustein nach folgender Übertragungsfunktion:

im Laplace-Bereich: $OUTV(s) / INV(s) = TD / (1+TM_LAG*s)$

Das Zeitverhalten des Differenzierers wird durch die Differentiationszeit TD und die Verzögerungszeit TM_LAG festgelegt. Die zugehörige Sprungantwort ist im folgenden Bild dargestellt.

Sprungantwort

Die Bilder 2-12 und 2-13 zeigen Ihnen die Sprungantwort von DIF (mit bzw. ohne Verzögerung)

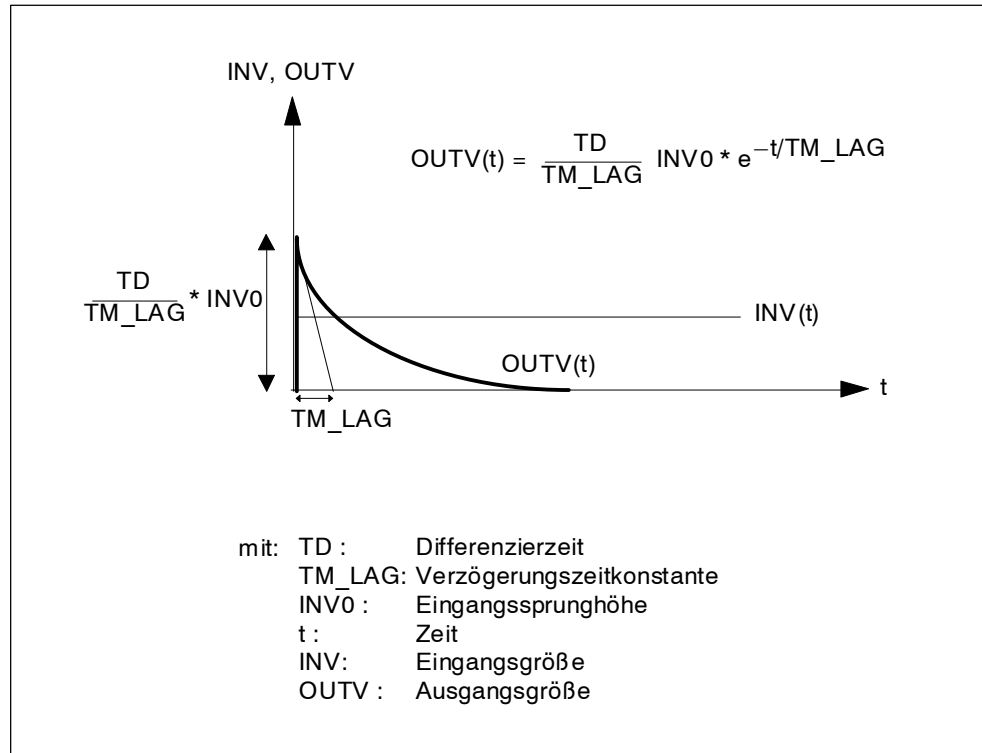


Bild 2-12 Sprungantwort von DIF

Ist $TM_LAG \leq CYCLE/2$ parametrierung, arbeitet der Differenzierer ohne Verzögerung. Ein Eingangssprung wird mit dem Faktor $TD/CYCLE$ auf den Ausgang gegeben. Nach einem Zyklus geht der Ausgang wieder auf 0.0 zurück.

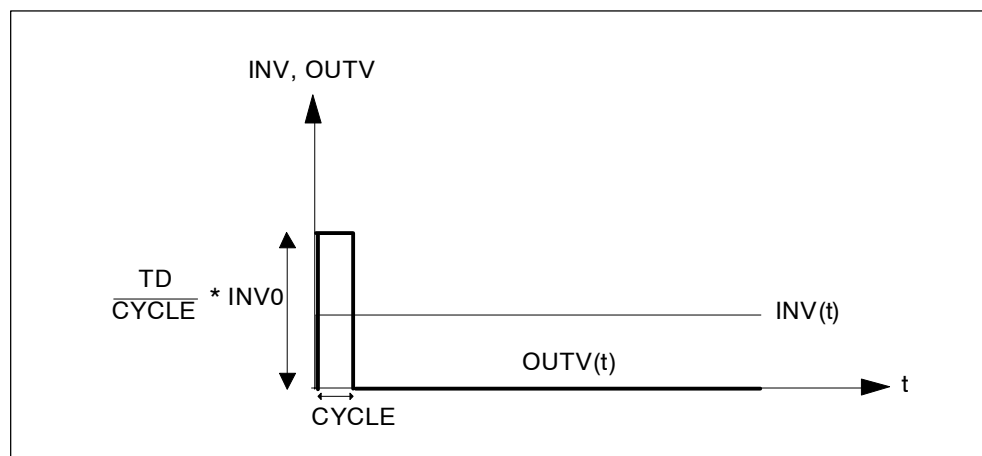


Bild 2-13 Sprungantwort von DIF ohne Verzögerung

Bausteininterne Begrenzungen

Die Differenzierzeit wird auf die Abtastzeit nach unten begrenzt. Die Verzögerungszeit wird auf die halbe Abtastzeit nach unten begrenzt.

$$TD_{\text{intern}} = \text{CYCLE} \quad \text{für } TD < \text{CYCLE}$$

$$TM_LAG_{\text{intern}} = \text{CYCLE}/2 \quad \text{für } TM_LAG < \text{CYCLE}/2$$

Die anderen Werte der Eingangsparameter werden im Baustein nicht begrenzt; eine Prüfung der Parameter findet nicht statt.

2.1.7 ERR_MON: error-signal monitoring (Regeldifferenzüberwachung)

Anwendungsbereich

Der Baustein wird zur Regeldifferenzbildung und -überwachung eingesetzt.

Blockschaltbild

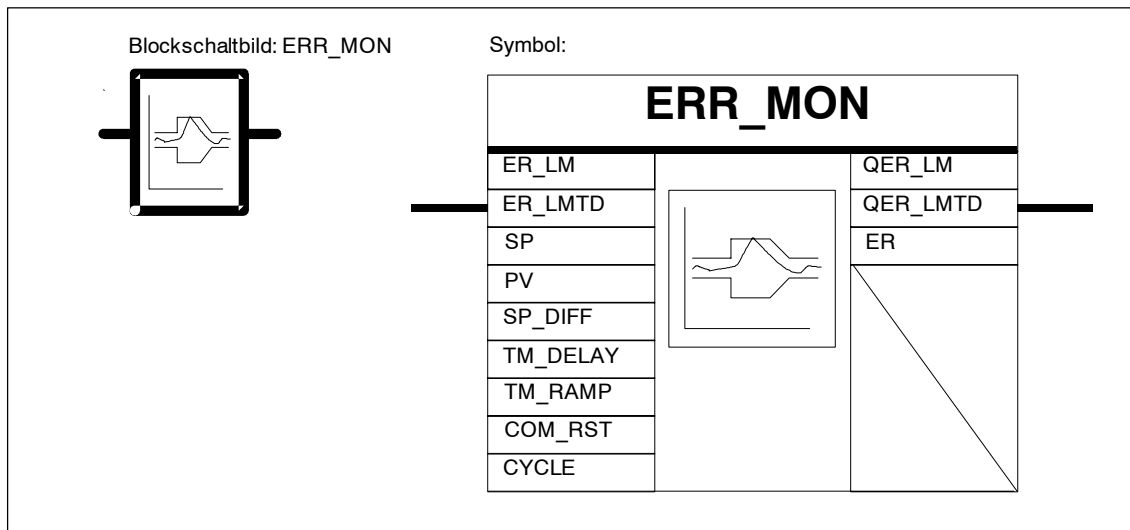


Bild 2-14 ERR_MON, Blockschaltbild und Symbol

Eingangsparameter

Die folgende Tabelle zeigt Ihnen Datentyp und Struktur der Eingangsparameter von ERR_MON.

Tabelle 2-17 Eingangsparameter von ERR_MON

| Daten- typ | Parameter | Kommentar englisch deutsch | zulässiger Wertebereich | Vorbelegung |
|---------------|-----------|--|---------------------------------------|-------------|
| REAL | ER_LM | error-variable limit Grenzwert der Regeldifferenz | tech. Wert. > 0.0 und < ER_LMTD | 10.0 |
| REAL | ER_LMTD | error-signal limit during time delay Grenzwert der Regeldifferenz während Einschaltverzögerung | tech. Wert. > ER_LM | 100.0 |
| REAL | SP | setpoint variable Sollwert | technischer Wertebereich | 0.0 |
| REAL | PV | process variable Regelgröße | technischer Wertebereich | 0.0 |
| REAL | SP_DIFF | setpoint differenz Sollwertänderung | tech. Wert. > 0.0 | 10.0 |
| TIME | TM_DELAY | time delay of the monitoring signal Einschaltverzögerung des Überwa- chungssignals | | T#60s |
| TIME | TM_RAMP | time constant of ramp Rampenzeitkonstante | | T#60s |
| BOOL | COM_RST | complete restart Neustart | | FALSE |
| TIME | CYCLE | sample time Abtastzeit | ≥ 1ms | T#1s |

Ausgangsparameter

Die folgende Tabelle zeigt Ihnen Datentyp und Struktur der Ausgangsparameter von ERR_MON.

Tabelle 2-18 Ausgangsparameter von ERR_MON

| Daten- typ | Parameter | Kommentar englisch deutsch | Vorbelegung |
|---------------|-----------|--|-------------|
| BOOL | QER_LM | error-signal limit reached Grenzwert der Regeldifferenz angesprochen | FALSE |
| BOOL | QER_LMTD | error-signal limit during time delay reached Grenzwert der Regeldifferenz während der Einschaltverzögerung und der Rampe angesprochen | FALSE |
| REAL | ER | error-signal Regeldifferenz | 0.0 |

Neustart

Bei Neustart werden die Meldesignale QER_LM und QER_LMTD und der Regeldifferenzausgang ER zurückgesetzt.

Normalbetrieb

Der Baustein enthält keine vom Normalbetrieb abweichende Betriebsarten.

Bausteininterne Begrenzungen

Die Werte der Eingangsparameter werden im Baustein nicht begrenzt; eine Prüfung der Parameter findet nicht statt.

2.1.8 INTEG: integrator (Integrierer)

Anwendungsbereich

Prozeßgrößen werden dynamisch integriert. So wird z.B. aus der Geschwindigkeit der zurückgelegte Weg berechnet. Der Integrierer ist einsetzbar zum Aufbau eines Reglers.

Blockschaltbild

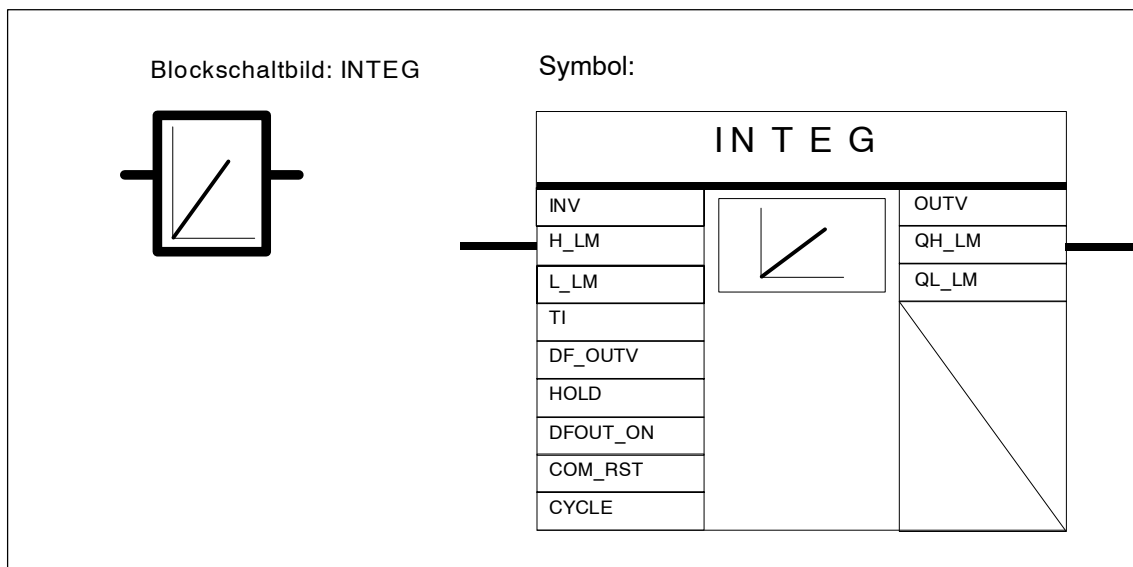


Bild 2-16 INTEG, Blockschaltbild und Symbol

Funktionsbeschreibung

Der Baustein integriert die Eingangsgröße über der Zeit und begrenzt das Integral auf eine vorgebbare obere und untere Begrenzung. Die Begrenzung der Ausgangsgröße wird über Meldebits angezeigt.

Eingangsparameter

Die folgende Tabelle zeigt Ihnen Datentyp und Struktur der Eingangsparameter von INTEG.

Tabelle 2-19 Eingangsparameter von INTEG

| Daten- typ | Parameter | Kommentar englisch deutsch | zulässiger Wertebereich | Vorbelegung |
|---------------|-----------|--|-----------------------------|-------------|
| REAL | INV | input variable Eingangsgröße | technischer Wertebereich | 0.0 |
| REAL | H_LM | high limit obere Begrenzung | tech. Wert. > L_LM | 100.0 |
| REAL | L_LM | low limit untere Begrenzung | tech. Wert. < H_LM | 0.0 |
| TIME | TI | reset time Integrationszeit | ≥ CYCLE | T#25s |
| REAL | DF_OUTV | default output variable Vorbesetzung der Ausgangsgröße | technischer Wertebereich | 0.0 |
| BOOL | HOLD | integrator hold integrator einfrieren | | FALSE |
| BOOL | DFOUT_ON | default output variable on Vorbesetzung der Ausgangsgröße ein | | FALSE |
| BOOL | COM_RST | complete restart Neustart | | FALSE |
| TIME | CYCLE | sample time Abtastzeit | ≥ 1ms | T#1s |

Ausgangsparameter

Die folgende Tabelle zeigt Ihnen Datentyp und Struktur der Ausgangsparameter von INTEG.

Tabelle 2-20 Ausgangsparameter von INTEG

| Daten- typ | Parameter | Kommentar englisch deutsch | Vorbelegung |
|---------------|-----------|---|-------------|
| REAL | OUTV | output variable Ausgangsgröße | 0.0 |
| BOOL | QH_LM | high limit reached obere Begrenzung angesprochen | FALSE |
| BOOL | QL_LM | low limit reached untere Begrenzung angesprochen | FALSE |

Neustart

Bei Neustart wird der Ausgang OUTV auf 0.0 rückgesetzt. Wenn DFOUT_ON = TRUE gesetzt ist wird DF_OUTV ausgegeben. Die Begrenzung des Ausgangs und deren Anzeige ist auch bei Neustart wirksam. Beim Übergang in den Normalbetrieb integriert der Baustein von OUTV aus weiter.

Soll der Integrierer bei Neustart von einem bestimmten Arbeitspunkt aus gestartet werden, so muß am Eingang DF_OUTV der Arbeitspunkt eingetragen werden. Beim Aufruf des Bausteins in der Neustarroutine muß DFOUT_ON = TRUE gesetzt werden und in der Weckalarmebene auf DFOUT_ON = FALSE wieder zurückgesetzt werden.

Normalbetrieb

Der Baustein enthält neben dem Normalbetrieb folgende Betriebsarten:

| Betriebsarten | DFOUT_ON | HOLD |
|-----------------------|----------|----------|
| Integrieren | FALSE | FALSE |
| Integrator einfrieren | FALSE | TRUE |
| Ausgang vorbelegen | TRUE | beliebig |

• Integrieren

Beim Integrieren arbeitet der Baustein nach folgender Übertragungsfunktion:

im Laplace-Bereich:

$$\text{OUTV}(s) / \text{INV}(s) = 1 / (T_I \cdot s)$$

Das Zeitverhalten des Integrators wird durch die Integrationszeit **TI** festgelegt. Die zugehörige Sprungantwort ist im folgenden Bild dargestellt.

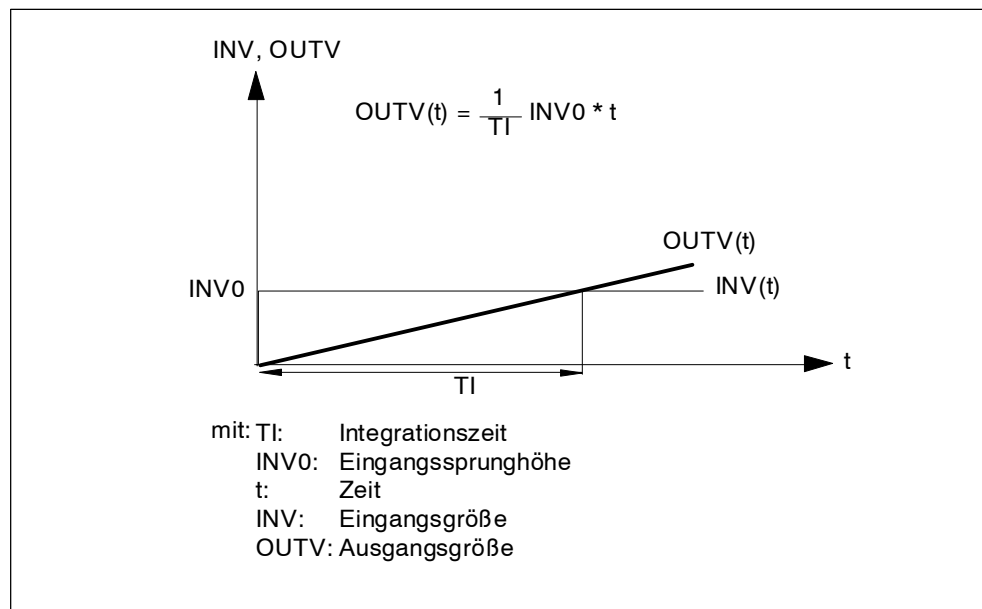


Bild 2-17 Sprungantwort von INTEG

Der Ausgang wie auch der Integratorspeicher werden an den vorgebbaren Begrenzungswerten H_{LM} und L_{LM} begrenzt. Liegt der Ausgang in der Begrenzung, wird dies an den Meldebits QH_{LM} und QL_{LM} angezeigt.

• Integrator einfrieren

Mit $HOLD = TRUE$ bleibt der Integrator auf seinem momentanen Ausgangswert $OUTV$ stehen. Beim Rücksetzen $HOLD = FALSE$ wird ausgehend vom momentanen Ausgangswert $OUTV$ weiterintegriert.

- **Ausgang vorbelegen**

Wenn DFOUT_ON = TRUE gesetzt ist, wird am Ausgang DF_OUTV ausgegeben. Die Begrenzung ist wirksam. Wird DF_OUTV_ON = FALSE zurückgesetzt, beginnt der Integrierer ausgehend vom Wert DF_OUTV zu integrieren.

Bausteininterne Begrenzungen

Die Integrationszeit wird auf die Abtastzeit nach unten begrenzt:

$$T_{\text{intern}} = \text{CYCLE} \quad \text{für } T_I < \text{CYCLE}$$

Die anderen Werte der Eingangsparameter werden im Baustein nicht begrenzt; eine Prüfung der Parameter findet nicht statt.

Beispiel

Bild 2-18 zeigt Ihnen ein Beispiel mit DFOUT_ON, HOLD und Begrenzung

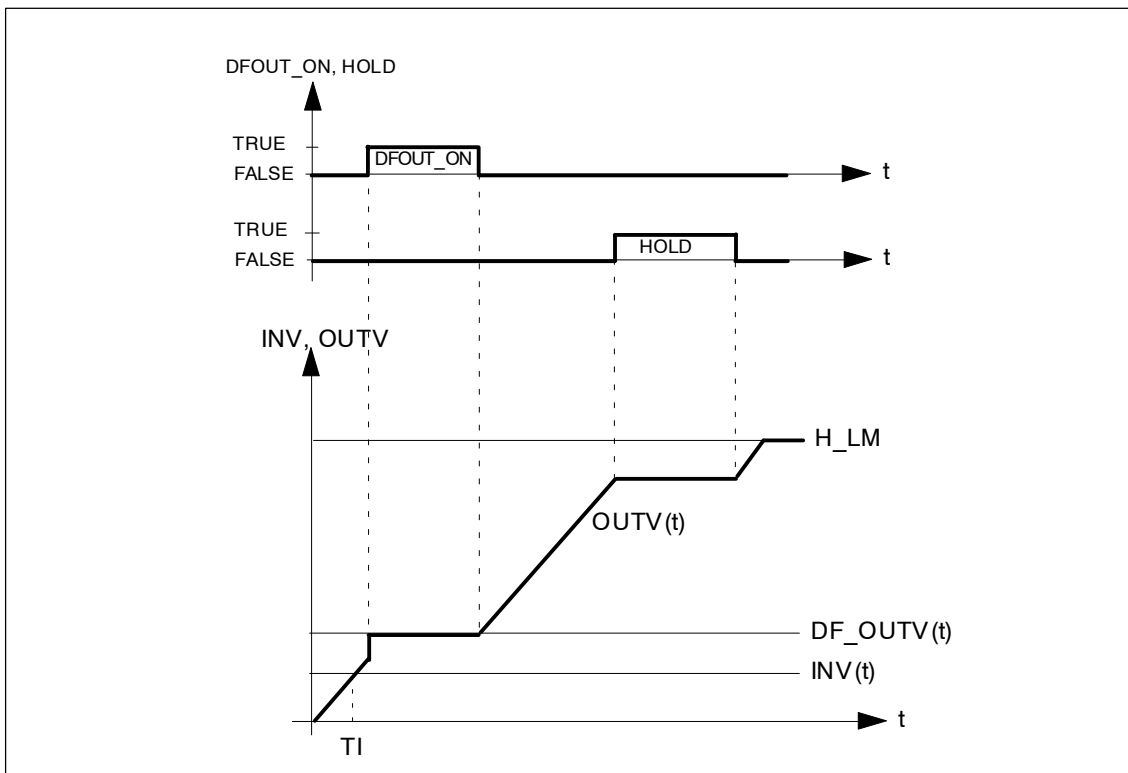


Bild 2-18 Beispiel mit DFOUT_ON, HOLD und Begrenzung

2.1.9 LAG1ST: first-order lag element (Verzögerungsglied 1. Ordnung)

Anwendungsbereich

Der Baustein ist einsetzbar als Verzögerungs- und Glättungsglied.

Blockschaltbild

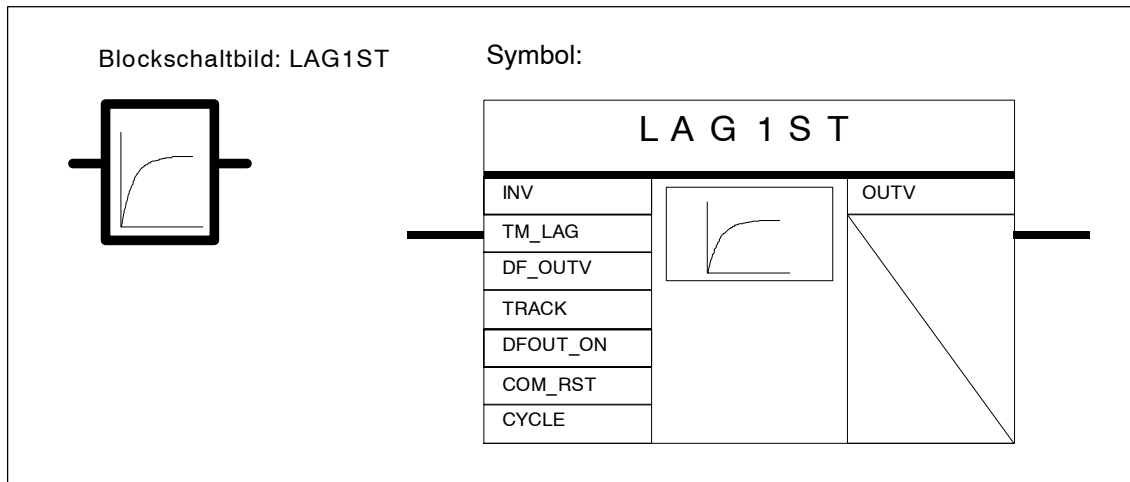


Bild 2-19 LAG1ST, Blockschaltbild und Symbol

Funktionsbeschreibung

Der Baustein glättet die Eingangsgröße nach der Verzögerung 1. Ordnung. Die Verzögerungszeit ist parametrierbar.

Eingangsparameter

Die folgende Tabelle zeigt Ihnen Datentyp und Struktur der Eingangsparameter von LAG1ST.

Tabelle 2-21 Eingangsparameter von LAG1ST

| Daten- typ | Parameter | Kommentar | englisch deutsch | zulässiger Wertebereich | Vorbelegung |
|---------------|-----------|---|---------------------|-----------------------------|-------------|
| REAL | INV | input variable Eingangsgröße | | technischer Wertebereich | 0.0 |
| TIME | TM_LAG | time lag Verzögerungszeit | | | T#25s |
| REAL | DF_OUTV | default output variable Vorbesetzung der Ausgangsgröße | | technischer Wertebereich | 0.0 |

Tabelle 2-21 Eingangparameter von LAG1ST, Fortsetzung

| Daten- typ | Parameter | Kommentar englisch deutsch | zulässiger Wertebereich | Vorbelegung |
|---------------|-----------|--|----------------------------|-------------|
| BOOL | TRACK | tracking OUTV=INV Nachführen OUTV=INV | | FALSE |
| BOOL | DFOUT_ON | default output variable on Vorbesetzung der Ausgangsgröße ein | | FALSE |
| BOOL | COM_RST | complete restart Neustart | | FALSE |
| TIME | CYCLE | sample time Abtastzeit | ≥1ms | T#1s |

Ausgangsparameter

Die folgende Tabelle zeigt Ihnen Datentyp und Struktur der Ausgangsparameter von LAG1ST.

Tabelle 2-22 Ausgangsparameter von LAG1ST

| Daten- typ | Parameter | Kommentar englisch deutsch | Vorbelegung |
|---------------|-----------|----------------------------------|-------------|
| REAL | OUTV | output variable Ausgangsgröße | 0.0 |

Neustart

Bei Neustart wird der Ausgang OUTV auf 0.0 rückgesetzt. Wenn DFOUT_ON = TRUE gesetzt ist wird DF_OUTV ausgegeben. Beim Übergang in den Normalbetrieb arbeitet der Baustein von OUTV aus weiter.

Normalbetrieb

Der Baustein enthält neben dem Normalbetrieb folgende Betriebsarten:

| Betriebsarten | DFOUT_ON | HOLD |
|--------------------|----------|----------|
| Glätten | FALSE | FALSE |
| Nachführen | FALSE | TRUE |
| Ausgang vorbelegen | TRUE | beliebig |

- **Glätten**

Beim Glätten arbeitet der Baustein nach folgender Übertragungsfunktion:

im Laplace-Bereich: $OUTV(s) / INV(s) = 1 / (1+TM_LAG*s)$

Das Zeitverhalten des Verzögerungsgliedes wird durch die Verzögerungszeit TM_LAG festgelegt. Die zugehörige Sprungantwort ist im folgenden Bild dargestellt.

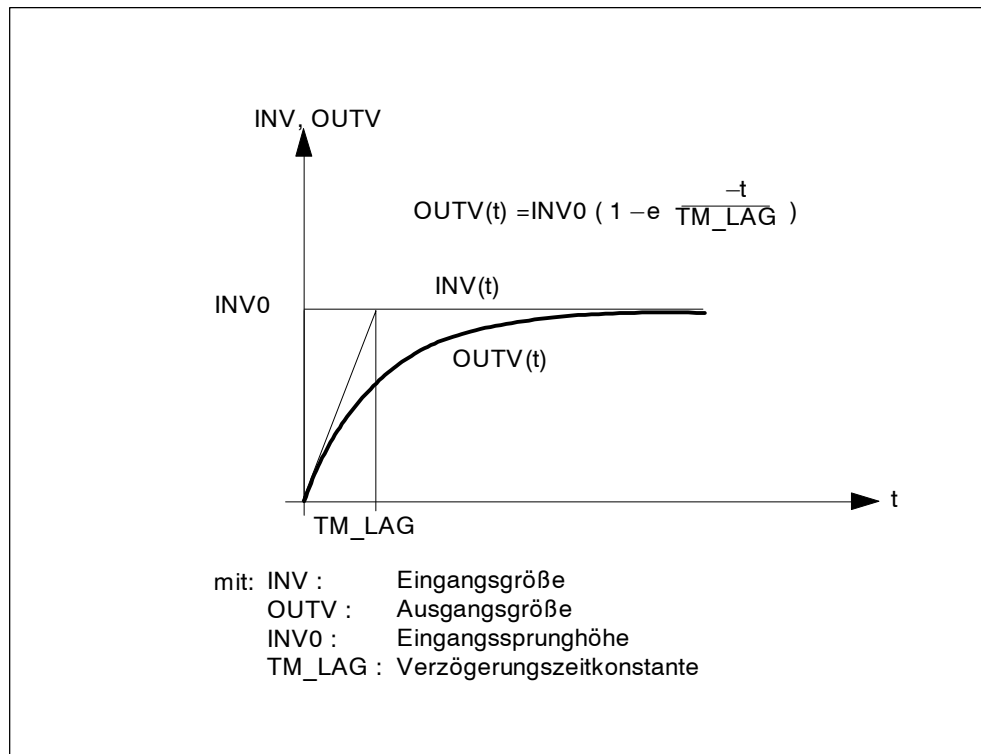


Bild 2-20 Sprungantwort von LAG1ST

- **Nachführen**

Mit $TRACK = TRUE$ wird der Eingangswert INV auf den Ausgang $OUTV$ durchgeschaltet.

- **Ausgang vorbelegen**

Wenn $DFOUT_ON = TRUE$ gesetzt ist, wird am Ausgang DF_OUTV ausgegeben. Wird $DF_OUTV_ON = FALSE$ zurückgesetzt, glättet das Verzögerungsglied ausgehend vom Wert DF_OUTV .

Bausteininterne Begrenzungen

Die Verzögerungszeit wird auf die halbe Abtastzeit nach unten begrenzt.

$$TM_LAG_{\text{intern}} = \text{CYCLE}/2 \quad \text{für } TM_LAG < \text{CYCLE}/2$$

Die anderen Werte der Eingangsparameter werden im Baustein nicht begrenzt; eine Prüfung der Parameter findet nicht statt.

Beispiel

Bild 2-21 zeigt Ihnen ein Beispiel mit DFOUT_ON und TRACK:

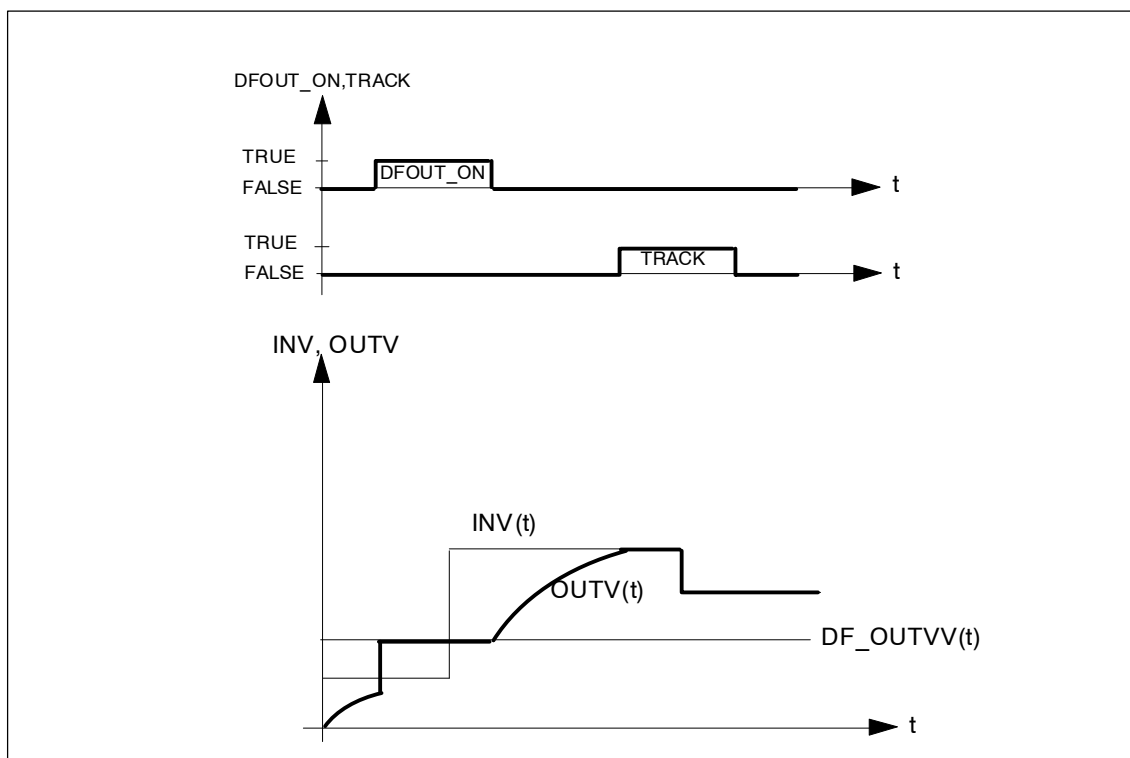


Bild 2-21 Beispiel mit DFOUT_ON und TRACK

2.1.10 LAG2ND: second-order lag element (Verzögerungsglied 2. Ordnung)

Anwendungsbereich

Der Baustein dient zur Simulation von Streckenanteilen für Vorsteuerungsregelungen und Zweikreisregelungen.

Blockschaltbild

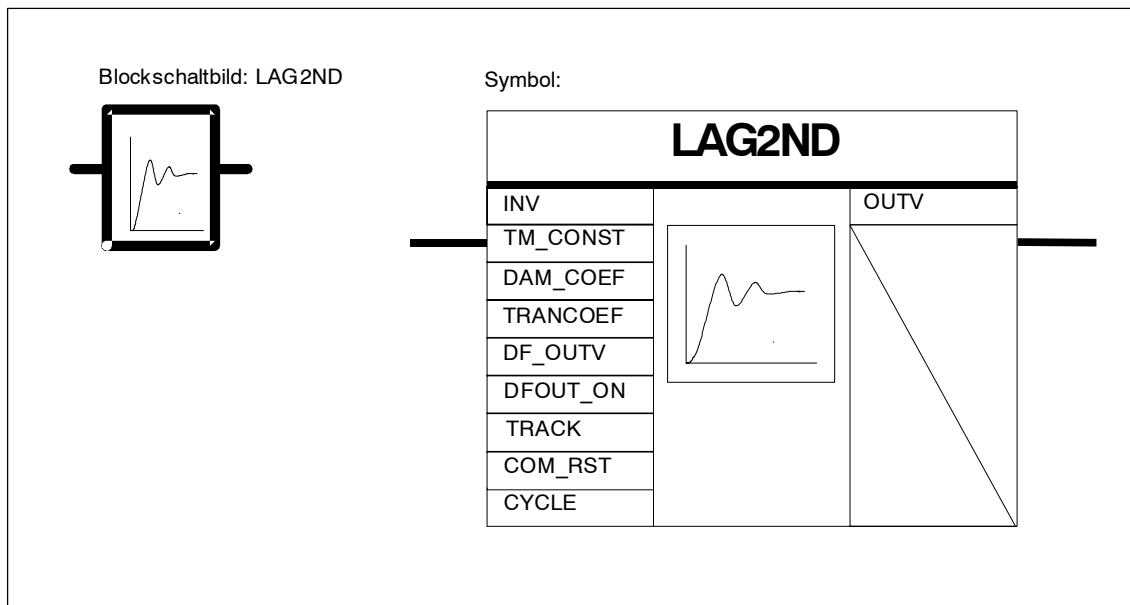


Bild 2-22 LAG2ND, Blockschaltbild und Symbol

Funktionsbeschreibung

Der Baustein realisiert ein schwingungsfähiges Verzögerungsglied 2. Ordnung.

Übertragungsfunktion

Die Übertragungsfunktion im Laplace-Bereich ist:

$$\text{OUTV}(s) / \text{INV}(s) = \text{TRANCOEF} / (1 + 2 \cdot \text{DAM_COEF} \cdot \text{TM_CONST} \cdot s + \text{TM_CONST}^2 \cdot s^2)$$

Wenn $\text{DAM_COEF} \geq 1$ ist (aperiodischer Fall), läßt sich das Übertragungsglied in einer Reihenschaltung zweier PT1-Glieder darstellen.

$$\text{OUTV}(s) / \text{INV}(s) = \text{TRANCOEF} / (1 + T1 \cdot s) \cdot 1 / (1 + T2 \cdot s)$$

Die Zeitkonstanten werden wie folgt umgerechnet:

$$T1 = TM_CONST (DAM_COEF + \sqrt{DAM_COEF^2-1})$$

$$T2 = TM_CONST (DAM_COEF - \sqrt{DAM_COEF^2-1})$$

Blockschaltbild

Bild 2-23 zeigt das Blockschaltbild der Funktion LAG2ND.

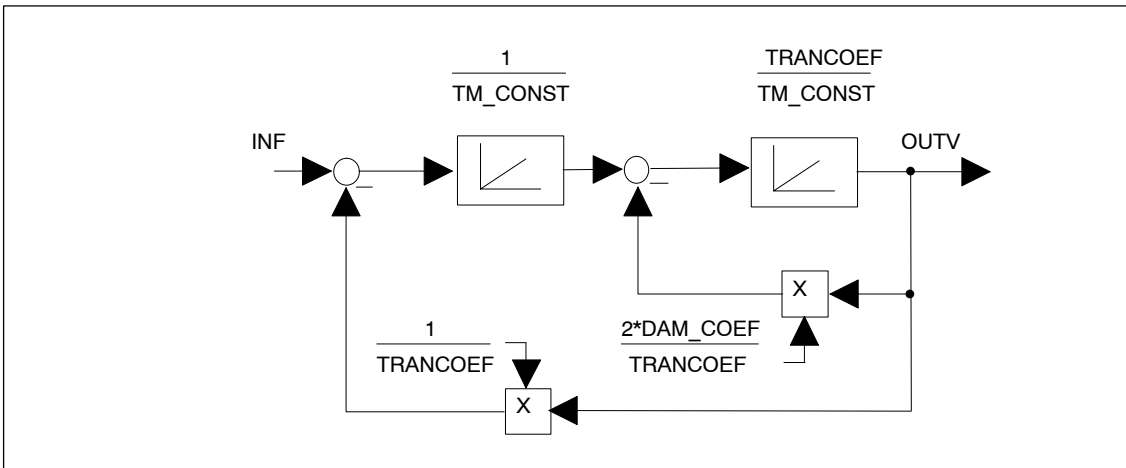


Bild 2-23 Aufbau des LAG2ND-Gliedes aus elementaren Übertragungsgliedern

Sprungantwort

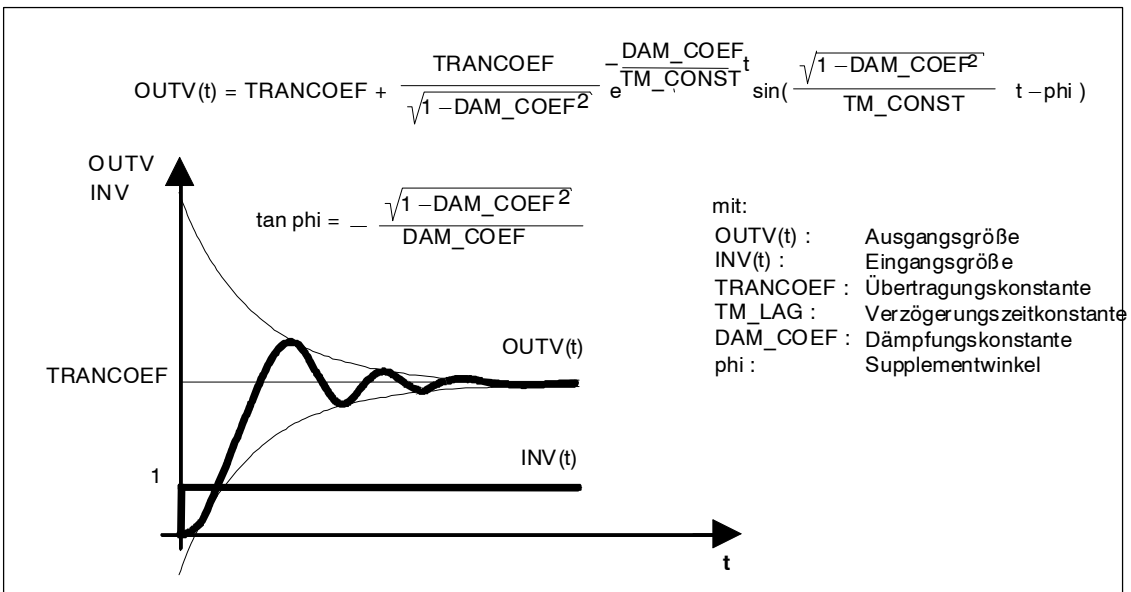


Bild 2-24 Sprungantwort des schwingungsfähigen LAG2ND-Gliedes

Eingangsparameter

Die folgende Tabelle zeigt Ihnen Datentyp und Struktur der Eingangsparameter von LAG2ND.

Tabelle 2-23 Eingangsparameter von LAG2ND

| Daten- typ | Parameter | Kommentar englisch deutsch | zulässiger Wertebereich | Vorbelegung |
|---------------|-----------|--|-----------------------------|-------------|
| REAL | INV | input variable Eingangsgröße | technischer Wertebereich | 0.0 |
| TIME | TM_CONST | time constant Zeitkonstante | | T#10s |
| REAL | DAM_COEF | damping coefficient Dämpfungskonstante | | 1.0 |
| REAL | TRANCOEF | transfer coefficient Übertragungskonstante | | 1.0 |
| REAL | DF_OUTV | default output variable Vorbesetzung der Ausgangsgröße | technischer Wertebereich | 0.0 |
| BOOL | DFOUT_ON | default output variable on Vorbesetzung der Ausgangsgröße ein | | FALSE |
| BOOL | TRACK | tracking OUTV=INV Nachführen OUTV=INV | | FALSE |
| BOOL | COM_RST | complete restart Neustart | | FALSE |
| TIME | CYCLE | sample time Abtastzeit | ≥ 1ms | T#1s |

Ausgangsparameter

Die folgende Tabelle zeigt Ihnen Datentyp und Struktur der Ausgangsparameter von LAG2ND.

Tabelle 2-24 Ausgangsparameter von LAG2ND

| Daten- typ | Parameter | Kommentar englisch deutsch | Vorbelegung |
|---------------|-----------|----------------------------------|-------------|
| REAL | OUTV | output variable Ausgangsgröße | 0.0 |

Neustart

Bei Neustart wird der Ausgang OUTV auf 0.0 gesetzt. Ist DFOUT_ON = TRUE, wird DF_OUTV ausgegeben.

Normalbetrieb

Der Baustein enthält neben dem Normalbetrieb folgende Betriebsarten:

- Nachführen

Ist TRACK =TRUE, wird OUTV = INV; die internen Vergangenheitswerte werden auf INV gesetzt.

- Ausgang vorbelegen

Ist DFOUT_ON =TRUE, wird DF_OUTV ausgegeben, die internen Vergangenheitswerte werden auf DF_OUTV gesetzt. DFOUT_ON hat höhere Priorität als TRACK.

Bausteininterne Begrenzungen

Die Zeitkonstante TM_CONST wird auf die halbe Abtastzeit nach unten begrenzt.

$$TM_CONST_{\text{intern}} = \text{CYCLE}/2 \quad \text{für } TM_CONST < \text{CYCLE}/2$$

Die anderen Werte der Eingangsparameter werden im Baustein nicht begrenzt; eine Prüfung der Parameter findet nicht statt.

2.1.11 LIMALARM: limit alarm (Grenzwertmelder)

Anwendungsbereich

Bei Regelungen können unzulässige oder gefährliche Betriebszustände auftreten, wenn Prozeßgrößen (z.B. Drehzahl, Druck, Temperatur, ...) über-/unterschritten werden. Die Über-/Unterschreitung von zulässigen Grenzen muß erfaßt und gemeldet werden, damit eine sinnvolle Reaktion erfolgen kann.

Blockschaltbild

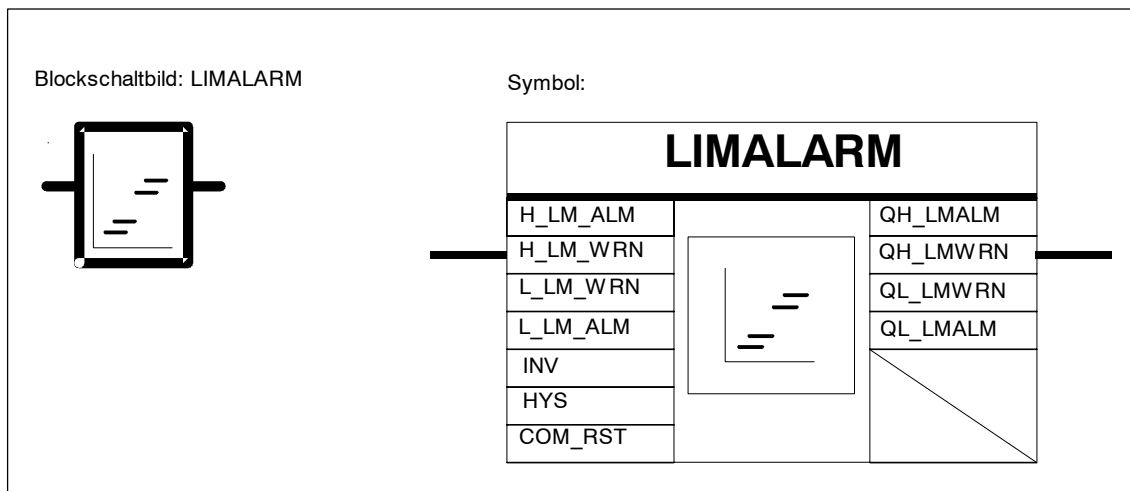


Bild 2-25 LIMALARM, Blockschaltbild und Symbol

Funktionsbeschreibung

Der Baustein überprüft die Eingangsgröße INV auf 4 vorgebbare Grenzen. Bei Erreichen und Überschreiten dieser Grenzen wird ein zugehöriges Grenzsinal ausgegeben. Die Ausschaltswelle der Grenzsinalne ist über eine Hysterese einstellbar.

Bild 2-26 zeigt Ihnen die Funktionsweise von Baustein LIMALARM:

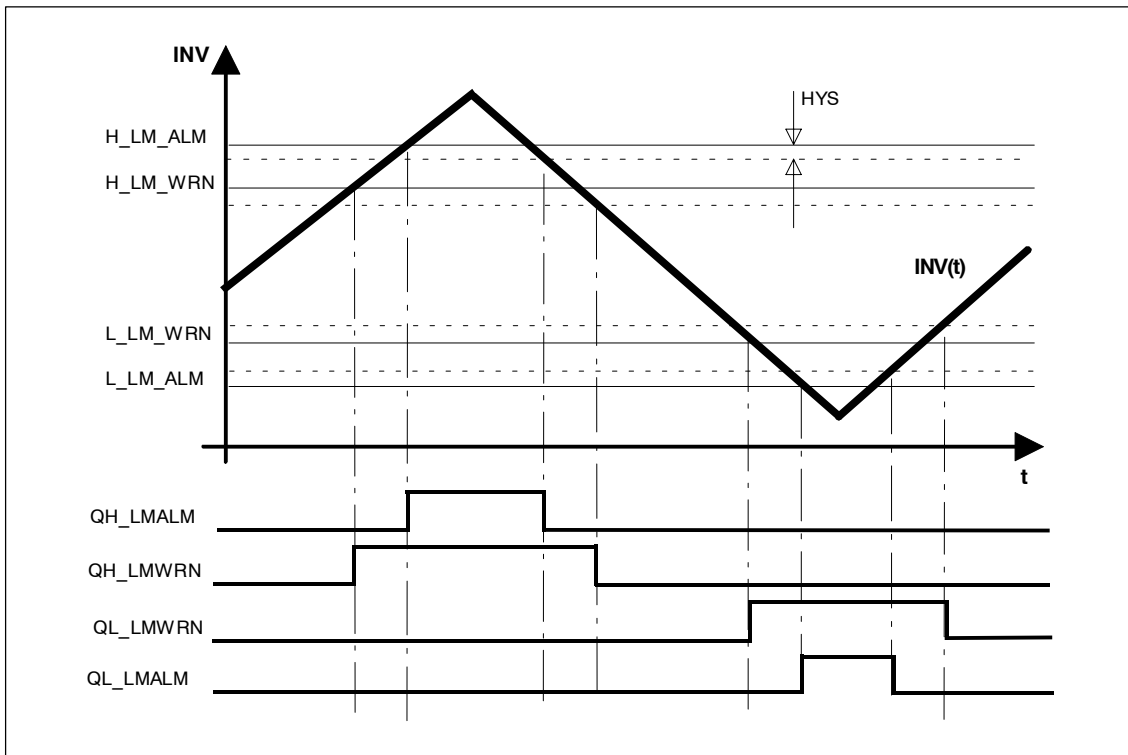


Bild 2-26 Funktionsweise von Baustein LIMALARM

Eingangsparameter

Die folgende Tabelle zeigt Ihnen Datentyp und Struktur der Eingangsparameter von LIMALARM.

Tabelle 2-25 Eingangsparameter von LIMALARM

| Daten- typ | Parameter | Kommentar englisch deutsch | zulässiger Wertebereich | Vorbelegung |
|---------------|-----------|--|-----------------------------|-------------|
| REAL | H_LM_ALM | high limit alarm oberer Grenzwert Alarm | tech. Wert. > H_LM_WRN | 100.0 |
| REAL | H_LM_WRN | high limit warning oberer Grenzwert Warnung | tech. Wert. > L_LM_WRN | 90.0 |
| REAL | L_LM_WRN | low limit warning unterer Grenzwert Warnung | tech. Wert. > L_LM_ALM | 10.0 |
| REAL | L_LM_ALM | low limit alarm unterer Grenzwert Alarm | tech. Wert. < L_LM_WRN | 0.0 |
| REAL | INV | input variable Eingangsgröße | technischer Wertebereich | 0.0 |

Tabelle 2-25 Eingangparameter von LIMALARM, Fortsetzung

| Datentyp | Parameter | Kommentar englisch deutsch | zulässiger Wertebereich | Vorbelegung |
|----------|-----------|----------------------------------|-----------------------------|-------------|
| REAL | HYS | hysteresis Hysterese | technischer Wertebereich | 1.0 |
| BOOL | COM_RST | complete restart Neustart | | FALSE |

Ausgangsparameter

Die folgende Tabelle zeigt Ihnen Datentyp und Struktur der Ausgangsparameter von LIMALARM.

Tabelle 2-26 Ausgangsparameter von LIMALARM

| Datentyp | Parameter | Kommentar englisch deutsch | Vorbelegung |
|----------|-----------|---|-------------|
| BOOL | QH_LMALM | high limit alarm reached oberer Grenzwert Alarm angesprochen | FALSE |
| BOOL | QH_LMWRN | high limit warning reached oberer Grenzwert Warnung angesprochen | FALSE |
| BOOL | QL_LMWRN | low limit warning reached unterer Grenzwert Warnung angesprochen | FALSE |
| BOOL | QL_LMALM | low limit alarm reached unterer Grenzwert Alarm angesprochen | FALSE |

Neustart

Bei Neustart werden alle Signalausgänge auf FALSE gesetzt.

Normalbetrieb

Der Baustein arbeitet nach folgenden Funktionen:

| | | | | |
|-----------------|------|------------|-----|-----------------------|
| QH_LMALM = TRUE | wenn | INV steigt | und | INV >= H_LM_ALM |
| | oder | INV sinkt | und | INV >= H_LM_ALM – HYS |
| QH_LMWRN = TRUE | wenn | INV steigt | und | INV >= H_LM_WRN |
| | oder | INV sinkt | und | INV >= H_LM_WRN – HYS |
| QL_LMWRN = TRUE | wenn | INV sinkt | und | INV <= L_LM_WRN |
| | oder | INV steigt | und | INV <= L_LM_WRN + HYS |
| QL_LMALM = TRUE | wenn | INV sinkt | und | INV <= L_LM_ALM |
| | oder | INV steigt | und | INV <= L_LM_ALM + HYS |

Für ein sinnvolles Arbeiten des Bausteins muß gelten:

$$L_LM_ALM < L_LM_WRN < H_LM_WRN < H_LM_ALM$$

An den Eingängen H_LM_ALM, H_LM_WRN, L_LM_WRN und L_LM_ALM werden die Grenzen eingestellt. Überschreitet die Eingangsgröße INV die Grenzen werden die Ausgangsmeldebite QH_LMALM, QH_LMWRN, QL_LMWRN und QL_LMALM gesetzt. Um ein schnelles Ein– Ausschalten der Meldebite zu vermeiden, muß der Eingangswert zusätzlich eine Hysterese HYS überwinden, bevor die Ausgänge rückgesetzt werden.

Bausteininterne Begrenzungen

Die Werte der Eingangsparameter werden im Baustein nicht begrenzt; eine Prüfung der Parameter findet nicht statt.

2.1.12 LIMITER: limiter (Begrenzung)

Anwendungsbereich

Bei dynamischer Vorgabe von Parametern (z.B. Berechnung von Sollwerten aus Prozeßgrößen) können diese für den Prozeß unzulässige Werte annehmen. Durch LIMITER können sie im zulässigen Bereich gehalten werden.

Blockschaltbild

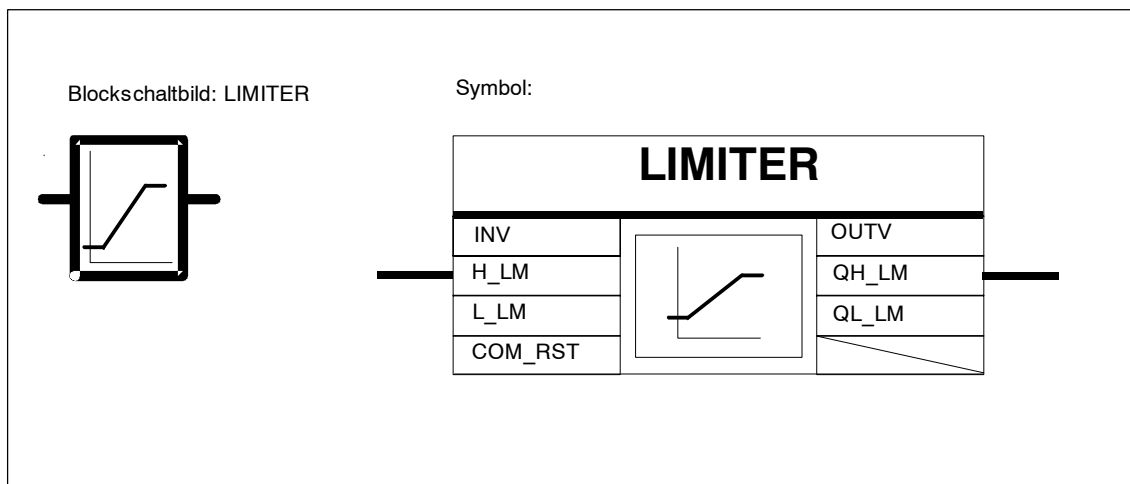


Bild 2-27 LIMITER, Blockschaltbild und Symbol

Funktionsbeschreibung

Der Baustein begrenzt die Ausgangsgröße OUTV auf eine vorgebbare obere und untere Begrenzung H_LM und L_LM, wenn die Eingangsgröße INV außerhalb dieser Begrenzungen liegt. Die Begrenzung von OUTV wird über die Ausgänge QH_LM und QL_LM gemeldet.

Bild 2-28 zeigt Ihnen die Funktionsweise von Baustein LIMITER:

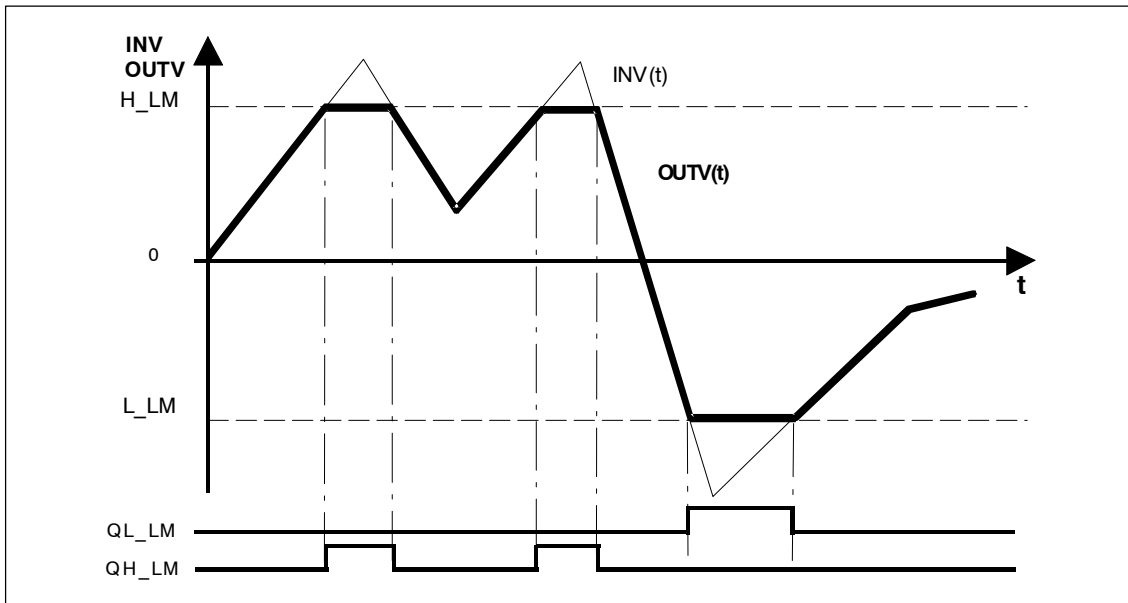


Bild 2-28 Funktionsweise von Baustein LIMITER

Eingangsparameter

Die folgende Tabelle zeigt Ihnen Datentyp und Struktur der Eingangsparameter von LIMITER.

Tabelle 2-27 Eingangsparameter von LIMITER

| Daten- typ | Parameter | Kommentar englisch deutsch | zulässiger Wertebereich | Vorbelegung |
|---------------|-----------|----------------------------------|------------------------------------|-------------|
| REAL | INV | input variable Eingangsgröße | technischer Wertebereich | 0.0 |
| REAL | H_LM | high limit obere Begrenzung | technischer Wertebereich > L_LM | 100.0 |
| REAL | L_LM | low limit obere Begrenzung | technischer Wertebereich < H_LM | 0.0 |
| BOOL | COM_RST | complete restart Neustart | | FALSE |

Ausgangsparameter

Die folgende Tabelle zeigt Ihnen Datentyp und Struktur der Ausgangsparameter von LIMITER.

Tabelle 2-28 Ausgangsparameter von LIMITER

| Daten- typ | Parameter | Kommentar englisch deutsch | Vorbelegung |
|---------------|-----------|---|-------------|
| REAL | OUTV | output variable Ausgangsgröße | 0.0 |
| BOOL | QH_LM | high limit reached obere Begrenzung angesprochen | FALSE |
| BOOL | QL_LM | low limit reached untere Begrenzung angesprochen | FALSE |

Neustart

Bei Neustart werden alle Signalausgänge auf FALSE gesetzt; am Ausgang OUTV wird 0.0 ausgegeben.

Normalbetrieb

Der Baustein arbeitet nach folgenden Funktionen:

$OUTV = H_LM,$ $QH_LM = TRUE, QL_LM = FALSE$ wenn $INV \geq H_LM$
 $OUTV = L_LM,$ $QH_LM = FALSE, QL_LM = TRUE$ wenn $INV \leq L_LM$
 $OUTV = INV,$ $QH_LM = FALSE, QL_LM = FALSE$ wenn $L_LM < INV < H_LM$

Für ein sinnvolles Arbeiten des Bausteins muß gelten: $L_LM < H_LM$

Bausteininterne Begrenzungen

Die Werte der Eingangsparameter werden im Baustein nicht begrenzt; eine Prüfung der Parameter findet nicht statt.

2.1.13 LMNGEN_C: output continuous PID controller (Ausgabe des kontinuierlichen PID-Reglers)

Anwendungsbereich

Der Baustein dient zum Aufbau eines kontinuierlichen PID-Reglers. Er beinhaltet die Stellwertbearbeitung des Reglers.

Blockschaltbild

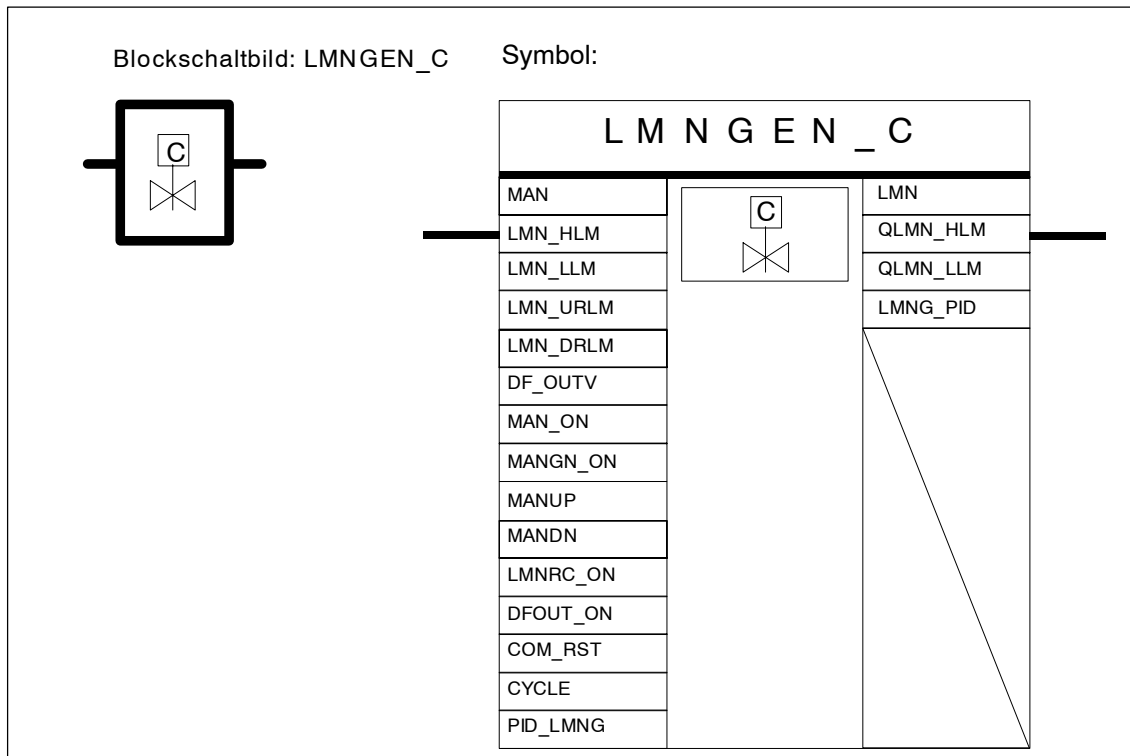


Bild 2-29 LMNGEN_C, Blockschaltbild und Symbol

Funktionsbeschreibung

Der Baustein beinhaltet die Hand-Automatik-Umschaltung. Im Handbetrieb kann der Handwert absolut vorgegeben oder per Schalterbetrieb vergrößert und verkleinert werden. Der Stellwert und die Steigung des Stellwertes können auf vorgebbare Werte begrenzt werden.

Der Baustein wird immer in Verbindung mit dem PID-Algorithmus Baustein angewandt.

• **kontinuierlicher PID-Regler: PID + LMNGEN_C**

Das Bild 2-30 zeigt Ihnen die Verschaltung des kontinuierlichen PID-Reglers.

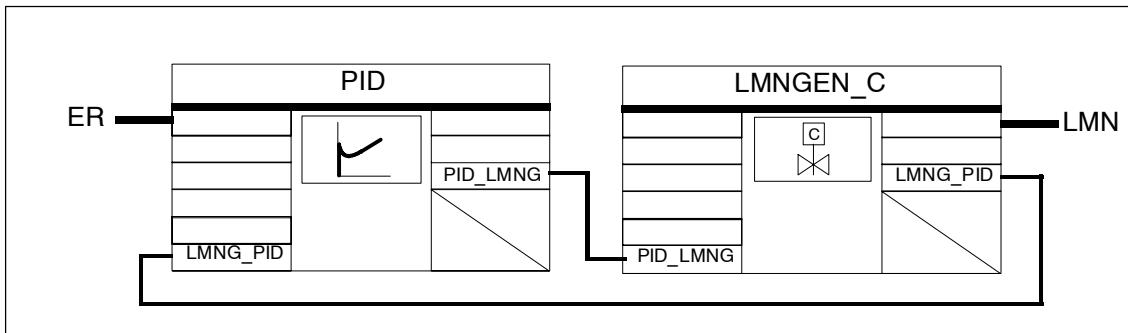


Bild 2-30 Verschaltung des kontinuierlichen PID-Reglers

Während der PID-Algorithmus in einer Weckalarmebene platziert wird, deren Zykluszeit der dominierenden Streckenzeitkonstanten angepaßt ist, kann der stgliedverarbeitende Baustein LMNGEN_C für Handeingriffe in eine schnellere Weckalarmebene untergebracht werden. Die Verschaltung erfolgt mit den strukturierten Ein-Ausgangsparameter PID_LMNG und LMNG_PID.

Eingangsparameter

Die folgende Tabelle zeigt Ihnen Datentyp und Struktur der Eingangsparameter von LMNGEN_C.

Tabelle 2-29 Eingangsparameter von LMNGEN_C

| Datentyp | Parameter | Kommentar | englisch deutsch | zulässiger Wertebereich | Vorbelegung |
|----------|-----------|---|---------------------|-----------------------------|-------------|
| REAL | MAN | manual value Handwert | | technischer Wertebereich | 0.0 |
| REAL | LMN_HLM | manipulated value high limit Stellwert obere Begrenzung | | tech. Wert. > LMN_LLM | 100.0 |
| REAL | LMN_LLM | manipulated value low limit Stellwert untere Begrenzung | | tech. Wert. < LMN_HLM | 0.0 |
| REAL | LMN_URLM | manipulated value up rate limit [1/s] Stellwert Anstiegsbegrenzung [1/s] | | > 0.0 | 10.0 |
| REAL | LMN_DRLM | manipulated value down rate limit [1/s] Stellwert Abstiegsbegrenzung [1/s] | | > 0.0 | 10.0 |
| REAL | DF_OUTV | default output variable Vorbereitung der Ausgangsgröße | | technischer Wertebereich | 0.0 |
| BOOL | MAN_ON | manual value on Handbetrieb einschalten | | | TRUE |
| BOOL | MANGN_ON | manual value generating on Handwertbedienung einschalten | | | FALSE |

Tabelle 2-29 Eingangsparmeter von LMNGEN_C, Fortsetzung

| Datentyp | Parameter | Kommentar englisch deutsch | zulässiger Wertebereich | Vorbelegung |
|----------|-----------|---|----------------------------|-------------|
| BOOL | MANUP | manual value up Handwert hoch | | FALSE |
| BOOL | MANDN | manual value down Handwert tief | | FALSE |
| BOOL | LMNRC_ON | manipulated value rate of change on Stellwertrampe einschalten | | FALSE |
| BOOL | DFOUT_ON | default output variable on Vorbesetzung der Ausgangsgröße ein | | FALSE |
| BOOL | COM_RST | complete restart Neustart | | FALSE |
| TIME | CYCLE | sample time Abtastzeit | ≥ 1ms | T#1s |
| STRUC | PID_LMNG | PID-LMNGEN interface PID-LMNGEN Schnittstelle | | |

Ausgangsparmeter

Die folgende Tabelle zeigt Ihnen Datentyp und Struktur der Ausgangsparmeter von LMNGEN_C.

Tabelle 2-30 Ausgangsparmeter von LMNGEN_C

| Datentyp | Parameter | Kommentar englisch deutsch | Vorbelegung |
|----------|-----------|--|-------------|
| REAL | LMN | manipulated value Stellwert | 0.0 |
| BOOL | QLMN_HLM | high limit of manipulated value reached obere Begrenzung des Stellwertes angesprochen | FALSE |
| BOOL | QLMN_LLM | low limit of manipulated value reached untere Begrenzung des Stellwertes angesprochen | FALSE |
| STRUC | LMNG_PID | PID-LMNGEN interface PID-LMNGEN Schnittstelle | |

Neustart

Bei Neustart wird der Vorbelegungswert DF_OUTV unabhängig vom Vorbelegungsbit DFOUT_ON auf den Ausgang LMN durchgeschaltet. Die Begrenzung des Ausgangs und deren Anzeige ist auch bei Neustart wirksam. Beim Übergang in den Normalbetrieb arbeitet der Baustein von DF_OUTV aus weiter.

Normalbetrieb

Der Baustein enthält neben dem Normalbetrieb folgende Betriebsarten:

| Betriebsarten | DFOUT_ON | MAN_ON |
|--------------------|----------|----------|
| Automatikbetrieb | FALSE | FALSE |
| Handbetrieb | FALSE | TRUE |
| Ausgang vorbelegen | TRUE | beliebig |

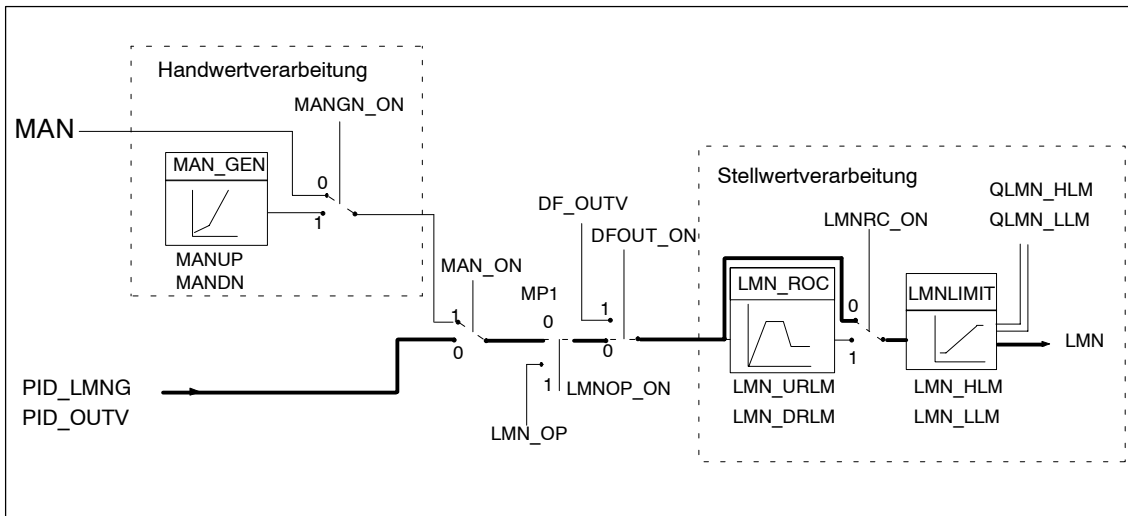


Bild 2-31 Blockschaltbild Stellwertverarbeitung des kontinuierlichen PID-Reglers

- **Automatikbetrieb**

Ist keine andere Betriebsart angewählt, wird der vom PID-Algorithmus ermittelte Wert in die Stellwertverarbeitung weitergeleitet. Die Umschaltung in den Automatikbetrieb ist stoßfrei, wenn der Stellwerthochlaufgeber (LMNRC_ON = TRUE) eingeschaltet ist.

- **Handbetrieb**

Mit dem Schalter MAN_ON kann der Anwender auf Handbetrieb umschalten und damit den Regelkreis unterbrechen. Ist MANGN_ON = TRUE, kann ausgehend vom momentanen Stellwert, dieser über die Schalter MANUP und MANDN innerhalb der Begrenzungen LMN_HLM und LMN_LLM vergrößert bzw. verkleinert werden. Die Änderungsgeschwindigkeit hängt wie folgt von den Begrenzungen ab:

während der ersten 3 s nach Setzen von MANUP bzw. MANDN:

$$dLMN/dt = (LMN_HLM - LMN_LLM) / 100 \text{ s}$$

danach: $dLMN/dt = (LMN_HLM - LMN_LLM) / 10 \text{ s}$

Ist MANGN_ON = FALSE und MAN_ON = TRUE, so wird der Eingangswert MAN als Stellwert durchgeschaltet.

An den Eingängen LMN_HLM und LMN_LLM müssen die obere und untere Begrenzung des Stellwertes eingegeben werden. Auch die Änderung des Stellwertes kann begrenzt werden. An den Eingängen LMN_URLM und LMN_DRLM werden die Stellwertan- bzw. abstiegsbegrenzungen eingestellt und über den Schalter LMNRC_ON aktiviert. Am Ausgang LMN erscheint der Stellwert. Eine Begrenzung des Stellwertes durch die Grenzen LMN_HLM und LMN_LLM werden von den Anzeigebits QLMN_HLM und QLMN_LLM gemeldet.

- **Ausgang vorbelegen**

Mit DFOUT_ON = TRUE wird der Vorbesetzungswert DF_OUTV am Ausgang LMN übernommen. Die Stellwertbegrenzungen und deren Anzeige ist wirksam. Die Umschaltung von bzw. nach "Ausgang vorbelegen" ist nur bei eingeschaltetem Stellwerthochlaufgeber (LMNRC_ON = TRUE) stoßfrei.

Bausteininterne Begrenzungen

Die Werte der Eingangsparameter werden im Baustein nicht begrenzt; eine Prüfung der Parameter findet nicht statt.

Stellwertbeeinflussung über die Projektiersoftware

LMN-Anzeige und -Einstellung im Kreisbild

Die Projektiersoftware hat eine eigene Schnittstelle zum Baustein LMNGEN_C. Es ist deshalb jederzeit möglich, (z.B. zu Testzwecken von einem PG/PC aus auf dem die Projektiersoftware geladen ist) den Stellgrößenzweig zu unterbrechen und eigene Stellwerte LMN_OP vorzugeben (Bild 2-32).

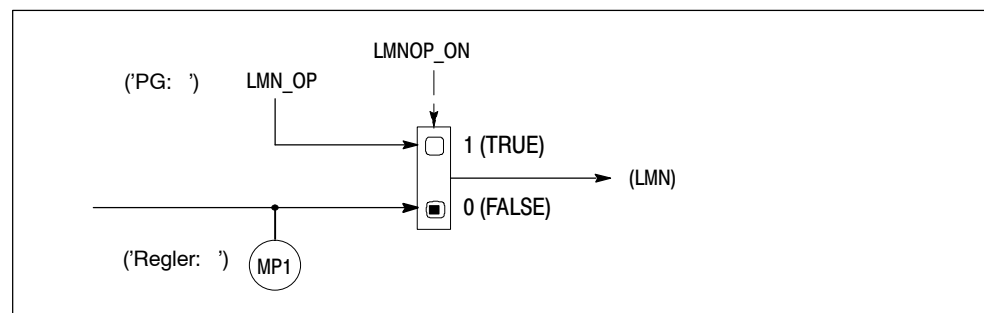


Bild 2-32 Eingriff in den Stellwertzweig über die Projektiersoftware

Im Kreisbild steht dafür ein Bedienfeld (bezeichnet mit Stellwert) zur Verfügung. Hier wird oben ('Regler: ') der aktuelle am Meßpunkt MP1 anliegende Stellwert ausgegeben. Im Feld darunter ('PG: ') wird der Parameter LMN_OP bedient.

Umschalten auf Stellwertvorgabe durch die Projektiersoftware

Steht der Schalter in der Projektiersoftware auf 'PG: ', dann wird im Regler-FB das Schaltsignal des Strukturschalters LMNOP_ON auf TRUE gesetzt und LMN_OP wird auf den Stellwert LMN durchgeschaltet.

Wenn im Stellgrößenzweig die Steigungsbegrenzung LMN_ROC eingeschaltet ist, kann stoßfrei zwischen den Schalterstellungen 'PG: ' und 'Regler: ' umgeschaltet werden. Der Wert, auf den zurückgeschaltet wird (MP1), kann im Anzeigefeld 'Regler: ' des Kreisbildes abgelesen werden. Nach Maßgabe der an LMN_ROC eingestellten Rampensteigung wird LMN danach auf diesen Wert zugeführt.

Diese Bedieneingriffe wirken erst dann auf den Prozeß, wenn sie durch Betätigen der Taste 'Senden' im Kreisbild zum Automatisierungssystem übertragen werden.

Die Parameter LMNOP_ON, LMN_OP und MP1 sind statische Variablen und an der Ein-/Ausgangsleiste nicht verfügbar. Die Parameter sollten nicht verschaltet und nur mit der Projektiersoftware bedient und beobachtet werden.

2.1.14 LMNGEN_S: output PID step controller (Ausgabe PID-Schrittregler)

Anwendungsbereich

Der Baustein dient zum Aufbau eines PID-Schrittreglers für integralwirkende Stellglieder (z.B. motorgetriebene Ventile). Er beinhaltet die Stellwertbearbeitung des Reglers. Der Schrittregler kann sowohl mit als auch ohne Stellungsrückmeldung arbeiten.

Blockschaltbild

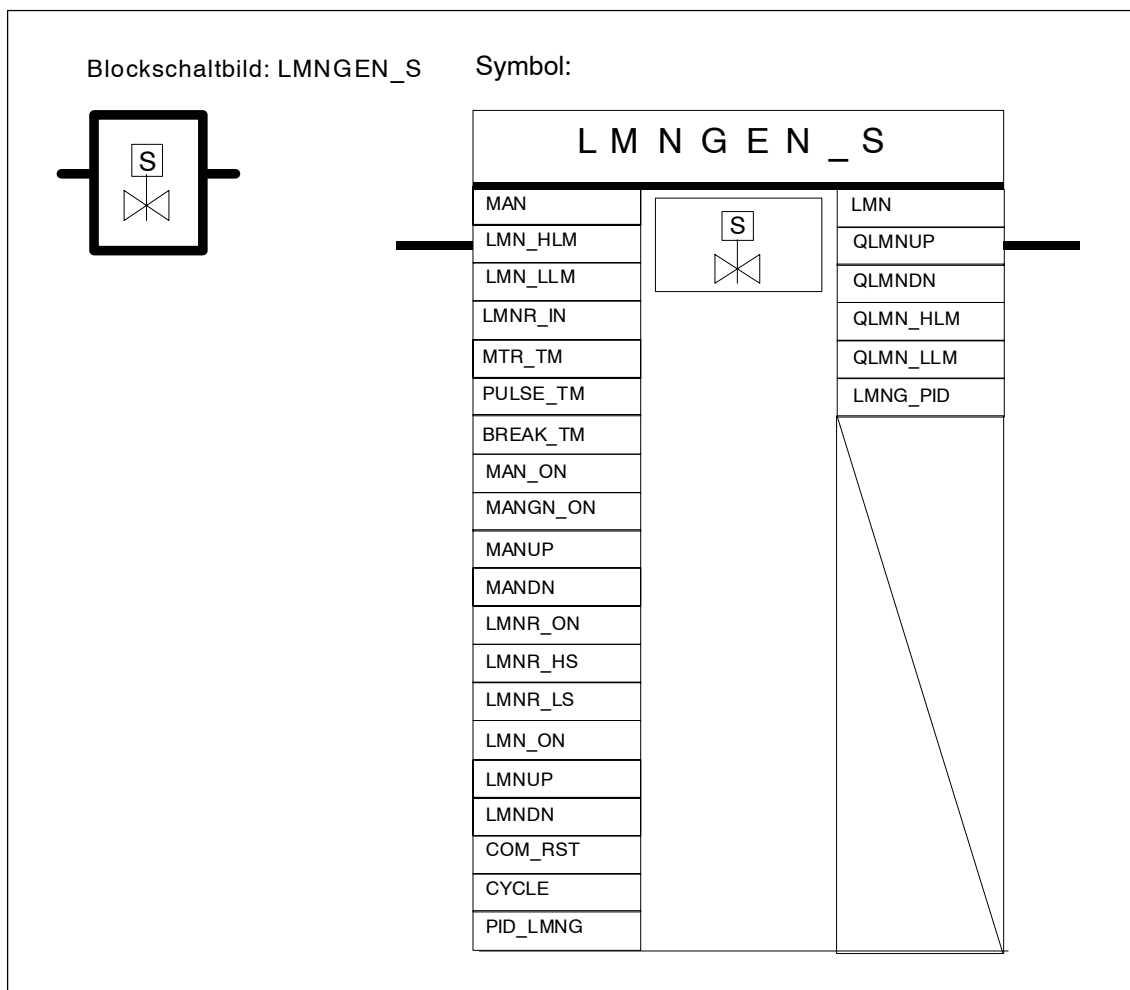


Bild 2-33 LMNGEN_S, Blockschaltbild und Symbol

Funktionsbeschreibung

Wenn eine Stellungsrückmeldung zur Verfügung steht, kann der Baustein als Stellungsregler benützt werden. Mit der Hand–Automatik–Umschaltung kann zwischen dem Stellwert vom PID–Algorithmus und einem Handwert umgeschaltet werden. Der Handwert kann absolut vorgegeben oder per Schalterbetrieb rauf und runter gefahren werden. Der Baustein bildet aus der Differenz zwischen dem Stellwert und der Stellungsrückmeldung über ein Dreipunktglied und Impulsformer die Impulse zur Ansteuerung des Stellventils. Über eine Adaption der Ansprechschwelle des Dreipunktgliedes wird die Schalthäufigkeit des Reglers reduziert.

Der Baustein arbeitet auch ohne Stellungsrückmeldung. Hierbei wird die Differenz aus dem I–Anteil des PID–Algorithmus und einer internen Stellungsrückmeldung in einem Integrator summiert und als Rückführungswert mit dem verbliebenen PD–Anteil verglichen. Die Differenz geht wiederum auf das Dreipunktglied und den Impulsformer, der die Impulse für das Stellventil bildet. Über eine Adaption der Ansprechschwelle des Dreipunktgliedes wird die Schalthäufigkeit des Reglers reduziert.

Der Baustein wird immer in Verbindung mit dem PID–Algorithmus Baustein angewandt.

- **PID–Schrittregler: PID + LMNGEN_S**

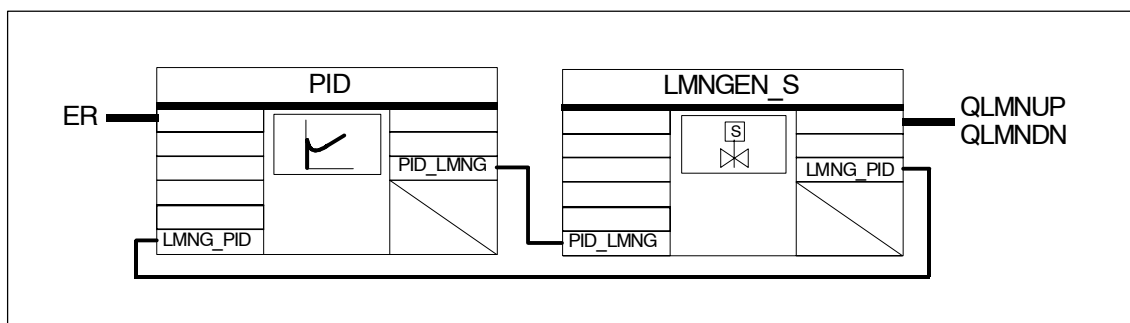


Bild 2-34 Verschaltung des PID–Schrittreglers

Während der PID–Algorithmus in einer Weckalarmebene platziert wird, deren Zykluszeit der dominierenden Streckenzeitkonstanten angepaßt ist, kann der stellungsgliedverarbeitende Baustein LMNGEN_S für Handeingriffe in eine schnellere Weckalarmebene untergebracht werden. Die Verschaltung erfolgt mit den strukturierten Ein–Ausgangsparameter PID_LMNG und LMNG_PID.

Eingangsparameter

Die folgende Tabelle zeigt Ihnen Datentyp und Struktur der Eingangsparameter von LMNGEN_S.

Tabelle 2-31 Eingangsparameter von LMNGEN_S

| Datentyp | Parameter | Kommentar englisch deutsch | zulässiger Wertebereich | Vorbelegung |
|----------|-----------|--|-----------------------------|-------------|
| REAL | MAN | manual value Handwert | technischer Wertebereich | 0.0 |
| REAL | LMN_HLM | manipulated value high limit Stellwert obere Begrenzung | tech. Wert. > LMN_LLM | 100.0 |
| REAL | LMN_LLM | manipulated value low limit Stellwert untere Begrenzung | tech. Wert. < LMN_HLM | 0.0 |
| REAL | LMNR_IN | repeated manipulated value Stellungsrückmeldung | technischer Wertebereich | 0.0 |
| TIME | MTR_TM | motor manipulated value Motorstellzeit | ≥ CYCLE | T#30s |
| TIME | PULSE_TM | minimum pulse time Mindestimpulsdauer | ≥ CYCLE | T#3s |
| TIME | BREAK_TM | minimum break time Mindestpausendauer | ≥ CYCLE | T#3s |
| BOOL | MAN_ON | manual value on Handbetrieb einschalten | | TRUE |
| BOOL | MANGN_ON | manual value generating on Handwertbedienung einschalten | | FALSE |
| BOOL | MANUP | manual value up Handwert hoch | | FALSE |
| BOOL | MANDN | manual value down Handwert tief | | FALSE |
| BOOL | LMNR_ON | repeated manipulated value on Stellungsrückmeldung einschalten | | FALSE |
| BOOL | LMNR_HS | high limit signal of repeated manipu- lated value oberes Anschlagssignal der Stellungs- rückmeldung | | FALSE |
| BOOL | LMNR_LS | low limit signal of repeated manipulated value unteres Anschlagssignal der Stellungs- rückmeldung | | FALSE |
| BOOL | LMNS_ON | manipulated signals on Handbetrieb der Stellwertsignale ein- schalten | | FALSE |
| BOOL | LMNUP | manipulated signal up Stellwertsignal Hoch | | FALSE |

Tabelle 2-31 Eingangparameter von LMNGEN_S

| Daten- typ | Parameter | Kommentar englisch deutsch | zulässiger Wertebereich | Vorbelegung |
|---------------|-----------|--|----------------------------|-------------|
| BOOL | LMNDN | manipulated signal down Stellwertsignal Tief | | FALSE |
| BOOL | COM_RST | complete restart Neustart | | FALSE |
| TIME | CYCLE | sample time Abtastzeit | ≥ 1ms | T#1s |
| STRUC | PID_LMNG | PID-LMNGEN interface PID-LMNGEN Schnittstelle | | |

Ausgangparameter

Die folgende Tabelle zeigt Ihnen Struktur der Ausgangparameter von LMNGEN_S.

Tabelle 2-32 Ausgangparameter von LMNGEN_S

| Daten- typ | Parameter | Kommentar englisch deutsch | Vorbelegung |
|---------------|-----------|--|-------------|
| REAL | LMN | manipulated value Stellwert | 0.0 |
| BOOL | QLMNUP | manipulated signal up Stellwertsignal up | FALSE |
| BOOL | QLMNDN | manipulated signal down Stellwertsignal Tief | FALSE |
| BOOL | QLMN_HLM | high limit of manipulated value reached obere Begrenzung des Stellwertes angesprochen | FALSE |
| BOOL | QLMN_LLM | low limit of manipulated value reached untere Begrenzung des Stellwertes angesprochen | FALSE |
| STRUC | LMNG_PID | PID-LMNGEN interface PID-LMNGEN Schnittstelle | |

Neustart

Bei Neustart werden alle Signalausgänge auf Null gesetzt.

Normalbetrieb

Der Baustein enthält neben dem Normalbetrieb folgende Betriebsarten:

| Betriebsarten | | LMNR_ON | LMN_ON | MAN_ON |
|---|--|---------|--------|----------|
| Schrittregler mit Stellungsrückmeldung | Automatikbetrieb | TRUE | FALSE | FALSE |
| | Stellwert-Handbetrieb | TRUE | FALSE | TRUE |
| | Handbetrieb der Stellwertausgangssignale | TRUE | TRUE | beliebig |
| Schrittregler ohne Stellungsrückmeldung | Automatikbetrieb | FALSE | FALSE | FALSE |
| | Handbetrieb der Stellwertausgangssignale | FALSE | TRUE | beliebig |

Da bei der Betriebsart "Schrittregler ohne Stellungsrückmeldung" der Regler keine Information über den Wert der Ventilstellung besitzt, macht es auch keinen Sinn einen Handwert vorzugeben. Bei Schrittreglern ohne Stellungsrückmeldung gibt es keine Betriebsart "Stellwert-Handbetrieb".



Warnung

Wenn keine Endlagensignale vorhanden sind, kann der Regler einen Ventilanschlag nicht erkennen und es besteht z.B. die Möglichkeit, dass der Regler Signale zum Öffnen des Ventils ausgibt, obwohl das Ventil sich am oberen Anschlag befindet. Sind keine Endlagensignale vorhanden, müssen die Eingänge LMNR_HS und LMNR_LS mit FALSE belegt werden.

Schrittregler mit Stellungsrückmeldung

Steht eine Stellungsrückmeldung zur Verfügung muß LMNR_ON = TRUE gesetzt werden. Der Baustein arbeitet dann als Stellungsregler.

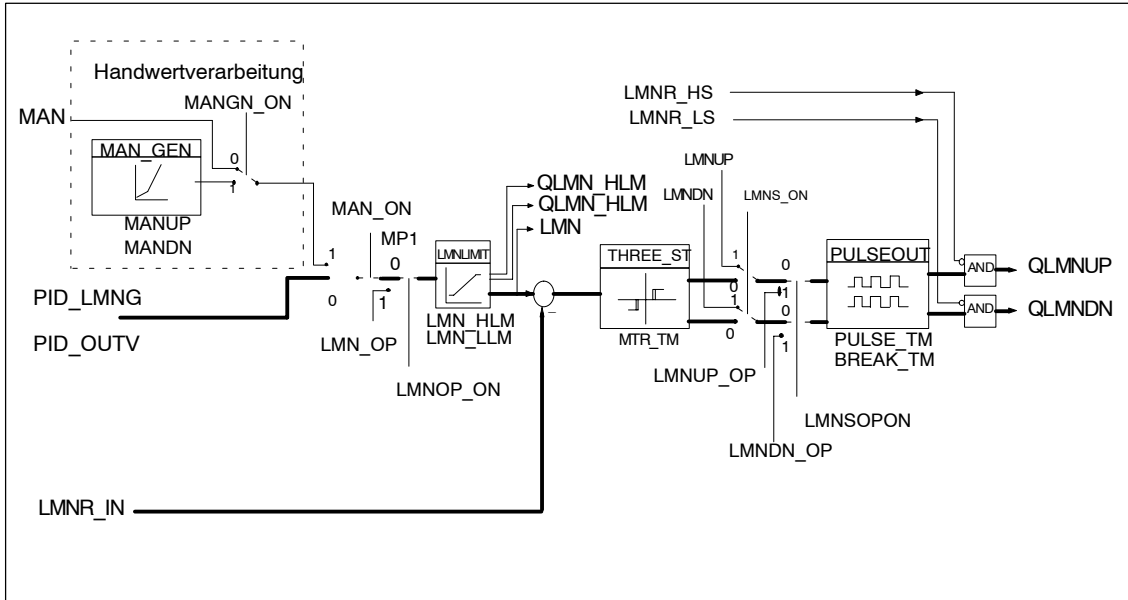


Bild 2-35 Schrittregler mit Stellungsrückmeldung

- **Stellwert-Handbetrieb**

In der Betriebsart Stellungsregelung kann der Anwender mit dem Schalter MAN_ON auf Handbetrieb umschalten und damit entweder absolut über den Eingang MAN oder über den Handwertgeber MAN_GEN einen Stellwert vorgeben.

Ist MANGN_ON = TRUE, kann ausgehend vom momentanen Stellwert dieser über die Schalter MANUP und MANDN innerhalb der Begrenzungen LMN_HLM und LMN_LLM vergrößert bzw. verkleinert werden. Die Änderungsgeschwindigkeit hängt wie folgt von den Begrenzungen ab:

während der ersten 3 Sekunden nach Setzen von MANUP bzw. MANDN:

$$dLMN/dt = (LMN_HLM - LMN_LLM) / 100 \text{ s}$$

danach: $dLMN/dt = (LMN_HLM - LMN_LLM) / 10 \text{ s}$

Ist MANGN_ON = FALSE und MAN_ON = TRUE, so wird der Eingangswert MAN als Stellwert durchgeschaltet.

- **Handbetrieb der Stellwertausgangssignale**

Ist LMNS_ON = TRUE, können die binären Ausgangssignale direkt beeinflusst werden. Über die Schalter LMNUP und LMNDN werden die Stellsignalausgänge QLMNUP und QLMNDN gesetzt. Die Mindestimpulszeit PULSE_TM und Mindestpausendauer BREAK_TM werden mitberücksichtigt. Ist einer der Endlagenschalter LMNR_HS bzw. LMNR_LS gesetzt, so wird das entsprechende Ausgangssignal QLMNUP bzw. QLMNDN gesperrt.

- **Automatikbetrieb**

Der Stellwert vom PID-Algorithmus PID_LMNG.PID_OUTV wird auf vorgebbare Werte LMN_HLM und LMN_LLM begrenzt. Die Differenz zwischen Stellwert und Stellungsrückmeldung wird auf ein Dreipunktglied mit Hysterese geschaltet. Die Abschaltswelle wird aus der Motorstellzeit berechnet. Zur Reduzierung der Schalthäufigkeit wird die Ansprechschwelle adaptiert. Der Impulsformer PULSEOUT gewährleistet, dass die Mindestimpuls- und Pausendauer eingehalten werden. Ist einer der Endlagenschalter LMNR_HS bzw. LMNR_LS gesetzt, so wird das entsprechende Ausgangssignal QLMNUP bzw. QLMNDN gesperrt.

Zur Reduzierung der Schalzhäufigkeit wird die Anspeichschwelle adaptiert. Der Impulsformer PULSEOUT gewährleistet, dass die Mindestimpuls- und Pausendauer eingehalten werden. Ist einer der Endlagenschalter LMNR_HS bzw. LMNR_LS gesetzt, so wird das entsprechende Ausgangssignal QLMNUP bzw. QLMNDN gesperrt.

Simulation der Stellungsrückmeldung

Die Simulation der Stellungsrückmeldung erfolgt durch einen Integrierer mit der Motorstellzeit MTR_TM als Integrationszeitkonstante. Durch Umschaltung des Parameter LMNRS_ON von FALSE auf TRUE wird die Simulation mit dem Startwert LMNRSVAL gestartet. Steht LMNRS_ON auf FALSE, so wird am Parameter LMNR_SIM der Startwert LMNRSVAL ausgegeben. Die Parameter LMNRS_ON, LMNRSVAL und LMNR_SIM liegen im statischen Variablenbereich. Sie werden bedient von der Projektiersoftware. Ist LMNR_HS = TRUE wird die Integration nach oben gesperrt. Ist LMNR_LS = TRUE wird die Integration nach unten gesperrt.



Warnung

Die Stellungsrückmeldung wird lediglich simuliert. Sie muß nicht übereinstimmen mit der realen Stellungsrückmeldung des Stellventils.

Die PID-Parameter-Optimierung der Projektiersoftware benötigt die Stellungsrückmeldung als Eingangsgröße.

Ist eine reale Stellungsrückmeldung vorhanden, so sollte diese auch genutzt werden.

Bausteininterne Begrenzungen

Bausteinintern werden keine Werte begrenzt; eine Prüfung der Parameter findet nicht statt.

Stellwertbeeinflussung über die Projektiersoftware

LMN-Anzeige und -Einstellung im Kreisbild

Die Projektiersoftware hat eine eigene Schnittstelle zum LMNGEN_S. Es ist deshalb jederzeit möglich, von einem PG/PC aus über die Projektiersoftware den Stellgrößenzweig zu unterbrechen und eigene Stellwerte vorzugeben (Bild 2-37).

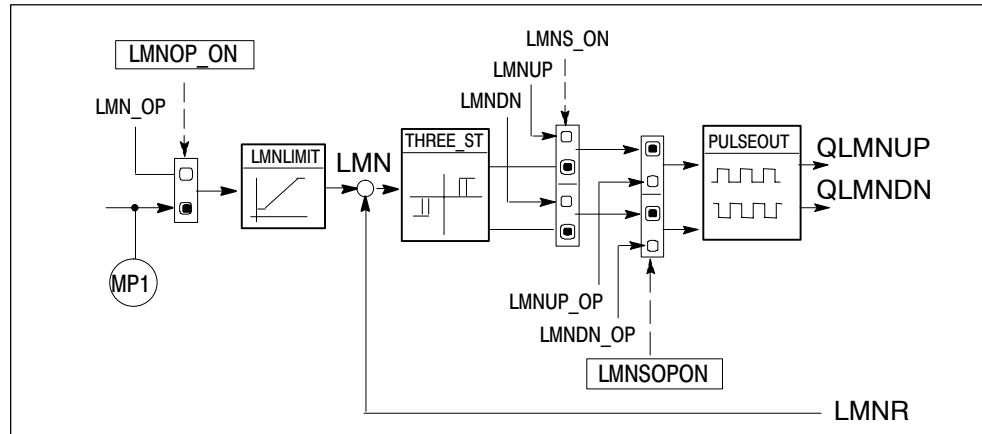


Bild 2-37 Eingriffe in den Stellwertzweig über die Projektiersoftware

Im Fenster des Kreisbildes steht dafür ein Bedienfeld- bezeichnet mit 'Stellwert' zur Verfügung. Hier wird oben ('Regler: ') der aktuelle am Meßpunkt MP1 anliegende Stellwert ausgegeben. Im Feld darunter ('PG: ') wird Parameter LMN_OP bedient.

Umschalten auf Stellwertvorgabe durch die Projektiersoftware

Steht der Schalter in der Projektiersoftware auf 'PG: ', dann wird im Regler-FB das Schaltsignal des Strukturschalters LMNOP_ON auf TRUE gesetzt und LMN_OP wird auf den Stellwert durchgeschaltet.

Ein Umschalten zwischen den Schalterstellungen 'PG: ' und 'Regler: ' ist nur dann stoßfrei, wenn im Stellwertzweig die Steigungsbegrenzung LMN_ROC eingeschaltet ist. Der Wert, auf den zurückgeschaltet wird (MP1), kann im Anzeigefeld 'Regler: ' des Kreisbildes abgelesen werden. Nach Maßgabe der an LMN_ROC eingestellten Rampensteigung wird LMN danach auf diesen Wert zugeführt

Steht der Schalter 'Regler:/PG: ' im Feld Stellsignale auf 'PG:', wird der Parameter LMNSOPON=TRUE gesetzt und die Stellsignalausgänge können über die Parameter LMNUP_OP (Hoch) bzw. LMNDN_OP (Tief) im Kreisbild bedient werden. Dies gilt für den Schrittreger mit und ohne Stellungsrückmeldung.

Diese Bedieneingriffe wirken jeweils dann auf den Prozeß, wenn Sie die Taste 'Senden' im Kreisbild betätigen.

Die Parameter LMNOP_ON, LMN_OP, MP1, LMNSOPON, LMNUP_OP und LMNDN_OP sind statische Variablen und an der Ein-/Ausgangsleiste nicht verfügbar. Die Parameter sollten nicht verschaltet und nur mit der Projektiersoftware bedient und beobachtet werden.

2.1.15 LP_SCHED: loop scheduler (Regleraufrufverteiler)

Anwendung

Müssen viele Regler mit unterschiedlichen Abtastzeiten, insbesondere träge Regelungen mit großen Abtastzeiten, aufgerufen werden, dann reicht der Umfang des Prioritätsklassenmodells in Bezug auf nutzbare Weckalarmebenen nicht aus. Der Regleraufrufverteiler (LP_SCHED) macht es nun möglich, in einer Weckalarmebene mehrere Regler mit unterschiedlichen Abtastzeiten unterzubringen. Die einzelnen Regler werden dabei mit ihren Abtastzeiten zyklisch aufgerufen.

Der Einsatz des Aufrufverteilers ist nicht obligatorisch. Die Regler-FBs können auch ohne die Verteilungsfunktion direkt aus dem OB aufgerufen werden.

Blockschaltbild

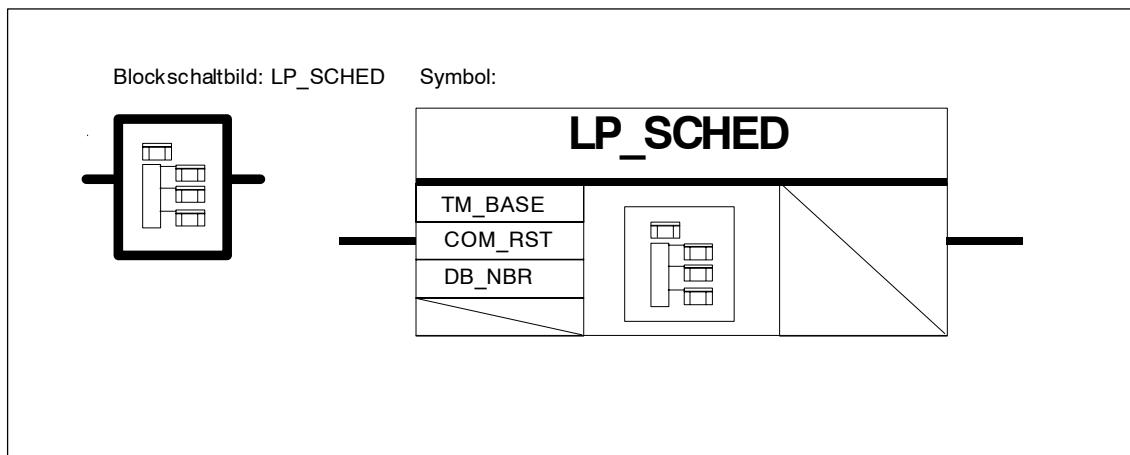
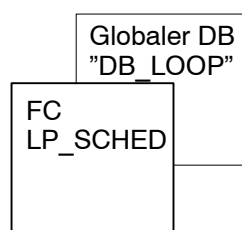


Bild 2-38 LP_SCHED, Blockschaltbild und Symbol

Funktionsbeschreibung

Die Aufrufverteilung mehrerer Regler in einer Weckalarmebene ist in der Funktion LP_SCHED realisiert. Der Baustein muß vor allen Regelkreisen aufgerufen werden. Die Daten für die einzelnen Regleraufrufe sind in einem globalen Datenbaustein (DB_LOOP) hinterlegt.



Der Aufrufverteiler bearbeitet den globalen Datenbaustein und setzt abhängig von der Reihenfolge und den parametrisierten Abtastzeiten der Regler die ENABLE-Bits. Der Zeittakt der Weckalarmebene wird dadurch untersetzt. Die einzelnen Regelkreise in dieser Weckalarmebene werden entsprechend ihrer eingestellten Abtastzeit aufgerufen und bearbeitet. Nach dem Bausteinaufruf muß das ENABLE-Bit wieder zurückgesetzt werden. Die Bausteinaufrufe und das Rücksetzen der ENABLE-Bits müssen programmiert werden.

Der Aufruf einzelner Regelkreise kann manuell gesperrt werden. Auch lassen sich einzelne Regelkreise rücksetzen (Neustart).



Hinweis

Der Baustein überprüft weder, ob ein globaler DB mit der Nummer DB_NBR wirklich vorhanden ist, noch, ob der Parameter GLP_NBR (größte Regelkreisnummer) zur Länge des Datenbausteins paßt. Bei inkorrektur Parametrierung geht die CPU mit "internem Systemfehler" in STOP.

Eingangsparameter

Die folgende Tabelle zeigt Ihnen Datentyp und Struktur der Eingangsparameter von LP_SCHED.

Tabelle 2-33 Eingangsparameter von LP_SCHED

| Datentyp | Parameter | Kommentar englisch deutsch | zulässiger Wertebereich |
|----------|-----------|--|-------------------------|
| TIME | TM_BASE | time base Zeitbasis | > =1ms |
| BOOL | COM_RST | complete restart Neustart | |
| BLOCK_DB | DB_NBR | data block number Datenbausteinnummer | |

Ausgangsparameter

Die folgende Tabelle zeigt Ihnen Datentyp und Struktur der Ausgangsparameter von LP_SCHED.

Tabelle 2-34

| Daten- typ | Parameter | Kommentar englisch deutsch |
|---|-----------|----------------------------------|
| Der Baustein besitzt keine Ausgangsparameter. | | |

Neustart

Bei COM_RST = TRUE werden folgende Vorbelegungen eingestellt:

Aktuelle Regelkreisnummer: ALP_NBR = 0

Die Aufrufdaten aller Regelkreise bis **GLB_NBR** werden wie folgt vorbelegt:

Freigabe: ENABLE = NOT MAN_DIS

Abtastzeit: CYCLE = GV(MAN_CYC)

Neustart: COM_RST = TRUE

Lokale Aufrufnummer: ILP_COU = 0

GV(MAN_CYC) = MAN_CYC wird gerundet auf ein ganzzahliges Vielfaches von TM_BASE*GLP_NBR

Globaler Datenbaustein "DB_LOOP"

Die folgende Tabelle zeigt Ihnen den Globalen Datenbaustein DB_LOOP für den Regleraufruf.

Tabelle 2-35

| Datentyp | Parameter | Kommentar englisch deutsch | zulässiger Wertebereich | Vorbelegung |
|----------|--------------------------|---|----------------------------|-------------|
| INT | GLP_NBR | greatest loop number größte Regelkreisnummer | 1 – 256 | 2 |
| INT | ALP_NBR | actual loop number aktuelle Regelkreisnummer | 0 – 256 | 0 |
| TIME | LOOP_DAT[1]. MAN_CYC | loop data [1] manual sample time Regelkreisdaten [1] manuelle Abtastzeit | >=1ms | T#1s |
| BOOL | LOOP_DAT[1]. MAN_DIS | loop data [1] manual loop disable Regelkreisdaten [1] manuell Regleraufruf sperren | | FALSE |
| BOOL | LOOP_DAT[1]. MAN_CRST | loop data [1] manual complete restart Regelkreisdaten [1] manuell Neustart setzen | | FALSE |
| BOOL | LOOP_DAT[1]. ENABLE | loop data [1] enable loop Regelkreisdaten [1] Reglerfreigabe | | FALSE |
| BOOL | LOOP_DAT[1]. COM_RST | loop data [1] complete restart Regelkreisdaten [1] Neustart | | FALSE |
| INT | LOOP_DAT[1]. ILP_COU | loop data [1] internal loop counter Regelkreisdaten [1] interner Regelkreiszähler | | 0 |
| TIME | LOOP_DAT[1]. CYCLE | loop data [1] sample time Regelkreisdaten [1] Abtastzeit | >=1ms | T#1s |

Der Regelkreisaufruf über LP_SCHED

Die Funktion LP_SCHED wird über drei Eingangsparameter in das Aufrufschema der CPU eingebunden. Am Eingang TM_BASE wird der Zeittakt der Weckalarmebene angegeben. Der Bearbeitungsaufwurf der zugeordneten Regelkreise geschieht über einen bedingten Bausteinaufruf aus einer Weckalarmebene (z. B. OB35). Hierbei werden die ENABLE-Bits im globalen Datenbaustein abgefragt.

Falls der Aufruf aus der Neustartebene heraus erfolgt, wird der Eingang COM_RST = TRUE gesetzt. In der Weckalarmebene muß dieser Aufruf wieder auf FALSE zurückgesetzt werden. Der globale Datenbaustein (Tabelle 2-35) mit den zeitrelevanten Daten der Regelkreise in der betreffenden Weckalarmebene wird über den Eingangsparameter DB_NBR zugeordnet.

Parametrierung des Regleraufrufs (globaler DB)

Die Parametrierung des Regleraufrufverteilers muß ohne Unterstützung der Projektiersoftware vorgenommen werden.

Der Datenbaustein (DB_LOOP) enthält sowohl einen Parameter, der die Anzahl der Regelkreise festlegt, die in der betreffenden Weckalarmebene insgesamt zu bearbeiten sind (max. 256), als auch einen Parameter, der die aktuell bearbeiteten Regelkreise anzeigt:

| | |
|---------|--|
| GLP_NBR | größte Regelkreisnummer |
| ALP_NBR | Nummer des im Zyklus bearbeiteten Regelkreises |

Die Nummer des jeweiligen Regelkreises ergibt sich aus der Platzierung seiner Aufrufdaten in der Abfolge der Eintragungen im DB.

Die Aufrufdaten der einzelnen Regelkreise liegen strukturiert im Feld LOOP_DAT. Sollen Regelkreise hinzugefügt werden, so muß die Feldlänge von LOOP_DAT angepaßt werden. Hierzu müssen Sie in der Deklarationsansicht den Datentyp ARRAY anpassen. Bei z.B. 10 Regelkreisen müssen Sie ARRAY[1..10] eintragen. Weiterhin müssen Sie in der Datenansicht den Parameter GLP_NBR anpassen. Er darf nie größer als die Feldlänge sein.

Die Parameter COM_RST und CYCLE in den Aufrufdaten müssen mit den entsprechenden Eingangsparametern am FB des aufgerufenen Regelkreises verschaltet werden. Diese Verschaltung ist vom Anwender zu programmieren. Ist der Parameter ENABLE gesetzt, soll der entsprechende Regelkreis aufgerufen werden. Nach dem Regleraufruf muß das ENABLE-Bit zurückgesetzt werden. Der bedingte Regleraufruf und das Rücksetzen des ENABLE-Bits müssen vom Anwender programmiert werden.

Mit Hilfe der manuell einstellbaren Parameter MAN_CYC/MAN_DIS/MAN_CRST läßt sich steuern, ob ein Regelkreises aufgerufen oder nicht aufgerufen werden kann. Diese Aufrufdaten können online - d.h. bei laufendem Betrieb - geändert werden, sofern dabei nur die Parameter überschrieben und nicht der ganze DB neu generiert wird. Es bedeutet:

| | |
|----------|---|
| MAN_CYC | Abtastzeit des betreffenden Reglers (wird in CYCLE auf ganzzahliges Vielfaches von TM_BASE * GLP_NBR gerundet). |
| MAN_DIS | Sperrung des Regleraufrufs, |
| MAN_CRST | Neustart für diesen Regler. |

Die Aufrufbearbeitung

Der betreffende Regelkreis wird, abhängig vom Wert des ENABLE-Signals der Regleraufrufdaten, gemäß vorhergehender DB-Parametrierung bearbeitet.

Der Datenbaustein wird von oben nach unten abgearbeitet. Pro Zyklus geht der Aufrufverteiler um eine Regelkreisnummer (ALP_NBR) in der Abfolge des DB weiter. Der interne Zähler ILP_COU wird dabei jeweils um 1 heruntergezählt. Ist ILP_COU = 0, dann setzt der Aufrufverteiler das ENABLE-Bit des betreffenden Regelkreises. Das Rücksetzen der ENABLE-Bits muß der Anwender nach dem Regleraufruf programmieren.

Der Parameter MAN_CYC wird bei der Bearbeitung auf CYCLE übertragen:

$$\text{CYCLE} = \text{GV} (\text{MAN_CYC}), \quad \text{GV} = \text{ganzzahliges Vielfaches}$$

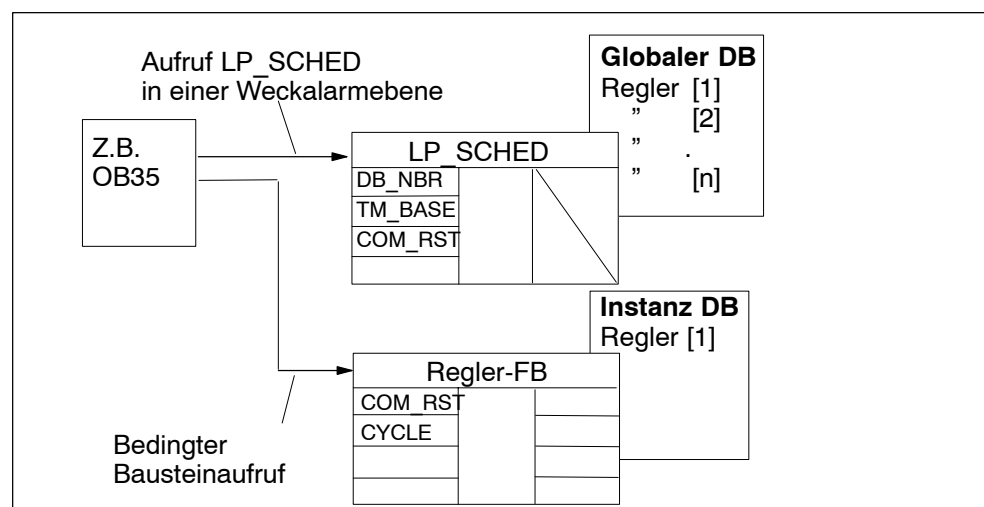


Bild 2-39 Prinzip des Regleraufrufs über den Aufrufverteiler LP_SCHED

- Einzelne Regelkreise sperren:
Setzt man durch Parametrieren im DB das Bit "MAN_DIS", dann wird das ENABLE-Bit auf FALSE zurückgesetzt und der betreffende Regelkreis von der Bearbeitung im Aufrufverteiler ausgeschlossen.
- Einzelne Regelkreise rücksetzen (Neustart):
Setzt man durch Parametrieren im DB das Bit "MAN_CRST", dann wird COM_RST = TRUE und MAN_CRST anschließend zurückgesetzt. Der betreffende Regelkreis wird mit seiner Neustartroutine bearbeitet. Im nächsten Aufrufzyklus wird automatisch auch COM_RST = FALSE zurückgesetzt.

Hinweis

Wird ein Regelkreis eingefügt oder gelöscht, d.h. wird der ganze DB neu generiert, ohne dass der Aufrufverteiler im Neustart bearbeitet werden soll, dann müssen die internen Regelkreiszähler (ILP_COU[n]) und der Parameter für die aktuelle Regelkreisnummer ALP_NBR mit Null vorbelegt werden.

Bedingungen für Regelkreisaufruf über LP_SCHED

Um zu gewährleisten, dass die Zeitabstände zwischen den Aufrufen eines bestimmten Reglers konstant bleiben und um die CPU gleichmäßig auszulasten, darf pro Zeittakt der Weckalarmebene nur ein Regelkreis bearbeitet werden. Bei der Parametrierung der Abtastzeiten MAN_CYC müssen deshalb in Relation zum Zeittakt (TM_BASE) folgende Bedingungen eingehalten werden:

- Die Bearbeitungszeiten der einzelnen Regelkreise müssen kleiner sein als der Zeittakt (TM_BASE) der Weckalarmebene.
- Die Abtastzeit eines Regelkreises (MAN_CYC) muß ein ganzzahliges Vielfaches (GV) des Produkts aus Zeitbasis und Anzahl der zu bearbeitenden Regler (GLP_NBR) sein:

$$\text{MAN_CYC} = \text{GV} (\text{TM_BASE} * \text{GLP_NBR}).$$

Beispiel für Aufrufverteilung

Das folgende Beispiel zeigt die Aufrufreihenfolge von vier Regelkreisen in einer Weckalarmebene (Bild 2-40). Pro Einheit der Zeitbasis wird immer nur ein Regelkreis bearbeitet. Die Aufrufreihenfolge und damit die Verschiebezeiten (VZ1 ... VZ5) ergeben sich aus der Reihenfolge der Aufrufdaten innerhalb des globalen DB.

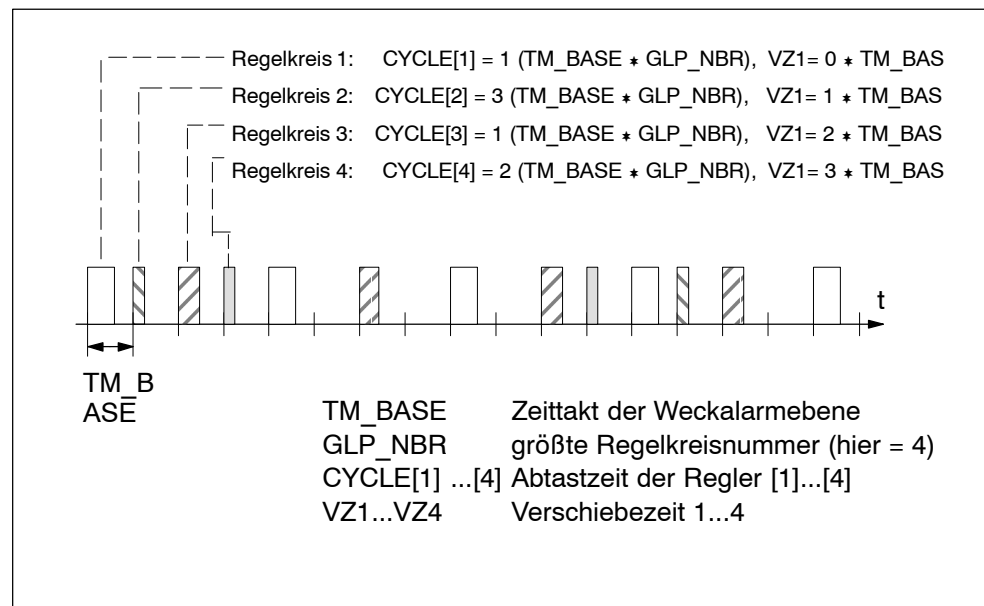


Bild 2-40 Aufrufreihenfolge von vier unterschiedlich häufig bearbeiteten Regelkreisen

Bausteininterne Begrenzungen

Die Werte der Eingangsparameter werden im Baustein nicht begrenzt; eine Prüfung der Parameter findet nicht statt.

2.1.16 NONLIN: nonlinear static function (nichtlineare statische Kennlinie)

Anwendungsbereich

Mit NONLIN kann ein Eingangswert, z.B. Meßwert von einem Thermoelement, über eine vorgebbare Kennlinie (Polygonzug) angepaßt werden.

Blockschaltbild

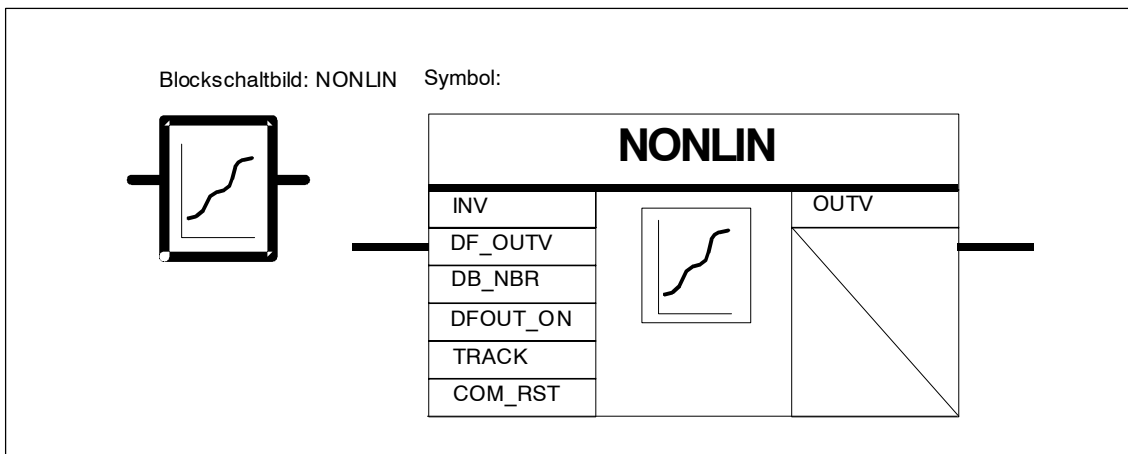
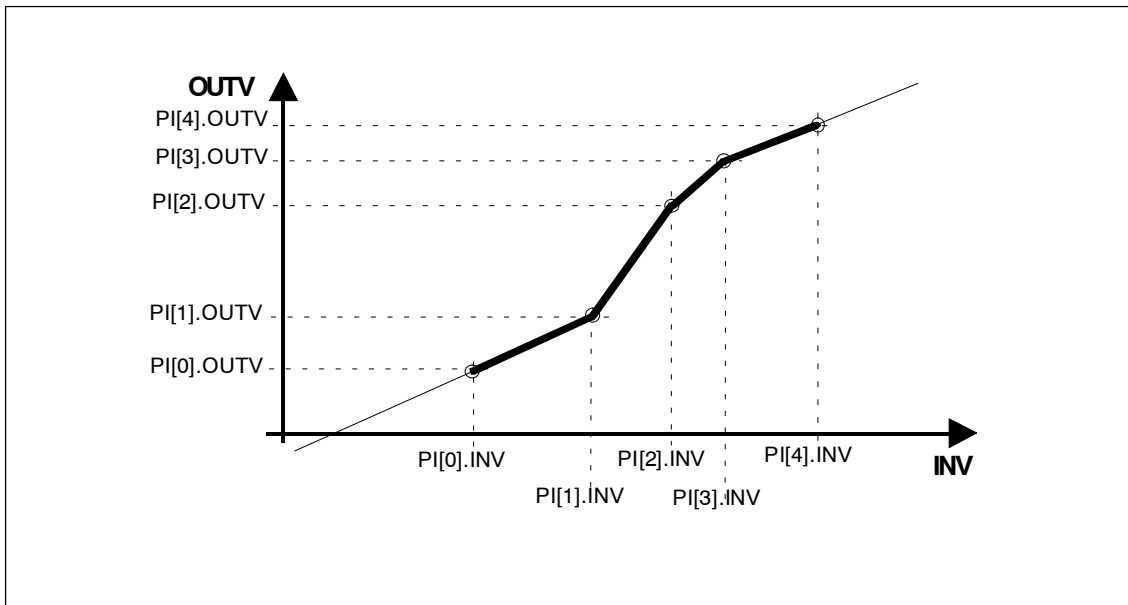


Bild 2-41 NONLIN, Blockschaltbild und Symbol

Funktionsbeschreibung

Die Stützpunkte des Polygonzuges sind in einem globalen Datenbaustein abgelegt. NONLIN weist dem momentanen Eingangswert den entsprechenden Ausgangswert auf dem Polygonzug zu. Liegt der Eingangswert unterhalb von Stützpunkt 0, so wird mit der Steigung des Stützpunktpaares 0/1 extrapoliert. Liegt er oberhalb des letzten Stützpunktes, so wird mit der Steigung des letzten Stützpunktpaares extrapoliert. Damit der Baustein ein sinnvolles Ergebnis liefert, müssen die Werte der Stützpunkte in aufsteigender Reihenfolge streng monoton steigend sein.

Über Steuereingänge kann ein parametrierter Wert oder die Eingangsgröße direkt als Ausgangsgröße ausgegeben werden.

Bild 2-42 $OUTV = f(INV)$ **Hinweis**

Der Baustein überprüft weder, ob ein globaler DB mit der Nummer DB_NBR wirklich vorhanden ist, noch ob der Parameter DB_NBR.NBR_PTS (Anzahl der Stützpunkte) zur Länge des Datenbausteins paßt. Bei falscher Parametrierung geht die CPU mit der Meldung "interner Systemfehler" in den Betriebszustand STOP.

Eingangsparameter

Die folgende Tabelle zeigt Ihnen Datentyp und Struktur der Eingangsparameter von NONLIN.

Tabelle 2-36 Eingangsparameter von NONLIN

| Datentyp | Parameter | Kommentar englisch deutsch | zulässiger Wertebereich | Vorbelegung |
|----------|-----------|---|-----------------------------|-------------|
| REAL | INV | input variable Eingangsgröße | technischer Wertebereich | 0.0 |
| REAL | DF_OUTV | default output variable Vorbesetzung der Ausgangs- größe | technischer Wertebereich | 0.0 |
| BLOCK_DB | DB_NBR | data block number Datenbausteinnummer | | DB 1 |
| BOOL | DFOUT_ON | default output variable on Vorbesetzung der Ausgangs- größe ein | | FALSE |
| BOOL | TRACK | tracking OUTV=INV Nachführen OUTV=INV | | FALSE |
| BOOL | COM_RST | complete restart Neustart | | FALSE |

Ausgangsparameter

Die folgende Tabelle zeigt Ihnen Datentyp und Struktur der Ausgangsparameter von NONLIN.

Tabelle 2-37 Ausgangsparameter von NONLIN

| Daten- typ | Parameter | Kommentar englisch deutsch | Vorbelegung |
|---------------|-----------|----------------------------------|-------------|
| REAL | OUTV | output variable Ausgangsgröße | 0.0 |

Globaler Datenbaustein

Die folgende Tabelle zeigt Ihnen Datentyp und Parameter von **DB_NBR** (Vorbelegung mit 5 Stützpunkten).

| Daten-typ | Parameter | Kommentar englisch deutsch | zulässiger Wertebereich | Vorbelegung |
|-----------|-------------------|--|-----------------------------|-------------|
| INT | DB_NBR.NBR_PTS | Größte Stützpunkt-Nummer highest coordinate | 1 – 255 | 4 |
| REAL | DB_NBR.PI[0].OUTV | output variable [0] Ausgangswert [0] Startpunkt 0 | technischer Wertebereich | 0.0 |
| REAL | DB_NBR.PI[0].INV | input variable [0] Eingangswert [0] Startpunkt 0 | technischer Wertebereich | 0.0 |
| REAL | DB_NBR.PI[1].OUTV | output variable [1] Ausgangswert [1] Stützpunkt 1 | technischer Wertebereich | 0.0 |
| REAL | DB_NBR.PI[1].INV | input variable [1] Eingangswert [1] Stützpunkt 1 | technischer Wertebereich | 0.0 |
| REAL | DB_NBR.PI[2].OUTV | output variable [2] Ausgangswert [2] Stützpunkt 2 | technischer Wertebereich | 0.0 |
| REAL | DB_NBR.PI[2].INV | input variable [2] Eingangswert [2] Stützpunkt 2 | technischer Wertebereich | 0.0 |
| REAL | DB_NBR.PI[3].OUTV | output variable [3] Ausgangswert [3] Stützpunkt 3 | technischer Wertebereich | 0.0 |
| REAL | DB_NBR.PI[3].INV | input variable [3] Eingangswert [3] Stützpunkt 3 | technischer Wertebereich | 0.0 |
| REAL | DB_NBR.PI[4].OUTV | output variable [4] Ausgangswert [4] Stützpunkt 4 | technischer Wertebereich | 0.0 |
| REAL | DB_NBR.PI[4].INV | input variable [4] Eingangswert [4] Stützpunkt 4 | technischer Wertebereich | 0.0 |

Neustart

Bei Neustart wird OUTV = 0.0 ausgegeben.

Ist DFOUT_ON=TRUE so wird DF_OUTV ausgegeben.

Normalbetrieb

Im Normalbetrieb wird durch Interpolation am aktuellen Stützpunktpaar der Ausgangswert berechnet.

DFOUT_ON und TRACK haben folgenden Einfluß auf OUTV:

| Betriebsarten | DFOUT_ON | TRACK | OUTV |
|-----------------------|----------|----------|---------------|
| Ausgang vorbelegen | TRUE | beliebig | DF_OUTV |
| Nachführen | FALSE | TRUE | OUTV = INV |
| Normalbetrieb | FALSE | FALSE | OUTV = f(INV) |

- **Ausgang vorbelegen**

Wenn DFOUT_ON = TRUE gesetzt ist, wird am Ausgang DF_OUTV ausgegeben, die Änderung von OUTV erfolgt sprungförmig. Bei Umschaltung nach DFOUT_ON = FALSE erfolgt die Änderung von OUTV ebenfalls sprungförmig.

- **Nachführen**

Bei TRACK=TRUE wird der Eingangswert direkt ausgegeben (OUTV=INV). Die Änderung erfolgt wie bei DFOUT_ON sprungförmig. TRACK hat niedrigere Priorität als DFOUT_ON.

Bausteininterne Begrenzungen

Die Werte der Eingangsparameter werden im Baustein nicht begrenzt; eine Prüfung der Parameter findet nicht statt.

2.1.17 NORM: physical norm (physikalische Normierung)

Anwendungsbereich

Bei Istwerten liegt der vom Geber gelieferte Wert oft in einem für den Anwender ungünstigen Bereich (z.B. 0 bis 10V entsprechen 0 bis 1200°C oder 0 bis 10V entsprechen 0 bis 3000U/min). Durch Anpassung von Soll- oder Istwert liegen beide Prozeßgrößen im gleichen Wertebereich.

Blockschaltbild

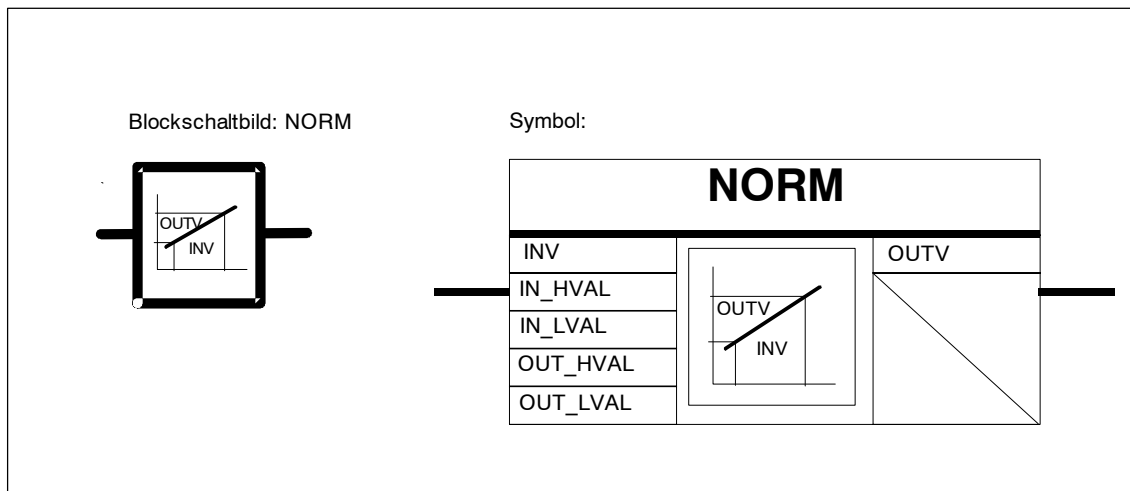


Bild 2-43 NORM, Blockschaltbild und Symbol

Funktionsbeschreibung

Der Baustein normiert die Eingangsgröße in eine Ausgangsgröße mit einem anderen Wertebereich. Die Normierungsgerade wird durch 2 Punkte definiert.

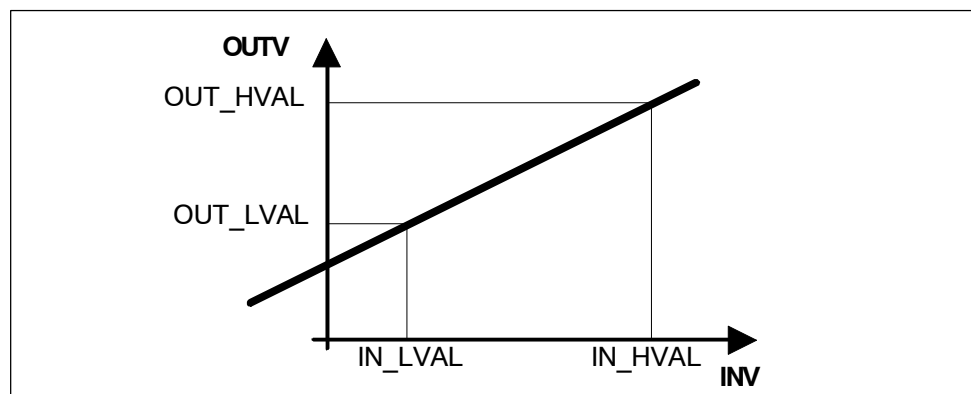


Bild 2-44 Normierungsgerade

Eingangsparameter

Die folgende Tabelle zeigt Ihnen Datentyp und Struktur der Eingangsparameter von NORM.

Tabelle 2-38 Eingangsparameter von NORM

| Daten- typ | Parameter | Kommentar englisch deutsch | zulässiger Wertebereich | Vorbelegung |
|---------------|-----------|---|-----------------------------|-------------|
| REAL | INV | input variable Eingangsgröße | technischer Wertebereich | 0.0 |
| REAL | IN_HVAL | physical input value high physikalischer Eingangswert unten | tech. Wert. > IN_LVAL | 100.0 |
| REAL | OUT_HVAL | physical output value high physikalischer Ausgangswert unten | tech. Wert. > OUT_LVAL | 100.0 |
| REAL | IN_LVAL | physical input value low physikalischer Eingangswert oben | tech. Wert. < IN_HVAL | 0.0 |
| REAL | OUT_LVAL | physical output value low physikalischer Ausgangswert oben | tech. Wert. < OUT_HVAL | 0.0 |

Ausgangsparameter

Die folgende Tabelle zeigt Ihnen Datentyp und Struktur der Ausgangsparameter von NORM.

Tabelle 2-39 Ausgangsparameter von NORM

| Daten- typ | Parameter | Kommentar englisch deutsch | Vorbelegung |
|---------------|-----------|----------------------------------|-------------|
| REAL | OUTV | output variable Ausgangsgröße | 0.0 |

Neustart

Der Baustein besitzt keine Neustarroutine.

Normalbetrieb

Die Eingangsgröße INV wird an der Normierungsgeraden in die Ausgangsgröße OUTV überführt. Die Normierungsgerade wird durch die Punkte IN_HVAL, IN_LVAL, OUT_HVAL und OUT_LVAL eindeutig bestimmt.

Bausteininterne Begrenzungen

Die Werte der Eingangsparameter werden im Baustein nicht begrenzt; eine Prüfung der Parameter findet nicht statt.

2.1.18 OVERRIDE: override controlling (Override Regelung)

Anwendungsbereich

Der Baustein wird benötigt zur Realisierung einer Ablöse- (override) Regelung.

Blockschaltbild

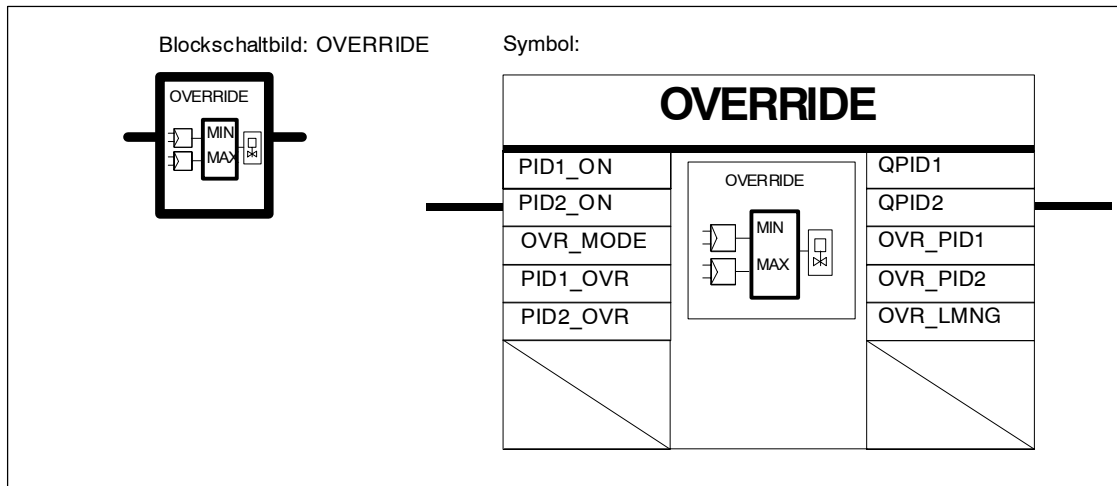


Bild 2-45 OVERRIDE, Blockschaltbild und Symbol

Funktionsbeschreibung

Zwei PID-Regler werden auf ein Stellungsglied geschaltet. Das Maximum (OVR_MODE=FALSE) bzw. Minimum (OVR_MODE=TRUE) der beiden PID-Regler wird auf das Stellungsglied durchgeschaltet. Hierzu werden 2 PID-Bausteine über den Baustein OVERRIDE mit einem der Stellwertverarbeitungsbausteine LMNGEN_C bzw. LMNGEN_S verbunden.

Wahrheitstabelle der Schalter PID1_ON, PID2_ON und OVR_MOD:

| PID1_ON | PID2_ON | OVR_MOD | Funktion |
|---------|---------|---------|--|
| 1 | 0 | | nur PID1-Algorithmus ist wirksam |
| 0 | 1 | | nur PID2-Algorithmus ist wirksam |
| 0 | 0 | 0 | das Maximum von PID1 und PID2 wird durchgeschaltet |
| 1 | 1 | 0 | das Maximum von PID1 und PID2 wird durchgeschaltet |
| 0 | 0 | 1 | das Minimum von PID1 und PID2 wird durchgeschaltet |
| 1 | 1 | 1 | das Minimum von PID1 und PID2 wird durchgeschaltet |

Die Funktion wird unabhängig von diesem Signalzustand bearbeitet.

Welcher PID-Algorithmus aktiv ist, wird an den Ausgängen QPID1 und QPID2 angezeigt.

Eingangsparameter

Die folgende Tabelle zeigt Ihnen Datentyp und Struktur der Eingangsparameter von OVERRIDE.

Tabelle 2-40 Eingangsparameter von OVERRIDE

| Daten- typ | Parameter | Kommentar englisch deutsch | zulässiger Wertebereich | Vorbe- legung |
|---------------|-----------|--|----------------------------|------------------|
| BOOL | PID1_ON | pid controller 1 on PID-Regler 1 ein | | FALSE |
| BOOL | PID2_ON | pid controller 2 on PID-Regler 2 ein | | FALSE |
| BOOL | OVR_MODE | override mode FALSE=maximum, TRUE=minimum Ablösemodus FALSE=Maximum, TRUE=Minimum | | FALSE |
| STRUC | PID1_OVR | PID-LMNGEN interface PID-LMNGEN Schnittstelle | | |
| STRUC | PID2_OVR | PID-LMNGEN interface PID-LMNGEN Schnittstelle | | |

Ausgangsparameter

Die folgende Tabelle zeigt Ihnen Struktur der Ausgangsparameter von OVERRIDE.

Tabelle 2-41 Ausgangsparameter von OVERRIDE

| Daten- typ | Parameter | Kommentar englisch deutsch | Vorbe- legung |
|---------------|-----------|--|------------------|
| BOOL | QPID1 | pid controller 1 activ PID-Regler 1 aktiv | |
| BOOL | QPID2 | pid controller 2 activ PID-Regler 2 aktiv | |
| STRUC | OVR_LMNG | PID-LMNGEN interface PID-LMNGEN Schnittstelle | |

Neustart

Der Baustein hat keine Neustarroutine.

Normalbetrieb

Der Baustein enthält keine vom Normalbetrieb abweichende Betriebsarten.

Bausteininterne Begrenzungen

Die Werte der Eingangsparameter werden im Baustein nicht begrenzt; eine Prüfung der Parameter findet nicht statt.

Beispiel

Bild 2-46 zeigt Ihnen ein Verschaltungsbeispiel einer Override-Regelung:

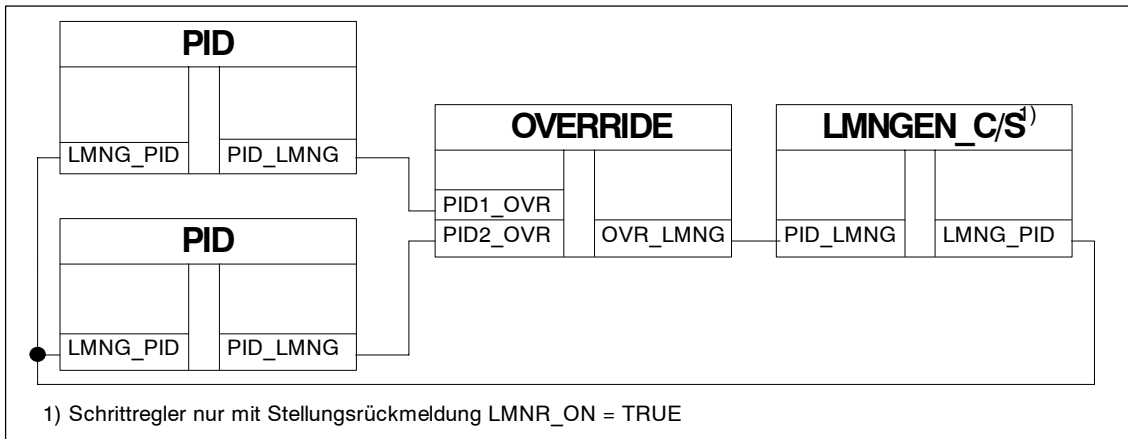


Bild 2-46 Verschaltungsbeispiel einer Override-Regelung

2.1.19 PARA_CTL: parameter controlling (Parametersteuerung)

Anwendungsbereich

Der Baustein wird bei Reglerstrukturen mit Parameterumschaltungen verwendet, wenn für unterschiedliche Arbeitsbereiche optimale Reglerparameter benötigt werden.

Blockschaltbild

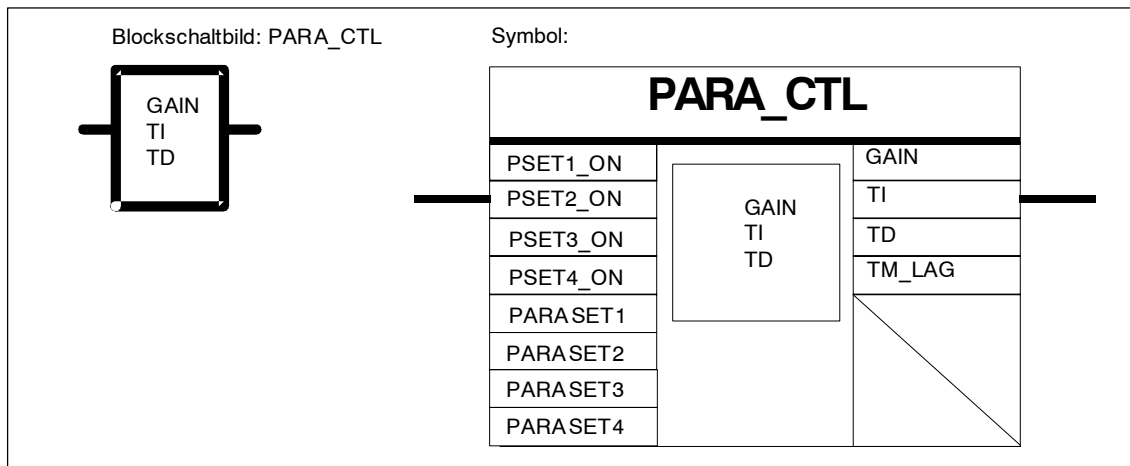


Bild 2-47 PARA_CTL, Blockschaltbild und Symbol

Funktionsbeschreibung

Mehrere Reglerparametersätze (GAIN, TI, TD und TM_LAG) können in einen PID-Regler übertragen werden.

Über die Schalter PSET1_ON...PSET4_ON kann einer der 4 Parametersätze auf die Ausgänge GAIN, TI, TD und TM_LAG durchgeschaltet werden.

wenn PSET1_ON = 1 gilt

$$\begin{aligned} \text{GAIN} &= \text{PARASET1.GAIN} \\ \text{TI} &= \text{PARASET1.TR} \\ \text{TD} &= \text{PARASET1.TD} \\ \text{TM_LAG} &= \text{PARASET1.TM_LAG} \end{aligned}$$

wenn PSET2_ON = 1 gilt

$$\begin{aligned} \text{GAIN} &= \text{PARASET2.GAIN} \\ \text{TI} &= \text{PARASET2.TR} \\ \text{TD} &= \text{PARASET2.TD} \\ \text{TM_LAG} &= \text{PARASET2.TM_LAG} \end{aligned}$$

wenn PSET3_ON = 1 gilt

$$\begin{aligned} \text{GAIN} &= \text{PARASET3.GAIN} \\ \text{TI} &= \text{PARASET3.TR} \\ \text{TD} &= \text{PARASET3.TD} \\ \text{TM_LAG} &= \text{PARASET3.TM_LAG} \end{aligned}$$

wenn PSET4_ON = 1 gilt GAIN = PARASET4.GAIN
 TI = PARASET4.TR
 TD = PARASET4.TD
 TM_LAG = PARASET4.TM_LAG

Sind 2 oder mehrere Schalter gesetzt, so hat immer der Schalter mit der kleinsten Parameternummer die höchste Priorität. Ist kein Schalter gesetzt so wird der 1. Parametersatz übertragen.

Eingangsparameter

Die folgende Tabelle zeigt Ihnen Datentyp und Struktur der Eingangsparameter von PARA_CTL.

Tabelle 2-42 Eingangsparameter von PARA_CTL

| Datentyp | Parameter | Kommentar | englisch deutsch | zulässiger Wertebereich | Vorbelegung |
|----------|-----------|--|---------------------|----------------------------|-------------|
| BOOL | PSET1_ON | set number 1 Parametersatznummer 1 übertragen | | | FALSE |
| BOOL | PSET2_ON | set number 2 Parametersatznummer 2 übertragen | | | FALSE |
| BOOL | PSET3_ON | set number 3 Parametersatznummer 3 übertragen | | | FALSE |
| BOOL | PSET4_ON | set number 4 Parametersatznummer 4 übertragen | | | FALSE |
| STRUCT | PARASET1 | parameter set 1 Reglerparametersatz 1 | | | – |
| STRUCT | PARASET2 | parameter set 2 Reglerparametersatz 2 | | | – |
| STRUCT | PARASET3 | parameter set 3 Reglerparametersatz 3 | | | – |
| STRUCT | PARASET4 | parameter set 4 Reglerparametersatz 4 | | | – |

Ausgangsparameter

Die folgende Tabelle zeigt Ihnen Datentyp und Struktur der Ausgangsparameter von PARA_CTL.

Tabelle 2-43 Ausgangsparameter von PARA_CTL

| Datentyp | Parameter | Kommentar englisch deutsch | Vorbelegung |
|----------|-----------|--|-------------|
| REAL | GAIN | proportional gain Proportionalbeiwert | 1.0 |
| TIME | TI | reset time Integrationszeit | 10s |
| TIME | TD | derivative time Differenzierzeit | 5s |
| TIME | TM_LAG | time lag Verzögerungszeit | 2s |

Reglerparametersätze

Die folgende Tabelle zeigt Ihnen Datentyp und Parameter.

Tabelle 2-44

| Daten- typ | Parameter | Kommentar englisch deutsch | zulässiger Wertebereich | Vorbelegung |
|---------------|-----------------|--|----------------------------|-------------|
| REAL | PARASET1.GAIN | proportional gain 1 Proportionalbeiwert 1 | | 1.0 |
| TIME | PARASET1.TI | reset time 1 Integrationszeit 1 | | T#10s |
| TIME | PARASET1.TD | derivative time 1 Differenzierzeit 1 | | T#5s |
| TIME | PARASET1.TM_LAG | time lag 1 Verzögerungszeit 1 | | T#2s |
| ... | ... | | ... | ... |
| REAL | PARASET4.GAIN | proportional gain 4 Proportionalbeiwert 4 | | 1.0 |
| TIME | PARASET4.TI | reset time 4 Integrationszeit 4 | | T#10s |
| TIME | PARASET4.TD | derivative time 4 Differenzierzeit 4 | | T#5s |
| TIME | PARASET4.TM_LAG | time lag 4 Verzögerungszeit 4 | | T#2s |

Neustart

Der Baustein hat keine Neustarroutine.

Normalbetrieb

Der Baustein enthält keine vom Normalbetrieb abweichende Betriebsarten.

Bausteininterne Begrenzungen

Die Werte der Eingangsparameter werden im Baustein nicht begrenzt; eine Prüfung der Parameter findet nicht statt.

Beispiel

Wird ein stoßfreies Umschalten zwischen den Parametersätzen gewünscht, muß beim Parameter GAIN ein Hochlaufgeber (ROC_LIM) zwischen dem PI-Baustein und dem Baustein PARA_CTL geschaltet werden (siehe Bild 2-48).

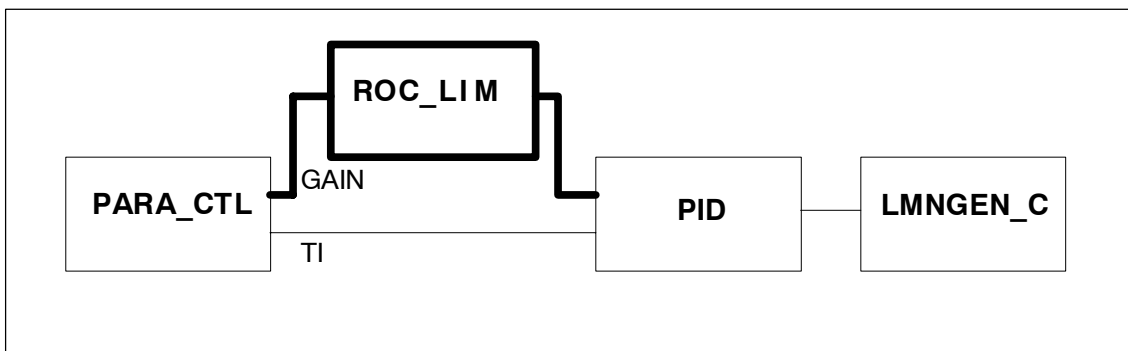


Bild 2-48 Stoßfreies Umschalten zwischen den Parametersätzen

2.1.20 PID: PID–algorithm (PID–Algorithmus)

Anwendungsbereich

Der Baustein beinhaltet den PID–Algorithmus zum Aufbau der folgenden Reglertypen:

- Kontinuierliche PID–Regler:
PID + LMNGEN_C
- PID–Impulsregler für proportionale Stellglieder:
PID + LMNGEN_C + PULSEGEN
- PID–Schrittregler für integrierende Stellglieder:
PID + LMNGEN_S

Blockschaltbild

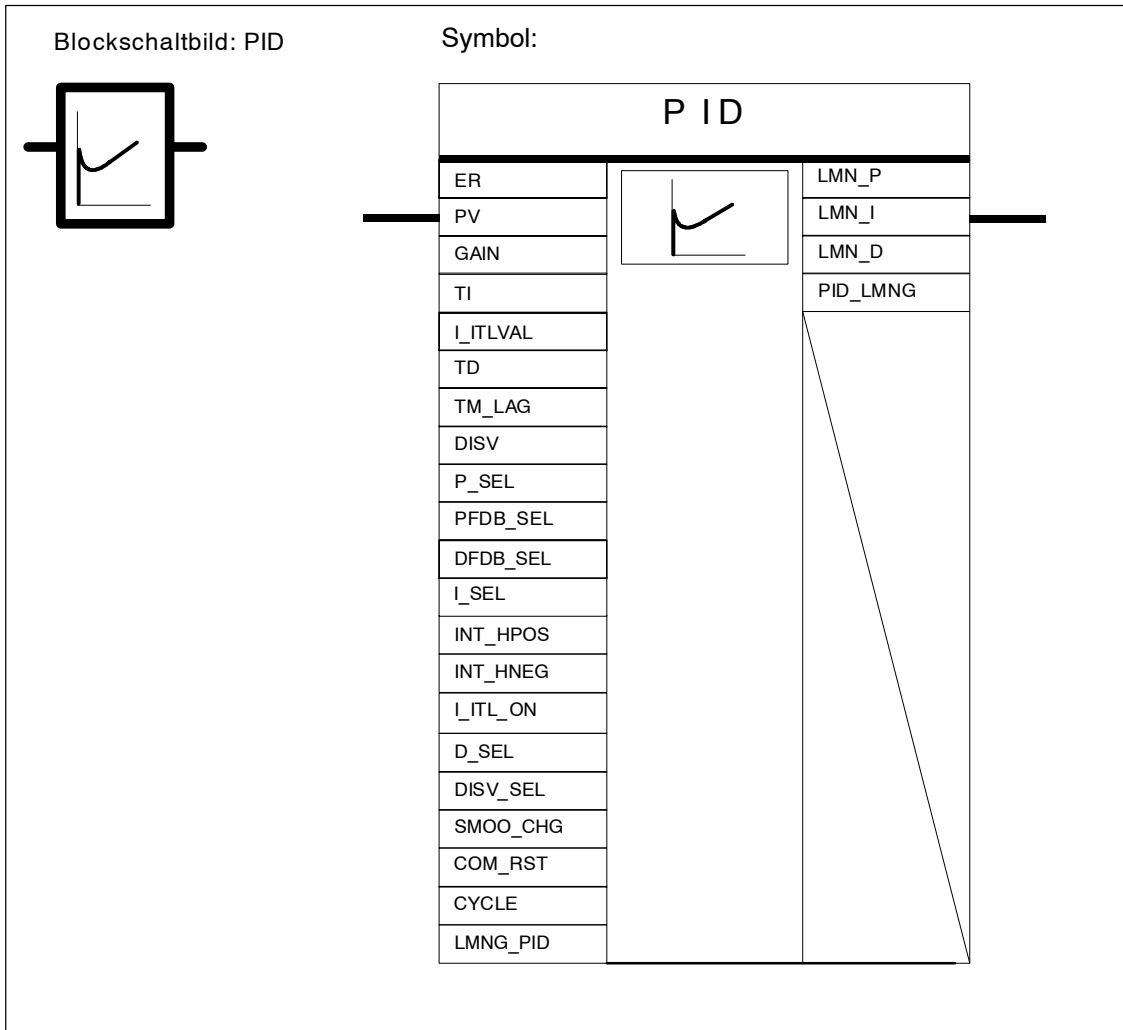


Bild 2-49 PID, Blockschaltbild und Symbol

Funktionsbeschreibung

Der Baustein realisiert den PID-Algorithmus. Er ist in reiner Parallelstruktur ausgeführt und wirkt ausschließlich als Stellungsalgorithmus. Die Proportional-, Integral- und Differentialanteile können einzeln zu- bzw. abgeschaltet werden. Damit lassen sich P-, PI-, PD- und PID-Regler parametrieren.

Die Berechnung der P- und D-Anteile können in die Rückführung gelegt werden. Durch das Verlegen des P- und des D-Anteils in den Rückführungszweig wird das Führungsverhalten "stoßfrei" bei gleicher Schnelligkeit der Ausregelung von Störgrößen. Auf die übliche Anwendung eines Sollwertintegrators zur Vermeidung von Sollwertsprüngen kann meist verzichtet werden.

Während der PID-Baustein in einer Weckalarmebene platziert wird, deren Zykluszeit der dominierenden Streckenzeitkonstante angepaßt ist, können die staggliedverarbeitenden Bausteine (LMNGEN_C bzw. LMNGEN_S) in einer schnelleren Weckalarmebene untergebracht werden. Der Aufruf des Impulsformers PULSE-GEN ist in Kap. 2.1.21 beschrieben.

Eingangsparameter

Die folgende Tabelle zeigt Ihnen Datentyp und Struktur der Eingangsparameter von PID.

Tabelle 2-45 Eingangsparameter von PID

| Daten- typ | Parameter | Kommentar englisch deutsch | zulässiger Wertebereich | Vorbelegung |
|---------------|-----------|--|-----------------------------|-------------|
| REAL | ER | error signal Regeldifferenz | technischer Wertebereich | 0.0 |
| REAL | PV | process variable Regelgröße (P- bzw. D-Anteil in Rück- führung) | technischer Wertebereich | 0.0 |
| REAL | GAIN | proportional gain Proportionalbeiwert | | 1.0 |
| TIME | TI | reset time Integrationszeit | | T#20s |
| REAL | I_ITLVAL | initialization value of the integral action Initialisierungswert für I-Anteil | technischer Wertebereich | 0.0 |
| TIME | TD | derivative time Differenzierzeit | | T#10s |
| TIME | TM_LAG | time lag of the derivative action Verzögerungszeit des D-Anteils | | T#2s |
| REAL | DISV | disturbance variable Störgröße | technischer Wertebereich | 0.0 |
| BOOL | P_SEL | proportional action on P-Anteil einschalten | | TRUE |
| BOOL | PFDB_SEL | proportional action in feedback path on P-Anteil in Rückführung schalten | | FALSE |

Tabelle 2-45 Eingangsparmeter von PID, Fortsetzung

| Daten- typ | Parameter | Kommentar englisch deutsch | zulässiger Wertebereich | Vorbelegung |
|---------------|-----------|---|----------------------------|-------------|
| BOOL | DFDB_SEL | derivative action in feedback path on D-Anteil in Rückführung schalten | | FALSE |
| BOOL | I_SEL | integral action on I-Anteil einschalten | | TRUE |
| BOOL | INT_HPOS | integral action hold in positive direction I-Anteil in positiver Richtung einfrieren | | FALSE |
| BOOL | INT_HNEG | integral action hold in negative direction I-Anteil in negativer Richtung einfrieren | | FALSE |
| BOOL | I_ITL_ON | initialization of the integral action I-Anteil setzen | | FALSE |
| BOOL | D_SEL | derivative action on D-Anteil einschalten | | FALSE |
| BOOL | DISV_SEL | disturbance variable on Störgröße aufschalten | | TRUE |
| BOOL | SMOO_CHG | smooth changeover from the manual mode to the automatic mode sanfter Übergang von Hand-Betrieb auf Automatik-Betrieb | | TRUE |
| BOOL | COM_RST | complete restart Neustart | | FALSE |
| TIME | CYCLE | sample time Abtastzeit | $\geq 1\text{ms}$ | T#1s |
| STRUC | LMNG_PID | PID-LMNGEN interface PID-LMNGEN Schnittstelle | | |

Ausgangsparameter

Die folgende Tabelle zeigt Ihnen Datentyp und Struktur der Ausgangsparameter von PID.

Tabelle 2-46 Ausgangsparameter von PID

| Daten- typ | Parameter | Kommentar englisch deutsch | Vorbelegung |
|---------------|-----------|---------------------------------------|-------------|
| REAL | LMN_P | proportionality component P-Anteil | 0.0 |
| REAL | LMN_I | integral component I-Anteil | 0.0 |

Tabelle 2-46 Ausgangsparameter von PID, Fortsetzung

| Daten- typ | Parameter | Kommentar englisch deutsch | Vorbelegung |
|-----------------------|------------------|---|--------------------|
| REAL | LMN_D | derivative component D-Anteil | 0.0 |
| STRUC | PID_LMNG | PID-LMNGEN interface PID-LMNGEN Schnittstelle | |

Neustart

Bei Neustart werden alle Aus- und Durchgangparameter mit Null vorbelegt.

Normalbetrieb

Der Baustein enthält neben dem Normalbetrieb folgende Betriebsarten:

| Betriebsarten | P_SEL | I_SEL | D_SEL |
|---------------|-------|-----------------|-------|
| P-Regelung | TRUE | TRUE bzw. FALSE | FALSE |
| PI-Regelung | TRUE | TRUE | FALSE |
| PD-Regelung | TRUE | FALSE | TRUE |
| PID-Regelung | TRUE | TRUE | TRUE |

P-Regelung

Hierbei ist der D-Anteil abgeschaltet. Am I-Anteil kann über den Initialisierungswert I_ITLVAL ein Arbeitspunkt vorgegeben werden (I_SEL = TRUE und I_ITL_ON = TRUE). Soll der Arbeitspunkt immer auf konstant Null gehalten werden, so kann der I-Anteil abgeschaltet werden.

Übertragungsfunktion

Die Regeldifferenz ER wird mit GAIN multipliziert.

Sprungantwort

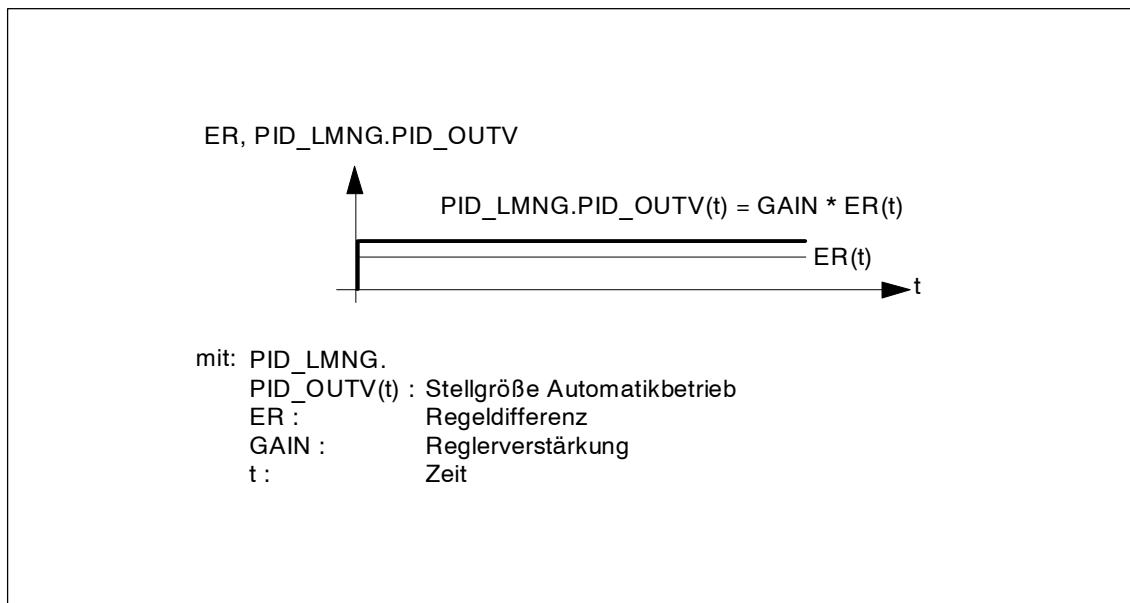


Bild 2-50 Sprungantwort bei P-Regelung

PI-Regelung

Hierbei ist der D-Anteil abgeschaltet.

Übertragungsfunktion

Die Regeldifferenz **ER** wird multipliziert und integriert. Die Übertragungsfunktion im Laplace-Bereich lautet:

$$\text{PID_LMNG.PID_OUTV}(s) / \text{ER}(s) = \text{GAIN} * (1 + 1 / (\text{TI} * s))$$

Sprungantwort

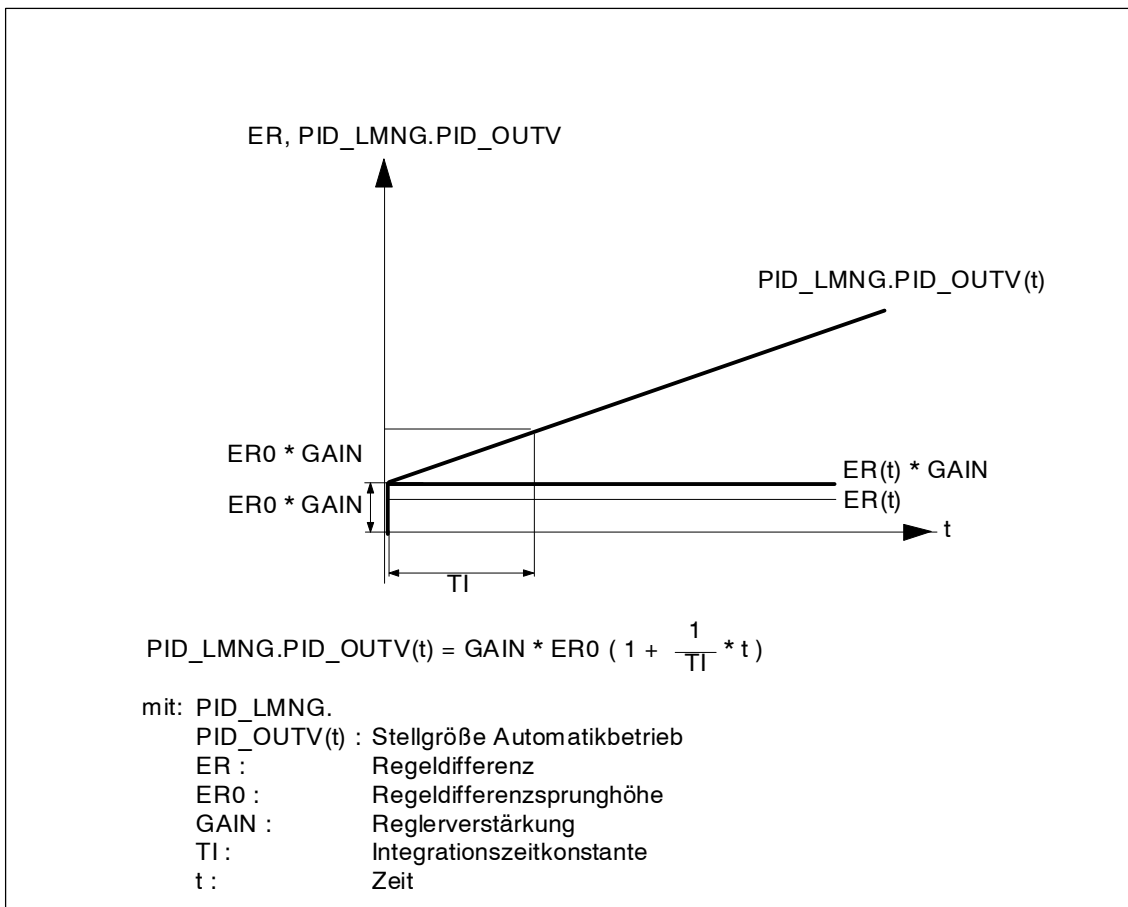


Bild 2-51 Sprungantwort bei PI-Regelung

PD–Regelung

Hierbei ist der I–Anteil abgeschaltet.

Übertragungsfunktion

Die Regeldifferenz **ER** wird multipliziert und differenziert. Die Übertragungsfunktion im Laplace–Bereich lautet:

$$\text{PID_LMNG.PID_OUTV}(s) / \text{ER}(s) = \text{GAIN} * (1 + \text{TD} * s / (1 + \text{TM_LAG} * s))$$

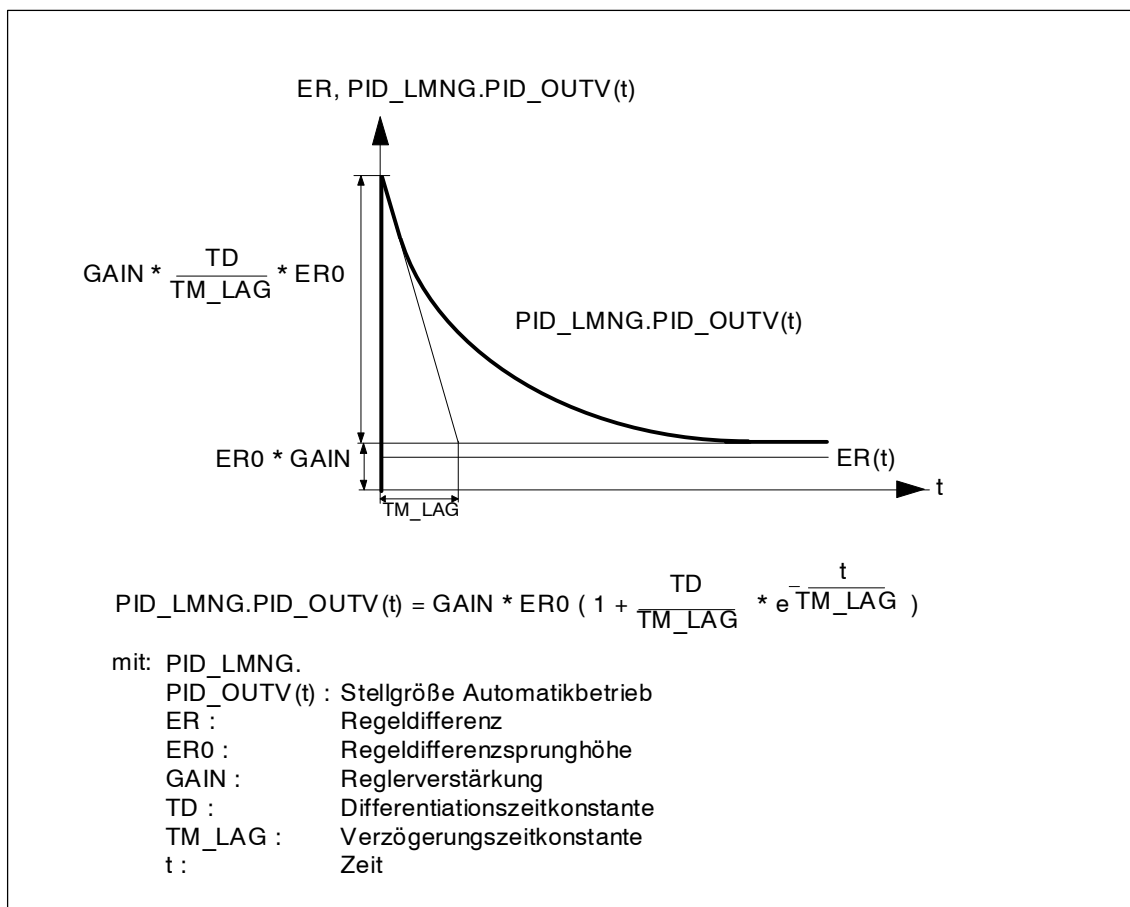
Sprungantwort

Bild 2-52 Sprungantwort bei PD–Regelung

PID-Regelung

P-, I- und D-Anteil sind eingeschaltet.

Übertragungsfunktion

Die Regeldifferenz ER wird multipliziert, integriert und differenziert. Die Übertragungsfunktion im Laplace-Bereich lautet:

$$\text{PID_LMNG.PID_OUTV}(s) / \text{ER}(s) = \text{GAIN} * (1 + 1 / (\text{TI} * s) + \text{TD} * s / (1 + \text{TM_LAG} * s))$$

Sprungantwort

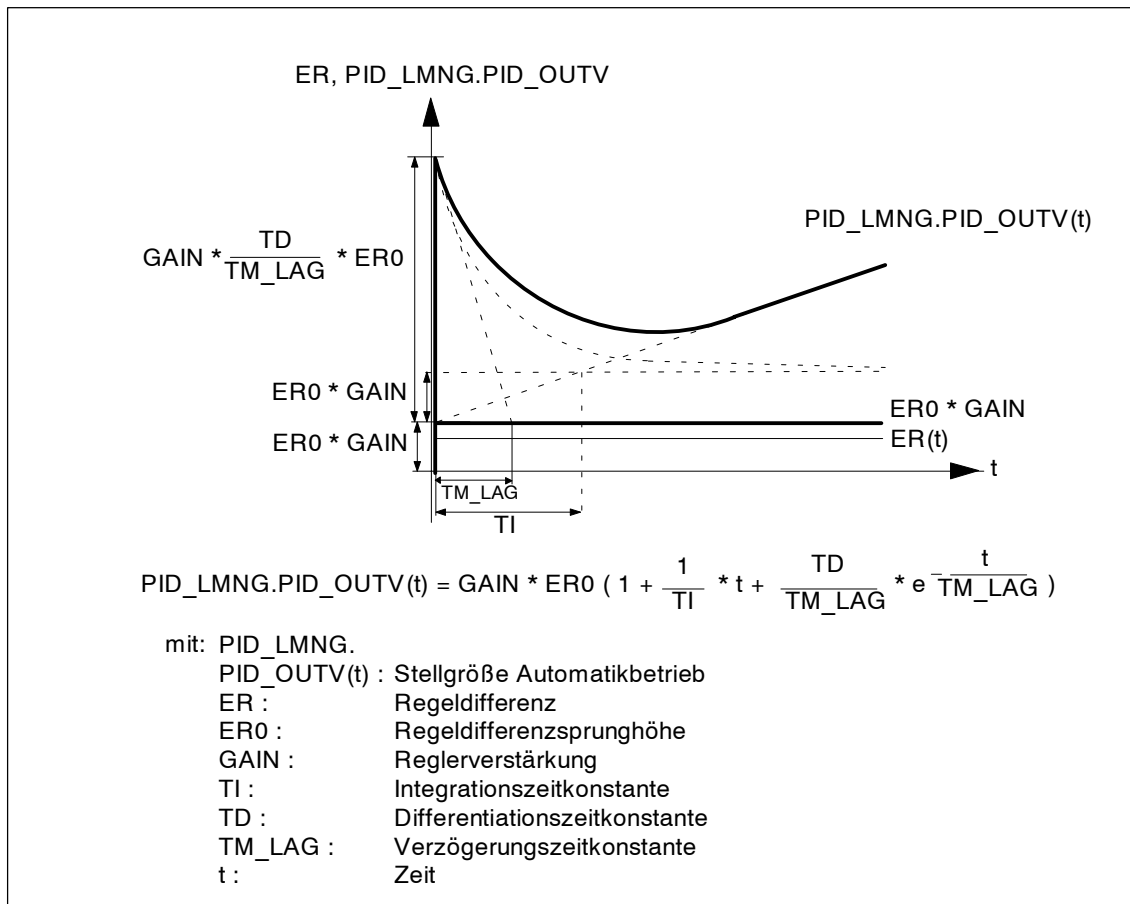


Bild 2-53 Sprungantwort bei PID-Regelung

Blockschaltbild

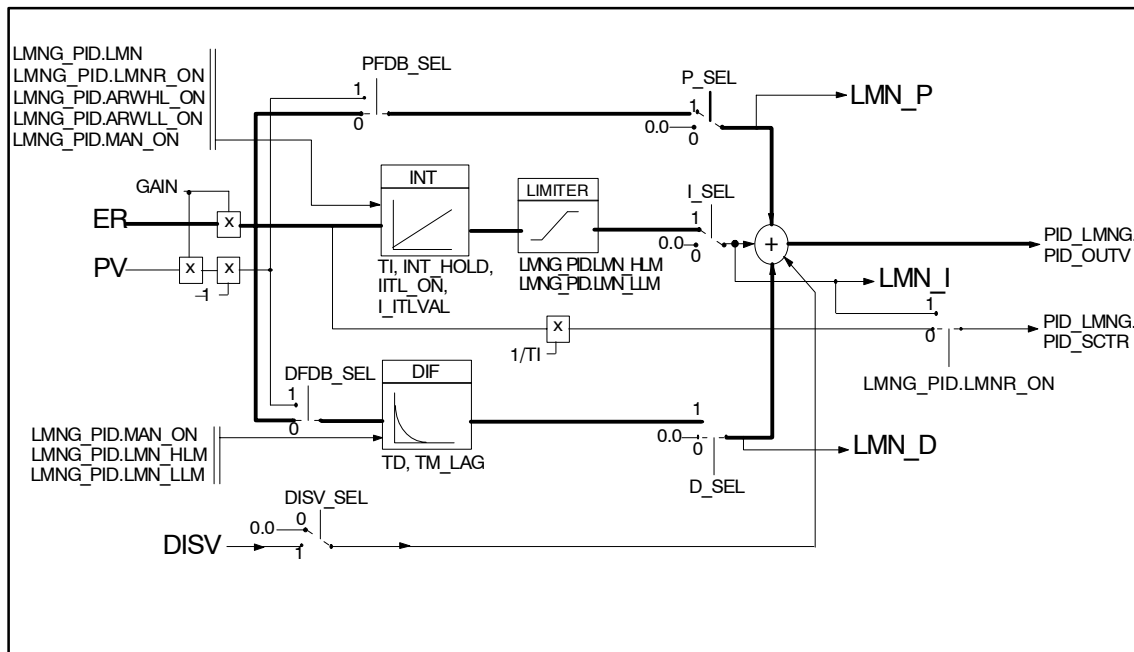


Bild 2-54 Blockschaltbild für PID-Algorithmus

P-Anteil

Der P-Anteil kann wahlweise mit dem Schalter P_SEL zu bzw. abgeschaltet werden. Über den Schalter PFDB_SEL kann er in die Rückführung gelegt werden. Das heißt, als Eingang für den P-Anteil wird die Regelgröße PV verwendet. Der P-Anteil stellt das Produkt aus Regeldifferenz ER (bei P-Anteil in Rückführung ist es die Regelgröße PV) und dem Proportionalbeiwert GAIN dar.

I-Anteil

Der I-Anteil kann wahlweise mit dem Schalter I_SEL ein- bzw. ausgeschaltet werden. Bei ausgeschaltetem Zustand wird der I-Anteil und der interne Speicher des Integrators auf Null gesetzt. Mit INT_HOLD kann der I-Anteil eingefroren werden. Die Integrationszeit wird durch die Integrationszeitkonstante TI bestimmt. Der Anwender hat die Möglichkeit den Integrator mit einem eigenen Wert zu belegen. Über den Schalter I_ITL_ON wird der Wert am Eingang I_ITLVAL in den Integrator übertragen. Bei Stellwertbegrenzungen bleibt der I-Anteil auf dem Altwert stehen (Anti reset wind-up).

D–Anteil

Der D–Anteil kann über den Schalter D_SEL zu– bzw. abgeschaltet werden. Mit dem Schalter DFDB_SEL kann er in die Rückführung gelegt werden. Die Eingangsgröße des D–Anteils ist nun die Regelgröße PV. Das Zeitverhalten wird durch die Vorhaltzeit TD bestimmt. Im Differenzierer ist eine Verzögerung 1. Ordnung integriert. Über TM_LAG wird die Verzögerungszeit eingegeben.

Besonders bei schnellen Strecken und eingeschaltetem D–Anteil kann es zu einer unzulässigen Störwelligkeit kommen. In diesem Fall ist eine Verbesserung des Regelverhaltens mit der im D–Anteil integrierten Dämpfung möglich. Meist genügt bereits ein kleines TM_LAG, um den gewünschten Erfolg zu erzielen.

Störgrößenaufschaltung

Additiv zum Stellwert kann eine Störgröße DISV aufgeschaltet werden. Über den Schalter DISV_SEL kann sie wahlweise zu– bzw. abgeschaltet werden.

Bausteininterne Begrenzungen

Die Integrationszeit wird auf die halbe Abtastzeit nach unten begrenzt.

Die Differenzierzeit wird auf die Abtastzeit nach unten begrenzt.

Die Verzögerungszeit wird auf die halbe Abtastzeit nach unten begrenzt.

$$\begin{array}{ll} T_{\text{intern}} = \text{CYCLE}/2 & \text{für } T_I < \text{CYCLE}/2 \\ T_{D\text{intern}} = \text{CYCLE} & \text{für } T_D < \text{CYCLE} \\ T_{M_LAG\text{intern}} = \text{CYCLE}/2 & \text{für } T_{M_LAG} < \text{CYCLE}/2 \end{array}$$

Die anderen Werte der Eingangsparameter werden im Baustein nicht begrenzt; eine Prüfung der Parameter findet nicht statt.

2.1.21 PULSEGEN: pulse generator (Impulsformer)

Anwendungsbereich

Der Baustein dient zum Aufbau eines PID-Reglers mit Impulsausgang für proportionale Stellglieder. Damit lassen sich Dreipunktregler und Zweipunktregler mit Pulsbreitenmodulation realisieren.

Blockschaltbild

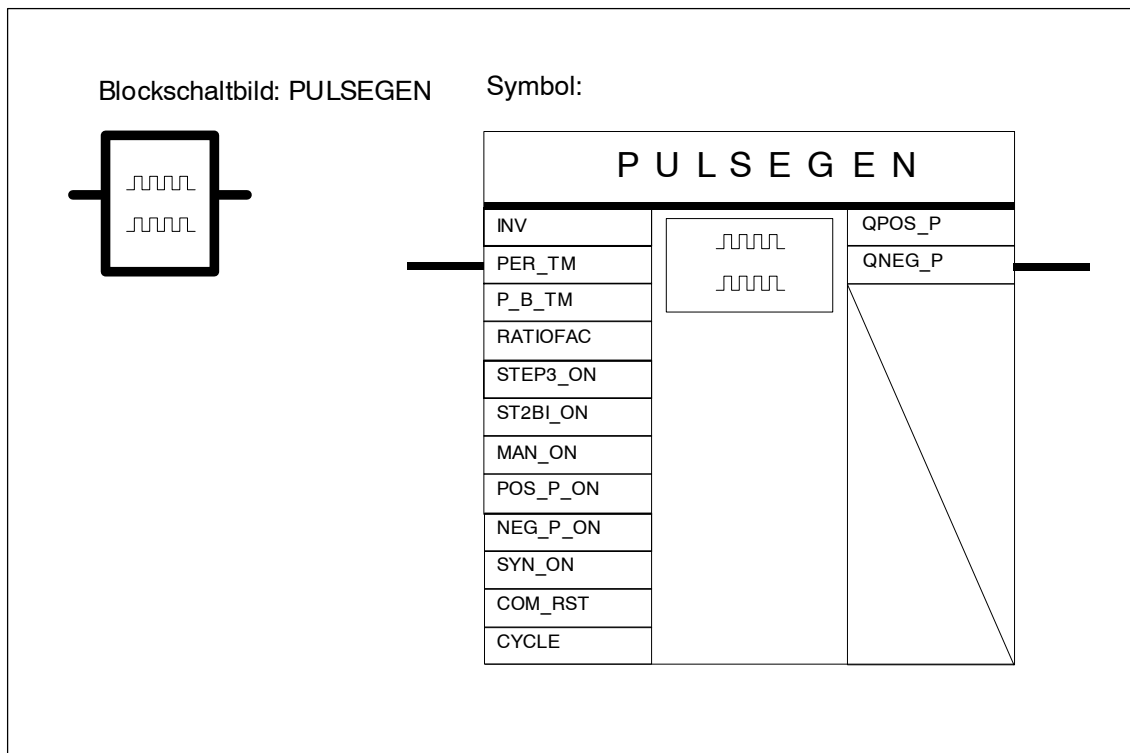


Bild 2-55 PULSEGEN, Blockschaltbild und Symbol

Funktionsbeschreibung

Der Baustein transformiert die Eingangsgröße INV mittels der Pulsbreitenmodulation in eine Impulsfolge mit konstanter Periodendauer PER_TM. Die Periodendauer entspricht der Zykluszeit mit der die Eingangsgröße aktualisiert wird. Die Dauer eines Impulses pro Periodendauer ist proportional der Eingangsgröße. Eine Eingangsgröße von 30% bedeutet also: Ein positiver Impuls der Dauer $0,3 \cdot$ Periodendauer, kein Impuls für $0,7 \cdot$ Periodendauer. Die Impulsdauer wird zu Beginn jeder Periode neu berechnet.

Bild 2-56 zeigt Ihnen die Pulsbreitenmodulation.

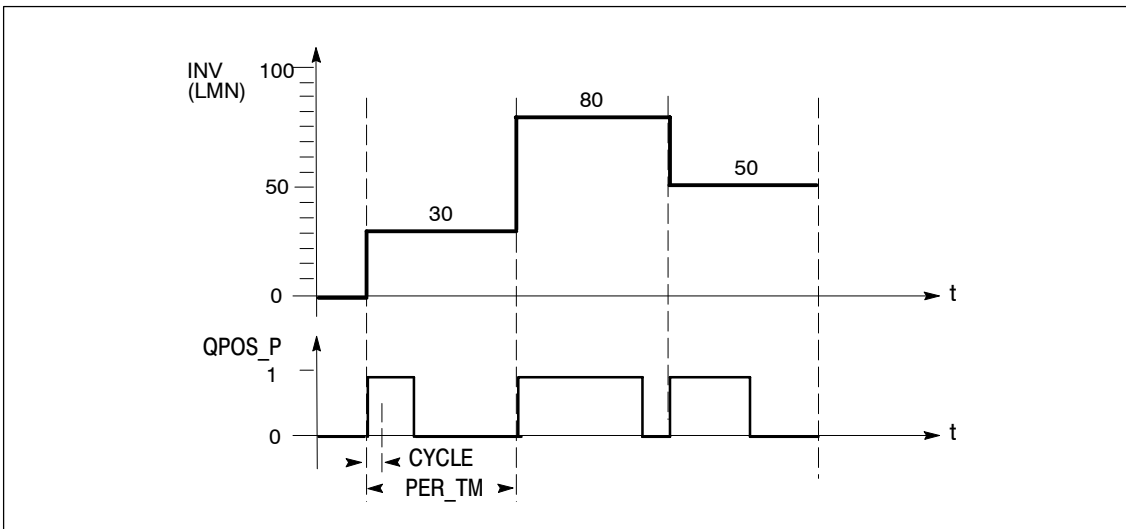


Bild 2-56 Pulsbreitenmodulation

Zur Entlastung des Stellgliedes kann eine minimale Impuls- bzw. Pausendauer parametrierbar sein.

In der Betriebsart Dreipunktregelung können über einen Verhältnissfaktor unterschiedliche Zeiten beim Heizen und Kühlen kompensiert werden.

Der Baustein wird meistens in Verbindung mit dem kontinuierlichen Regler angewandt.

- Zwei- oder Dreipunkt PID-Regler: PID + LMNGEN_C + PULSEGEN

Bild 2-57 zeigt Ihnen die Verschaltung des PID-Impulsreglers

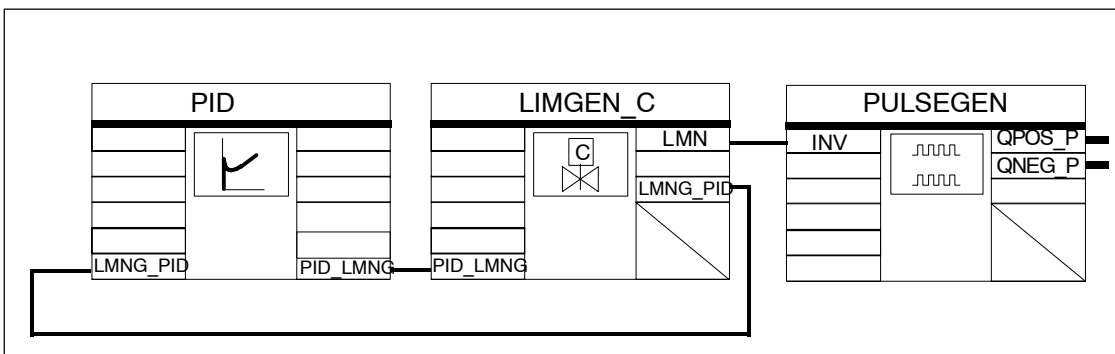


Bild 2-57 Verschaltung des PID-Impulsreglers

Eingangsparameter

Die folgende Tabelle zeigt Ihnen Datentyp und Struktur der Eingangsparameter von PULSEGEN.

Tabelle 2-47 Eingangsparameter von PULSEGEN

| Datentyp | Parameter | Kommentar englisch deutsch | zulässiger Wertebereich | Vorbelegung |
|----------|-----------|---|------------------------------|-------------|
| REAL | INV | input variable Eingangsvariable | -100,0..100,0 (%) | 0.0 |
| TIME | PER_TM | period time Periodendauer | PER_TM >= 20*CYCLE | T#1s |
| TIME | P_B_TM | minimum pulse/break time Mindestimpuls- bzw. Mindestpausendauer | P_B_TM >= CYCLE | T#50ms |
| REAL | RATIOFAC | ratio factor Verhältnissfaktor | 0,1..10,0 (dimensionslos) | 1.0 |
| BOOL | STEP3_ON | three step control on Dreipunktregelung einschalten | | TRUE |
| BOOL | ST2BI_ON | two step control for bipolar manipulated value range on Zweipunktregelung für bipolaren Stellwertbereich einschalten | | FALSE |
| BOOL | MAN_ON | manual mode on Handbetrieb einschalten | | FALSE |
| BOOL | POS_P_ON | positive pulse on positiver Impuls ein | | FALSE |
| BOOL | NEG_P_ON | negative pulse on negativer Impuls ein | | FALSE |
| BOOL | SYN_ON | synchronisation on Synchronisation einschalten | | TRUE |
| BOOL | COM_RST | complete restart Neustart | | FALSE |
| TIME | CYCLE | sample time Abtastzeit | CYCLE >= 1ms | T#10ms |

Ausgangsparameter

Die folgende Tabelle zeigt Ihnen Datentyp und Struktur der Ausgangsparameter von PULSEGEN.

Tabelle 2-48 Ausgangsparameter von PULSEGEN

| Datentyp | Parameter | Kommentar englisch deutsch | Vorbelegung |
|----------|-----------|--|-------------|
| BOOL | QPOS_P | output positive pulse Ausgangssignal positiver Impuls | FALSE |
| BOOL | QNEG_P | output negative pulse Ausgangssignal negativer Impuls | FALSE |

Neustart

Bei Neustart werden alle Signalausgänge auf Null gesetzt.

Genauigkeit der Pulsformung

Die programmtechnische Realisierung dieser Funktion in der CPU erfordert innerhalb einer Impulsperiode des betreffenden Reglerausgangs n mal zu zyklischen Zeitpunkten eine Entscheidung über den aktuellen Status des Binärsignals. Je größer n , desto genauer ist die Pulsbreitenmodulation.

Während der kontinuierliche Regler PID+LMGEN_C in einer langsamen Weckalarmebene plaziert wird, deren Zykluszeit der dominierenden Streckenzeitkonstante angepaßt ist, muß der PULSEGEN Baustein in einer schnelleren Weckalarmebene untergebracht werden. Je schneller die Weckalarmebene ist, desto genauer kann der Stellwert ausgegeben werden. Bei einer 100 mal schnelleren Weckalarmebene erreicht man eine Auflösung von 1% des Stellwertbereiches.

Automatische Synchronisation

Es besteht die Möglichkeit, die Impulsausgabe mit dem Regler-FB automatisch zu synchronisieren. Damit ist gewährleistet, dass ein geänderter Wert der Stellgröße LMN(t) schnellstmöglich als proportional geänderte Pulsdauer des Binärsignals ausgegeben wird.

Der Impulsformer wertet jeweils im Zeitabstand der Periodendauer PER_TM die Eingangsgröße INV aus. Da aber INV in einer langsameren Weckalarmebene berechnet wird, sollte der Impulsformer möglichst schnell nach der Aktualisierung von INV mit der Umwandlung des diskreten Wertes in ein Impulssignal beginnen. Dazu kann der Baustein den Start der Periode nach folgendem Verfahren selbst synchronisieren. Hat sich INV geändert und befindet sich der Bausteinaufruf nicht im ersten oder in den letzten zwei Aufrufzyklen einer Periode, so wird eine Synchronisation vorgenommen. Die Impulsdauer wird neu berechnet und beim nächsten Zyklus mit einer neuen Periode wird mit der Ausgabe begonnen.

Die Periodendauer PER_TM muß der Abtastzeit CYCLE des Reglers entsprechen.

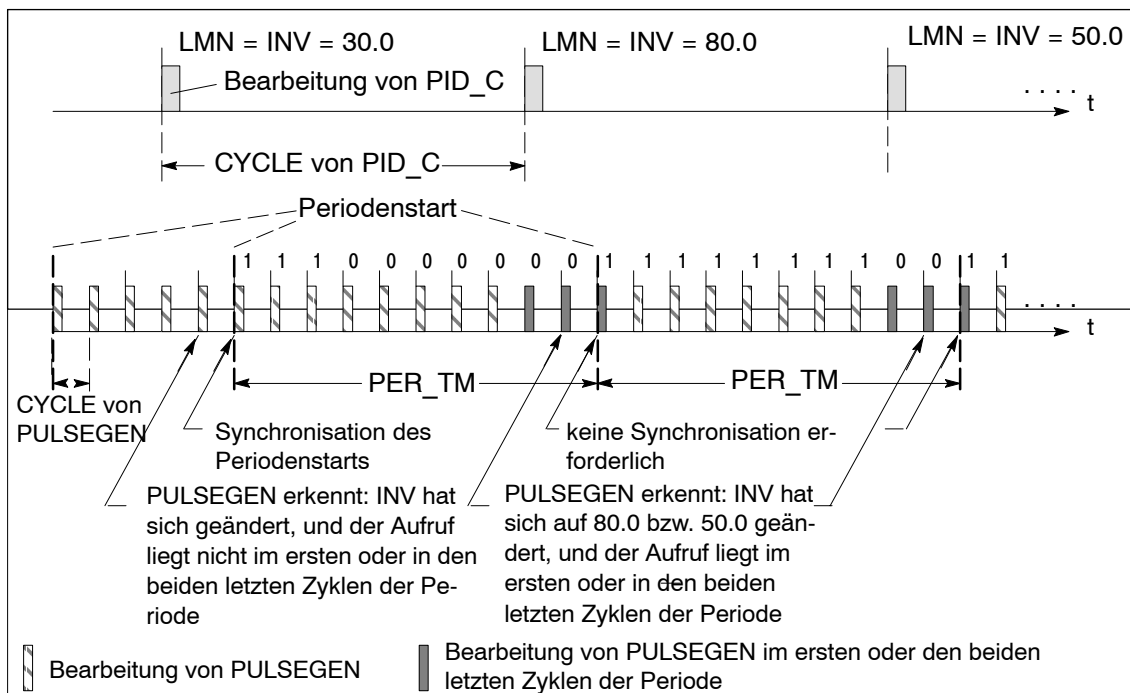


Bild 2-58 Synchronisation des Periodenstarts

Die automatische Synchronisation läßt sich am Eingang "SYN_ON" (= FALSE) abschalten.

Hinweis

Durch den Beginn der neuen Periode wird der Altwert von INV (d.h. von LMN) nach erfolgter Synchronisation mehr oder weniger genau auf das Impulssignal abgebildet.

Betriebsarten des Reglers mit Impulsausgang

Je nach Parametrierung des Impulsformers können PID-Regler mit Dreipunktverhalten oder mit bipolarem bzw. unipolarem Zweipunktausgang konfiguriert werden. Nachstehende Tabelle zeigt die Einstellung der Schalterkombinationen für die möglichen Betriebsarten.

| Betrieb- | Schalter | MAN_ON | STEP3_ON | ST2BI_ON |
|---|----------|--------|----------|----------|
| Dreipunktregelung | | FALSE | TRUE | beliebig |
| Zweipunktregelung mit bipolarem Stellbereich (-100 % ... 100 %) | | FALSE | FALSE | TRUE |
| Zweipunktregelung mit unipolarem Stellbereich (0 % ... 100 %) | | FALSE | FALSE | FALSE |
| Handbetrieb | | TRUE | beliebig | beliebig |

Dreipunktregelung

In der Betriebsart "Dreipunktregelung" können drei Zustände des Stellsignals erzeugt werden, z.B. je nach Stellglied und Prozeß: Mehr - Aus - Weniger, Vorwärts - Stop - Rückwärts, Heizen - Aus - Kühlen usw. Nach Anforderung des zu regelnden Prozesses werden die Zustandswerte der binären Ausgangssignale QPOS_P und QNEG_P den jeweiligen Betriebszuständen des Stellgliedes zugeordnet. Die Tabelle zeigt zwei Beispiele.

| | heizen vorwärts | aus stop | kühlen rückwärts |
|--------|--------------------|-------------|---------------------|
| QPOS_P | TRUE | FALSE | FALSE |
| QNEG_P | FALSE | FALSE | TRUE |

Eine Dimensionierung der Mindestimpuls- bzw. Mindestpausendauer P_B_TM kann sehr kurze Ein- oder Ausschaltzeiten, die die Lebensdauer von Schaltgliedern und Stelleinrichtungen beeinträchtigen, verhindern. Dabei wird der proportionalen Ausgangskennlinie, über die die Impulsdauer berechnet wird, ein Ansprechwert für die Impulsausgabe überlagert.

Hinweis

Kleine Absolutwerte der Eingangsgröße LMN, die eine Impulsdauer kleiner als P_B_TM erzeugen würden, werden unterdrückt. Für große Eingangswerte, die eine Impulsdauer größer als PER_TM - P_B_TM erzeugen würden, wird die Impulsdauer auf 100 % bzw. -100 % gesetzt.

Es werden Einstellwerte $P_B_TM \leq 0,1 * PER_TM$ empfohlen.

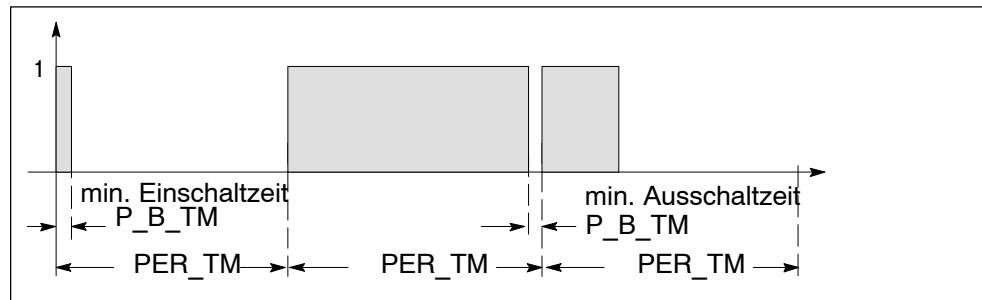


Bild 2-59 Schaltverhalten des Impulsausgangs

Die Dauer der positiven oder negativen Impulse errechnet sich aus Eingangsgröße (in %) mal Impulsdauer:

$$\text{Impulsdauer} = \frac{\text{INV}}{100} * \text{PER_TM}[\text{s}]$$

Durch die Unterdrückung von Mindestimpuls- bzw. -Pausendauer erhält die Umformkennlinie Knickpunkte im Anfangs- und Endbereich (Bild 2-60).

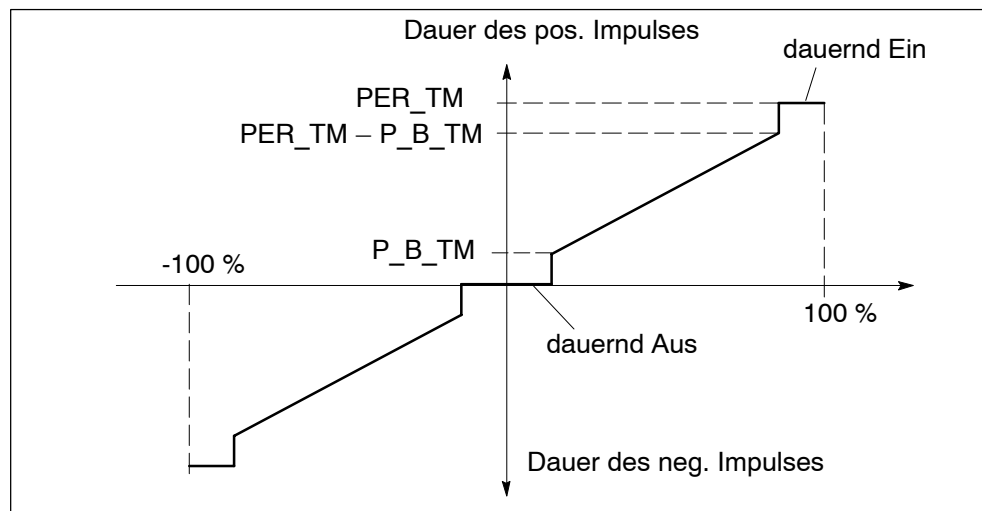


Bild 2-60 Symmetrische Kennlinie des Dreipunktreglers (Verhältnissfaktor = 1)

Dreipunktregelung unsymmetrisch

Über den Verhältnissfaktor **RATIOFAC** kann das Verhältnis der Dauer von positiven zu negativen Impulsen verändert werden. Bei einem thermischen Prozeß lassen sich damit z.B. unterschiedliche Streckenzeitkonstanten für Heizen und Kühlen berücksichtigen.

Soll bei gleichem Absolutwert der Eingangsgröße **|INV|** die Impulsdauer am negativen Impulsausgang gegenüber dem positiven Impuls verkürzt werden, so ist ein Verhältnissfaktor kleiner 1 einzustellen (Bild 2-61):

pos. Impuls > neg. Impuls: **RATIOFAC < 1**

$$\text{Impulsdauer negativ: } \frac{\text{INV}}{100} * \text{PER_TM} * \text{RATIOFAC}$$

$$\text{Impulsdauer positiv: } \frac{\text{INV}}{100} * \text{PER_TM}$$

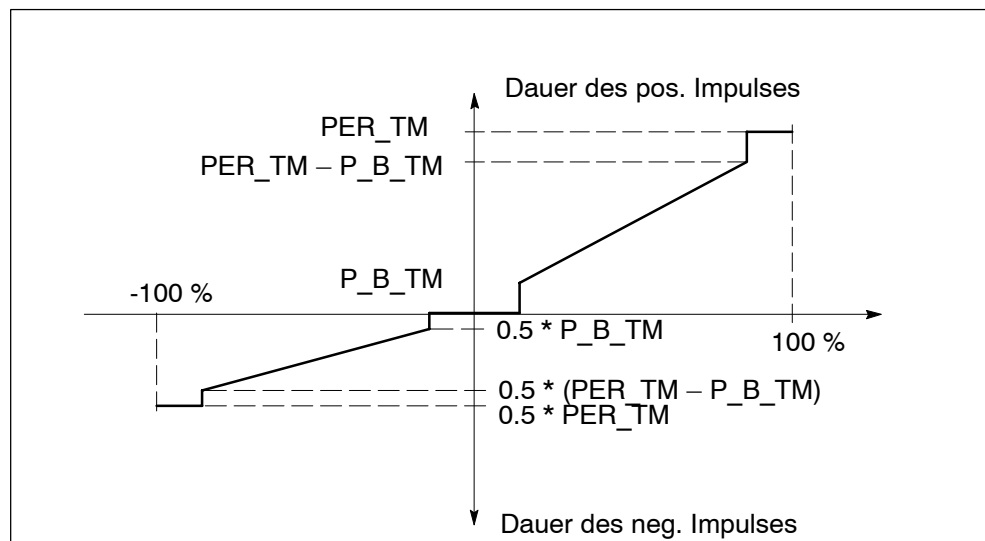


Bild 2-61 Unsymmetrische Kennlinie des Dreipunktreglers (Verhältnisfaktor = 0.5)

Soll umgekehrt bei gleichem Absolutwert der Eingangsgröße $|INV|$ die Impulsdauer am positiven Impulsausgang gegenüber dem negativen Impuls verkürzt werden, so ist ein Verhältnisfaktor größer 1 einzustellen:

pos. Impuls < neg. Impuls: $RATIOFAC > 1$

$$\text{Impulsdauer negativ: } \frac{INV}{100} * PER_TM$$

$$\text{Impulsdauer positiv: } \frac{INV * PER_TM}{100 * RATIOFAC}$$

Der Verhältnisfaktor beeinflusst auch die Mindestimpuls- bzw. Pausendauer. Rechnerisch bedeutet dies, dass bei $RATIOFAC < 1$ der Ansprechwert für negative Impulse mit dem Verhältnisfaktor multipliziert und bei $RATIOFAC > 1$ der Ansprechwert für positive Impulse durch den Verhältnisfaktor dividiert wird.

Zweipunktregelung

Bei der Zweipunktregelung wird nur der positive Impulsausgang QPOS_P von PULSEGEN mit dem betreffenden Ein/Aus-Stellglied verbunden. Je nach genutztem Stellbereich, $LMN = -100.0 \dots 100.0 \%$ oder $LMN = 0.0 \dots 100.0 \%$, hat der Zweipunktregler einen bipolaren oder einen unipolaren Stellbereich.

Bei unipolarer Betriebsart darf die Eingangsgröße INV nur Werte zwischen 0.0 und 100% annehmen.

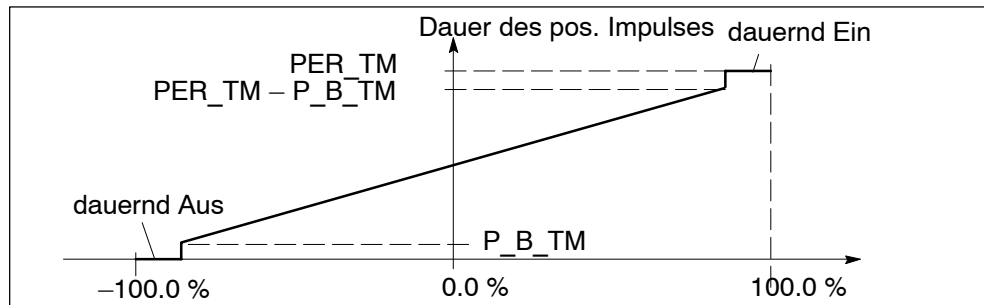


Bild 2-62 Zweipunktregler mit bipolarem Stellbereich (-100 %...100 %)

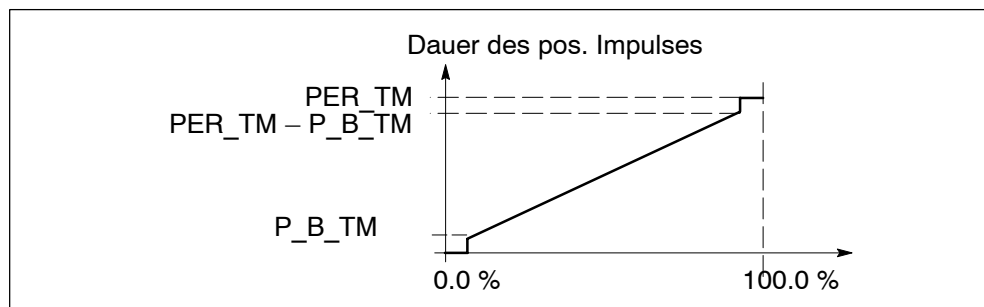


Bild 2-63 Zweipunktregler mit unipolarem Stellbereich (0 %...100 %)

An QNEG_P steht das negierte Ausgangssignal zur Verfügung, falls die Verschaltung des Zweipunktreglers im Regelkreis ein logisch invertiertes Binärsignal für die Stellimpulse erfordert.

| | Ein | Aus |
|--------|-------|-------|
| QPOS_P | TRUE | FALSE |
| QNEG_P | FALSE | TRUE |

Handbetrieb bei Zwei- bzw. Dreipunkt-Regelung

Im Handbetrieb (MAN_ON = TRUE) können die Binärausgänge des Dreipunkt- bzw. Zweipunktreglers über die Signale POS_P_ON und NEG_P_ON unabhängig von INV gesetzt werden.

| | POS_P_ON | NEG_P_ON | QPOS_P | QNEG_P |
|-------------------|----------|----------|--------|--------|
| Dreipunktregelung | FALSE | FALSE | FALSE | FALSE |
| | TRUE | FALSE | TRUE | FALSE |
| | FALSE | TRUE | FALSE | TRUE |
| | TRUE | TRUE | FALSE | FALSE |
| Zweipunktregelung | FALSE | beliebig | FALSE | TRUE |
| | TRUE | beliebig | TRUE | FALSE |

Bausteininterne Begrenzungen

Die Werte der Eingangsparameter werden im Baustein nicht begrenzt; eine Prüfung der Parameter findet nicht statt.

2.1.22 RMP_SOAK: ramp soak (Zeitplangeber)

Anwendungsbereich

Der Zeitplangeber dient vornehmlich als Sollwertgeber für einen Regler, mit dem in Abhängigkeit vom zeitlichen Verlauf des Prozesses unterschiedliche Sollwerte vorgegeben werden.

Blockschaltbild

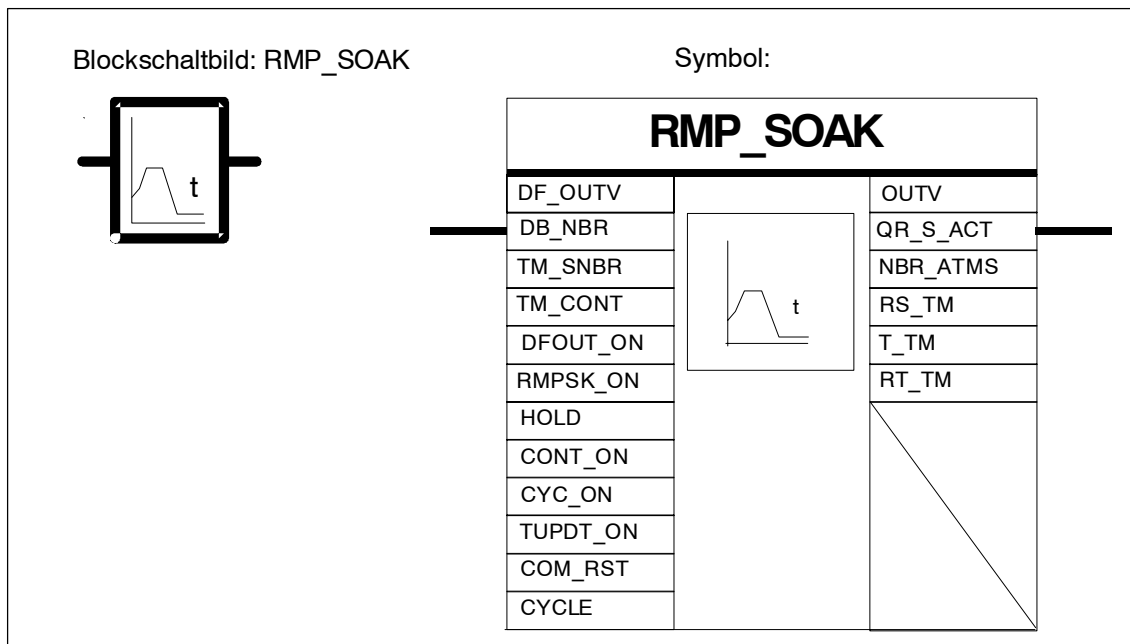


Bild 2-64 RMP_SOAK, Blockschaltbild und Symbol

Funktionsbeschreibung

Mit dem Baustein lassen sich Fahrkurven, deren Stützpunkte in einem globalen Datenbaustein abgelegt sind, ausführen; in jedem Aufrufzyklus werden Werte nach einem Zeitplan ausgegeben.

Zwischen den Stützpunkten wird interpoliert.

Über Steuereingänge können die Betriebsarten

- Ausgang vorbelegen
 - Zeitplangeber einschalten
 - Bearbeitung anhalten
 - Bearbeitungsschritt und –zeit vorgeben
 - zyklischen Betrieb einschalten
 - Gesamtzeit und –restzeit aktualisieren
- angewählt werden.

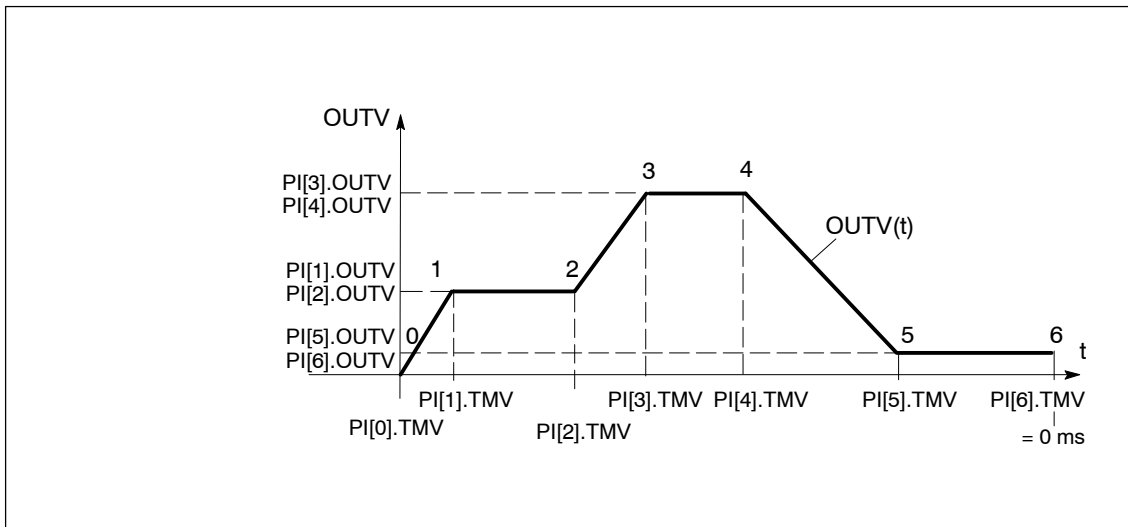


Bild 2-65 Beispiel einer Fahrkurve mit Startpunkt und 6 Stützpunkten

Bei n Stützpunkten ist der Zeitwert für Stützpunkt $n=0$ ms (Bearbeitungsende).

Hinweis

Der Baustein überprüft weder, ob ein globaler DB mit der Nummer DB_NBR wirklich vorhanden ist, noch ob der Parameter $DB_NBR.NBR_PTS$ (Anzahl der Stützpunkte) zur Länge des Datenbausteins paßt. Bei falscher Parametrierung geht die CPU mit der Meldung "interner Systemfehler" in den Betriebszustand STOP.

Eingangsparameter

Die Tabelle zeigt Ihnen Datentyp und Struktur der Eingangsparameter von RMP_SOAK.

Tabelle 2-49 Eingangsparameter von RMP_SOAK

| Datentyp | Parameter | Kommentar englisch deutsch | zulässiger Wertebereich | Vorbelegung |
|----------|-----------|---|--------------------------------------|-------------|
| REAL | DF_OUTV | default output variable Vorbereitung der Ausgangsgröße | technischer Wertebereich | 0.0 |
| BLOCK_DB | DB_NBR | data block number Datenbausteinnummer, globa (Stützpunkt- werte) | CPU abhängig | DB 1 |
| INT | TM_SNBR | time slice number Nr. des nächsten Stützpunktes zum Weiterma- chen | 0 – 255 | 0 |
| TIME | TM_CONT | time to continue (instant) Restzeit zum Weitermachen bis zum Stütz- punkt TM_SNBR | technischer Wertebereich | T#0s |
| BOOL | DFOUT_ON | default output variable on Vorbereitung der Ausgangsgröße ein | | FALSE |
| BOOL | RMPSK_ON | ramp soak on Zeitplangeber einschalten | | FALSE |
| BOOL | HOLD | hold output variable Einfrieren der Ausgangsgröße | | FALSE |
| BOOL | CONT_ON | continue Weitermachen | | FALSE |
| BOOL | CYC_ON | cycle replication on zyklische Wiederholung einschalten | | FALSE |
| BOOL | TUPDT_ON | total time update on Gesamtzeit nach Verändern der Zeitwerte neu berechnen (verlängerte Bausteinlaufzeit !) | | FALSE |
| BOOL | COM_RST | complete restart Neustart | | FALSE |
| TIME | CYCLE | sample time Abtastzeit | technischer Wertebereich ≥ 1ms | T#1s |

Ausgangsparameter

Die Tabelle zeigt Datentyp und Struktur der Ausgangsparameter von RMP_SOAK.

Tabelle 2-50 Ausgangsparameter von RMP_SOAK

| Daten- typ | Parameter | Kommentar englisch deutsch | Vorbe- legung |
|---------------|-----------|--|------------------|
| REAL | OUTV | output variable Ausgangsgröße | 0.0 |
| BOOL | QR_S_ACT | ramp soak active Zeittabelle wird bearbeitet | FALSE |
| INT | NBR_ATMS | number of acting time slice aktuelle Stützpunktnummer (Stützpunkt, der angefahren wird) | 0 |
| TIME | RS_TM | residual slice time aktuelle Restzeit | T#0s |
| TIME | T_TM | total time globale Daten : Gesamtzeit Σ TMVI | T#0s |
| TIME | RT_TM | residual total time globale Daten : Gesamtrestzeit Σ TMVI | T#0s |

Die Stützpunkte sowie die Anzahl der Stützpunkte NBRPTS sind in einem globalen Datenbaustein DB_NBR) abgelegt. Die Ausgabe beginnt mit Stützpunkt 0 und endet mit Stützpunkt NBR_PTS.

Globale Daten

(Vorbelegung mit Startpunkt und 4 Stützpunkten)

Tabelle 2-51

| Daten- typ | Parameter DB_NBR. | Kommentar englisch deutsch | zulässiger Wertebereich | Vorbelegung |
|---------------|----------------------|--|-----------------------------|-------------|
| INT | NBR_PTS | number of points – 1 Anzahl der Stützpunkte –1 | 1 – 255 | 4 |
| REAL | PI[0].OUTV | output variable [0] Ausgangswert [0] Startpunkt | technischer Wertebereich | 0.0 |
| TIME | PI[0].TMV | output time value [0] Zeitwert [0] Startpunkt | | T#1s |
| REAL | PI[1].OUTV | output variable [1] Ausgangswert [1] Stützpunkt 1 | technischer Wertebereich | 0.0 |
| TIME | PI[1].TMV | output time value [1] Zeitwert [1] Stützpunkt 1 | | T#1s |
| REAL | PI[2].OUTV | output variable [2] Ausgangswert [2] Stützpunkt 2 | technischer Wertebereich | 0.0 |

Tabelle 2-51 , Fortsetzung

| Daten-typ | Parameter DB_NBR. | Kommentar englisch deutsch | zulässiger Wertebereich | Vorbelegung |
|-----------|-------------------|--|--------------------------|-------------|
| TIME | PI[2].TMV | output time value [2] Zeitwert [2] Stützpunkt 2 | | T#1s |
| REAL | PI[3].OUTV | output variable [3] Ausgangswert [3] Stützpunkt 3 | technischer Wertebereich | 0.0 |
| TIME | PI[3].TMV | output time value [3] Zeitwert [3] Stützpunkt 3 | | T#1s |
| REAL | PI[4].OUTV | output variable [4] Ausgangswert [4] Stützpunkt 4 | technischer Wertebereich | 0.0 |
| TIME | PI[4].TMV | output time value [4] Zeitwert [4] Stützpunkt 4 | 0 ms | T#0s |

Neustart

Bei Neustart wird der Ausgang OUTV auf 0.0 rückgesetzt. Bei DFOUT_ON=TRUE wird DF_OUTV ausgegeben. Die Zeitabschnitte (0 bis NBRPTS-1) zwischen den Stützpunkten 0 bis NBRPTS werden aufsummiert und unter T_TM zur Verfügung gestellt. Der Ausgang QR_S_ACT wird zurückgesetzt; die Ausgänge NBR_ATMS und RS_TM werden mit 0 vorbesetzt.

Normalbetrieb

- Die Stützpunktparameter NBR_PTS, PI[i].TMV und PI[i].OUTV werden in einem globalen Datenbaustein hinterlegt.
- Der Parameter PI[i].TMV ist im Format TIME anzugeben.
- Die Zählweise der Stützpunkt-Werte und -Zeiten ist in folgendem Bild deutlicher zu erkennen:

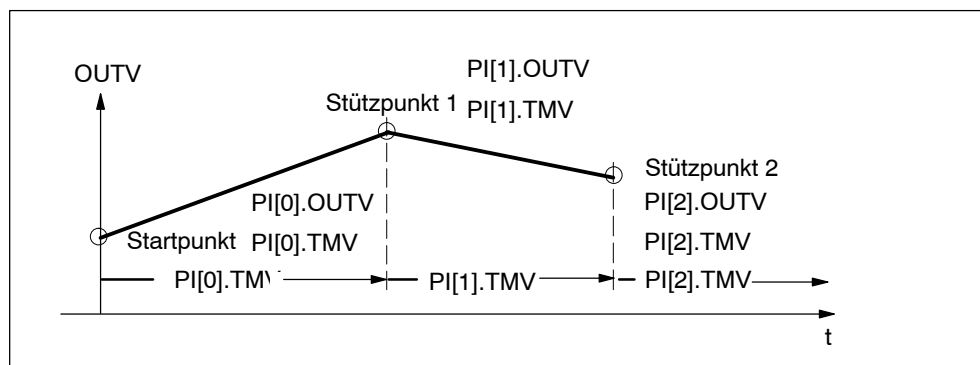


Bild 2-66 Zählweise der Stützpunkte und Zeiten

Der Zeitplangeber interpoliert im Normalbetrieb nach folgender Funktion mit $0 \leq n < (\text{NBR_PTS} - 1)$:

$$\text{OUTV}(t) = \text{PI}[n + 1] \cdot \text{OUTV} - \frac{\text{RS_TM}}{\text{PI}[n] \cdot \text{TMV}} (\text{PI}[n + 1] \cdot \text{OUTV} - \text{PI}[n] \cdot \text{OUTV})$$

Betriebsarten des Zeitplangebers

Über entsprechende Beeinflussung der Steuereingänge können folgende Betriebszustände und Betriebsarten des Zeitplangebers realisiert werden:

1. Zeitplangeber einschalten für einmaligen Durchlauf
2. Ausgang des Zeitplangebers mit festem Wert vorbelegen
3. Zyklischen Betrieb des Zeitplangebers einschalten
4. Bearbeitung des Zeitplangebers anhalten
5. Bearbeitungsschritt und -zeit vorgeben (die Restzeit RS_TM und die Stützpunktnummer TM_SNBR werden neu definiert)
6. Gesamtbearbeitungszeit und Gesamtrestzeit aktualisieren

Betriebsarten

Zur Einstellung einer gewünschten Betriebsart gilt für die Wertigkeit der Steuereingänge folgende Tabelle:

Tabelle 2-52 Betriebsarten des Zeitplangebers (RMP_SOAK)

| Betriebsart | RMPS K_ON | DFOUT _ON | RMP _HOLD | CONT _ON | CYC _ON | TUPDT _ON | Ausgangssignal OUTV |
|--|--------------|--------------|--------------|-------------|------------|--------------|---|
| 1. Zeitplangeber ein | TRUE | FALSE | FALSE | | FALSE | | OUTV(t) Endwert wird nach Bearbeitungsende gehalten. |
| 2. Ausgang vorbelegen | TRUE | TRUE | | | | | DF_OUTV |
| 3. Zykl. Betrieb ein | TRUE | FALSE | FALSE | | TRUE | | OUTV(t) nach Ende: automa- tischer Start |
| 4. Zeitplangeber anhalten | TRUE | FALSE | TRUE | FALSE | | | aktueller Wert von OUTV(t) wird gehalten *) |
| 5. Bearbeitungsschritt und -zeit vorgeben | TRUE | FALSE | TRUE | TRUE | | | OUTV (alt) *) |
| | | | FALSE | | | | Es wird mit neu vorgegebenen Werten weitergefahren. |
| 6. Gesamtzeit aktualisieren | | | | | | FALSE | beeinflusst OUTV nicht |
| | | | | | | TRUE | beeinflusst OUTV nicht |

*) Bis zum nächsten Stützpunkt hat die Fahrkurve nicht die vom Anwender parametrisierte Form!



Die jeweils eingestellte Betriebsart wird unabhängig von der Wertigkeit der Steuersignale in den schraffierten Feldern ausgeführt.

Einschalten des Zeitplangebers

Mit dem Wechsel RMPSK_ON von FALSE nach TRUE wird die Zeitplangeberfunktion eingeschaltet. Nach Erreichen des letzten Stützpunktes ist die Fahrkurve beendet. Bei erneutem Start der Funktion durch den Bediener muß RMPSK_ON zunächst auf 'FALSE' und anschließend wieder auf 'TRUE' gesetzt werden.

Ausgang vorbelegen, Fahrkurve starten

Soll die Fahrkurve mit einem bestimmten Ausgabewert beginnen, dann muß $DFOUT_ON = TRUE$ sein. In diesem Falle steht am Ausgang des Zeitplangebers der Signalwert DF_OUTV an.

Hinweis

Das Signal für die Ausgabe des konstanten Sollwertes $DFOUT_ON$ hat höhere Priorität als das Startsignal für den Zeitplangeber $RMPSK_ON$.

Nach dem Umschalten $DFOUT_ON = FALSE$ wird $OUTV$ linear vom eingestellten Sollwert (DF_OUTV) zum Ausgangswert der aktuellen Stützpunktnummer $PI[NBR_ATMS].OUTV$ gefahren.

Die interne Zeitbearbeitung läuft auch bei durchgeschaltetem festen Sollwert ($RMPSK_ON = TRUE$ und $DFOUT_ON = TRUE$) weiter.

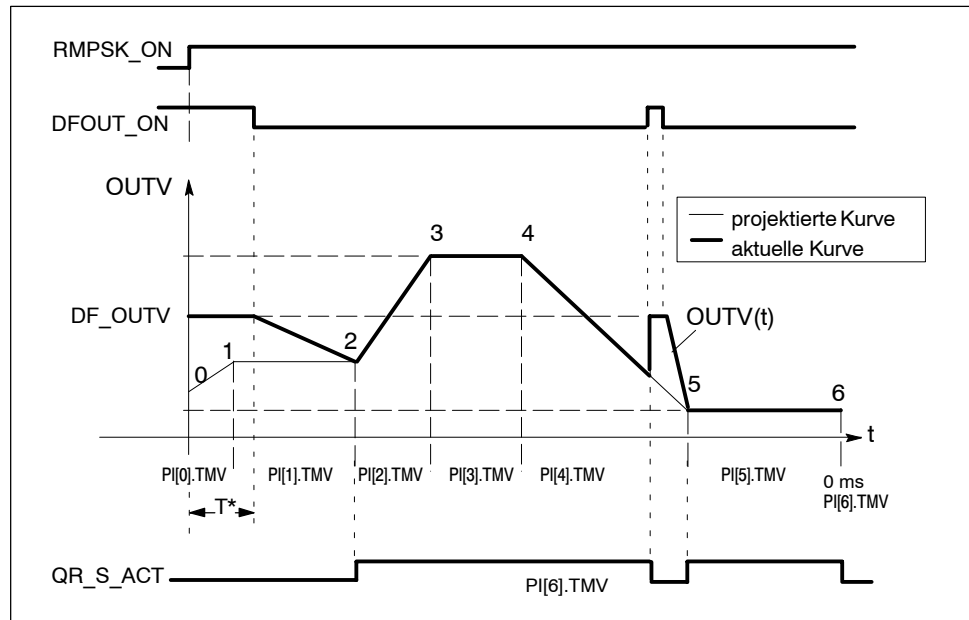


Bild 2-67 Beeinflussung der Fahrkurve durch das Vorbelegungssignal $DFRMP_ON$

Beim Start der Fahrkurve $RMPSK_ON = TRUE$ wird solange der feste Sollwert DF_OUTV ausgegeben, bis $DFOUT_ON$ nach der Dauer von T^* von $TRUE$ nach $FALSE$ wechselt (Bild 2–67). Zu diesem Zeitpunkt ist die Zeit $PI[0].TMV$ und ein Teil der Zeit $PI[1].TMV$ abgelaufen. $OUTV$ wird von DF_OUTV nach $PI[2].OUTV$ d.h. zum Stützpunkt 2 gefahren.

Erst ab dem Stützpunkt 2 wird die projektierte Fahrkurve ausgegeben, d.h. das Ausgangssignal QR_S_ACT erhält den Wert $TRUE$. Bei einem Wechsel des Vorbelegungssignals $DFOUT_ON$ während der Fahrkurvenbearbeitung springt der ausgegebene Wert $OUTV$ ohne Verzögerung auf DF_OUTV .

Zyklischer Betrieb eingeschaltet

Ist die Betriebsart 'Zyklische Wiederholung' (CYC_ON = TRUE) eingeschaltet, dann kehrt der Zeitplangeber nach Ausgabe des letzten Stützpunktwertes automatisch zum Startpunkt zurück und beginnt einen neuen Durchlauf.

Zwischen dem letzten Stützpunkt und dem Startpunkt wird nicht interpoliert. Für stoßfreien Übergang muß gelten: $PI[NBR_PTS].OUTV = PI[0].OUTV$.

Zeitplangeber anhalten

Mit RMP_HOLD = TRUE wird der Wert der Ausgangsgröße (inkl. der Zeitbearbeitung) eingefroren. Bei Rücksetzen RMP_HOLD = FALSE wird an der Unterbrechungsstelle $PI[x].TMV$ fortgefahren.

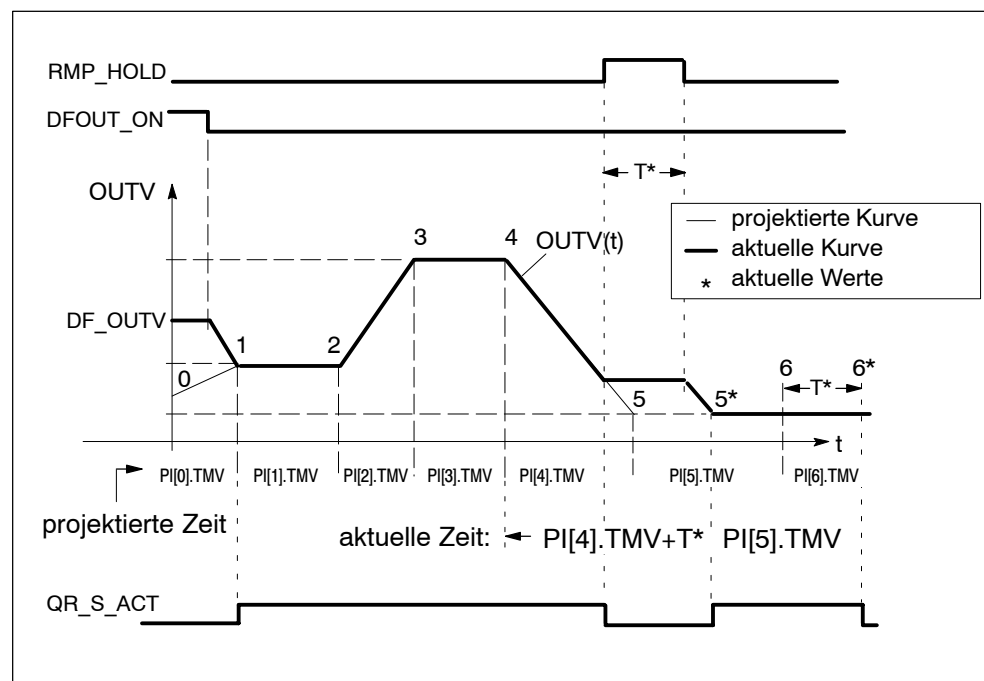


Bild 2-68 Beeinflussung der Fahrkurve durch das Haltesignal RMP_HOLD

Die Bearbeitungszeit der Fahrkurve verlängert sich um die Haltezeit T^* . Die Fahrkurve hat vom Stützpunkt bis zum Signalwechsel von RMP_HOLD (FALSE \rightarrow TRUE) und vom Stützpunkt 5* bis zum Stützpunkt 6* den projektierten Verlauf, d.h. das Ausgangssignal QR_S_ACT hat den Wert TRUE (Bild 2-68).

Ist das Bit $CONT_ON$ gesetzt, fährt der angehaltene Zeitplangeber an der vorgegebenen Stelle TM_CONT mit der Fahrkurvenbearbeitung fort.

Gesamtzeit und Gesamtrestzeit aktualisieren

In jedem Zyklus werden die aktuelle Stützpunktnummer `NBR_ATMS`, die aktuelle Restzeit bis zum Erreichen des Stützpunktes `RS_TM`, die Gesamtzeit `T_TM` und die Gesamtrestzeit bis zum Erreichen des Fahrkurvenendes `RT_TM` aktualisiert.

Bei Online-Änderungen von `PI[n].TMV` ändert sich die Gesamtzeit und die Gesamtrestzeit der Fahrkurve. Da die Berechnung von `T_TM` und `RT_TM` bei vielen Stützpunkten die Bearbeitungszeit des Funktionsbausteins stark vergrößert, wird sie nur nach Neustart oder bei `TUPDT_ON = TRUE` durchgeführt. Die Zeitabschnitte `PI[0 ... NBR_PTS].TMV` zwischen den einzelnen Stützpunkten werden aufsummiert und am Ausgang Gesamtzeit `T_TM` und Gesamtrestzeit `RT_TM` angezeigt.

Bitte beachten Sie, dass die Ermittlung der Gesamtzeiten relativ viel CPU-Laufzeit erfordert!

Bausteininterne Begrenzungen

Die Werte der Eingangsparameter werden im Baustein nicht begrenzt; eine Prüfung der Parameter findet nicht statt.

2.1.23 ROC_LIM: rate of change limiter (Rampe bzw. Hochlaufgeber)

Anwendungsbereich

Rampenfunktionen werden dann verwendet, wenn der Prozeß keinen Eingangssprung verträgt. Dies ist zum Beispiel der Fall, wenn zwischen Motor und anzutreibender Last ein Getriebe zwischengeschaltet ist, und ein zu schnelles Ansteigen der Motordrehzahl zu einer Überlastung des Getriebes führen würde.

Blockschaltbild

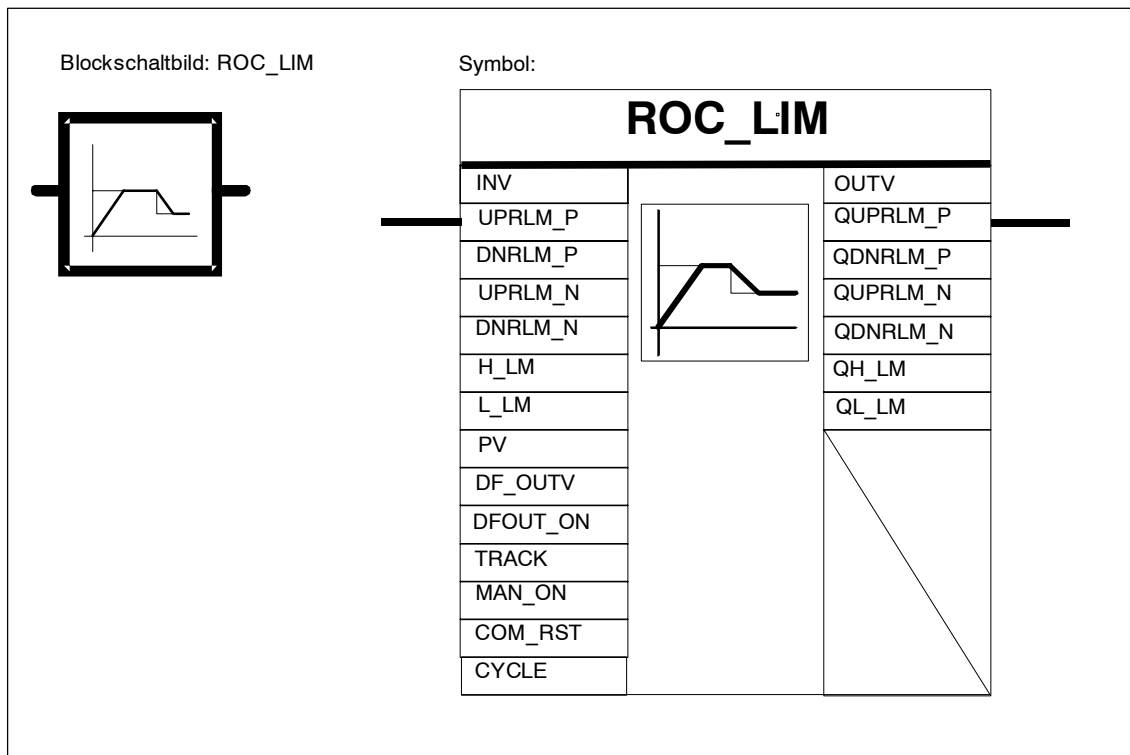


Bild 2-70 ROC_LIM, Blockschaltbild und Symbol

Funktionsbeschreibung

Der Baustein begrenzt die Änderungsgeschwindigkeit eines Ausgangswertes. Aus Sprungfunktionen werden Rampenfunktionen. Für den positiven/negativen Wertebereich der Eingangs- und der Ausgangsgröße sind je zwei Rampen (steigende und fallende Werte) parametrierbar. Über Steuereingänge können die Betriebsarten

- Ausgang vorbelegen
- Nachführen
- Stoßfreie Automatik-Hand-Umschaltung

angewählt werden.

Die Ausgangsgröße kann durch zwei parametrierbare Grenzen begrenzt werden. Eine Begrenzung der Anstiegs-/Abfallgeschwindigkeit sowie die Begrenzung der Ausgangsgröße wird über Ausgänge gemeldet.

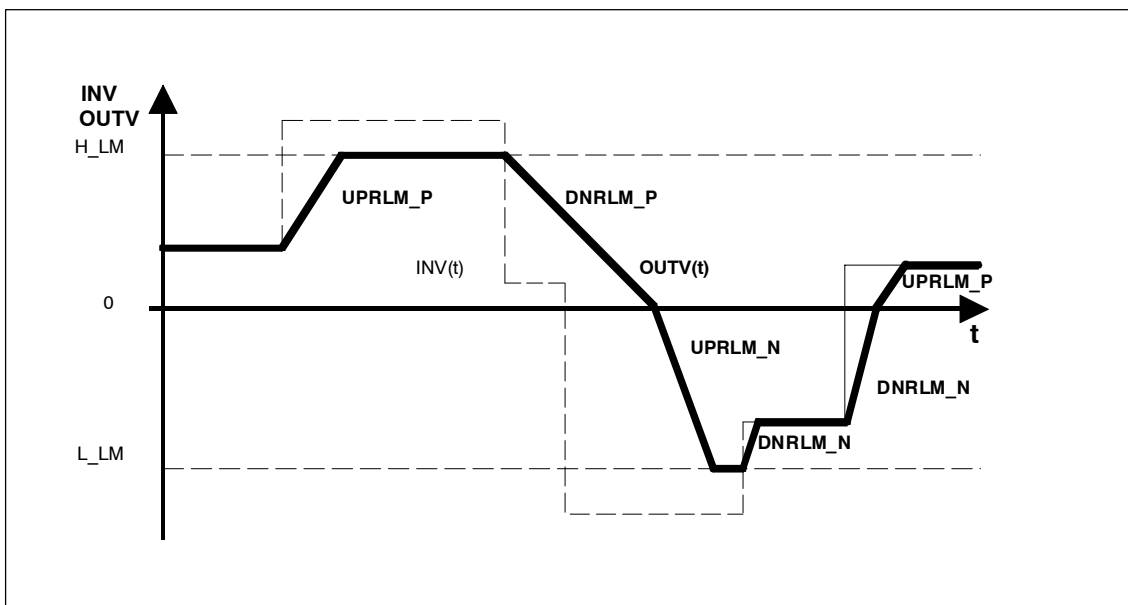


Bild 2-71 Rampenfunktionsverlauf

Die Kennzeichnung der Rampen erfolgt nach folgendem Schema:

| | |
|------------------------------|---------|
| OUTV > 0 und OUTV steigend | UPRLM_P |
| OUTV > 0 und OUTV fallend | DNRLM_P |
| OUTV < 0 und OUTV steigend | UPRLM_N |
| OUTV < 0 und OUTV fallend | DNRLM_N |

Eingangsparameter

Die folgende Tabelle zeigt Ihnen Datentyp und Struktur der Eingangsparameter von ROC_LIM.

Tabelle 2-53

| Daten- typ | Parameter | Kommentar englisch deutsch | zulässiger Wertebereich | Vorbelegung |
|---------------|-----------|--|-----------------------------|-------------|
| REAL | INV | input variable Eingangsgröße | technischer Wertebereich | 0.0 |
| REAL | UPRLM_P | up rate limit in positiv range Anstiegsbegrenzung im positiven Bereich Inkrement pro Sekunde [1/s] | > 0.0 | 10.0 |
| REAL | DNRLM_P | down rate limit in positiv range Abstiegsbegrenzung im positiven Bereich Inkrement pro Sekunde [1/s] | > 0.0 | 10.0 |
| REAL | UPRLM_N | up rate limit in negativ range Anstiegsbegrenzung im negativen Bereich Inkrement pro Sekunde [1/s] | > 0.0 | 10.0 |
| REAL | DNRLM_N | down rate limit in negativ range Abstiegsbegrenzung im negativen Bereich Inkrement pro Sekunde [1/s] | > 0.0 | 10.0 |
| REAL | H_LM | high limit obere Begrenzung | tech. Wert. > L_LM | 100.0 |
| REAL | L_LM | low limit untere Begrenzung | tech. Wert. < H_LM | 0.0 |
| REAL | PV | process variable Regelgröße | technischer Wertebereich | 0.0 |
| REAL | DF_OUTV | default output variable Vorbesetzung der Ausgangsgröße | technischer Wertebereich | 0.0 |
| BOOL | DFOUT_ON | default output variable on Vorbesetzung der Ausgangsgröße ein | | FALSE |
| BOOL | TRACK | tracking OUTV=INV Nachführen OUTV=INV | | FALSE |
| BOOL | MAN_ON | manual mode on Handbetrieb einschalten | | FALSE |
| BOOL | COM_RST | complete restart Neustart | | FALSE |
| TIME | CYCLE | sample time Abtastzeit | | T#1s |

Ausgangsparameter

Die folgende Tabelle zeigt Ihnen Datentyp und Struktur der Ausgangsparameter von ROC_LIM.

Tabelle 2-54

| Daten- typ | Parameter | Kommentar englisch deutsch | Vorbelegung |
|---------------|-----------|--|-------------|
| REAL | OUTV | output variable Ausgangsgröße | 0.0 |
| BOOL | QUPRLM_P | up rate limit in positiv range reached Anstiegsbegrenzung im positiven Bereich angesprochen | FALSE |
| BOOL | QDNRLM_P | down rate limit in positiv range reached Abstiegsbegrenzung im positiven Bereich angesprochen | FALSE |
| BOOL | QUPRLM_N | up rate limit in negativ range reached Anstiegsbegrenzung im negativem Bereich angesprochen | FALSE |
| BOOL | QDNRLM_N | down rate limit in negativ range reached Abstiegsbegrenzung im negativen Bereich angesprochen | FALSE |
| BOOL | QH_LM | high limit reached obere Begrenzung angesprochen | FALSE |
| BOOL | QL_LM | low limit reached untere Begrenzung angesprochen | FALSE |

Neustart

Bei Neustart wird der Ausgang OUTV auf 0.0 rückgesetzt. Wenn DFOUT_ON = TRUE gesetzt ist, wird DF_OUTV ausgegeben. Alle Signalausgänge werden auf FALSE gesetzt.

Normalbetrieb

Die Steigungen sind Begrenzungsgeraden und beziehen sich auf einen Anstieg/ Abfall pro Sekunde; wird z.B. für UPRLM_P = 10.0 parametrier, so wird bei einer Abtastzeit von 1s/100ms/10ms bei jedem Bausteinaufruf, wenn INV > OUTV ist, 10.0/1.0/0.1 zu OUTV addiert, bis INV erreicht ist. Eine Begrenzung der Ausgangsgröße nach oben und unten ist möglich, wenn die Eingangsgröße H_LM über- oder L_LM unterschreitet. (Ausnahme: Handbetrieb MAN_ON=TRUE; siehe Beispiele Bild 2-72)

Ist eine der Begrenzungen verletzt, so wird dies an den Ausgängen QUPRLM_P, QDNRLM_P, QUPRLM_N, QDNRLM_N, QH_LM und QL_LM angezeigt.

- **Ausgang vorbelegen**

Wenn DFOUT_ON = TRUE gesetzt ist, wird am Ausgang DF_OUTV ausgegeben; bei Änderung von TRUE nach FALSE wird OUTV von DF_OUTV nach INV und bei Änderung von FALSE nach TRUE wird OUTV von INV nach DF_OUTV gefahren.

- **Nachführen**

Zum Nachführen OUTV = INV wird das Bit TRACK = TRUE gesetzt; da die Eingangsgröße direkt zur Ausgangsgröße durchgeschaltet wird, werden auch Sprünge der Eingangsgröße ausgegeben.

- **Stoßfreie Hand–Automatik–Umschaltung**

Hierzu muss der Rampenbaustein im Sollwertzweig direkt vor der Regeldifferenz eingebaut werden. Die Regelgröße wird am Eingang PV und das Hand–Automatik–Bit am Eingang MAN_ON verschaltet. Beim Umschalten auf Betriebsart Hand MAN_ON = TRUE wird der am Eingang PV anliegende Wert sofort zum Ausgang OUTV durchgeschaltet. Da Sollwert und Regelgröße gleich sind, wird die Regeldifferenz zu Null und der Regler befindet sich in einem stationären Ruhezustand. Beim Rücksetzen in den Automatikbetrieb MAN_ON = FALSE fährt der Ausgang OUTV vom momentan anliegenden Wert PV rampenförmig zum Eingangswert INV. Dies bewirkt ein stoßfreies Umschalten der Reglerstruktur von Hand– in den Automatikbetrieb. (siehe Bild 2–72)

Ausgang vorbelegen ist bei MAN_ON=TRUE niederprior und wird nicht bearbeitet.

Bausteininterne Begrenzungen

Die Werte der Eingangsparameter werden im Baustein nicht begrenzt; eine Prüfung der Parameter findet nicht statt.

Beispiel

Bei `MAN_ON`, `TRACK`, `DFOUT_ON` gleich `FALSE` ergeben sich folgende Verläufe der Signalausgänge:

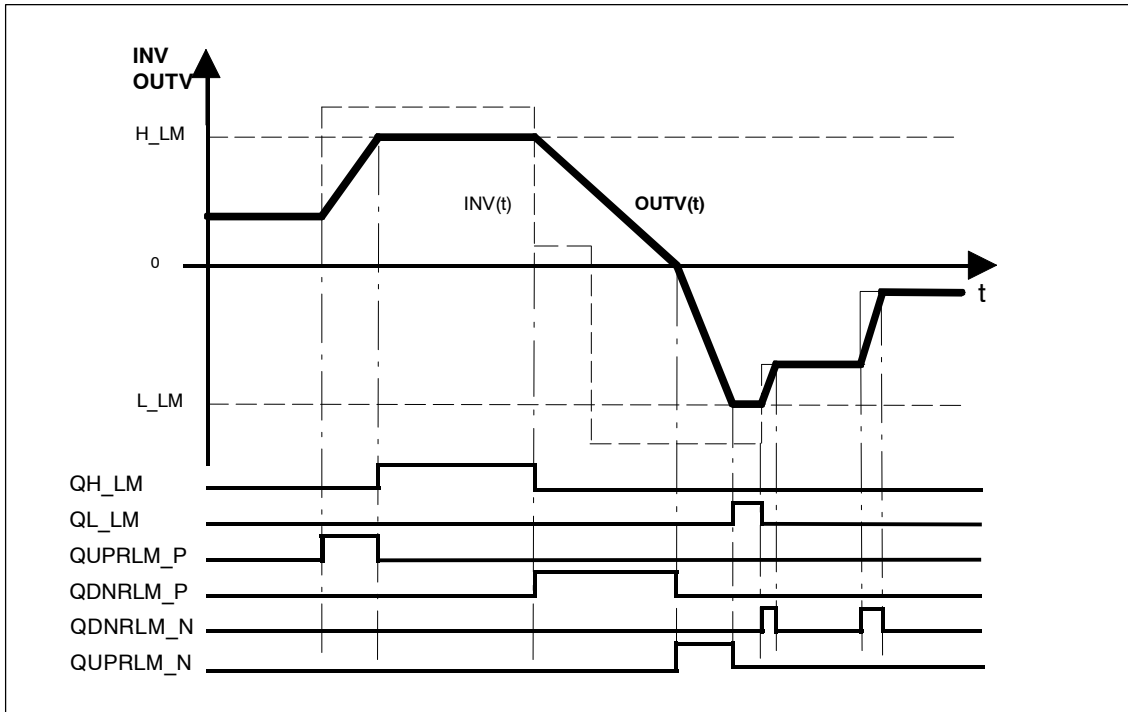


Bild 2-72 Beispiel (`MAN_ON`, `TRACK`, `DFOUT_ON` =`FALSE`; $L_LM < 0.0 < H_LM$)

Bei `MAN_ON` sind die Begrenzungen `H_LM` und `L_LM` unwirksam; bei `TRACK` wird der Eingangswert unverändert ausgegeben. Ist `DFOUT_ON` = `TRUE`, so wird immer `DF_OUTV` ausgegeben.

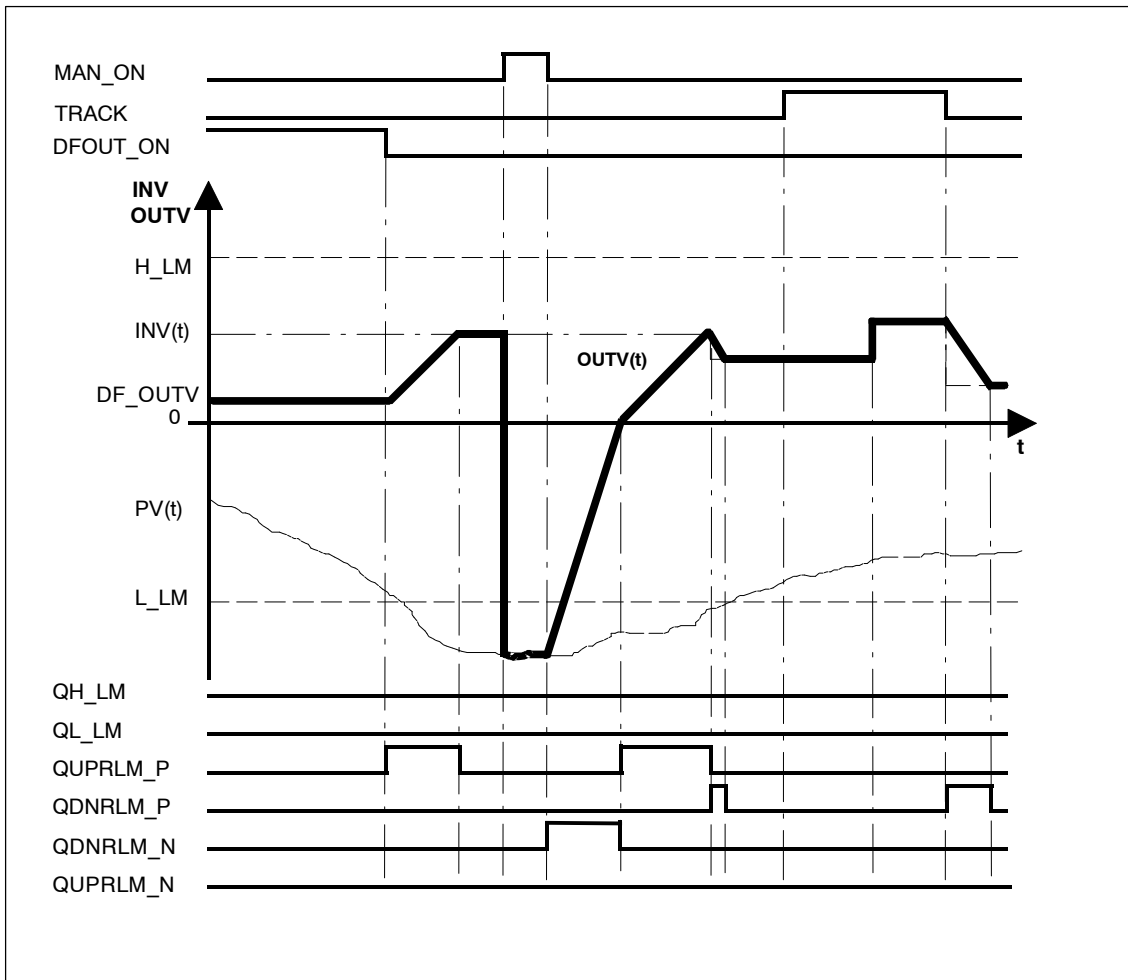


Bild 2-73 Beispiel ($L_LM < 0.0 < H_LM$)

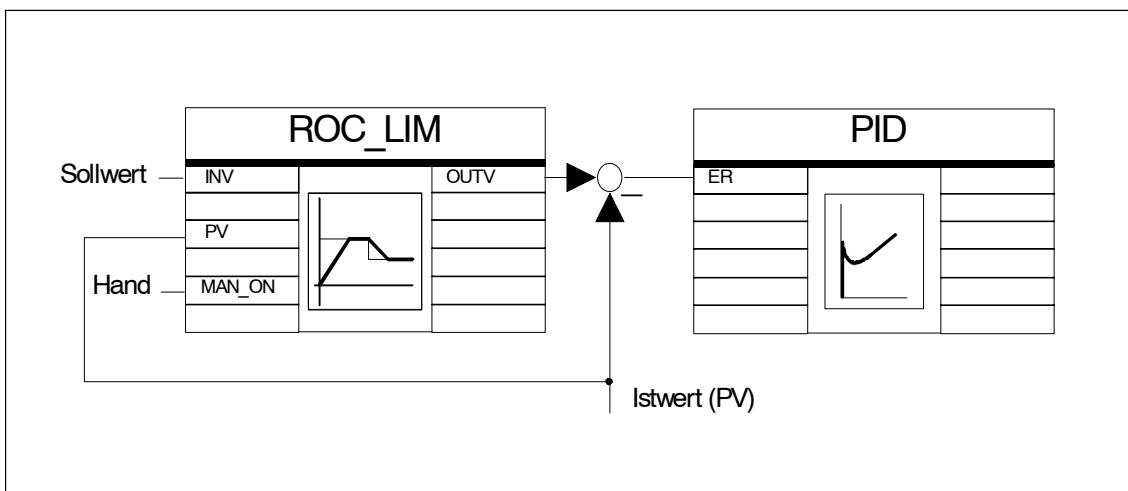


Bild 2-74 Beispiel Stoßfreie Hand-Automatik-Umschaltung

Beispiel einer Rampe, die nur im positiven Wertebereich eingesetzt ist.

Eingangsparameter

Die folgende Tabelle zeigt Ihnen Datentyp und Struktur der Eingangsparameter.

Tabelle 2-55 Eingangsparameter

| Datentyp | Parameter | Kommentar : englisch deutsch | Parametrierung |
|----------|-----------|--|----------------|
| REAL | INV | input variable Eingangsgröße | 0.0 |
| REAL | UPRLM_P | up rate limit in positiv range Anstiegsbegrenzung im positiven Bereich Inkrement pro Sekunde | 10.0 |
| REAL | DNRLM_P | down rate limit in positiv range Abstiegsbegrenzung im positiven Bereich Inkrement pro Sekunde | 5.0 |
| REAL | UPRLM_N | up rate limit in negativ range Anstiegsbegrenzung im negativen Bereich Inkrement pro Sekunde | 0.0 |
| REAL | DNRLM_N | down rate limit in negativ range Abstiegsbegrenzung im negativen Bereich Inkrement pro Sekunde | 0.0 |
| REAL | H_LM | high limit obere Begrenzung | 85.5 |
| REAL | L_LM | low limit untere Begrenzung | 27.0 |
| REAL | PV | process variable Regelgröße | 0.0 |
| REAL | DF_OUTV | default output variable Vorbereitung der Ausgangsgröße | 46.15 |
| BOOL | DFOUT_ON | default output variable on Vorbereitung der Ausgangsgröße ein | FALSE |
| BOOL | TRACK | tracking OUTV=INV Nachführen OUTV=INV | FALSE |
| BOOL | MAN_ON | manual mode on Handbetrieb einschalten | FALSE |

Tabelle 2-55 Eingangparameter, Fortsetzung

| Datentyp | Parameter | Kommentar : englisch deutsch | Parametrierung |
|----------|-----------|------------------------------------|----------------|
| BOOL | COM_RST | complete restart Neustart | FALSE |
| TIME | CYCLE | sample time Abtastzeit | T#1s |

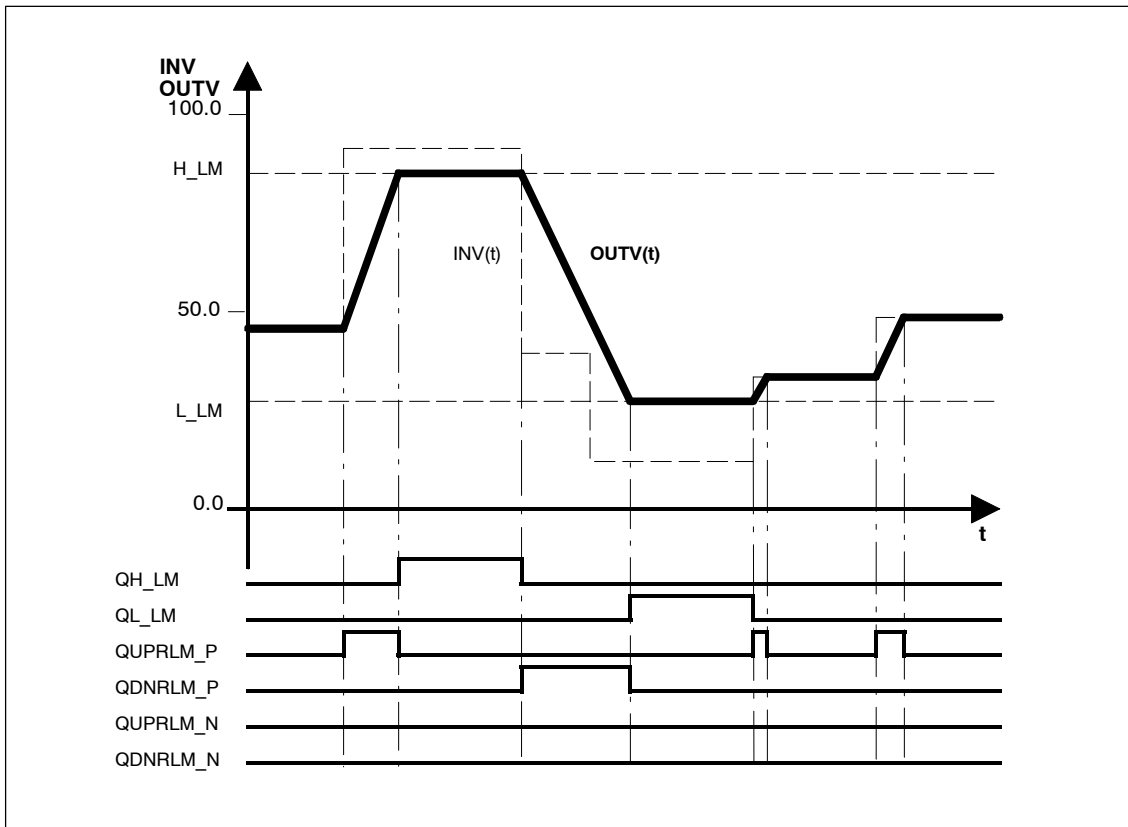


Bild 2-75 Beispiel Arbeitsbereich nur mit Werten ≥ 0.0

2.1.24 SCALE: linear scaling (lineare Skalierung)

Anwendungsbereich

Bei Istwerten liegt der vom Geber gelieferte Wert oft in einem für den Anwender ungünstigen Bereich. (z.B. 0 bis 10V entsprechen 0 bis 1200°C oder 0 bis 10V entsprechen 0 bis 3000U/min). Durch Anpassung von Soll- oder Istwert liegen beide Prozeßgrößen im gleichen Wertebereich.

Blockschaltbild

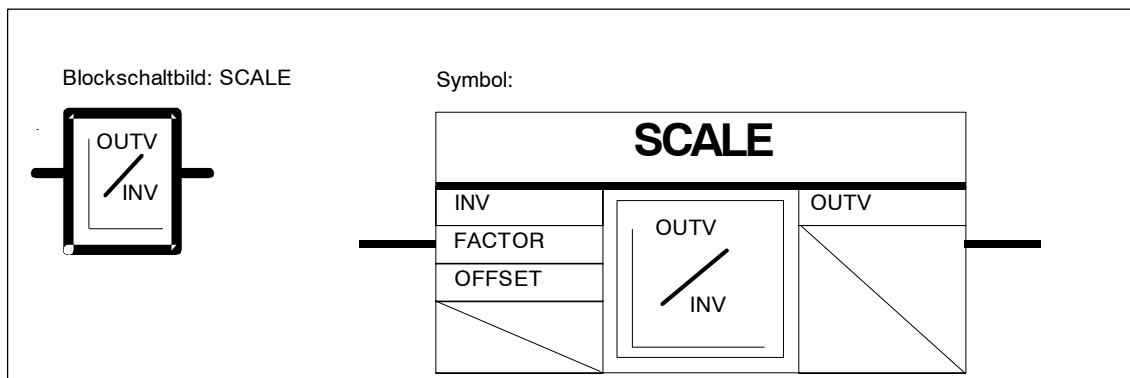


Bild 2-76 SCALE, Blockschaltbild und Symbol

Funktionsbeschreibung

Der Baustein normiert eine analoge Größe. Die Normierungsgerade wird durch die Steigung (FACTOR) und dem Abstand zwischen OUTV bei INV=0 und der Koordinatenachse OUTV=0 definiert.

Algorithmus

$$\text{OUTV} = \text{INV} * \text{FACTOR} + \text{OFFSET}$$

Eine analoge Größe INV wird an der Normierungsgerade in die Ausgangsgröße OUTV überführt. Die Normierungsgerade wird durch die Größen FACTOR und OFFSET eindeutig bestimmt.

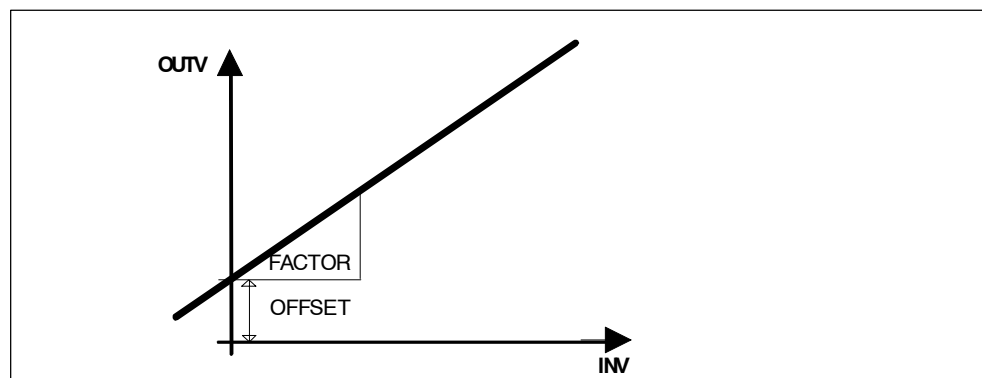


Bild 2-77 Normierungsgerade mit Begrenzung

Eingangsparameter

Die folgende Tabelle zeigt Ihnen Datentyp und Struktur der Eingangsparameter von SCALE.

Tabelle 2-56 Eingangsparameter von SCALE

| Daten- typ | Parameter | Kommentar englisch deutsch | zulässiger Wertebereich | Vorbelegung |
|---------------|-----------|-------------------------------------|-----------------------------|-------------|
| REAL | INV | input variable Eingangsgröße | technischer Wertebereich | 0.0 |
| REAL | FACTOR | scaling factor Skalierungsfaktor | | 1.0 |
| REAL | OFFSET | offset Offset | technischer Wertebereich | 0.0 |

Ausgangsparameter

Die folgende Tabelle zeigt Ihnen Datentyp und Struktur der Ausgangsparameter von SCALE.

Tabelle 2-57 Ausgangsparameter von SCALE

| Daten- typ | Parameter | Kommentar englisch deutsch | Vorbelegung |
|---------------|-----------|----------------------------------|-------------|
| REAL | OUTV | output variable Ausgangsgröße | 0.0 |

Neustart

Der Baustein hat keine Neustarroutine.

Normalbetrieb

Der Baustein enthält keine vom Normalbetrieb abweichende Betriebsarten.

Bausteininterne Begrenzungen

Die Werte der Eingangsparameter werden im Baustein nicht begrenzt; eine Prüfung der Parameter findet nicht statt.

2.1.25 SP_GEN: setpoint generator (Sollwertgeber)

Anwendungsbereich

Zur Sollwertvorgabe von Hand kann mit dem Baustein SP_GEN über zwei Eingänge ein Ausgangswert verändert werden. Für eine feinstufige Änderung sollte der Baustein eine Abtastzeit von $\leq 100\text{ms}$ haben.

Blockschaltbild

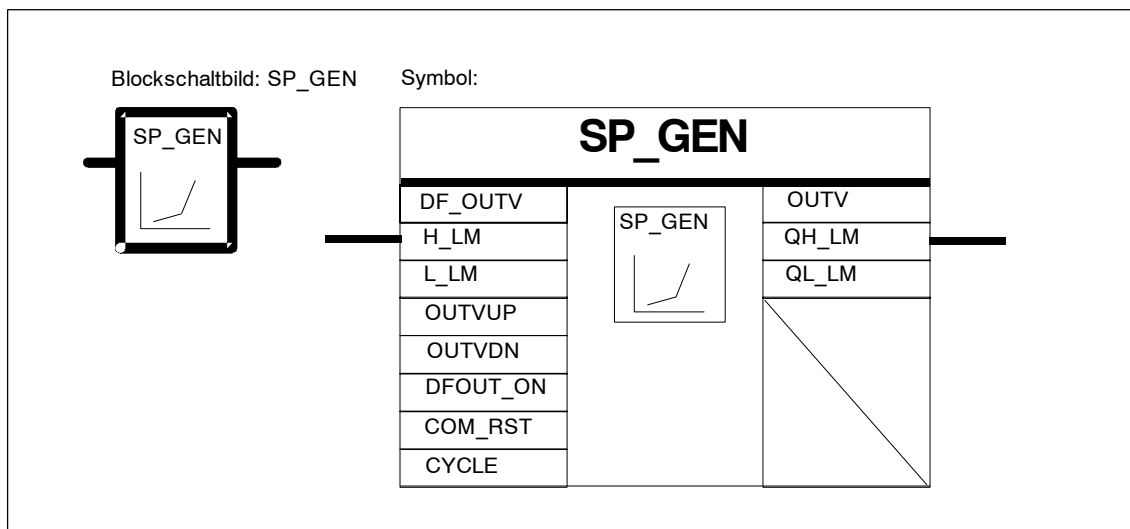


Bild 2-78 SP_GEN, Blockschaltbild und Symbol

Funktionsbeschreibung

Über die Eingänge OUTVUP und OUTVDN kann die Ausgangsgröße OUTV innerhalb der Begrenzungen H_LM und L_LM kontinuierlich vergrößert bzw. verkleinert werden. Die Änderungsgeschwindigkeit hängt von der Betätigungsdauer von OUTVUP und OUTVDN ab.

Während der ersten 3 Sekunden nach Setzen von OUTVUP bzw. OUTVDN ist sie

$$dV/dt = (H_LM - L_LM) / 100 \text{ sec}; \text{ danach ist sie}$$

$$dV/dt = (H_LM - L_LM) / 10 \text{ sec}.$$

Der Wert von OUTV beträgt $L_LM \leq OUTV \leq H_LM$; wird OUTV begrenzt, so wird eine Meldung ausgegeben.

Über DFOUT_ON kann OUTV mit DF_OUTV vorbelegt werden.

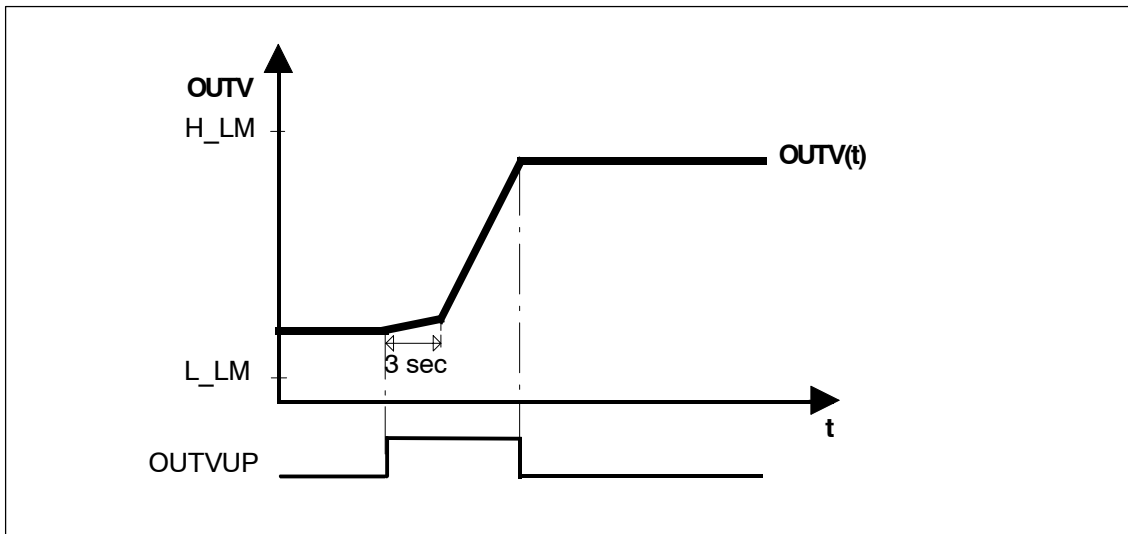


Bild 2-79 Veränderung des Ausgangswertes durch Setzen von OUTVUP

Eingangsparameter

Die folgende Tabelle zeigt Ihnen Datentyp und Struktur der Eingangsparameter von SP_GEN.

Tabelle 2-58 Eingangsparameter von SP_GEN

| Daten- typ | Parameter | Kommentar englisch deutsch | Baustein- interne Wertbe- grenzung | Vorbelegung |
|---------------|-----------|--|--|-------------|
| REAL | DF_OUTV | default output variable Vorbesetzung der Ausgangsgröße | technischer Wertebereich | 0.0 |
| REAL | H_LM | high limit obere Begrenzung | tech. Wert. > L_LM | 100.0 |
| REAL | L_LM | low limit untere Begrenzung | tech. Wert. < H_LM | 0.0 |
| BOOL | OUTVUP | output variable up OUTV Hoch | | FALSE |
| BOOL | OUTVDN | output variable down OUTV Tief | | FALSE |
| BOOL | DFOUT_ON | default output variable on Vorbesetzung der Ausgangsgröße ein | | FALSE |
| BOOL | COM_RST | complete restart Neustart | | FALSE |
| TIME | CYCLE | sample time Abtastzeit | | T#100ms |

Ausgangsparameter

Die folgende Tabelle zeigt Ihnen Datentyp und Struktur der Ausgangsparameter von SP_GEN.

Tabelle 2-59 Ausgangsparameter von SP_GEN

| Daten- typ | Parameter | Kommentar englisch deutsch | Vorbelegung nach Neustart |
|---------------|-----------|---|------------------------------|
| REAL | OUTV | output variable Ausgangsgröße | 0.0 |
| BOOL | QH_LM | high limit reached obere Begrenzung angesprochen | FALSE |
| BOOL | QL_LM | low limit reached untere Begrenzung angesprochen | FALSE |

Neustart

Bei Neustart wird der Ausgang OUTV auf 0.0 gesetzt. Wenn DFOUT_ON = TRUE ist wird DF_OUTV ausgegeben. Die Begrenzungen sind auch bei Neustart wirksam.

Normalbetrieb

DFOUT_ON, OUTVDN und OUTVUP haben folgenden Einfluß auf OUTV:

| DFOUT_ON | OUTVDN | OUTVUP | OUTV |
|----------|----------|----------|------------------|
| TRUE | beliebig | beliebig | DF_OUTV |
| FALSE | TRUE | TRUE | OUTV unverändert |
| | FALSE | TRUE | OUTV steigend |
| | TRUE | FALSE | OUTV fallend |
| | FALSE | FALSE | OUTV unverändert |

- **Ausgang vorbelegen (DFOUT_ON=TRUE)**

Wenn DFOUT_ON = TRUE gesetzt ist, wird am Ausgang DF_OUTV ausgegeben. Ist der Wert von DF_OUTV größer/kleiner H_LM/L_LM so wird er auf H_LM/L_LM begrenzt und QH_LM/QL_LM=TRUE ausgegeben. Die Änderung von OUTV erfolgt sprunghaft. Die Umschaltung nach DFOUT_ON = FALSE ist stoßfrei.

2.1.26 SPLT_RAN: split-ranging (Bereichsaufteilung)

Anwendungsbereich

Der Baustein wird benötigt zur Realisierung einer Bereichsauswahl- (splitrange) Regelung.

Der Stellwertbereich eines PID-Reglers wird aufgesplittet in mehrer Einzelbereiche. Pro Einzelbereich muß der Baustein einmal aufgerufen werden und mit einem der Stellwertverarbeitungsbausteine LMNGEN_C bzw. LMNGEN_S verbunden werden.

Blockschaltbild

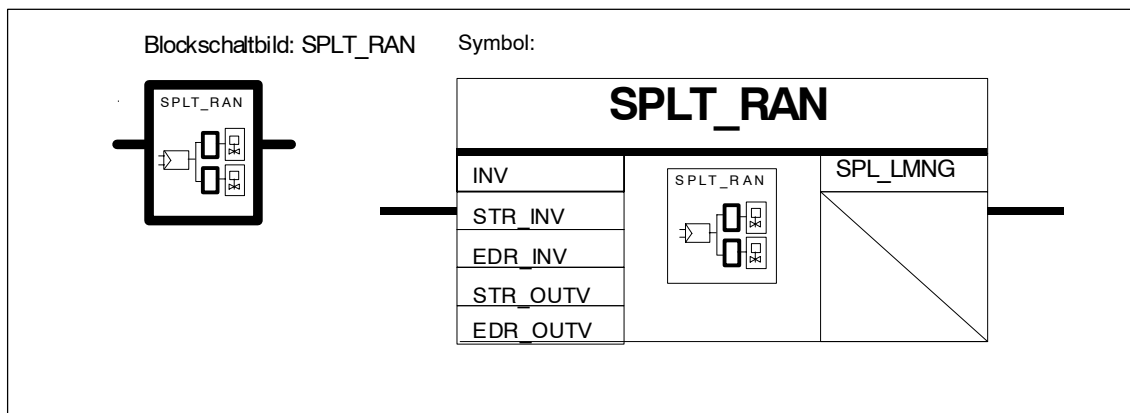


Bild 2-81 SPLT_RAN, Blockschaltbild und Symbol

Funktionsbeschreibung

Ein Eingangswert wird innerhalb eines parametrisierten Bereiches, festgelegt durch STR_INV und EDR_INV, abgebildet auf einen Ausgangswert mit einem parametrisierten Bereich, festgelegt durch STR_OUTV und EDR_OUTV. (siehe Bild 2-82)

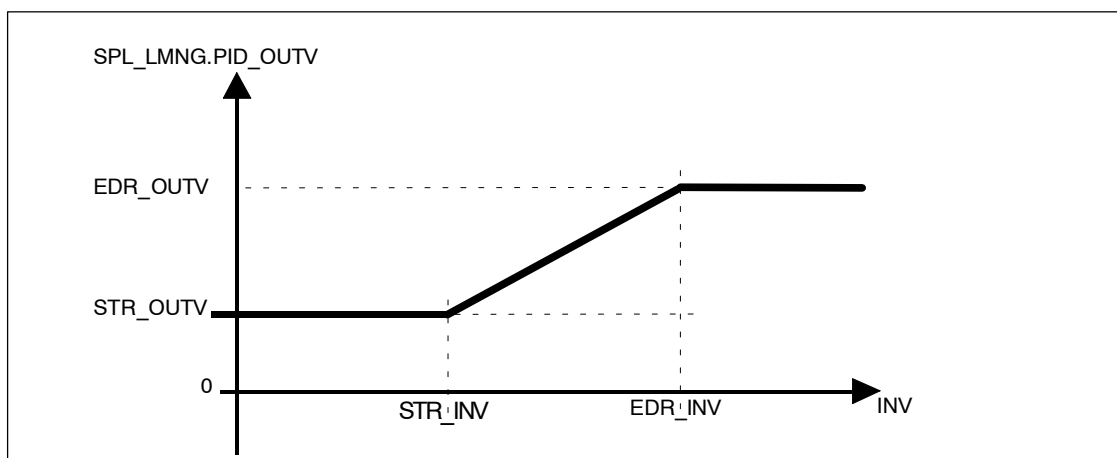


Bild 2-82

Eingangsparameter

Die folgende Tabelle zeigt Ihnen Datentyp und Struktur der Eingangsparameter von SPLT_RAN.

Tabelle 2-60 Eingangsparameter von SPLT_RAN

| Daten- typ | Parameter | Kommentar englisch deutsch | zulässiger Wertebereich | Vorbelegung |
|---------------|-----------|--|-----------------------------|-------------|
| REAL | INV | input variable Eingangsvariable | technischer Wertebereich | 0.0 |
| REAL | STR_INV | start of range INV Bereichsanfang INV | technischer Wertebereich | 0.0 |
| REAL | EDR_INV | end of range INV Bereichsende INV | technischer Wertebereich | 50.0 |
| REAL | STR_OUTV | start of range OUTV Bereichsanfang OUTV | technischer Wertebereich | 0.0 |
| REAL | EDR_OUTV | end of range OUTV Bereichsende OUTV | technischer Wertebereich | 100.0 |

Ausgangsparameter

Die folgende Tabelle zeigt Ihnen Datentyp und Struktur der Ausgangsparameter von SPLT_RAN.

Tabelle 2-61 Ausgangsparameter von SPLT_RAN

| Daten- typ | Parameter | Kommentar englisch deutsch | Vorbelegung |
|---------------|-----------|--|-------------|
| STRUC | SPL_LMNG | PID-LMNGEN interface PID-LMNGEN Schnittstelle | |

Neustart

Der Baustein hat keine Neustarroutine.

Normalbetrieb

Der Baustein enthält keine vom Normalbetrieb abweichenden Betriebsarten.

Bausteininterne Begrenzungen

Die Werte der Eingangsparameter werden im Baustein nicht begrenzt; eine Prüfung der Parameter findet nicht statt.

Beispiel

Der Ausgangswert des PID-Bausteins wird durch SPLT_RAN auf zwei Stellwertverarbeitungs-Bausteine LMNGEN_C und LMNGEN_S aufgeteilt.

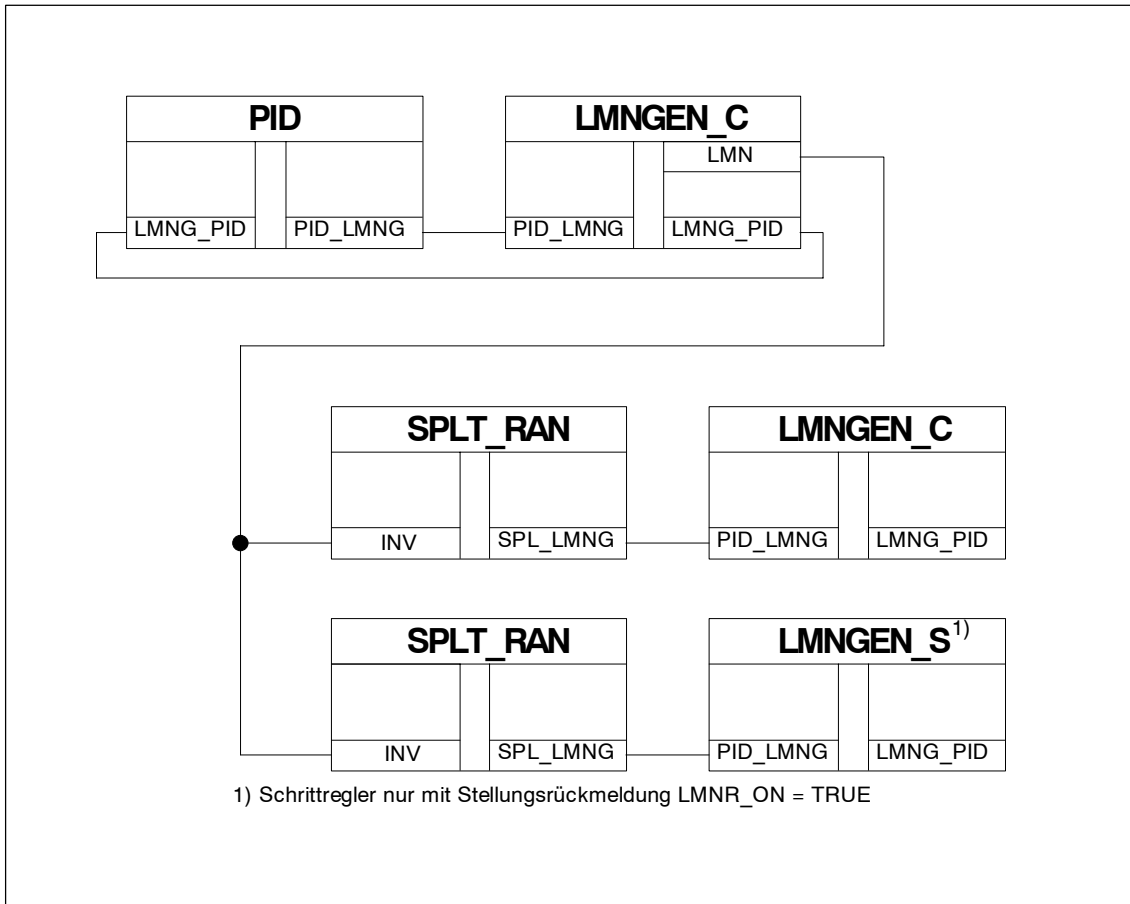


Bild 2-83 Verbindung von SPLT_RAN mit PID und LMNGEN_S

2.1.27 SWITCH: switch (Analogwertschalter)

Anwendungsbereich

Der Baustein findet Verwendung als Eingangs- und/oder Ausgangsmultiplexer von 2 Ein-/Ausgangsgrößen.

Blockschaltbild

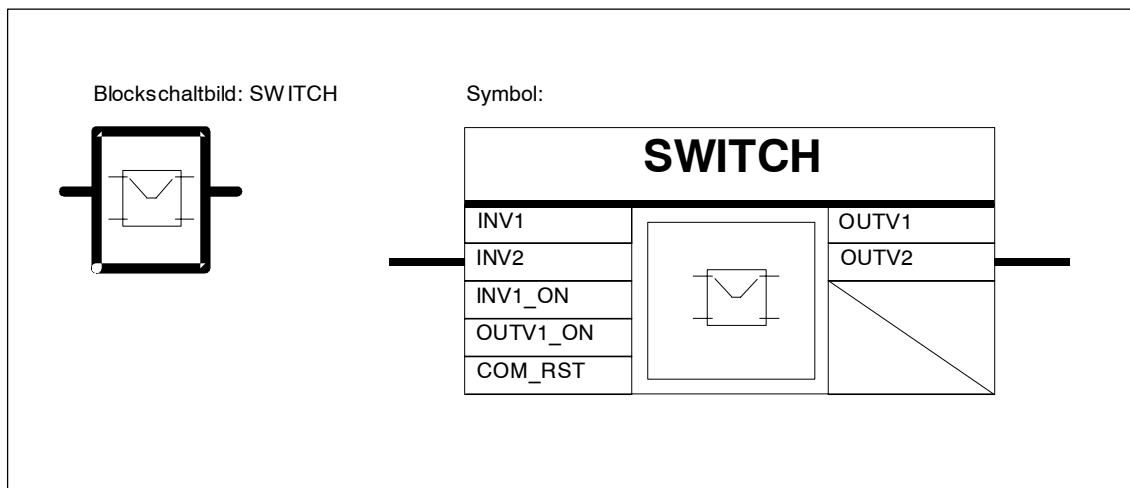


Bild 2-84 SWITCH, Blockschaltbild und Symbol

Funktionsbeschreibung

Der Baustein schaltet einen von 2 analogen Eingangswerten auf einen von 2 Ausgangswerten, er arbeitet nach folgender Tabelle:

| INV1_ON | OUTV1_ON | OUTV1 | OUTV2 |
|---------|----------|-------------|-------------|
| 0 | 0 | unverändert | INV2 |
| 1 | 0 | unverändert | INV1 |
| 0 | 1 | INV2 | unverändert |
| 1 | 1 | INV1 | unverändert |

Eingangsparameter

Die folgende Tabelle zeigt Ihnen Datentyp und Struktur der Eingangsparameter von SWITCH.

Tabelle 2-62 Eingangsparameter von SWITCH

| Daten- typ | Parameter | Kommentar englisch deutsch | zulässiger Wertebereich | Vorbelegung |
|---------------|-----------|---|-----------------------------|-------------|
| REAL | INV1 | input variable 1 Eingangsgröße 1 | technischer Wertebereich | 0.0 |
| REAL | INV2 | input variable 2 Eingangsgröße 2 | technischer Wertebereich | 0.0 |
| BOOL | INV1_ON | connect through INV1 Eingangsgröße INV1 wird durchgeschaltet | | FALSE |
| BOOL | OUTV1_ON | connect through OUTV1 Ausgangsgröße OUTV1 wird durchgeschaltet | | FALSE |
| BOOL | COM_RST | complete restart Neustart | | FALSE |

Ausgangsparameter

Die folgende Tabelle zeigt Ihnen Datentyp und Struktur der Ausgangsparameter von SWITCH.

Tabelle 2-63 Ausgangsparameter von SWITCH

| Daten- typ | Parameter | Kommentar englisch deutsch | Vorbelegung |
|---------------|-----------|--------------------------------------|-------------|
| REAL | OUTV1 | output variable 1 Ausgangsgröße 1 | 0.0 |
| REAL | OUTV2 | output variable 2 Ausgangsgröße 2 | 0.0 |

Neustart

Bei Neustart werden OUTV1=0.0 und OUTV2=0.0 gesetzt.

Normalbetrieb

Der Baustein enthält keine vom Normalbetrieb abweichende Betriebsarten.

Bausteininterne Begrenzungen

Die Werte der Eingangsparameter werden im Baustein nicht begrenzt; eine Prüfung der Parameter findet nicht statt.

Beispiele

3.1 Anwendung von Modular PID Control

Übersicht

Mit Hilfe der Bausteine aus der Bibliothek ModPID von Modular PID Control können Sie Ihren speziellen Regler aufbauen.

Im Projekt zDt28_4_ModCon sind 12 Beispiele für Reglerstrukturen enthalten (EXAMPLE01 bis EXAMPLE12). In den Abschnitten 3.2 bis 3.13 finden Sie die Beschreibung dieser 12 Beispiele, die mit den in Kapitel 2 beschriebenen Bausteinen der Bibliothek ModPID aufgebaut sind.

Beispiele und ihre Anwendung

Tabelle 3-1 zeigt Ihnen eine Aufstellung der mitgelieferten Beispiele aus dem Projekt zDt28_4_ModCon.

Tabelle 3-1 Liste der Beispiele

| Beispiel | Funktionalität |
|-----------|---|
| EXAMPLE01 | Festwertregler mit schaltendem Ausgang für integrierende Stellglieder |
| EXAMPLE02 | Festwertregler mit kontinuierlichem Ausgang |
| EXAMPLE03 | Festwertregler mit schaltendem Ausgang für proportionale Stellglieder |
| EXAMPLE04 | Einschleifiger Verhältnisregler |
| EXAMPLE05 | Mehrschleifiger Verhältnisregler |
| EXAMPLE06 | Mischungsregler |
| EXAMPLE07 | Kaskadenregler |
| EXAMPLE08 | Regler mit Vorsteuerung |
| EXAMPLE09 | Regler mit Störgrößenaufschaltung |
| EXAMPLE10 | Bereichsauswahlregler |
| EXAMPLE11 | Ablöseregler |
| EXAMPLE12 | Mehrgrößenregler |

Anhand der Beispiele aus Tabelle 3-1 können Sie den Aufruf und die Verschaltung der wichtigsten Bausteine nachsehen.

Sie können das Beispiel, das am besten Ihrer gewünschten Reglerstruktur entspricht, als Vorlage kopieren und diese Vorlage durch Streichen oder Hinzufügen von Bausteinaufrufen und Verschaltungen modifizieren.

Hinweis

Nur die Beispiele 1 bis 3 haben eine Streckensimulation und sind deshalb ohne Prozeßanbindung lauffähig.

Die Beispiele 4 bis 12 erfordern eine Prozeßanbindung. Damit diese wirksam werden kann, müssen Sie die Bausteine (CRP_IN, LMNGEN_C, LMNGEN_S, SP_GEN ...) so umparametrieren, daß die Prozeßwerte durchgeschaltet werden.

Sie realisieren Ihre gewünschte Reglerstruktur (Anwender-FB) als FB mit lokalen Instanzen von FBs aus der Bibliothek ModPID. Ihr Anwender-FB beinhaltet den Bausteinaufruf und die Verschaltung von Ein- und Ausgangsparametern. Sie können Ihren Anwender-FB sowohl mit AWL als auch mit SCL erstellen.

Den so erstellten Regler (Anwender-FB) können Sie in einem zur Anwendung passenden Organisationsbaustein aufrufen.

Programmierbeispiel für AWL

Das folgende Beispiel zeigt, wie Sie in AWL die Bausteine aus der Bibliothek ModPID aufrufen und verschalten können.

| Adresse | Deklaration | Name | Type |
|---------|-------------|------------|-------|
| 0.0 | in | SP_UP | BOOL |
| 0.1 | in | SP_DOWN | BOOL |
| 2.0 | out | OUT | REAL |
| 6.0 | stat | DI_SP_GEN | FB 25 |
| 46.0 | stat | DI_ROC_LIM | FB 22 |

```

AWL
Netzwerk 1:
CALL  #DI_SP_GEN           //Bausteinaufruf
      OUTVUP := #SP_UP
      OUTVDN :=#SP_DOWN
L     #DI_SP_GEN.OUTV      //Verschaltung
T     #DI_ROC_LIM.INV
CALL  #DI_ROC_LIM         //Bausteinaufruf
      OUTV :=#OUT
BE
    
```

Programmierbeispiel für SCL

Das folgende Beispiel zeigt, wie Sie in SCL die Bausteine aus der Bibliothek Mod-PID aufrufen und verschalten können.

```

FUNCTION_BLOCK Anwender-FB
VAR_INPUT
    SP_UP:                bool := FALSE;
    SP_DOWN:              bool := FALSE;
END_VAR
VAR_OUTPUT
    OUT:                  real := 0.0;
END_VAR
VAR
    DI_SP_GEN:            SP_GEN;
    DI_ROC_LIM:           ROC_LIM;
END_VAR
BEGIN
    DI_SP_GEN(             //Bausteinanruf + Verschaltung
        OUTVUP := SP_UP,
        OUTVDN :=SP_DOWN);
    DI_ROC_LIM(           //Bausteinanruf + Verschaltung
        INV := DI_SP_GEN.OUTV);
    OUT := DI_ROC_LIM.OUTV; //Verschaltung
END_FUNCTION_BLOCK

```

Üben mit den Beispielen

Die Beispiele 1 bis 3 enthalten einen vollständigen Regelkreis. Sie eignen sich besonders zum "Üben".

Mit dem Standard-S7-Tool "Variable beobachten und steuern" können Sie mit einfachen Mitteln Regelparameter ändern und die Reaktion des im Beispiel simulierten Regelkreises beobachten.

Die Projektiersoftware bietet grafische Oberflächen mit Kreisbilddarstellung und Kurvenschreiber und erlaubt Regleroptimierung.

3.2 Beispiel 1: Festwertregler mit schaltendem Ausgang für integrierende Stellglieder mit Streckensimulation

Übersicht

Beispiel 1 hat den Namen EXAMPLE01 und besteht aus einem PID-Schrittregler (Festwertregler mit schaltendem Ausgang für integrierende Stellglieder) und einer simulierten Regelstrecke.

Regelkreis

Bild 3-1 zeigt den kompletten Regelkreis des Beispiels 1.

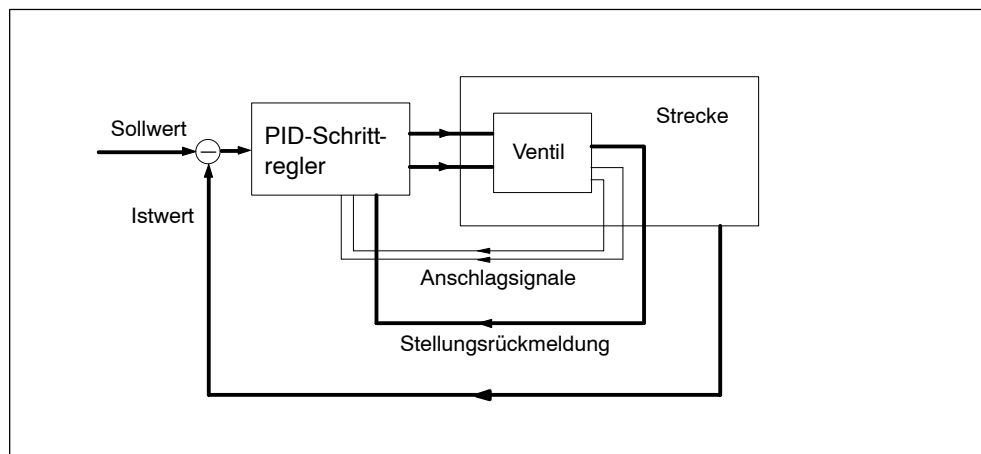


Bild 3-1 Regelkreis des Beispiels 1

Hinweis

Sie müssen den Parameter DB50.DI_LMNGEN_S.MAN_ON auf FALSE setzen, um mit der Funktion Kreisbild der Projektiersoftware arbeiten zu können.

Bausteinanruf und Verschaltung

Bild 3-2 zeigt den Bausteinanruf und die Verschaltung des Beispiel 1.##

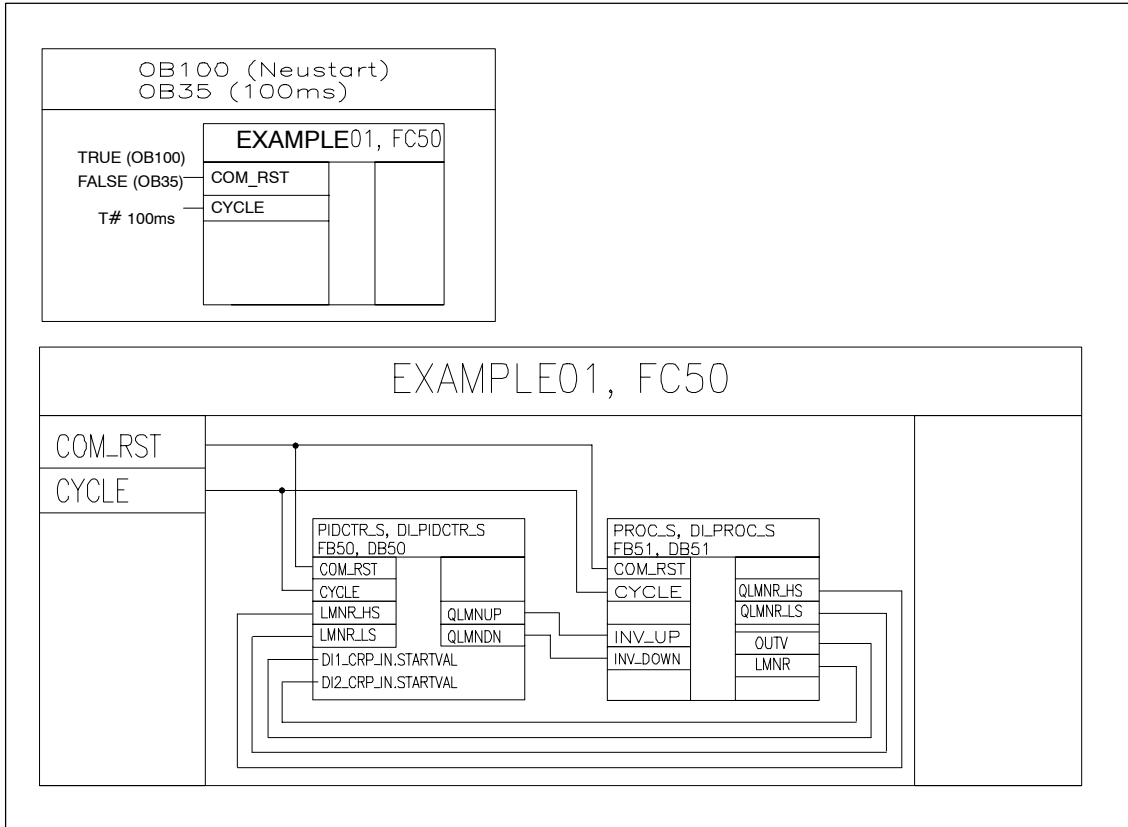


Bild 3-2 Bausteinanruf und Verschaltung des Beispiels 1

3.2.1 PIDCTR_S: Festwertregler mit schaltendem Ausgang für integrierende Stellglieder

Anwendungsbereich

Der Baustein PIDCTR_S realisiert einen PID-Schrittregler für integrierende Stellglieder (z. B. motorgetriebene Stellventile in verfahrenstechnischen Prozessen). Bild 3-3 zeigt die Bausteinverschaltung des PIDCTR_S.

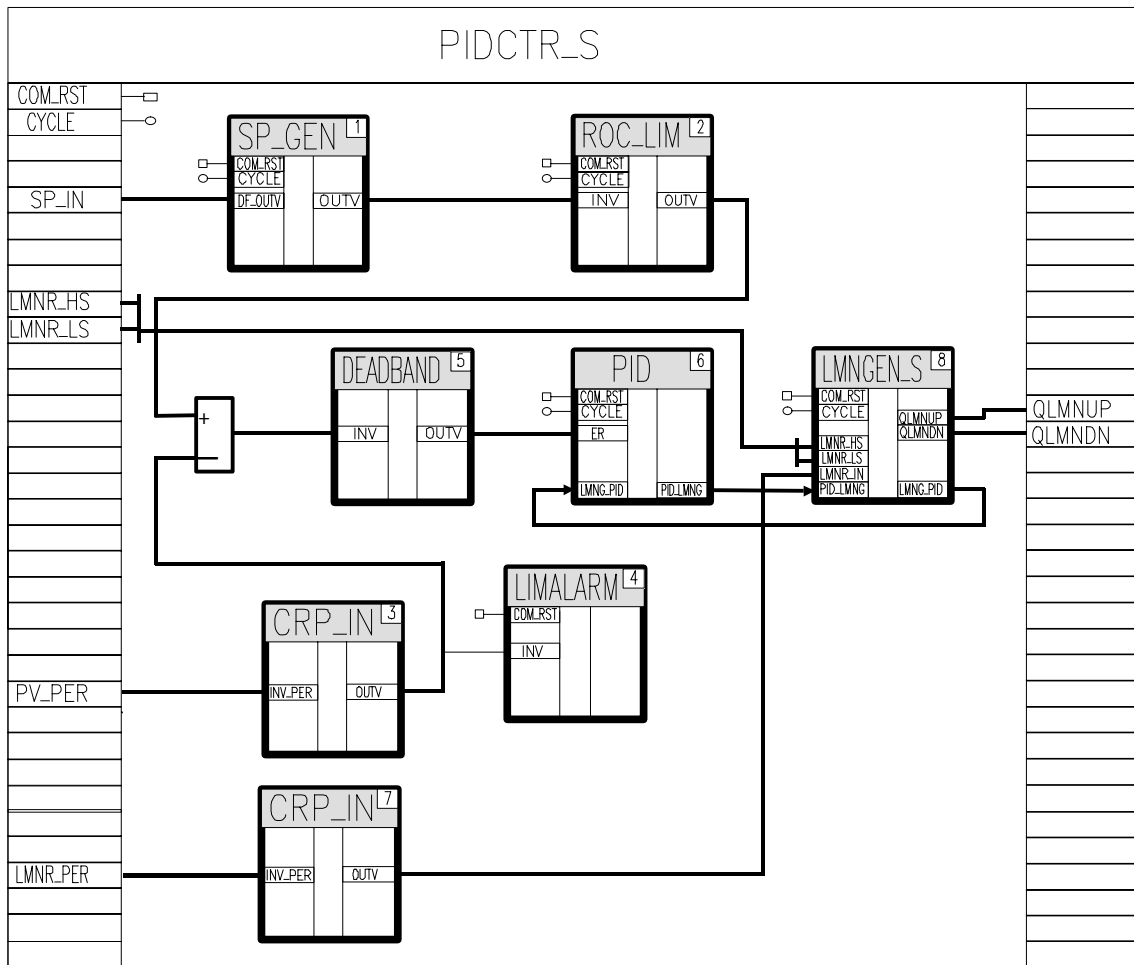


Bild 3-3 Bausteinverschaltung des PIDCTR_S

Funktionsbeschreibung

Der Sollwertgeber SP_GEN gibt den Sollwert vor, dessen Steigung vom Hochlaufgeber ROC_LIM begrenzt wird. Der Peripherie-Istwert wird mittels CRP_IN in einen Gleitpunktwert gewandelt und durch den Grenzwertmelder LIMALARM auf voreingestellte Grenzwerte überwacht. Die Regeldifferenz wird über ein Totbandglied DEADBAND auf den PID-Algorithmus PID geführt. Die Stellungsrückmeldung wird über einen zweiten CRP_IN-Baustein eingelesen. Der Stellwertverarbeitungsbaustein LMNGEN_S setzt die Ausgangssignale QLMNUP und QLMNDN.

Neustart

Bei Neustart wird jeder Einzelbaustein aufgerufen. Einzelbausteine mit Neustart-routine werden in ihrer Neustart-routine aufgerufen.

3.2.2 PROC_S: Regelstrecke für Schrittreger

Anwendungsbereich

Der Baustein PROC_S simuliert ein integrierendes Stellventil mit einer Verzögerungsstrecke 3. Ordnung.

Bild 3-4 zeigt das Blockschaltbild des PROC_S.

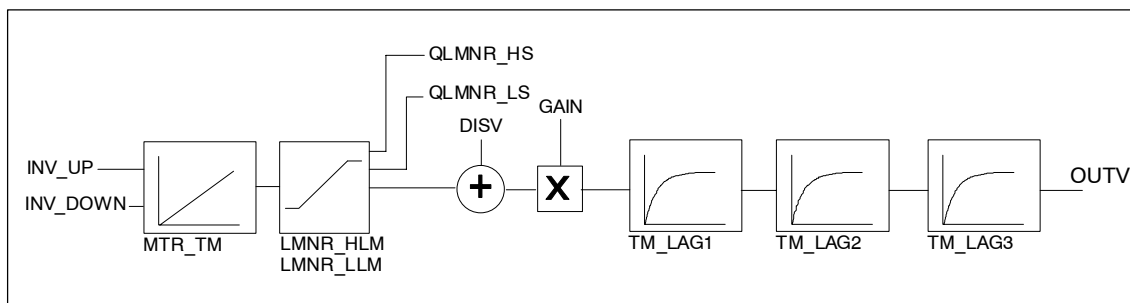


Bild 3-4 Blockschaltbild des PROC_S

Funktionsbeschreibung

Der Baustein bildet eine Reihenschaltung von einem integrierenden Stellventil und drei Verzögerungsgliedern 1. Ordnung nach. Zum Ausgang des Stellventils wird immer die Störgröße **DISV** hinzuaddiert. Die Motorstellzeit **MTR_TM** ist die Zeit, die das Ventil von Anschlag zu Anschlag benötigt.

Neustart

Bei Neustart werden sowohl die Ausgangsgröße **OUTV** als auch die internen Speichergrößen alle auf 0 gesetzt.

3.3 Beispiel 2: Festwertregler mit kontinuierlichem Ausgang mit Streckensimulation

Übersicht

Beispiel 2 hat den Namen EXAMPLE02 und besteht aus einem kontinuierlichen PID-Regler und einer simulierten Regelstrecke.

Regelkreis

Bild 3-5 zeigt den kompletten Regelkreis des Beispiels 2.

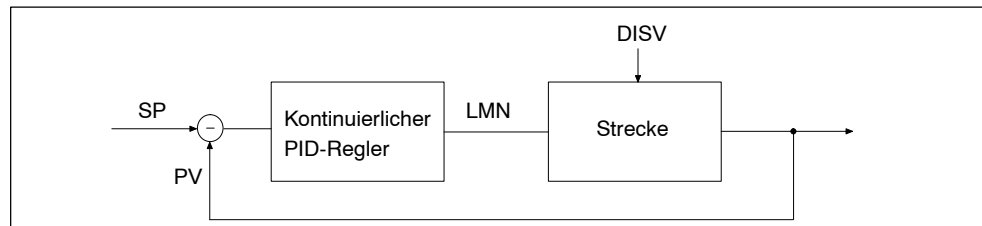


Bild 3-5 Regelkreis des Beispiels 2

Bausteinanruf und Verschaltung

Bild 3-6 zeigt den Bausteinanruf und die Verschaltung des Beispiels 2.

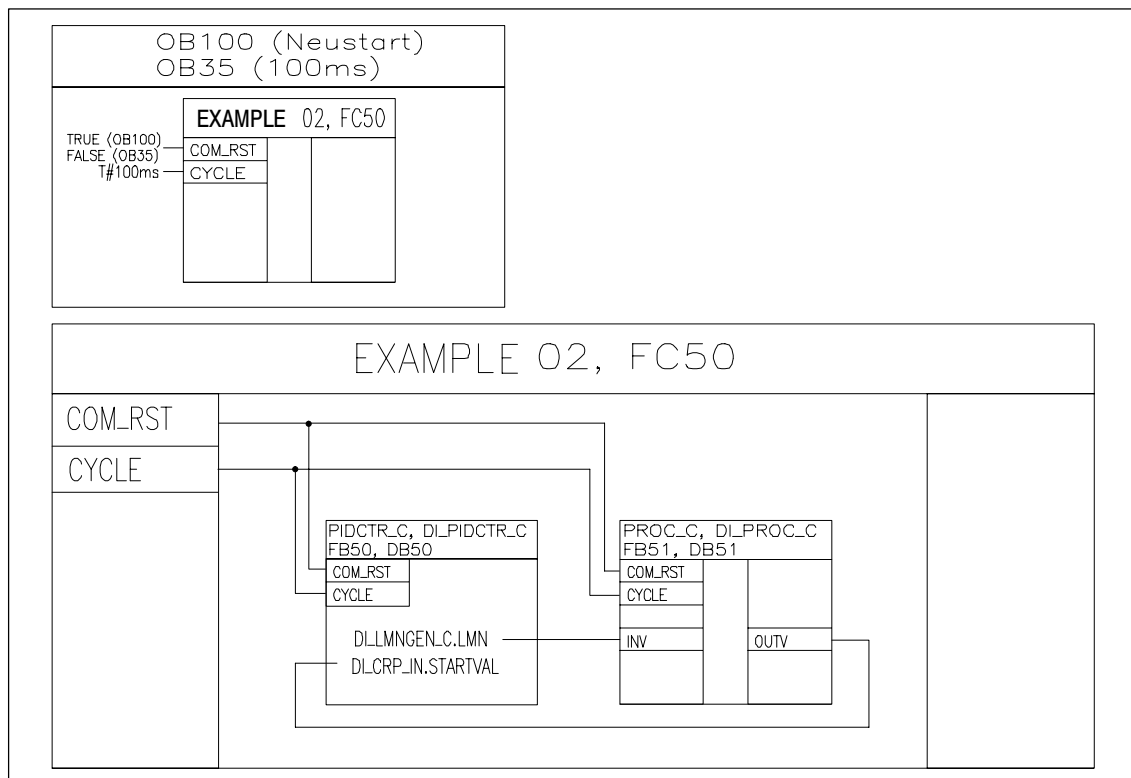


Bild 3-6 Bausteinanruf und Verschaltung des Beispiels 2

3.3.1 PIDCTR_C: Festwertregler mit kontinuierlichem Ausgang für integrierende Stellglieder

Anwendungsbereich

Der Baustein PIDCTR_C realisiert einen PID-Regler für kontinuierliche Stellglieder. Bild 3-7 zeigt die Bausteinverschaltung des PIDCTR_C.

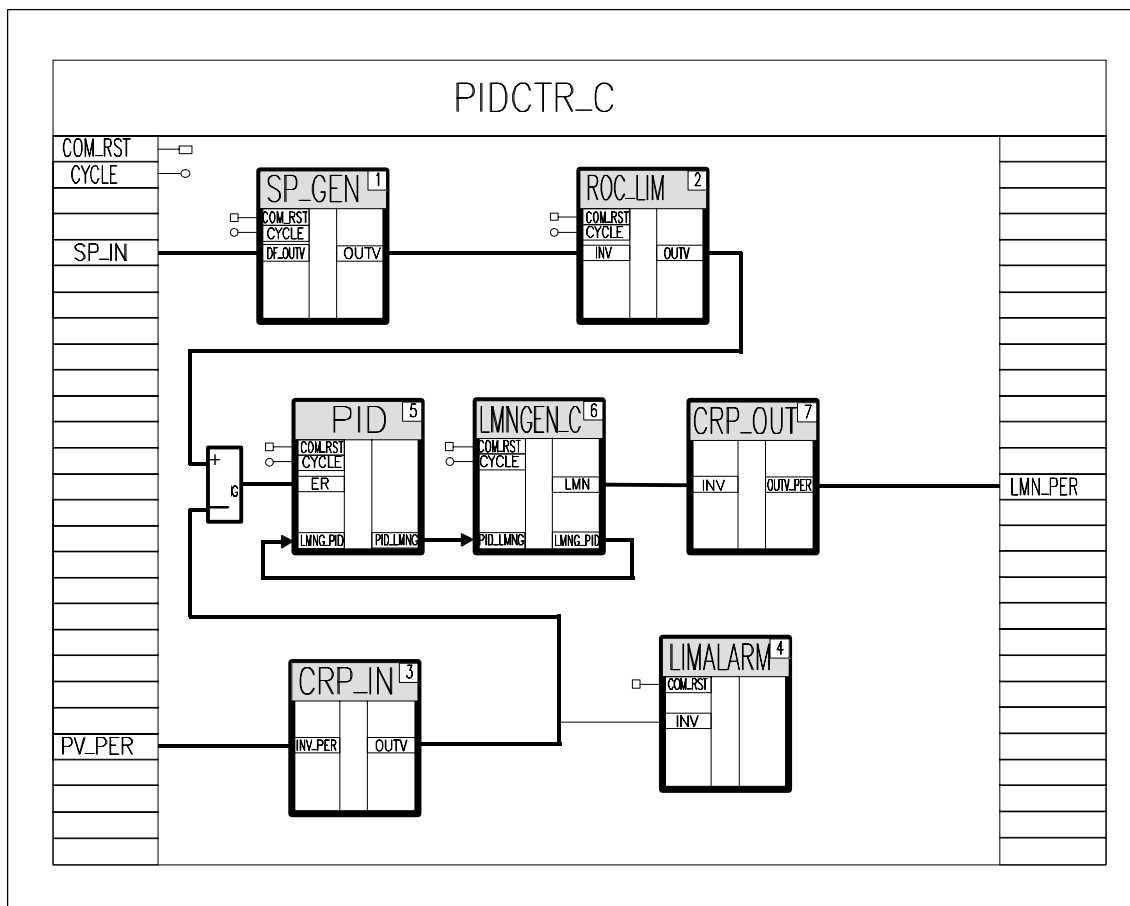


Bild 3-7 Bausteinverschaltung des PIDCTR_C

Funktionsbeschreibung

Der Sollwertgeber SP_GEN gibt den Sollwert vor, dessen Steigung vom Hochlaufgeber ROC_LIM begrenzt wird. Der Peripherie-Istwert wird mittels CRP_IN in einen Gleitpunktwert gewandelt und durch den Grenzwertmelder LIM_ALARM auf voreingestellte Grenzwerte überwacht. Die Regeldifferenz wird auf den PID-Algorithmus PID geführt. Der Stellwertverarbeitungsbaustein LMNGEN_C bildet die analoge Stellgröße LMN, die sich mit CRP_OUT in das Peripherieformat wandelt.

Neustart

Bei Neustart wird jeder Einzelbaustein aufgerufen. Einzelbausteine mit Neustartroutine werden in ihrer Neustartroutine aufgerufen.

3.3.2 PROC_C: Regelstrecke für kontinuierlichen Regler

Anwendungsbereich

Der Baustein PROC_C simuliert eine Verzögerungsstrecke 3. Ordnung.

Bild 3-8 zeigt das Blockschaltbild des PROC_C.

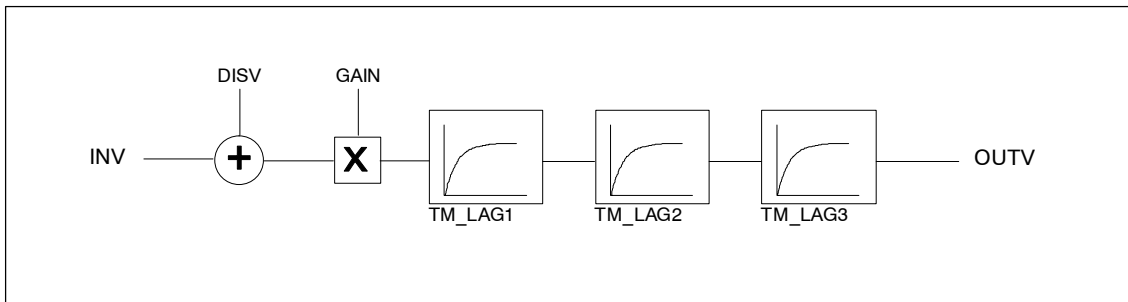


Bild 3-8 Blockschaltbild des PROC_C

Funktionsbeschreibung

Der Baustein bildet eine Reihenschaltung von 3 Verzögerungsgliedern 1. Ordnung. Zum Eingang INV wird immer die Störgröße **DISV** hinzuaddiert.

Neustart

Bei Neustart werden sowohl die Ausgangsgröße **OUTV** als auch die internen Speichergrößen alle auf 0 gesetzt.

3.4 Beispiel 3: Festwertregler mit schaltendem Ausgang für proportionale Stellglieder mit Streckensimulation

Übersicht

Beispiel 3 hat den Namen EXAMPLE03 und besteht aus einem kontinuierlichen PID-Regler mit Pulspausemodulation und einer simulierten Regelstrecke.

Regelkreis

Bild 3-9 zeigt den kompletten Regelkreis des Beispiels 3.

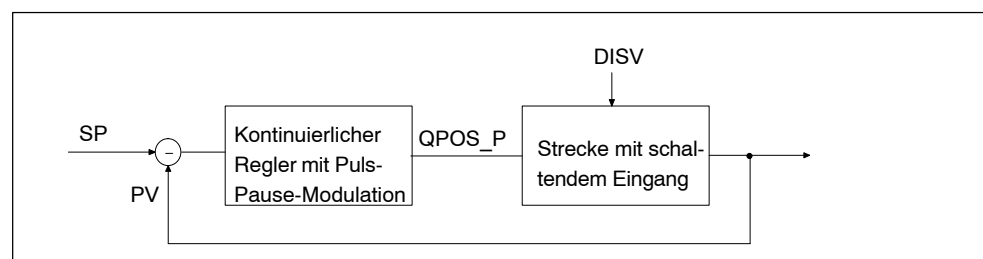


Bild 3-9 Regelkreis des Beispiels 3

Bausteinaufruf und Verschaltung

Bild 3-10 zeigt den Bausteinaufruf und die Verschaltung des Beispiels 3.

Hinweis

Die Zykluszeit des OB35 müssen Sie mittels "HW Konfig: Hardware konfigurieren" auf 10 ms einstellen.

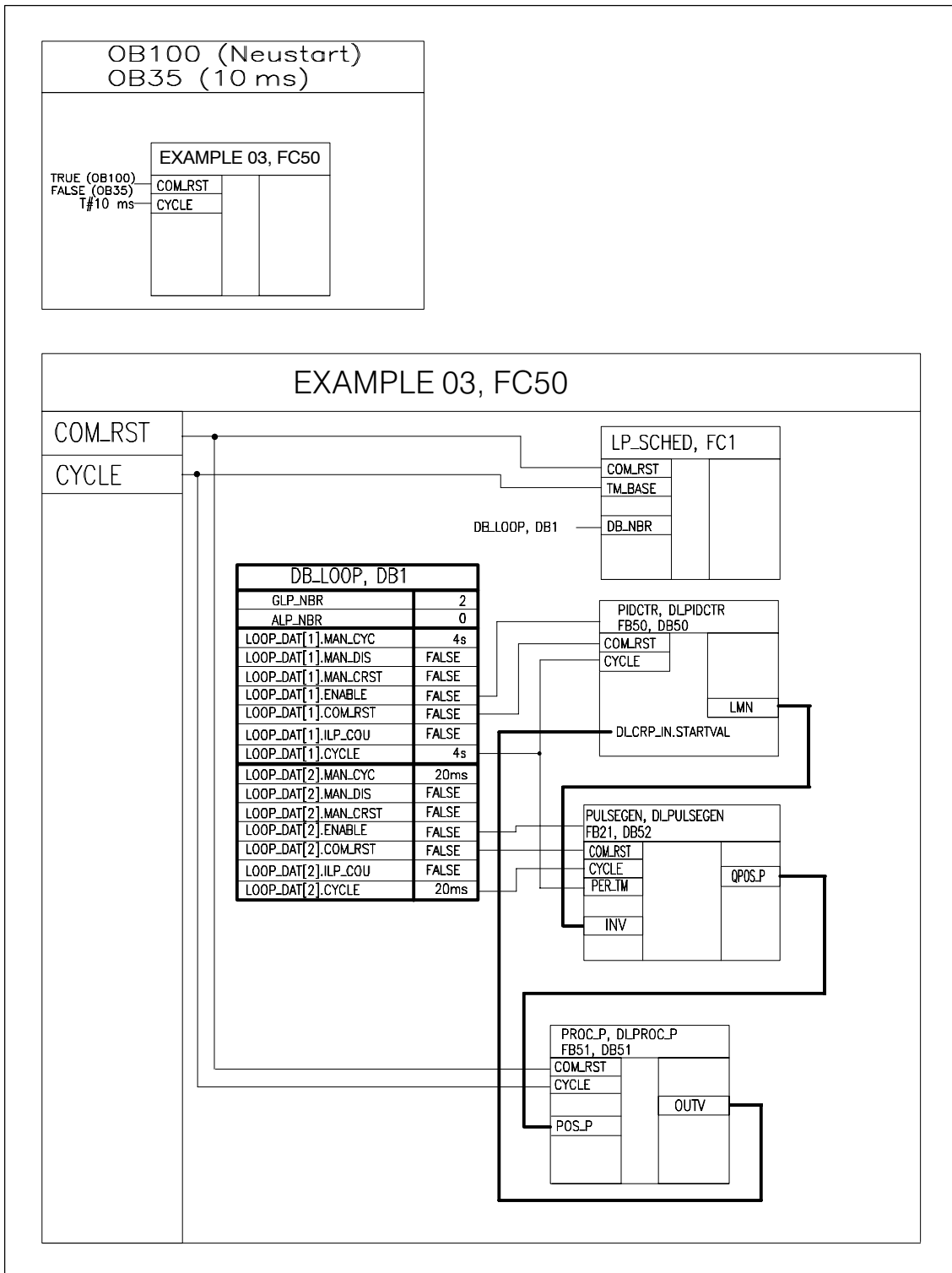


Bild 3-10 Bausteinaufruf und Verschaltung des Beispiels 3

3.4.1 PIDCTR: Führungsregler für kontinuierliche Regler mit Impulsformer

Anwendungsbereich

Der Baustein PIDCTR realisiert einen PID-Regler mit kontinuierlichem Ausgang. Er wird zur Berechnung der analogen Stellgröße innerhalb eines Puls-Pause-Reglers verwendet. Desweiteren kommt er als Führungsregler in Verhältnis-, Mischungsregelungen und in der Kaskadenregelung zum Einsatz. Bild 3–11 zeigt die Bausteinverschaltung des PIDCTR.

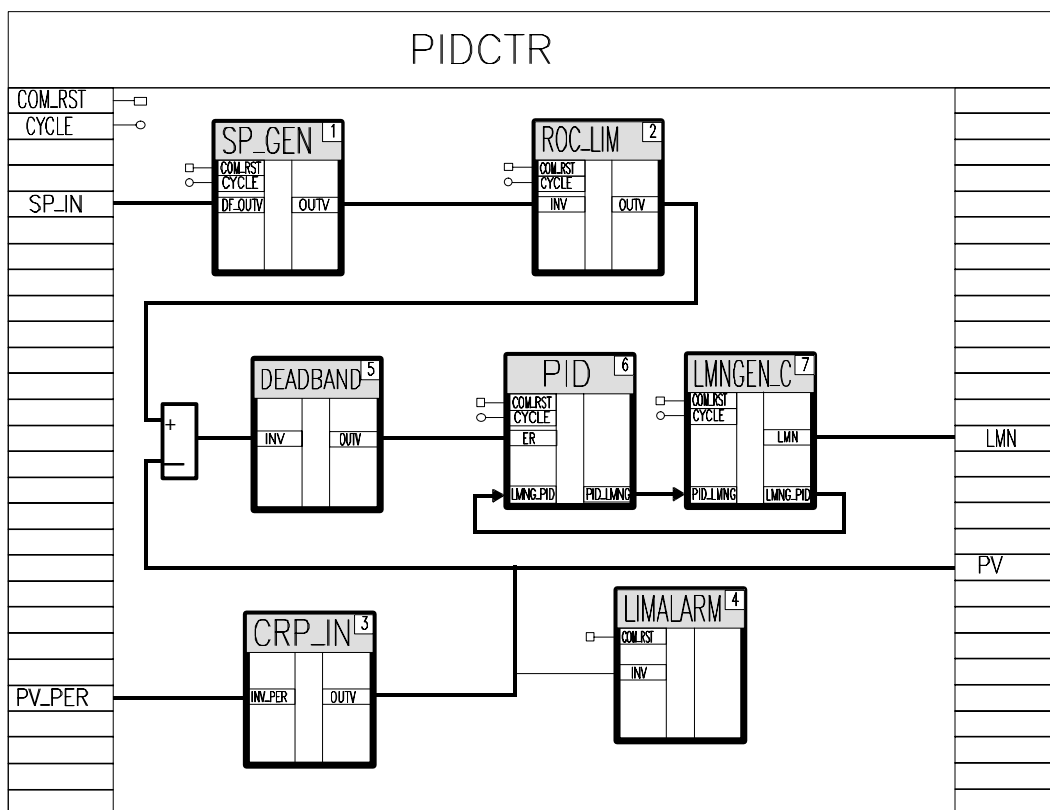


Bild 3-11 Bausteinverschaltung des PIDCTR

Funktionsbeschreibung

Der Sollwertgeber SP_GEN gibt den Sollwert vor, dessen Steigung vom Hochlaufgeber ROC_LIM begrenzt wird. Der Peripherie-Istwert wird mittels CRP_IN in einen Gleitpunktwert gewandelt und durch den Grenzwertmelder LIMALARM auf voreingestellte Grenzwerte überwacht. Die Regeldifferenz wird auf den PID-Algorithmus PID geführt. Der Stellwertverarbeitungsbaustein LMNGEN_C bildet die analoge Stellgröße LMN.

Neustart

Bei Neustart wird jeder Einzelbaustein aufgerufen. Einzelbausteine mit Neustartoutine werden in ihrer Neustartoutine aufgerufen.

3.4.2 PROC_P: Regelstrecke für kontinuierlichen Regler mit Impulsformer

Anwendungsbereich

Der Baustein PROC_P simuliert ein kontinuierliches Stellventil mit digitalem Eingang und einer Verzögerungsstrecke 3. Ordnung.

Bild 3-12 zeigt das Blockschaltbild des PROC_P.

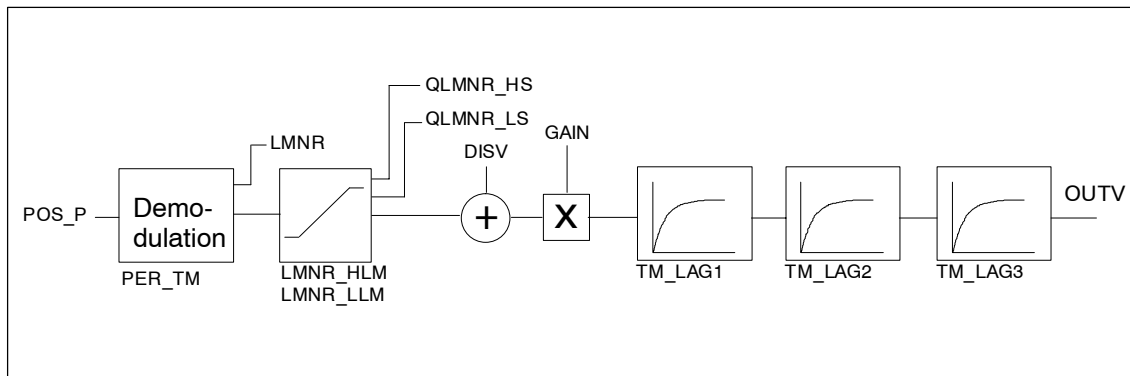


Bild 3-12 Blockschaltbild des PROC_P

Funktionsbeschreibung

Der Baustein wandelt die binären Eingangswerte der Pulsbreitenmodulation in kontinuierliche Analogwerte um und verzögert nach der Störgrößenaufschaltung das Ausgangssignal mit drei Verzögerungsgliedern 1. Ordnung.

Neustart

Bei Neustart werden sowohl die Ausgangsgröße **OUTV** als auch die internen Speichergrößen alle auf 0 gesetzt.

3.5 Beispiel 4: Einschleifiger Verhältnisregler (RATIOCTR)

Übersicht

Beispiel 4 hat den Namen EXAMPLE04 und ist ein einschleifiger Verhältnis-Regler. Eine simulierte Regelstrecke ist bei diesem Beispiel nicht enthalten.

Regelkreis

Bild 3-13 zeigt die Anwendung von Beispiel 4 in einem kompletten Regelkreis.

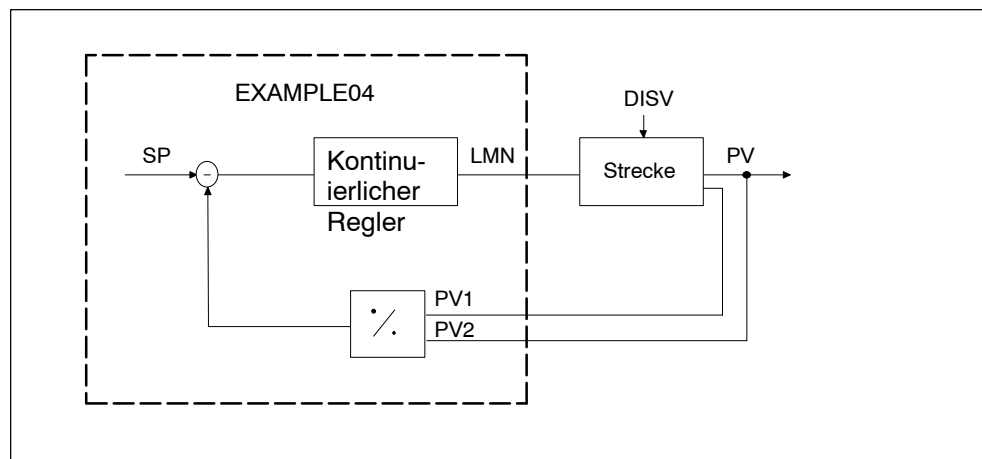


Bild 3-13 Regelkreis mit Beispiel 4

Bausteinanruf

Bild 3-14 zeigt den Bausteinanruf des Beispiels 4.

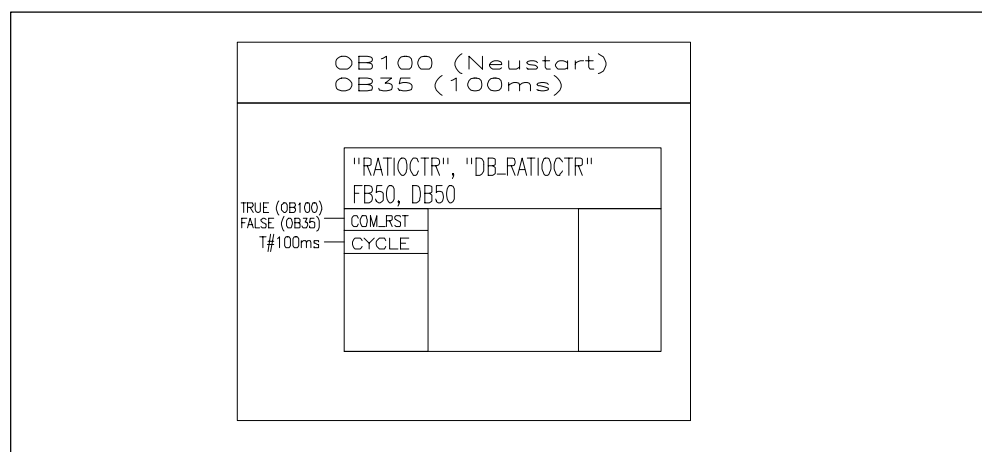


Bild 3-14 Bausteinanruf des Beispiels 4

Anwendungsbereich

Der Baustein RATIOCTR realisiert einen einschleifigen Verhältnisregler für kontinuierliche Stellglieder. Bild 3-15 zeigt die Bausteinverschaltung des RATIOCTR.

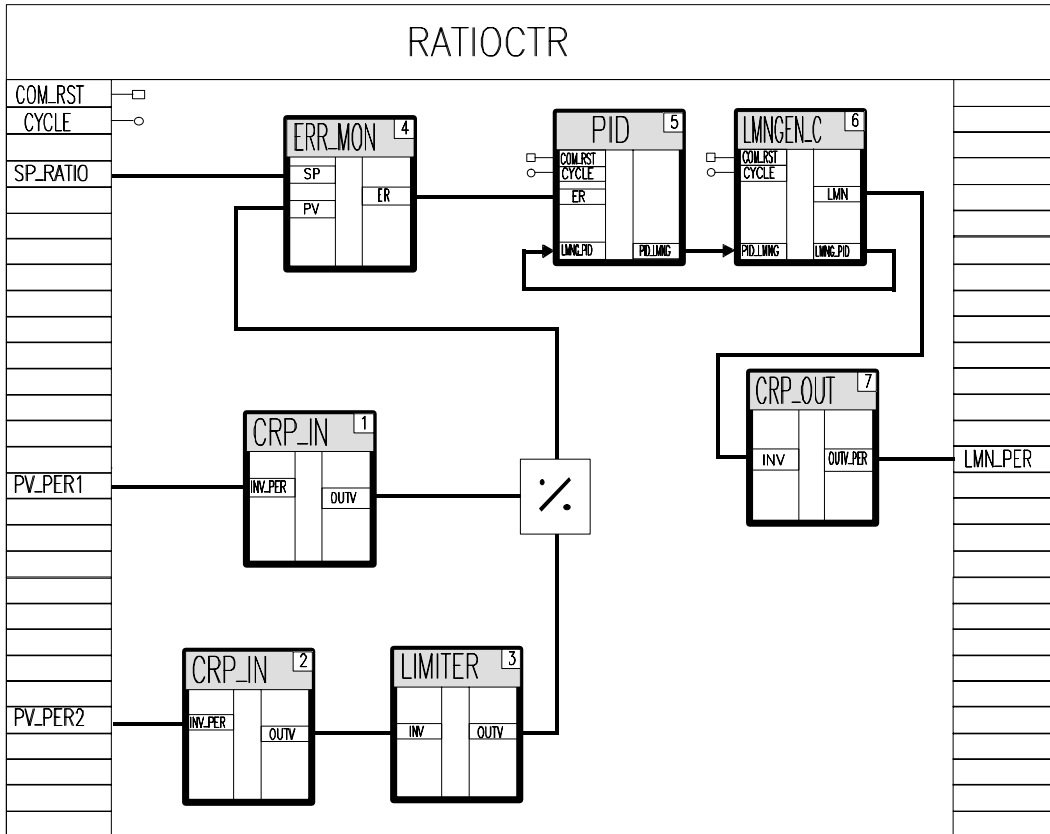


Bild 3-15 Bausteinverschaltung des RATIOCTR

Funktionsbeschreibung

Der Verhältnissollwert wird am Eingangsparameter SP_RATIO vorgegeben. Die Peripherie-Istwerte PV_PER1 und PV_PER2 werden mittels CRP_IN in Gleitpunktwerte gewandelt und das Verhältnis gebildet. Der Gleitpunktwert von PV_PER2 wird durch LIMITER so begrenzt, daß keine Division durch Null möglich ist. Die Regeldifferenz wird auf den PID-Algorithmus PID geführt. Der Stellwertverarbeitungsbaustein LMNGEN_C bildet die analoge Stellgröße LMN, die mittels CRP_OUT in das Peripherieformat gewandelt wird.

Neustart

Bei Neustart wird jeder Einzelbaustein aufgerufen. Einzelbausteine mit Neustartroutine werden in ihrer Neustartroutine aufgerufen.

3.6 Beispiel 5: Mehrschleifiger Verhältnisregler

Übersicht

Beispiel 5 hat den Namen EXAMPLE05 und ist ein mehrschleifiger Verhältnis-Regler. Es enthält einen Führungsregler und drei Folgeregler. Eine simulierte Regelstrecke ist bei diesem Beispiel nicht enthalten.

Regelkreis

Bild 3-16 zeigt die Anwendung von Beispiel 5 in einem kompletten Regelkreis.

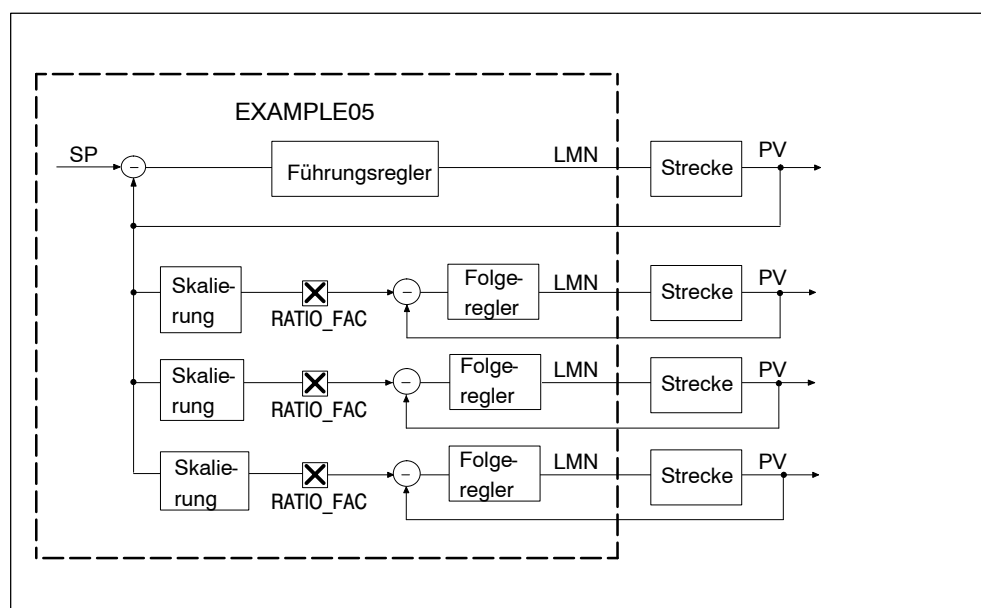


Bild 3-16 Regelkreis mit Beispiel 5

Bausteinufruf und Verschaltung

Bild 3-17 zeigt den Bausteinufruf und die Verschaltung des Beispiels 5.

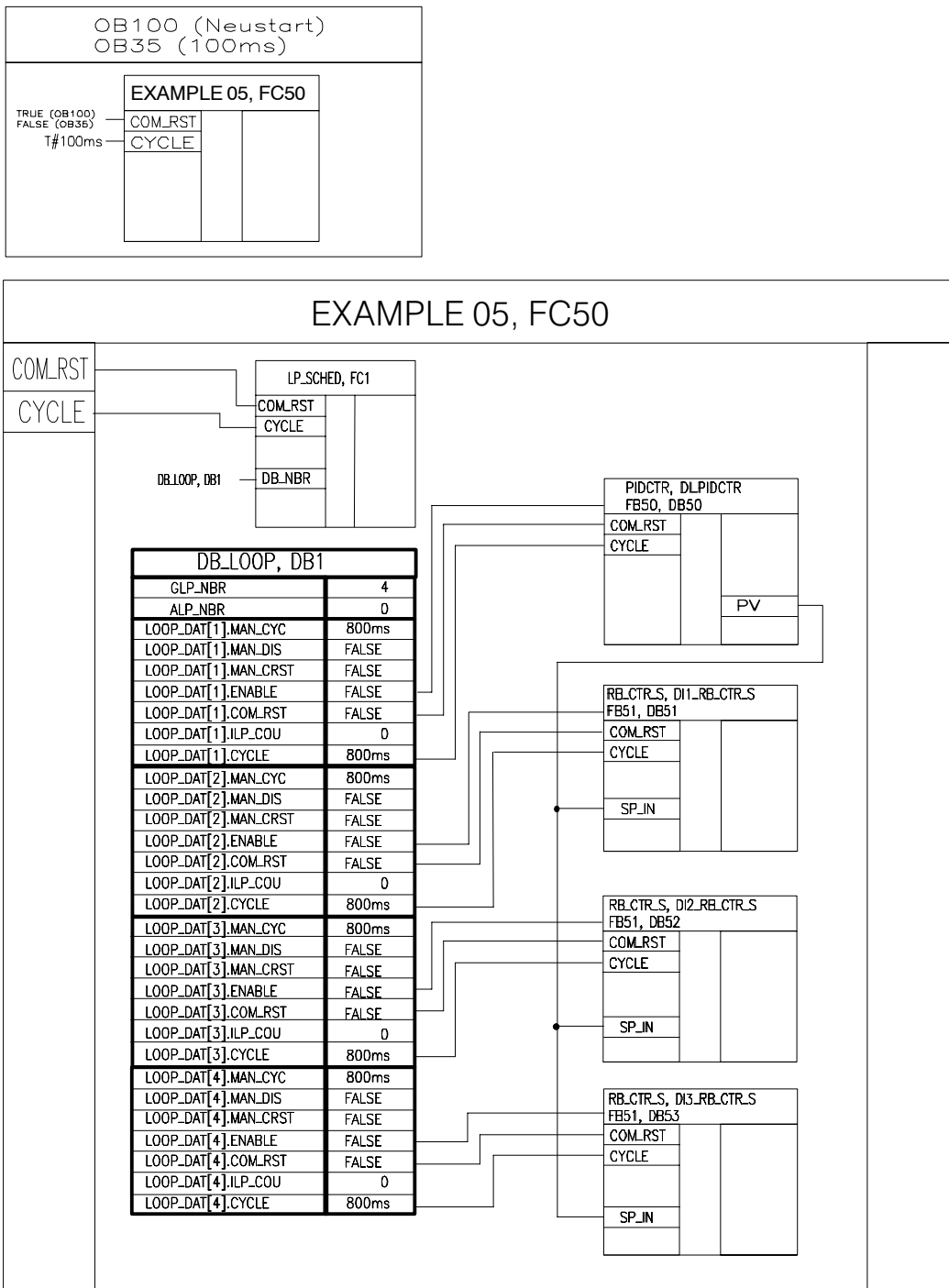


Bild 3-17 Bausteinufruf und Verschaltung des Beispiels 5

Führungsregler

Als Führungsregler wird der Baustein PIDCTR aus Beispiel 3 verwendet. Seine Funktionsweise ist in Abschnitt 3.4.1 auf Seite 3-13 beschrieben.

Folgeregler

Als Folgeregler wird der Baustein RB_CTR_S verwendet. Dieser Baustein ist ein PID-Schrittregler für integrierende Stellglieder, der als Folgeregler in einer mehrschleifigen Verhältnis- oder Mischungsregelung eingesetzt werden kann. Bild 3-18 zeigt die Bausteinverschaltung des RB_CTR_S.

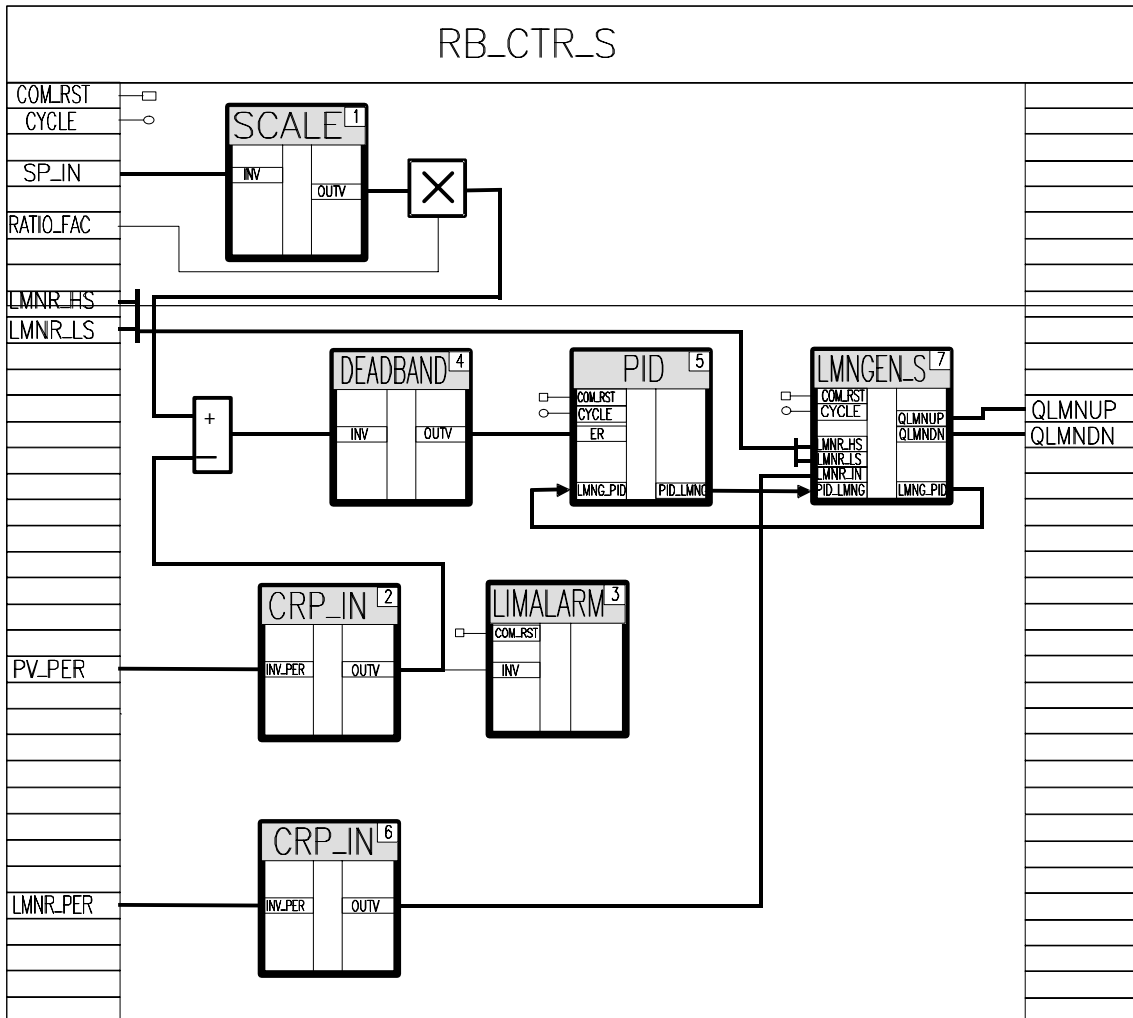


Bild 3-18 Bausteinverschaltung des RB_CTR_S

Funktionsbeschreibung des Folgereglers

Die Funktionsweise des Folgereglers RB_CTRL_S ist analog zu der des Schrittreglers PIDCTR_S aus Beispiel 1 (siehe Seite 3-4). Der Eingang des Folgereglers wird mittels einer Skalierung an den Ausgang der Regelstrecke angepasst und mit einem voreingestellten Verhältnis- bzw. Mischungsfaktor multipliziert.

Neustart

Bei Neustart wird jeder Einzelbaustein aufgerufen. Einzelbausteine mit Neustartroutine werden in ihrer Neustartroutine aufgerufen.

3.7 Beispiel 6: Mischungsregler

Übersicht

Beispiel 6 hat den Namen EXAMPLE06 und ist ein Mischungsregler. Es enthält einen Führungsregler und drei Folgeregler. Eine simulierte Regelstrecke ist bei diesem Beispiel nicht enthalten.

Regelkreis

Bild 3-19 zeigt die Anwendung von Beispiel 6 in einem kompletten Regelkreis.

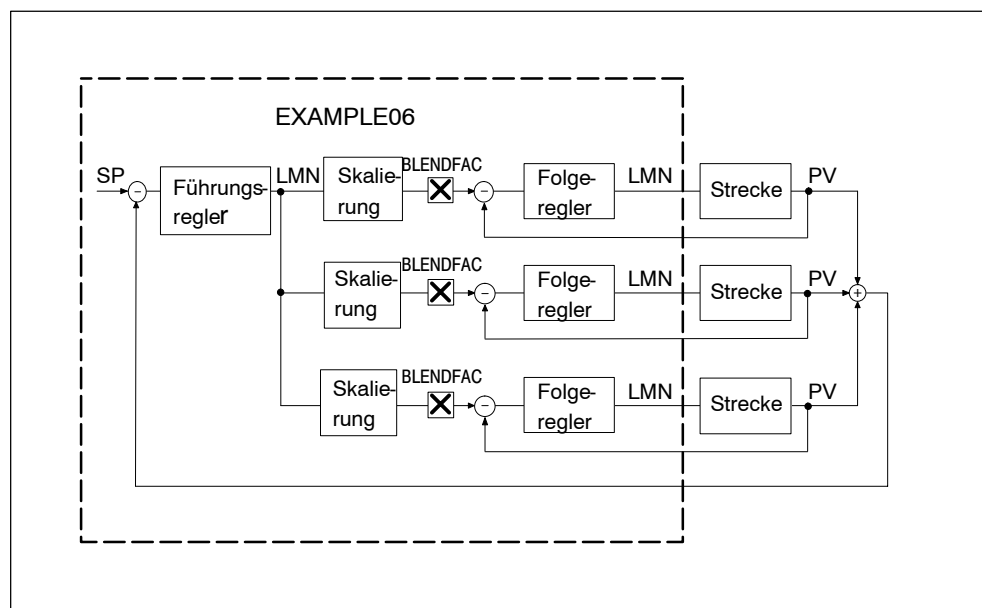


Bild 3-19 Regelkreis mit Beispiel 6

Bausteinufruf und Verschaltung

Bild 3-20 zeigt den Bausteinufruf und die Verschaltung des Beispiels 6.

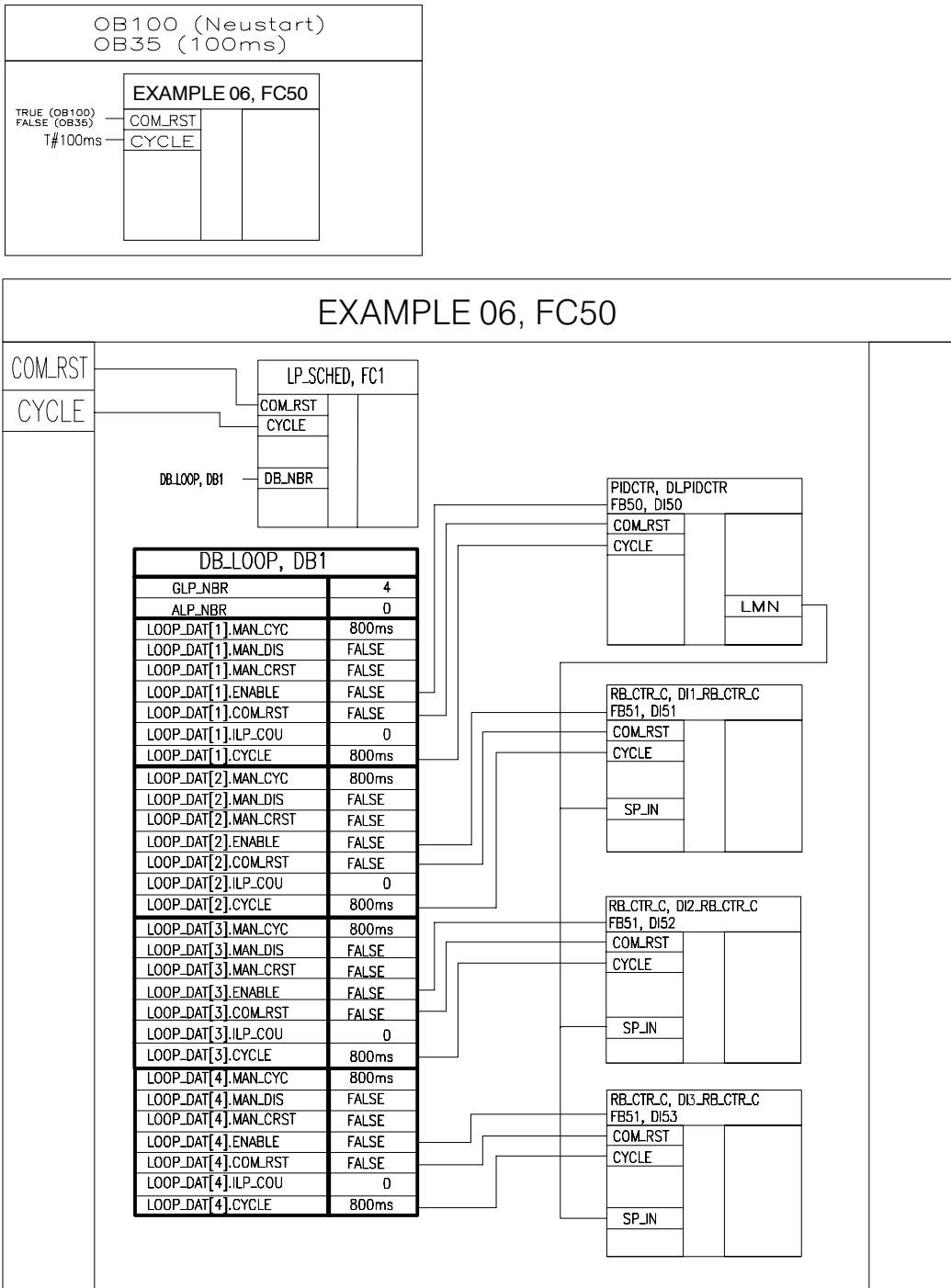


Bild 3-20 Bausteinufruf und Verschaltung des Beispiels 6

Führungsregler

Als Führungsregler wird der Baustein PIDCTR aus Beispiel 3 verwendet. Seine Funktionsweise ist in Abschnitt 3.4.1 auf Seite 3-13 beschrieben.

Folgeregler

Als Folgeregler wird der Baustein RB_CTR_C verwendet. Dieser Baustein ist ein kontinuierlicher PID-Regler, der als Folgeregler in einer mehrschleifigen Verhältnis- oder Mischungsregelung eingesetzt werden kann. Bild 3-21 zeigt die Bausteinverschaltung des RB_CTR_C.

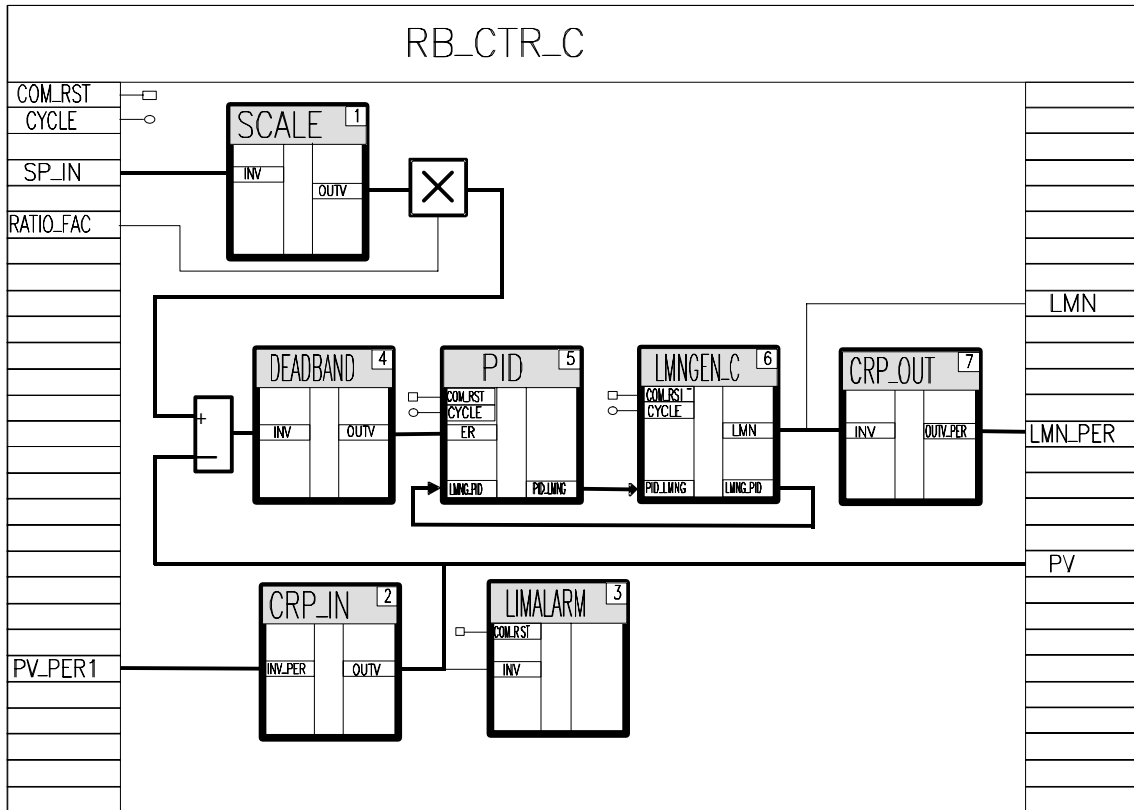


Bild 3-21 Bausteinverschaltung des RB_CTR_C

Funktionsbeschreibung des Folgereglers

Die Funktionsweise des Folgereglers RB_CTRL_C ist analog zu der des kontinuierlichen PID-Reglers PIDCTR_C aus Beispiel 2 (siehe Seite 3-8). Der Eingang des Folgereglers wird mittels einer Skalierung an den Ausgang des Führungsreglers angepaßt und mit einem voreingestellten Verhältnis- oder Mischungsfaktor multipliziert.

Neustart

Bei Neustart wird jeder Einzelbaustein aufgerufen. Einzelbausteine mit Neustartroutine werden in ihrer Neustartroutine aufgerufen.

3.8 Beispiel 7: Kaskadenregler

Übersicht

Beispiel 7 hat den Namen EXAMPLE07 und ist ein Kaskadenregler. Es enthält einen Führungsregler und einen Folgeregler. Eine simulierte Regelstrecke ist bei diesem Beispiel nicht enthalten.

Regelkreis

Bild 3-22 zeigt die Anwendung von Beispiel 7 in einem kompletten Regelkreis.

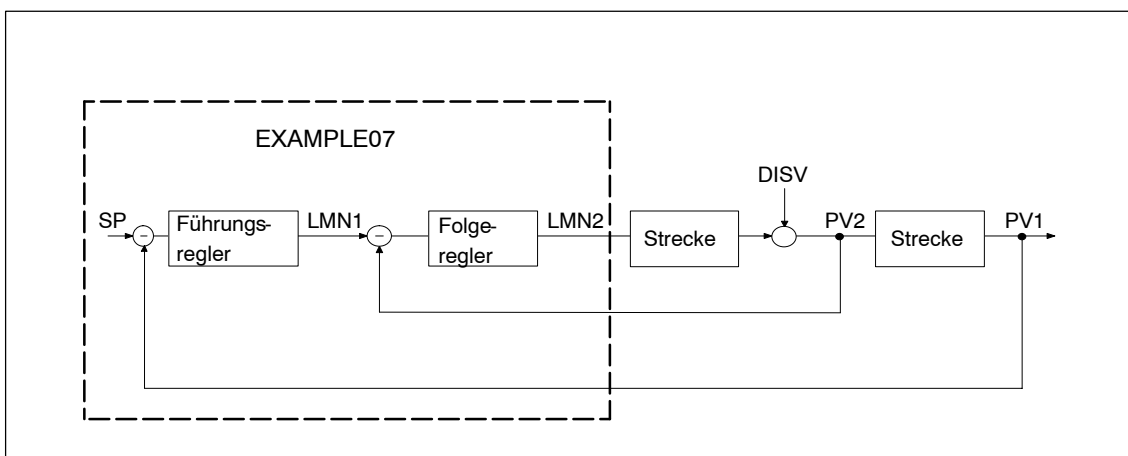


Bild 3-22 Regelkreis mit Beispiel 7

Bausteinufruf und Verschaltung

Bild 3-23 zeigt den Bausteinufruf und die Verschaltung des Beispiels 7.

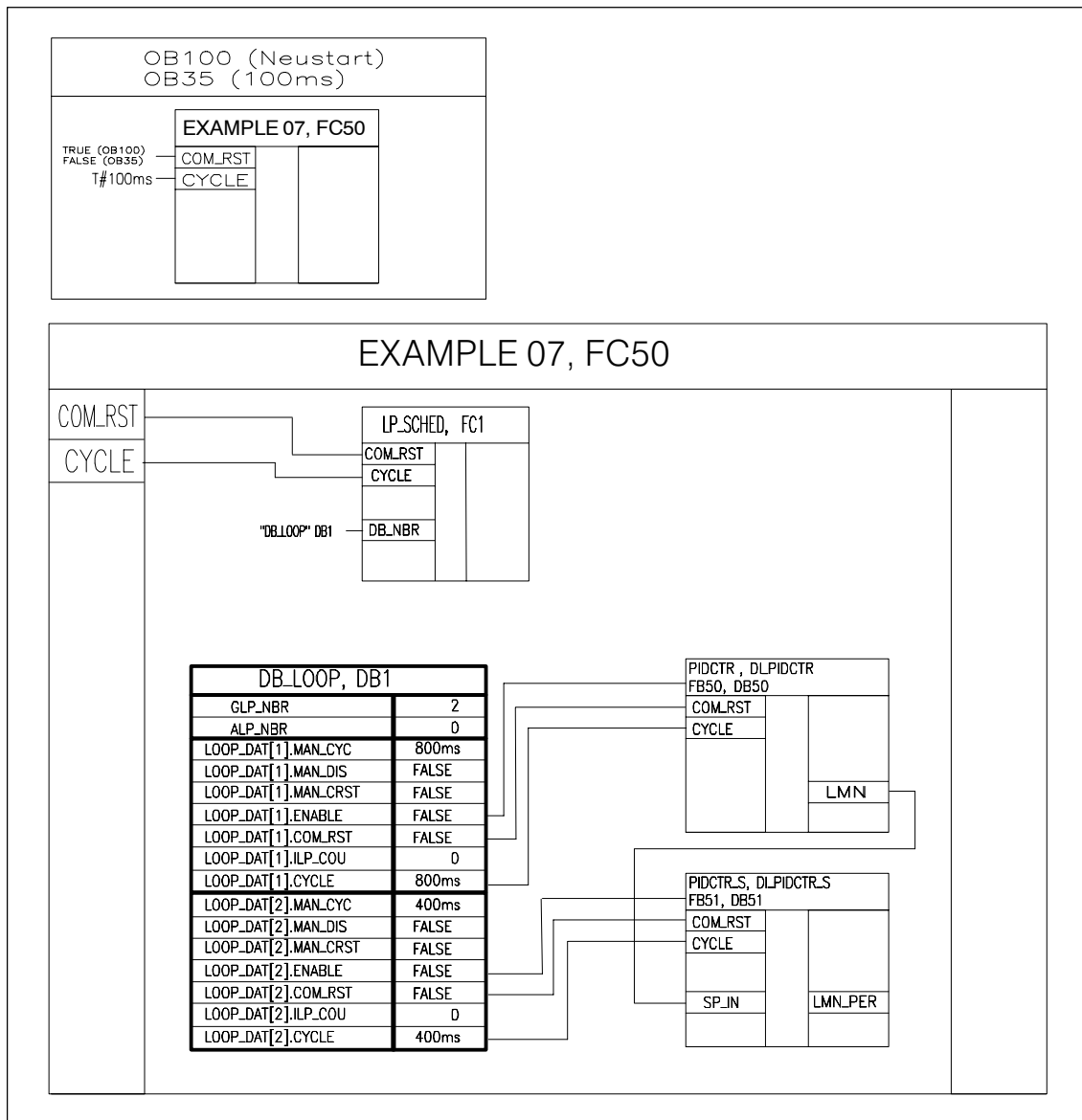


Bild 3-23 Bausteinufruf und Verschaltung des Beispiels 7

Führungsregler

Als Führungsregler wird der Baustein PIDCTR aus Beispiel 3 verwendet. Seine Funktionsweise ist in Abschnitt 3.4.1 beschrieben.

Folgeregler

Als Folgeregler wird der Baustein PIDCTR_S aus Beispiel 1 verwendet. Seine Funktionsweise ist in Abschnitt 3.2.1 auf Seite 3-6 beschrieben.

3.9 Beispiel 8: Regler mit Vorsteuerung (CTRC_PRE)

Übersicht

Beispiel 8 hat den Namen EXAMPLE08 und ist ein Regler mit Vorsteuerung. Eine simulierte Regelstrecke ist bei diesem Beispiel nicht enthalten.

Regelkreis

Bild 3-24 zeigt die Anwendung von Beispiel 8 in einem kompletten Regelkreis.

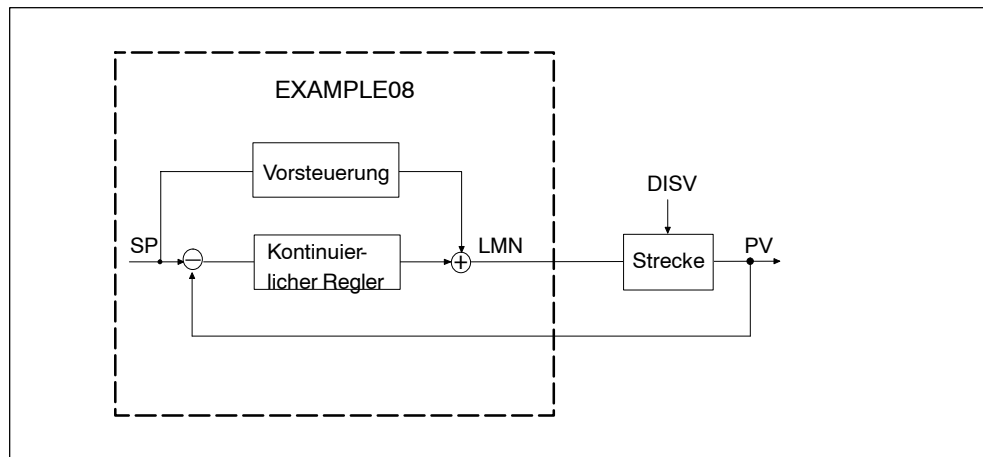


Bild 3-24 Regelkreis mit Beispiel 8

Bausteinanruf

Bild 3-25 zeigt den Bausteinanruf des Beispiels 8.

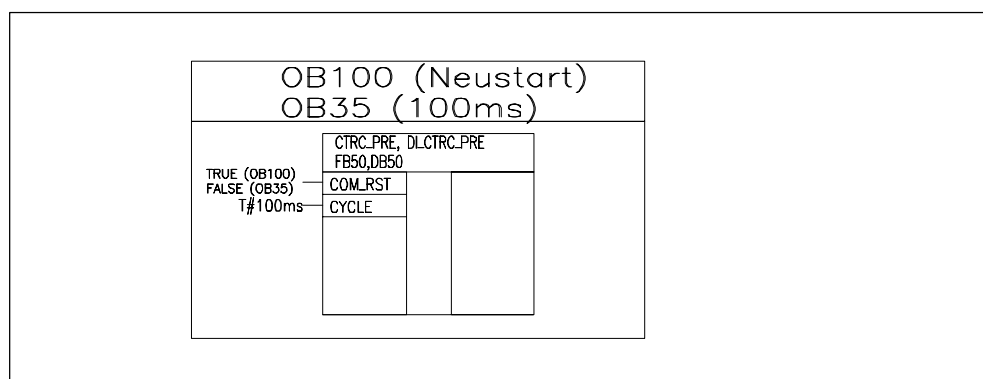


Bild 3-25 Bausteinanruf des Beispiels 8

Anwendungsbereich

Der Baustein CTRC_PRE ist ein kontinuierlicher PID-Regler mit Vorsteuerung. Bild 3-26 zeigt die Bausteinverschaltung des CTRC_PRE.

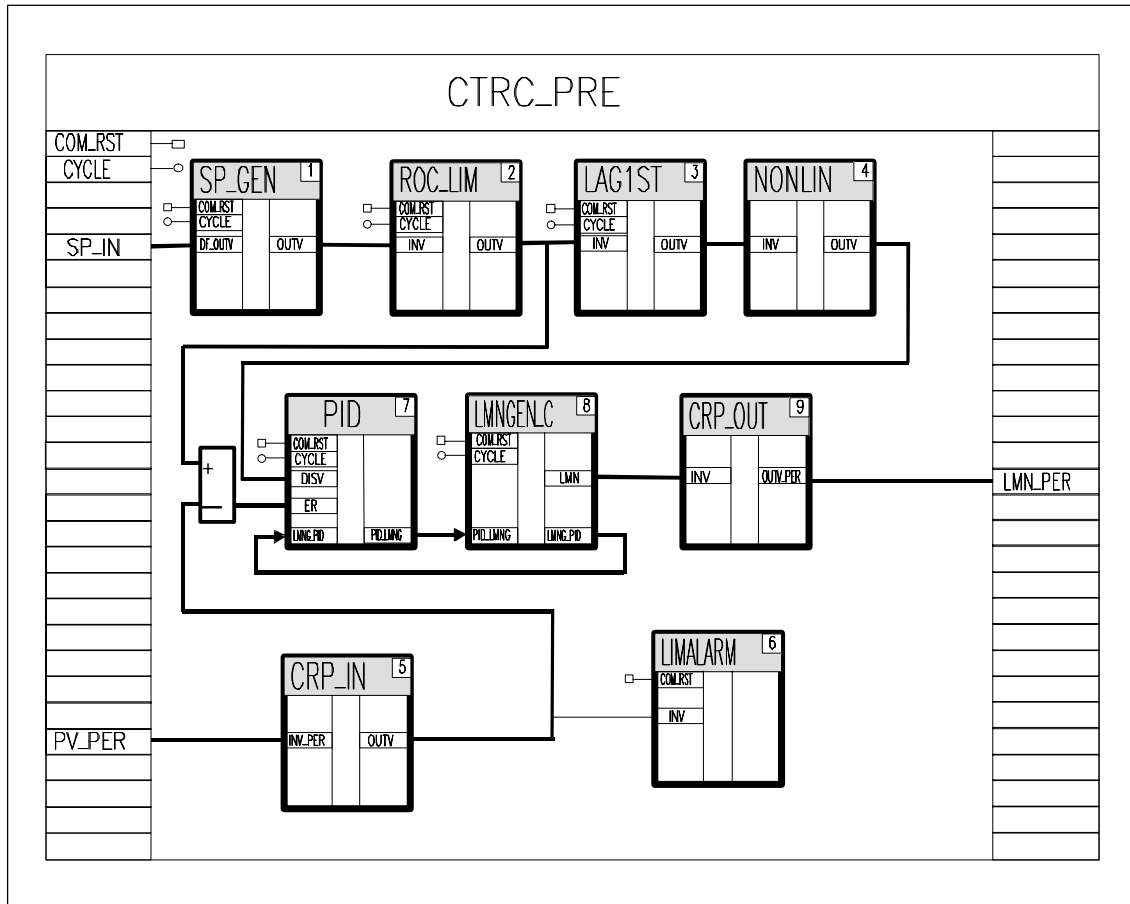


Bild 3-26 Bausteinverschaltung des CTRC_PRE

Funktionsbeschreibung

Die Funktionsweise des Reglers mit Vorsteuerung ist analog zu der des Festwertreglers mit kontinuierlichem Ausgang PIDCTR_C aus Beispiel 2. Die Vorsteuerung besteht aus einem Verzögerungsglied 1. Ordnung mit nichtlinearer statischer Kennlinie parallel zum PID-Algorithmus.

Neustart

Bei Neustart wird jeder Einzelbaustein aufgerufen. Einzelbausteine mit Neustartroutine werden in ihrer Neustartroutine aufgerufen.

3.10 Beispiel 9: Regler mit Störgrößenaufschaltung (CTR_C_FF)

Übersicht

Beispiel 9 hat den Namen EXAMPLE09 und ist ein Regler mit Störgrößenaufschaltung. Eine simulierte Regelstrecke ist bei diesem Beispiel nicht enthalten.

Regelkreis

Bild 3-27 zeigt die Anwendung von Beispiel 9 in einem kompletten Regelkreis.

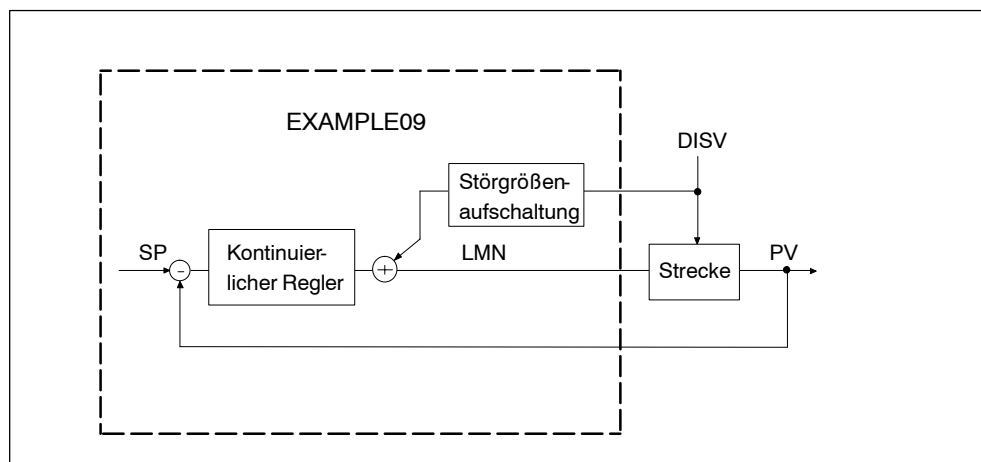


Bild 3-27 Regelkreis mit Beispiel 9

Bausteinanruf

Bild 3-28 zeigt den Bausteinanruf des Beispiels 9.

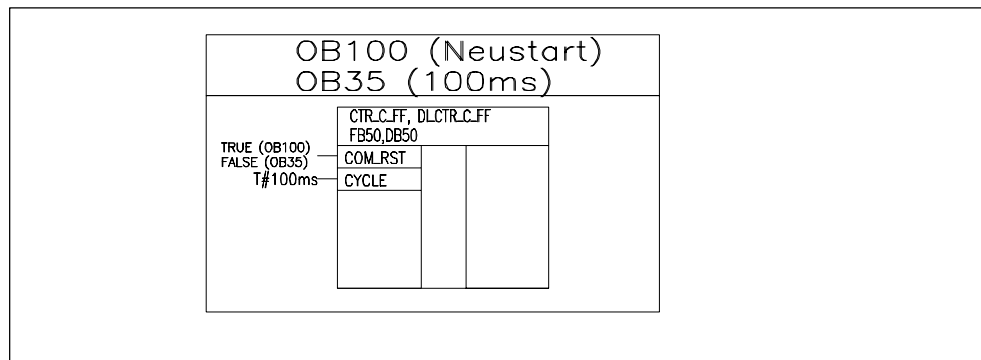


Bild 3-28 Bausteinanruf des Beispiels 9

Anwendungsbereich

Der Baustein CRT_C_FF ist ein PID-Regler mit Störgrößenaufschaltung für kontinuierliche Stellglieder. Bild 3-29 zeigt die Bausteinverschaltung des CTR_C_FF.

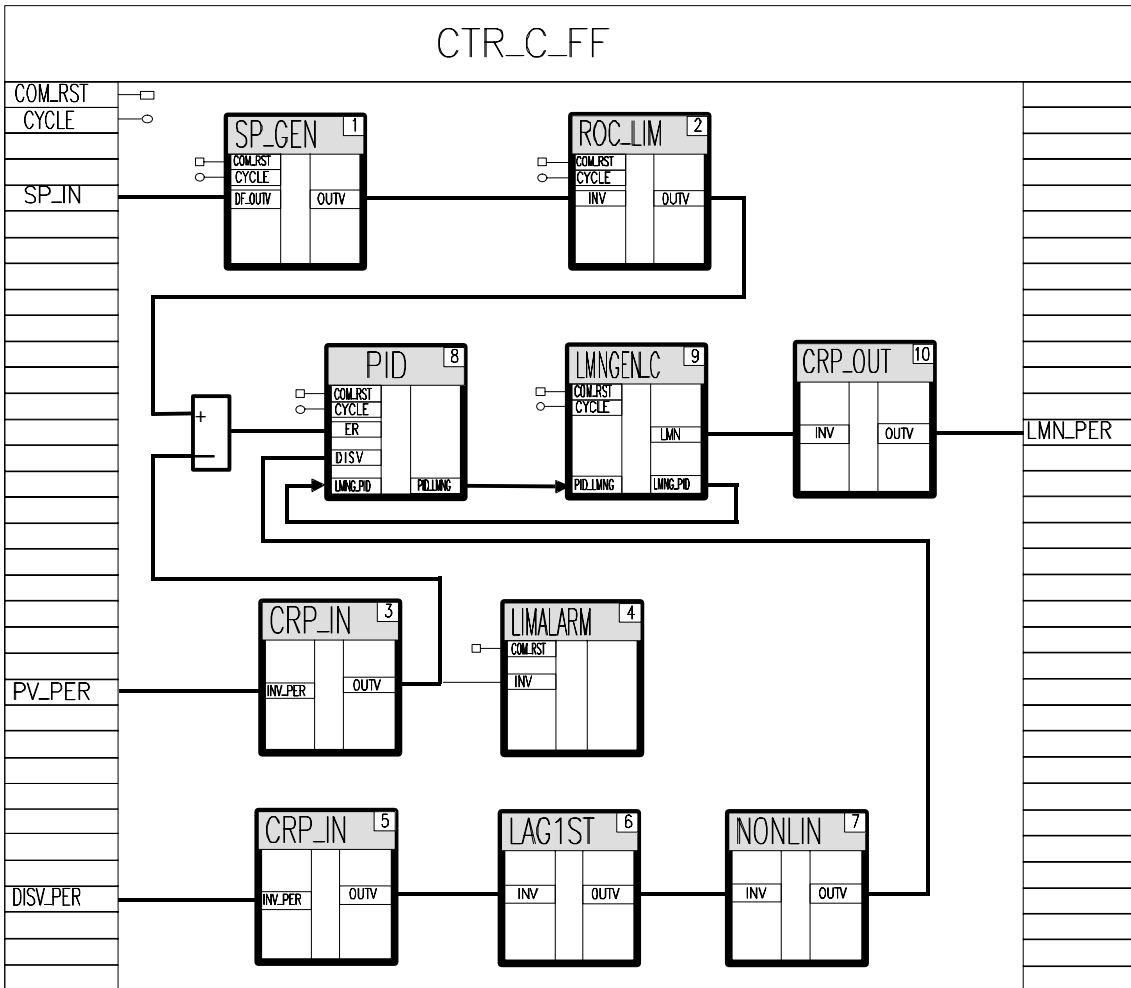


Bild 3-29 Bausteinverschaltung des CTR_C_FF

Funktionsbeschreibung

Der Sollwertgeber SP_GEN gibt den Sollwert vor, dessen Steigung vom Hochlaufgeber ROC_LIM begrenzt wird. Der Peripherie-Istwert wird mittels CRP_IN in einen Gleitpunktwert gewandelt und durch den Grenzwertmelder LIM_ALARM auf voreingestellte Grenzwerte überwacht. Die Regeldifferenz wird auf den PID-Algorithmus PID geführt. Die Peripherie-Störgröße wird mittels CRP_IN in einen Gleitpunktwert gewandelt, mit LAG1ST geglättet und mit NONLIN über eine Kennlinie linearisiert. Der Stellwertverarbeitungsbaustein LMNGEN_C bildet die analoge Stellgröße LMN, die mittels CRP_OUT in das Peripherieformat gewandelt wird.

Neustart

Bei Neustart wird jeder Einzelbaustein aufgerufen. Einzelbausteine mit Neustartroutine werden in ihrer Neustartroutine aufgerufen.

3.11 Beispiel 10: Bereichsauswahlregler (SPLITCTR)

Übersicht

Beispiel 10 hat den Namen EXAMPLE10 und ist ein Bereichsauswahlregler. Eine simulierte Regelstrecke ist bei diesem Beispiel nicht enthalten.

Regelkreis

Bild 3-30 zeigt die Anwendung von Beispiel 10 in einem kompletten Regelkreis.

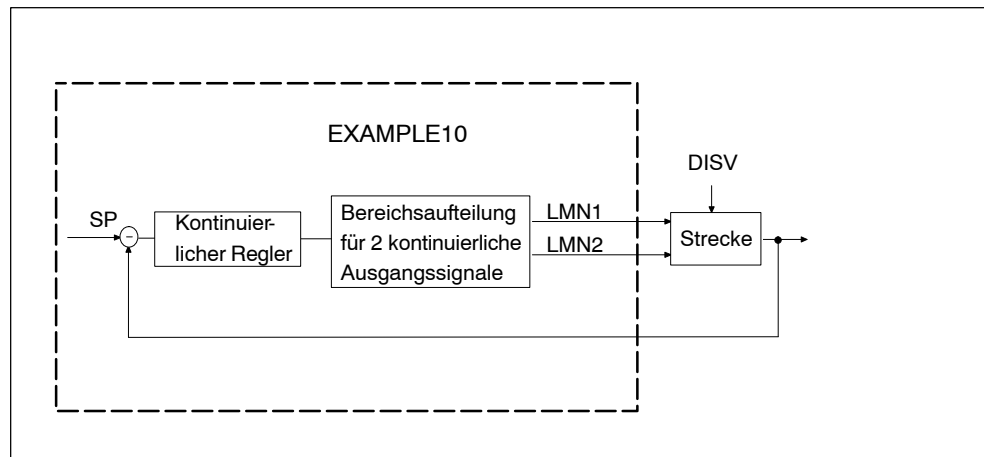


Bild 3-30 Regelkreis mit Beispiel 10

Bausteinanruf

Bild 3-31 zeigt den Bausteinanruf des Beispiels 10.

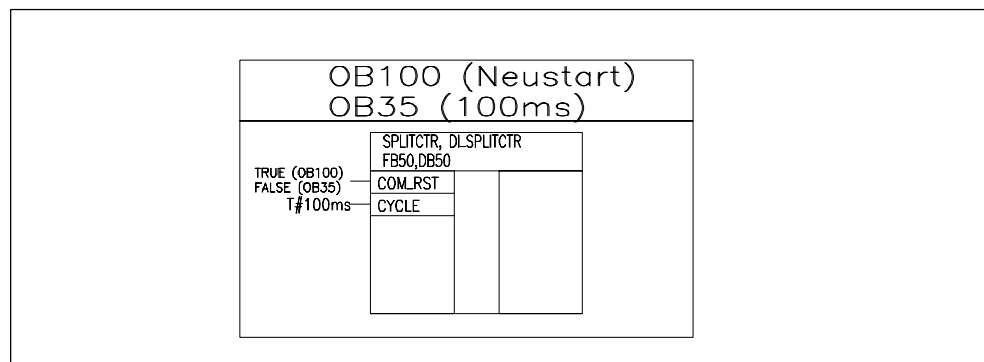


Bild 3-31 Bausteinanruf des Beispiels 10

Anwendungsbereich

Der Baustein SPLITCTR ist ein PID-Regler mit Bereichsauswahl für 2 kontinuierliche Stellglieder. Bild 3-32 zeigt die Bausteinverschaltung des SPLITCTR.

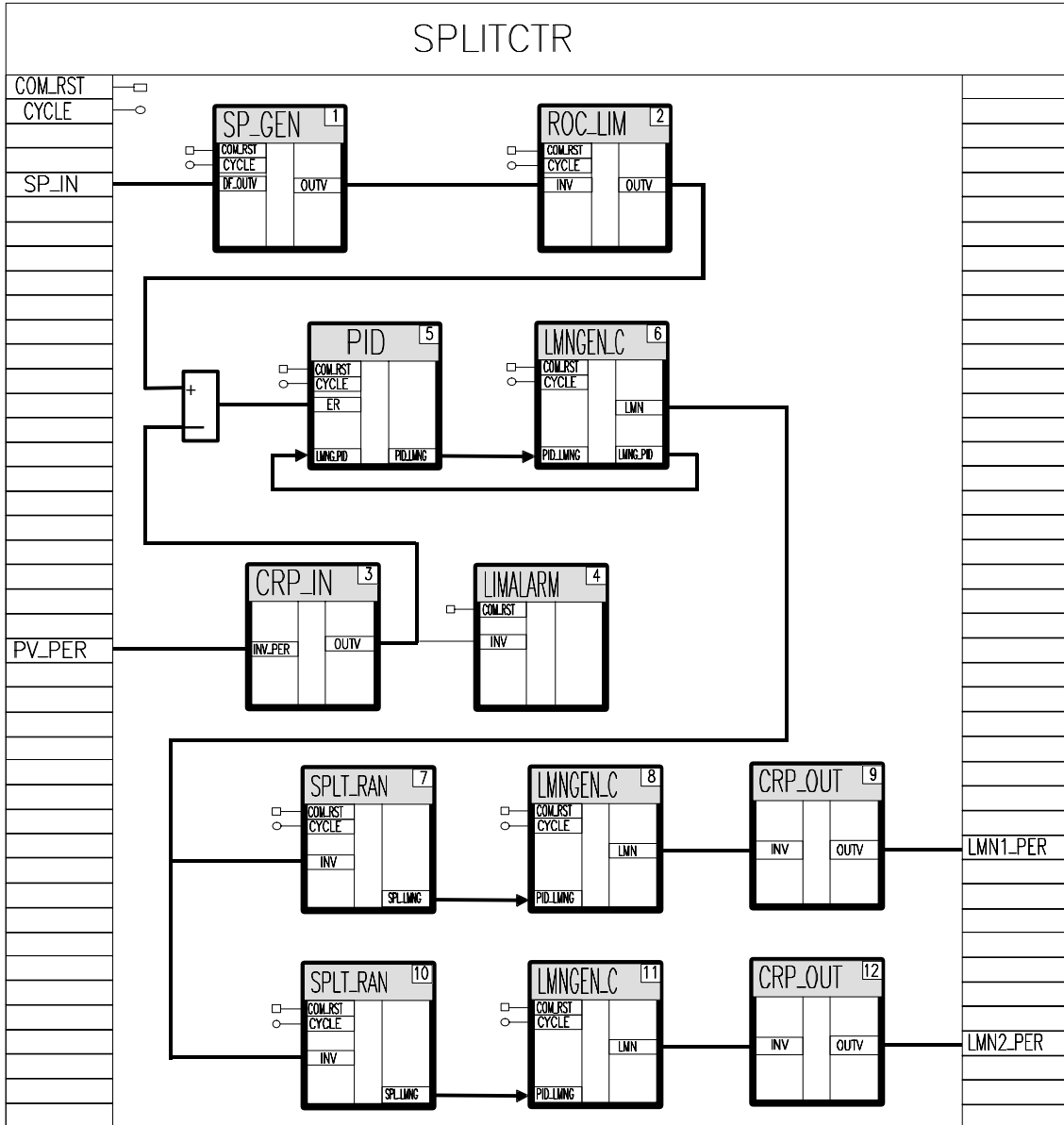


Bild 3-32 Bausteinverschaltung des SPLITCTR

Funktionsbeschreibung

Der Sollwertgeber SP_GEN gibt den Sollwert vor, dessen Steigung vom Hochlaufgeber ROC_LIM begrenzt wird. Der Peripherie-Istwert wird mittels CRP_IN in einen Gleitpunktwert gewandelt und durch den Grenzwertmelder LIMALARM auf voreingestellte Grenzwerte überwacht. Die Regeldifferenz wird auf den PID-Algorithmus PID geführt. Der Stellwertverarbeitungsbaustein LMNGEN_C bildet die analoge Stellgröße LMN. Der Stellwertbereich wird mittels 2 SPLT_RAN Bausteinen in 2 Bereiche zerlegt. Für jeden Bereich wird mit LMNGEN_C eine analoge Stellgröße ermittelt und mit CRP_OUT in das Peripherieformat gewandelt.

Neustart

Bei Neustart wird jeder Einzelbaustein aufgerufen. Einzelbausteine mit Neustartroutine werden in ihrer Neustartroutine aufgerufen.

3.12 Beispiel 11: Ablöseregler (OVR_CTR)

Übersicht

Beispiel 11 hat den Namen EXAMPLE11 und ist ein Ablöseregler. Eine simulierte Regelstrecke ist bei diesem Beispiel nicht enthalten.

Regelkreis

Bild 3-33 zeigt die Anwendung von Beispiel 11 in einem kompletten Regelkreis.

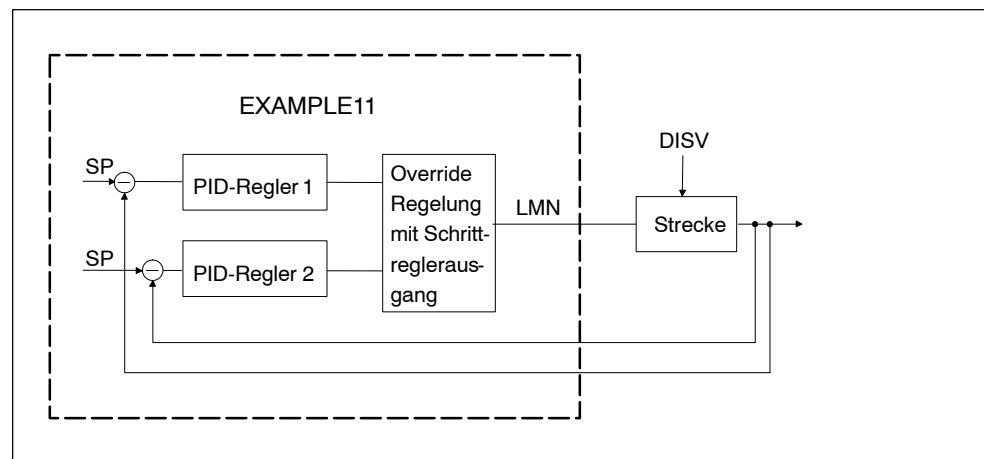


Bild 3-33 Regelkreis mit Beispiel 11

Bausteinanruf

Bild 3-34 zeigt den Bausteinanruf des Beispiels 11.

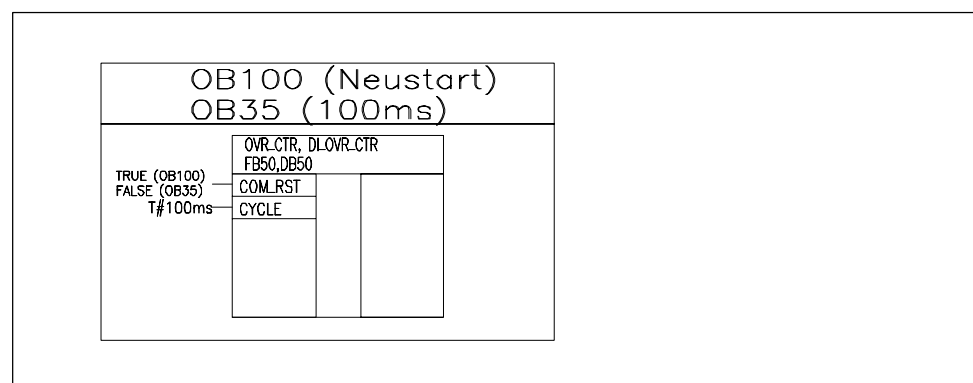


Bild 3-34 Bausteinanruf des Beispiels 11

Anwendungsbereich

Der Baustein OVR_CTR ist ein Ablöseregeler. Zwei PID-Regler werden auf einen Schritreglerausgang geschaltet. Bild 3-35 zeigt die Bausteinverschaltung des OVR_CTR.

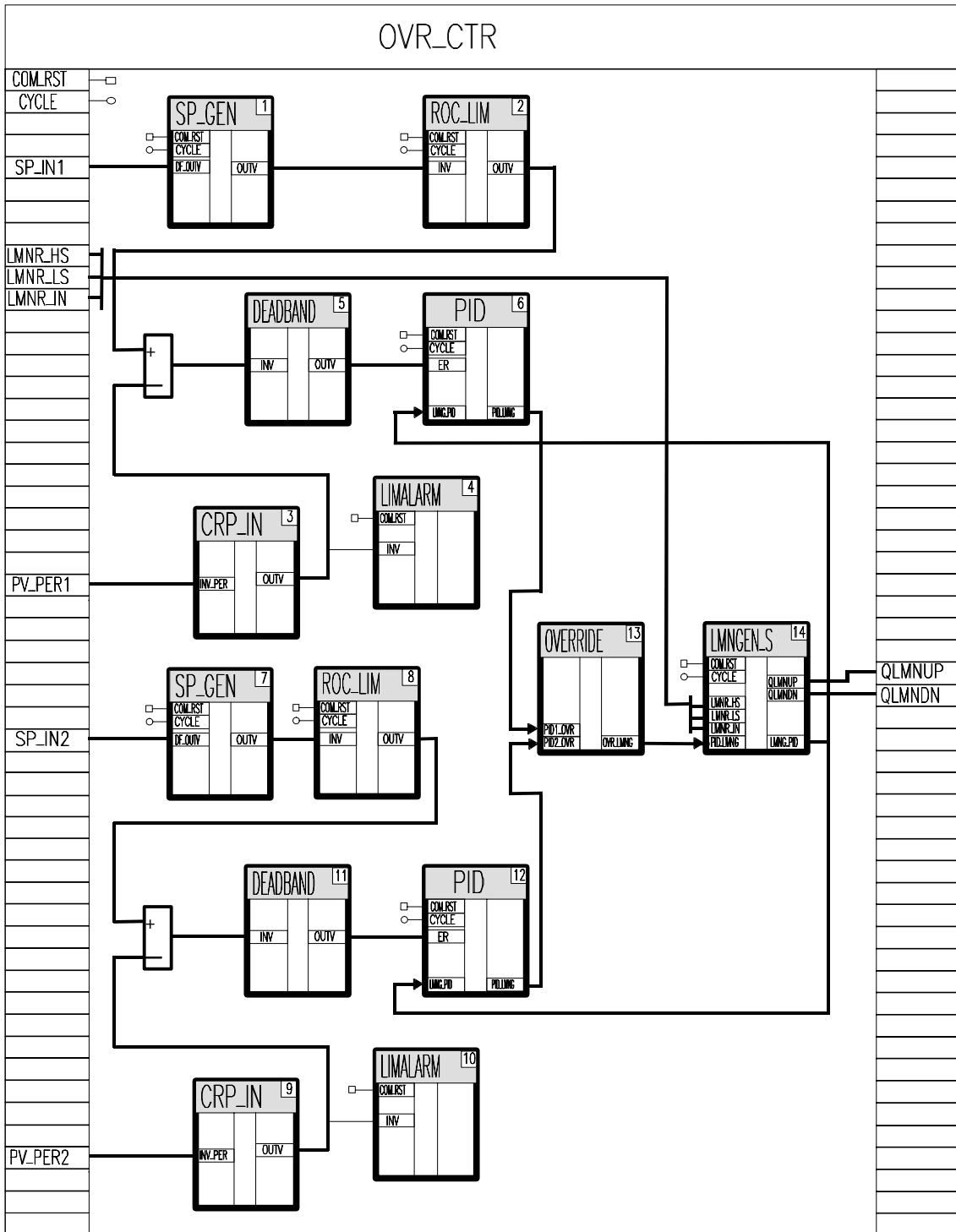


Bild 3-35 Bausteinverschaltung des OVR_CTR

Funktionsbeschreibung

Die Sollwertgeber SP_GEN geben die Sollwerte vor, deren Steigungen von den Hochlaufgebern ROC_LIM begrenzt werden. Die Peripherie-Istwerte werden mittels CRP_IN-Bausteinen in Gleitpunktwerte gewandelt und durch die Grenzwertmelder LIMALARM auf voreingestellte Grenzwerte überwacht. Die Regeldifferenzen werden auf die PID-Bausteine geführt. Die Stellwerte der zwei PID-Bausteine gelangen auf einen OVERRIDE-Baustein. Hier wird entweder das Maximum oder das Minimum der zwei Stellwerte bestimmt und auf den Stellwertverarbeitungsbaustein LMNGEN_S weitergeleitet. LMNGEN_S kann im Ablöseregler nur in der Betriebsart "Schrittregler mit Stellungsrückmeldung" arbeiten.

Neustart

Bei Neustart wird jeder Einzelbaustein aufgerufen. Einzelbausteine mit Neustartroutine werden in ihrer Neustartroutine aufgerufen.

3.13 Beispiel 12: Mehrgrößenregler

Übersicht

Beispiel 12 hat den Namen EXAMPLE12 und ist ein Ablöseregler. Eine simulierte Regelstrecke ist bei diesem Beispiel nicht enthalten.

Regelkreis

Bild 3-36 zeigt die Anwendung von Beispiel 12 in einem kompletten Regelkreis.

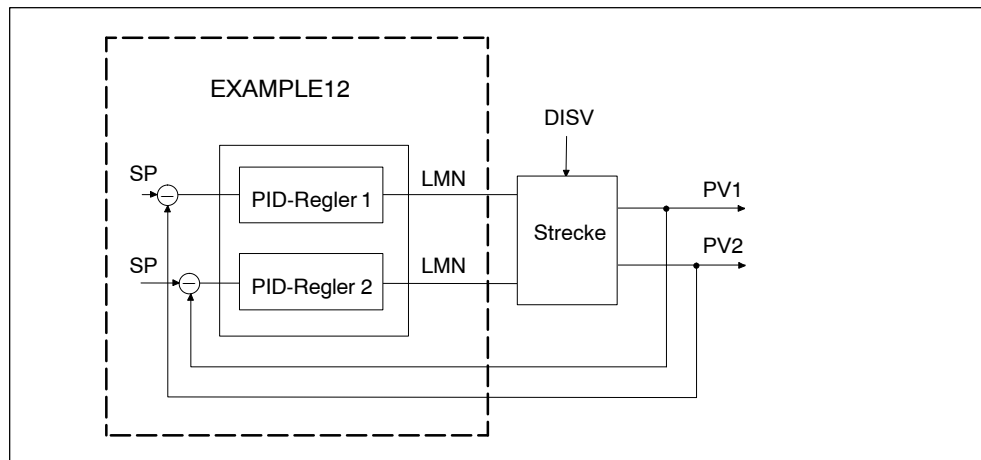


Bild 3-36 Regelkreis mit Beispiel 12

Bausteinanruf

Bild 3-37 zeigt den Bausteinanruf des Beispiels 12.

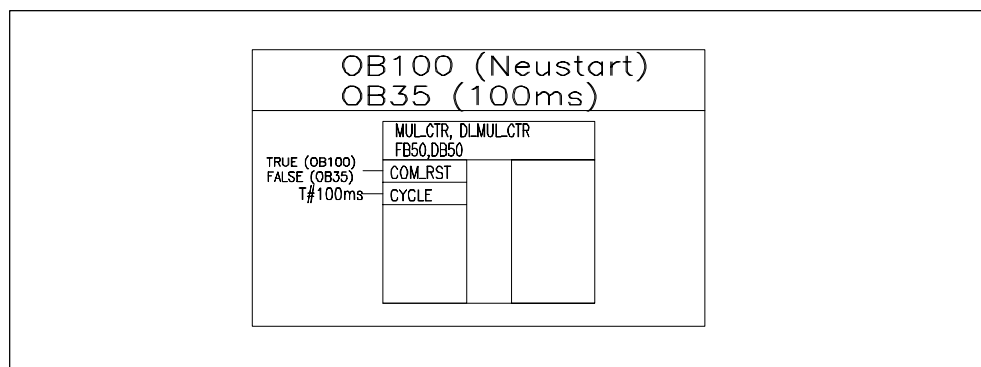


Bild 3-37 Bausteinanruf des Beispiels 12

Anwendungsbereich

Der Baustein MUL_CTR ist ein Mehrgrößenregler. Zwei PID-Regler mit kontinuierlichem Ausgang werden auf die Regelstrecke geschaltet. Bild 3-38 zeigt die Bausteinverschaltung des MUL_CTR.

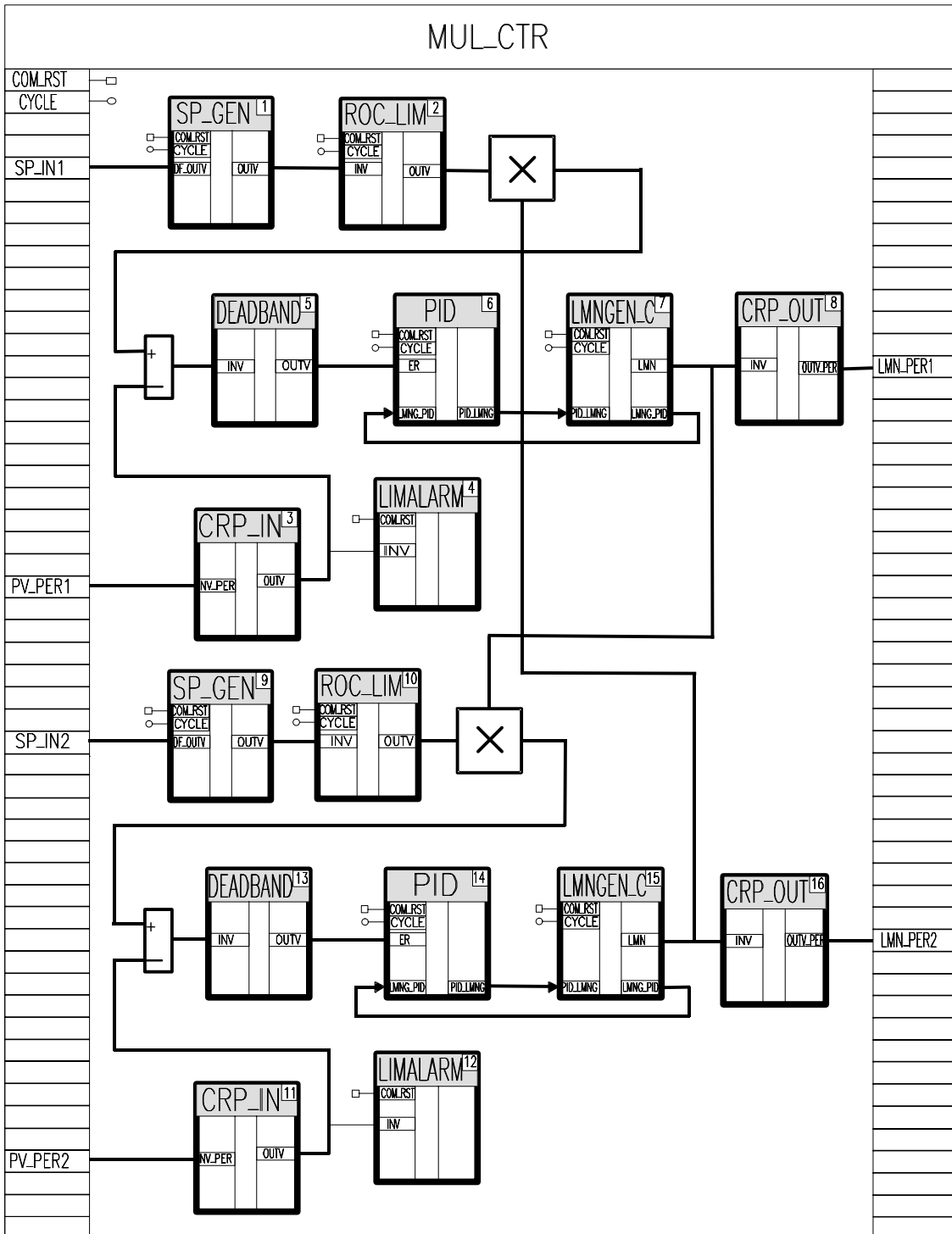


Bild 3-38 Bausteinverschaltung des MUL_CTR

Funktionsbeschreibung

Die Funktionsweise der einzelnen PID-Regler ist analog zu dem kontinuierlichen Regler PIDCTR_C in Abschnitt 3.3.1 auf Seite 3-9.

Zusätzlich wird jeweils der Stellwert des einen Reglers mit dem Sollwert des anderen Reglers multipliziert.

Neustart

Bei Neustart wird jeder Einzelbaustein aufgerufen. Einzelbausteine mit Neustartroutine werden in ihrer Neustartroutine aufgerufen.

4

Technische Daten

4.1 Laufzeiten

| Baustein- name | | | | CPU 315 CPU 315-2DP | Pin μ s | | | CPU 416-1 CPU 416- 2DP |
|-------------------|-------|------------|------------|------------------------|-------------|------------|------------|------------------------------|
| | | in μ s | in μ s | in μ s | | in μ s | in μ s | |
| A_DEAD_B | FB 1 | 170 | 160 | 130 | 31 | 30 | | 9 |
| CRP_IN | FB 2 | 60 | 60 | 60 | 21 | 30 | | 6 |
| CRP_OUT | FB 3 | 220 | 210 | 180 | 42 | 30 | | 12 |
| DEAD_T | FB 4 | 330 | 320 | 260 | 60 | 39 | | 17 |
| DEADBAND | FB 5 | 210 | 200 | 160 | 32 | 30 | | 10 |
| DIF | FB 6 | 710 | 690 | 550 | 92 | 55 | | 26 |
| ERR_MON | FB 7 | 350 | 340 | 270 | 49 | 34 | | 14 |
| INTEG | FB 8 | 510 | 500 | 400 | 74 | 44 | | 20 |
| LAG1ST | FB 9 | 670 | 650 | 520 | 94 | 56 | | 27 |
| LAG2ND | FB 10 | 1140 | 1110 | 880 | 158 | 86 | | 44 |
| LIMALARM | FB 11 | 610 | 590 | 470 | 65 | 41 | | 18 |
| LIMITER | FB 12 | 170 | 170 | 140 | 30 | 30 | | 9 |
| LMNGEN_C | FB 13 | 410 | 390 | 320 | 64 | 41 | | 18 |
| LMNGEN_S | FB 14 | 1470 | 1430 | 1160 | 184 | 115 | | 57 |
| NONLIN | FB 15 | 410 | 400 | 320 | 74 | 45 | | 20 |
| NORM | FB 16 | 430 | 420 | 330 | 67 | 42 | | 19 |
| OVERRIDE | FB 17 | 180 | 170 | 150 | 35 | 30 | | 9 |
| PARA_CTL | FB 18 | 150 | 140 | 120 | 30 | 30 | | 9 |
| PID | FB 19 | 1460 | 1420 | 1150 | 184 | 113 | | 56 |
| PULSEGEN | FB 20 | 200 | 200 | 170 | 50 | 33 | | 12 |
| RMP_SOAK | FB 21 | 200 | 200 | 160 | 40 | 30 | | 11 |
| ROC_LIM | FB 22 | 680 | 660 | 530 | 90 | 55 | | 24 |
| SCALE | FB 23 | 130 | 120 | 100 | 23 | 30 | | 8 |
| SPLT_RAN | FB 24 | 110 | 100 | 90 | 23 | 30 | | 7 |
| SP_GEN | FB 25 | 350 | 340 | 270 | 58 | 35 | | 15 |
| SWITCH | FB 26 | 90 | 80 | 70 | 25 | 30 | | 7 |
| LP_SCHED | FC 1 | 340 | 330 | 280 | 79 | 42 | | 26 |

4.2 Arbeitsspeicherbelegung

| Bausteinname | | FB-Länge im Speicher (in Bytes) | FB-Länge zur Laufzeit (in Bytes) | DB-Länge im Speicher (in Bytes) | DB-Länge zur Laufzeit (in Bytes) |
|---|-------|---------------------------------|----------------------------------|---------------------------------|----------------------------------|
| A_DEAD_B | FB 1 | 898 | 692 | 186 | 44 |
| CRP_IN | FB 2 | 182 | 70 | 122 | 20 |
| CRP_OUT | FB 3 | 206 | 96 | 114 | 14 |
| DEAD_T | FB 4 | 532 | 394 | 142 | 22 |
| DB_DEADT (bei 10 Vergangenheitswerten) | DB 3 | | | 138 | 40 |
| DEADBAND | FB 5 | 232 | 120 | 114 | 16 |
| DIF | FB 6 | 410 | 268 | 158 | 30 |
| ERR_MON | FB 7 | 558 | 360 | 206 | 52 |
| INTEG | FB 8 | 488 | 314 | 168 | 36 |
| LAG1ST | FB 9 | 534 | 368 | 156 | 30 |
| LAG2ND | FB 10 | 690 | 516 | 190 | 46 |
| LIMALARM | FB 11 | 390 | 240 | 152 | 28 |
| LIMITER | FB 12 | 262 | 140 | 124 | 20 |
| LMNGEN_C | FB 13 | 1576 | 1280 | 276 | 80 |
| LMNGEN_S | FB 14 | 2578 | 2152 | 360 | 110 |
| NONLIN | FB 15 | 826 | 672 | 138 | 18 |
| DB_NONLI (mit Startpunkt und 4 Stützstellen) | DB 4 | | | 146 | 42 |
| NORM | FB 16 | 234 | 122 | 130 | 24 |
| OVERRIDE | FB 17 | 362 | 214 | 146 | 28 |
| PARA_CTL | FB 18 | 406 | 232 | 234 | 82 |
| PID | FB 19 | 1560 | 1242 | 340 | 98 |
| PULSEGEN | FB 20 | 1110 | 872 | 190 | 34 |
| RMP_SOAK | FB 21 | 1706 | 1500 | 212 | 62 |
| DB_RMPSK (mit Startpunkt und 4 Stützstellen) | DB 2 | | | 146 | 42 |
| ROC_LIM | FB 22 | 1242 | 980 | 222 | 50 |
| SCALE | FB 23 | 136 | 32 | 114 | 16 |
| SPLT_RAN | FB 24 | 304 | 180 | 138 | 28 |
| SP_GEN | FB 25 | 658 | 484 | 164 | 40 |
| SWITCH | FB 26 | 238 | 116 | 118 | 18 |

| Bausteinname | | FB-Länge im Speicher (in Bytes) | FB-Länge zur Laufzeit (in Bytes) | DB-Länge im Speicher (in Bytes) | DB-Länge zur Laufzeit (in Bytes) |
|------------------------------|------|--|---|--|---|
| LP_SCHED | FC 1 | 1104 | 972 | 280 | 79 |
| DB_LOOP (bei 5 Regelkreisen) | DB 1 | | | 190 | 64 |

4.3 Schätzformeln

Laufzeit

Mit der folgenden Formel können Sie die Gesamtlaufzeit berechnen:

| |
|--|
| Laufzeit der aufgerufenen Bausteine (aus Modular PID Control) + Anzahl der Bausteinaufrufe * Konstante <hr style="width: 80%; margin-left: 0;"/> = Gesamtlaufzeit |
|--|

Die Anzahl der Bausteinaufrufe ermitteln Sie wie folgt:

| |
|---|
| aufgerufene Bausteine (aus Modular PID Control) + aufgerufene Anwender-FBs <hr style="width: 80%; margin-left: 0;"/> = Anzahl der Bausteinaufrufe |
|---|

Für die CPUs gelten folgende **Konstanten**:

| CPU | Konstante |
|-------------------------|-----------|
| CPU 313 | 105 µs |
| CPU 314 | 100 µs |
| CPU 315, CPU 315-2 DP | 80 µs |
| CPU 412-1, CPU 413-1 | 23 µs |
| CPU 414-1, CPU 414-2 DP | 12 µs |
| CPU 416-1, CPU 416-2 DP | 9 µs |

Speicherplatz

Die Speicherplatzabschätzung bezieht sich auf den Arbeitsspeicher.

| |
|---|
| Speicherplatz der verwendeten FBs (aus Modular PID Control) |
| + Speicherplatz für Instanz-DB-Daten aufgerufener FBs (aus Modular PID Control) |
| + Anzahl der Bausteinaufrufe * 120 Bytes |
| <hr/> |
| = Gesamtspeicherplatz |

Die Anzahl der Bausteinaufrufe ermitteln Sie wie folgt:

| |
|---|
| aufgerufene Bausteine (aus Modular PID Control) |
| + aufgerufene Anwender-FBs |
| <hr/> |
| = Anzahl der Bausteinaufrufe |

Projektiersoftware für Modular PID Control

5

Voraussetzungen

STEP 7 muß korrekt auf Ihrem PG/PC installiert sein.

Lieferform

Die Auslieferung der Software erfolgt auf CD.

Installation

So installieren Sie die Software:

1. Legen Sie die CD in das Laufwerk Ihres PC ein.
2. Starten Sie unter WINDOWS den Dialog zur Installation von Software durch Doppelklick auf das Symbol "Software" in "Systemsteuerung".
3. Wählen Sie im Dialog das Laufwerk und die Datei Setup.exe aus und starten Sie den Installationsvorgang.
Dabei wird die Projektiersoftware auf Ihrem PG/PC installiert.
4. Befolgen Sie Schritt für Schritt die Anweisungen, die Ihnen das Installationswerkzeug anzeigt.

Liesmich-Datei auslesen

Gegebenenfalls sind wichtige aktuelle Informationen über die gelieferte Software in einer Liesmich-Datei abgelegt. Diese Datei finden Sie im Startmenü von WINDOWS unter START > SIMATIC > Produkt-Hinweise > Deutsch.

Zweck

Die Projektiersoftware soll Ihnen die Inbetriebnahme Ihres Reglers erleichtern, so dass Sie sich verstärkt dem eigentlichen Regelungsproblem zuwenden können.

Funktionen der Projektiersoftware

Jede Funktion läuft in einem eigenen Fenster ab. Eine Funktion kann auch mehrfach aufgerufen werden, d.h. sie kann z.B. die Kreisbilder von mehreren Reglern simultan darstellen.

Regler beobachten

Mit Hilfe der Funktion Kurvenschreiber können Sie für von Ihnen ausgewählte Variablen des Regelkreises den Werteverlauf über einen definierten Zeitraum aufzeichnen und darstellen lassen. Es sind bis zu vier Variablen gleichzeitig darstellbar.

Mit der Funktion Kreisbild können Sie sich die relevanten Regelgrößen (Sollwert, Stellwert und Istwert) eines von Ihnen ausgewählten Reglers darstellen lassen.

Regler optimieren

Mit Hilfe der Funktion Regleroptimierung können Sie für einen vorgegebenen Regelkreis die optimale Reglereinstellung ermitteln. In einem experimentellen Verfahren werden die Kennparameter der Regelstrecke ermittelt. Daraus werden die optimalen Reglerparameter berechnet und zur weiteren Verwendung bereitgestellt.

Regler bedienen

Über die Funktion Kreisbild können Sie die relevanten Regelkreisgrößen des aktuell angezeigten Reglers verändern bzw. neu vorgeben.

Integrierte Hilfe

Zur Projektiersoftware gibt es eine integrierte Hilfe, die Sie unterstützt. Sie haben folgende Möglichkeiten, die integrierte Hilfe aufzurufen:

- Über den Menübefehl **Hilfe ► Hilfethemen**
- Durch Drücken der Taste F1
- Durch Anklicken des Hilfe-Buttons in den einzelnen Masken

Literaturverzeichnis

A

Ergänzende Literatur

Nachfolgend sind Bücher aufgelistet, in denen Sie Grundlagen der Regelungstechnik nachschlagen können:

/350/ Benutzerhandbuch: *SIMATIC 7*,
Standardregelung

/352/ J. Gißler, M. Schmid: Vom Prozeß zur Regelung. Analyse, Entwurf, Realisierung in der Praxis. Siemens AG. ISBN 3-8009-1551-0.

Index

A

Ablöseregler, 3-32
Adaption der Totzone, 2-4
analogen Ausgangswert wandeln, 2-11
analogen Eingangswert wandeln, 2-8
Analogwertschalter, 2-136
Arbeitsspeicherbelegung, 4-2
Aufrufbearbeitung, 2-70
Aufrufdaten, 2-1
Ausgabe des kontinuierlichen PID-Reglers,
2-48
Ausgabe PID-Schrittregler, 2-55
Ausgangsparameter
A_DEAD_B, 2-6
CRP_IN, 2-9
CRP_OUT, 2-12
DEAD_T, 2-15
DEADBAND, 2-18
DIF, 2-21
ERR_MON, 2-27
INTEG, 2-30
LAG1ST, 2-34
LAG2ND, 2-39
LIMALARM, 2-43
LIMITER, 2-47
LMNGEN_C, 2-50
LMNGEN_S, 2-58
LP_SCHED_S, 2-66
NONLIN, 2-74
NORM, 2-78
OVERRIDE, 2-80
PARA_CTL, 2-84
PID, 2-89
PULSEGEN, 2-100
RMP_SOAK, 2-110
ROC_LIM, 2-121
SCALE, 2-128
SP_GEN, 2-131
SPLT_RAN, 2-134
SWITCH, 2-137

B

Begrenzung, 2-45
Begrenzungen, bausteinintern, 2-71
Bereichsaufteilung, 2-133
Bereichsauswahlregler, 3-29
Blockschaltbild
A_DEAD_B, 2-2
CRP_IN, 2-8
CRP_OUT, 2-11
DEAD_T, 2-13
DEADBAND, 2-17
DIF, 2-20
ERR_MON, 2-24
INTEG, 2-28
LAG1ST, 2-33
LAG2ND, 2-37
LIMALARM, 2-41
LIMITER, 2-45
LMNGEN_C, 2-48
LMNGEN_S, 2-55
LP_SCHED, 2-65
NONLIN, 2-72
NORM, 2-77
OVERRIDE, 2-79
PARA_CTL, 2-82
PID, 2-87
PULSEGEN, 2-97
RMP_SOAK, 2-107
ROC_LIM, 2-118
SCALE, 2-127
SP_GEN, 2-129
SPLT_RAN, 2-133
SWITCH, 2-136

D

Differenzierer, 2-20

Dreipunktregelung, 2-102
Handbetrieb, 2-106
Kennlinie, 2-103
unsymmetrische Kennlinie, 2-103

E

Eingangsparameter
A_DEAD_B, 2-6
CRP_IN, 2-9
CRP_OUT, 2-12
DEAD_T, 2-14
DEADBAND, 2-18
DIF, 2-20
ERR_MON, 2-26
INTEG, 2-29
LAG1ST, 2-33
LAG2ND, 2-39
LIMALARM, 2-42
LIMITER, 2-46
LMNGEN_C, 2-49
LMNGEN_S, 2-57
LP_SCHED_S, 2-66
NONLIN, 2-74
NORM, 2-78
OVERRIDE, 2-80
PARA_CTL, 2-83
PID, 2-88
PULSEGEN, 2-99
RMP_SOAK, 2-109
ROC_LIM, 2-120
SCALE, 2-128
SP_GEN, 2-130
SPLT_RAN, 2-134
SWITCH, 2-137

F

Fahrkurve
anhalten, fortsetzen, 2-116
starten, 2-114
Festwertregler, 3-6
Festwertregler mit kontinuierlichem Ausgang,
3-9
Festwertregler mit schaltendem Ausgang
für integrierende Stellglieder, 3-4
für proportionale Stellglieder, 3-11

G

Grenzwertmelder, 2-41

H

Hardwareumgebung, 1-3
Hochlaufgeber, 2-118

I

Impulsausgang, Schaltverhalten, 2-102
Impulsformer, 2-97
automatische Synchronisation, 2-100
Betriebsarten, 2-101
Genauigkeit, 2-100
Integrierer, 2-28

K

Kaskadenregler, 3-23

L

Laufzeiten, 4-1
Schätzformel, 4-4
Liesmich-Datei, 5-1
lineare Skalierung, 2-127

M

Mehrgrößenregler, 3-35
Mindestimpulsdauer, 2-102
Mindestpausendauer, 2-102
Mischungsregler, 3-20
Modular PID Control, 1-1
Funktionsspektrum, 1-4
Grundfunktionen, 1-1
Konzept, 1-1
Softwareprodukt, 1-2
Softwareumgebung, 1-3
Teilfunktionen, 1-1

N

nichtlineare statische Kennlinie, 2-72
Normalbetrieb, 2-111

O

Override Regelung, 2-79

P

Parametersteuerung, 2-82
physikalische Normierung, 2-77
PID-Algorithmus, 2-86

R

Rampe, 2-118
Regeldifferenzüberwachung, 2-24
Regelkreisaufruf
 Bedingungen, 2-71
 Beispiel, 2-71
Regelkreisaufruf (LP_SCHED), 2-68
Regelkreisaufrufe sperren, 2-70
Regler mit Störgrößenaufschaltung, 3-27
Regler mit Vorsteuerung, 3-25
Regleraufrufverteiler, 1-1, 2-65
 Funktionsweise, 2-65
 Parametrierung, 2-69
Reglerstrukturen, Beispiele, 3-1

S

Schätzformeln, 4-4

Schrittregler, 3-7
Softwareumgebung, 1-3
Sollwertgeber, 2-129
Speicherplatz, 4-5

T

Technische Daten, 4-1
Teilfunktionen, 1-1
Tool
 integrierte Hilfe, 5-2
 Softwarevoraussetzungen, 5-1
Totzeit, 2-13
Totzone, 2-17
Totzone, adaptive, 2-2

V

Verhältnisregler
 einschleifig, 3-15
 mehrschleifig, 3-17
Verzögerungsglied 1. Ordnung, 2-33
Verzögerungsglied 2. Ordnung, 2-37

Z

Zeitplangeber, 2-107
 anhalten, 2-115
 Ausgang vorbelegen, 2-114
 Betriebsarten, 2-112, 2-113
 einschalten, 2-113
 Online-Änderungen, 2-117
 zyklischer Betrieb, 2-115
Zweipunktregelung, 2-105