

ADCO (2.2)

(ADaptive COntroller für Siemens S7-4xx und S7-3xx)

1. Einleitung	2
2. Konfiguration/Programmierung des Reglers	4
2.1. Eingänge:	5
2.2. Ausgänge:	9
3. Einstellen der Reglerparameter	11
4. Sichern der Reglerparameter	13
5. Kontinuierliche Adaption	13
6. Kaskadenregelkreis	14
7. Mehrbereichs-Regler	15
7.1. Eingänge:	17
7.2. Ausgänge:	17
8. Besonderheiten zum Betrieb auf S7-Steuerungen	18
8.1. Systemvoraussetzungen	18
8.2. Installation der Regler-Bibliothek	18
8.3. Installation der Beispiel-Applikation	19
8.4. Erklärung der Bausteine	19
8.5. Arbeiten mit der Test-Applikation	20
8.6. Nach dem Start des AG	20
8.7. Zyklischer Aufruf der Regler	20
8.8. Bedienen der Regler	20
8.9. Bedienung des ADCO (FB50, DB50 und DB10)	20
8.10. Um den Regler adaptieren zu lassen, verfahren Sie wie folgt :	21
8.11. Bedienung des ADMR (FB80, DB80, DB30)	21
8.12. Vorschläge zum testen und probieren	21
9. Tipps und Tricks	22

1. Einleitung

Das Einstellen (Parametrieren) von PID-Reglern erfolgt überwiegend durch sogenannte "Trial and Error"-Methoden. Dies erfordert spezielle Erfahrung und ist, insbesondere bei langsamen Prozessen (z.B. Temperaturstrecken), äußerst zeitaufwendig. Darüberhinaus wird meist nur eine "akzeptable" Regelgüte erzielt, die selbst aber noch relativ weit vom Optimum entfernt liegt. Ändert sich der Prozeß zeit- oder arbeitspunktabhängig, so wird die Reglereinstellung nochmals um ein vielfaches schwieriger.

In all diesen Punkten schafft der adaptive Regler ADCO Abhilfe. Er paßt sich schnell und automatisch an veränderliche Prozeßeigenschaften an, kann aber auch - als bessere Alternative zum PID-Regler - mit festen Parametern betrieben werden, indem die Adaption nach erstmaliger Reglerparameteroptimierung ausgeschaltet wird. Eine Nachoptimierung kann dann bei Bedarf zu einem späteren Zeitpunkt beliebig oft durchgeführt werden.

ADCO liefert eine hervorragende Regelgüte bei Prozessen mit reinen Verzögerungsgliedern, bei Prozessen mit Integralanteil und bei Prozessen mit beinahe beliebig großen Totzeitanteilen, die mit herkömmlichen Reglern nur sehr schwer zu regeln sind.

Im Unterschied zum PID-Regler liefert ADCO eine gleichermaßen optimale Regelgüte sowohl im Führungs- (Sollwertänderungen) als auch im Störverhalten (Störeinflüsse auf die Regelgröße).

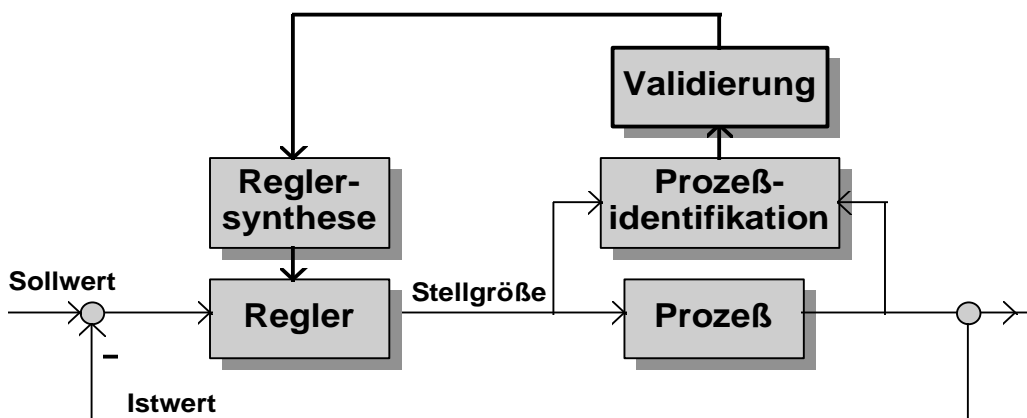


Bild 1.1: Blockstruktur des adaptiven Regelkreises

ADCO besteht im wesentlichen aus zwei Teilen:

- * Der Schätzalgorithmus zur Bestimmung des Prozeßmodells basiert auf einer Methode, die unter der Bezeichnung DSF (Discrete Square Root Filtering) oder SRIF (Square Root Information Filter) bekannt ist. Diese Prozedur errechnet ein parametrisches Modell des zu regelnden Prozesses, indem die Prozeßsignale (Prozeßvariable, Stellgröße) nach oben erwähnter Methode ausgewertet werden.

- * Die Regleroptimierung basiert auf dem geschätzten und durch eine Überwachungsebene validierten Prozeßmodell. Der Algorithmus realisiert einen optimalen Zustandsregler, der neben der aktuellen Abweichung der Prozeßvariablen vom Sollwert noch weitere Zustände, die eine Aussage über zukünftige Werte der Prozeßvariablen erlauben, ausregelt. Da dieser Regler wesentlich mehr Informationen über den Prozeß verarbeitet als ein PID-Regler, ist er diesem selbst bei einfachen Prozessen deutlich überlegen. Nach einer Sollwertänderung oder einer Störung auf den Prozeß werden alle Zustandsabweichungen auf den Wert 0 reduziert. Das Regelverhalten hängt von einem einzigen Einstellparameter (Reglersensitivität) ab, der Werte zwischen 0 und 101 annehmen kann. Die Standardbelegung für diesen Parameter beträgt 50 und muß bei den meisten Prozessen nicht angepaßt werden. Prinzipiell bedeutet eine Erhöhung des Parameters eine Erhöhung der Regleraktivität. Dies bedeutet, daß der Regler stärker eingreift, aber auch mehr Stellenergie benötigt.

Herausragende Vorteile gegenüber herkömmlichen Reglern:

- * wesentliche Zeiteinsparung bei der Reglerparametrierung
- * prinzipiell bessere Regelgüte
- * signifikant bessere Regelung bei Prozessen mit Integral- oder Totzeitanteilen
- * optimale Führungs- und Störgrößenregelung
- * Adaption an veränderliche Prozeßeigenschaften
- * asymptotisches Einschwingen der Regelgröße

ADCO kann als Funktionsbaustein in STEP-7-Programme integriert werden. Aufgrund der Anforderungen des Reglers an Speicherplatz und Rechenkapazität ist der adaptive Regelalgorithmus allerdings nur auf Steuerungen vom Typ S7-318 oder höher lauffähig.

Gegenüber Version 1.0 wurde die Regelgüte speziell bei Prozessen mit signifikanten Totzeitanteilen wesentlich verbessert.

2. Konfiguration/Programmierung des Reglers

Wie Standardfunktionsbausteine von STEP-7 wird auch der adaptive Zustandsregler durch Zuordnung der Ein-/Ausgänge auf Elemente von Datenbausteinen parametrisiert. Es wird empfohlen, pro Reglerinstanz einen Datenbaustein für die internen Instanzvariablen und einen anderen Datenbaustein für die Ein-/Ausgänge bereitzustellen. Die im untenstehenden Bild vorgeschlagenen Offsets für die Zuordnung der Ein-/Ausgänge sollten ebenfalls eingehalten werden.

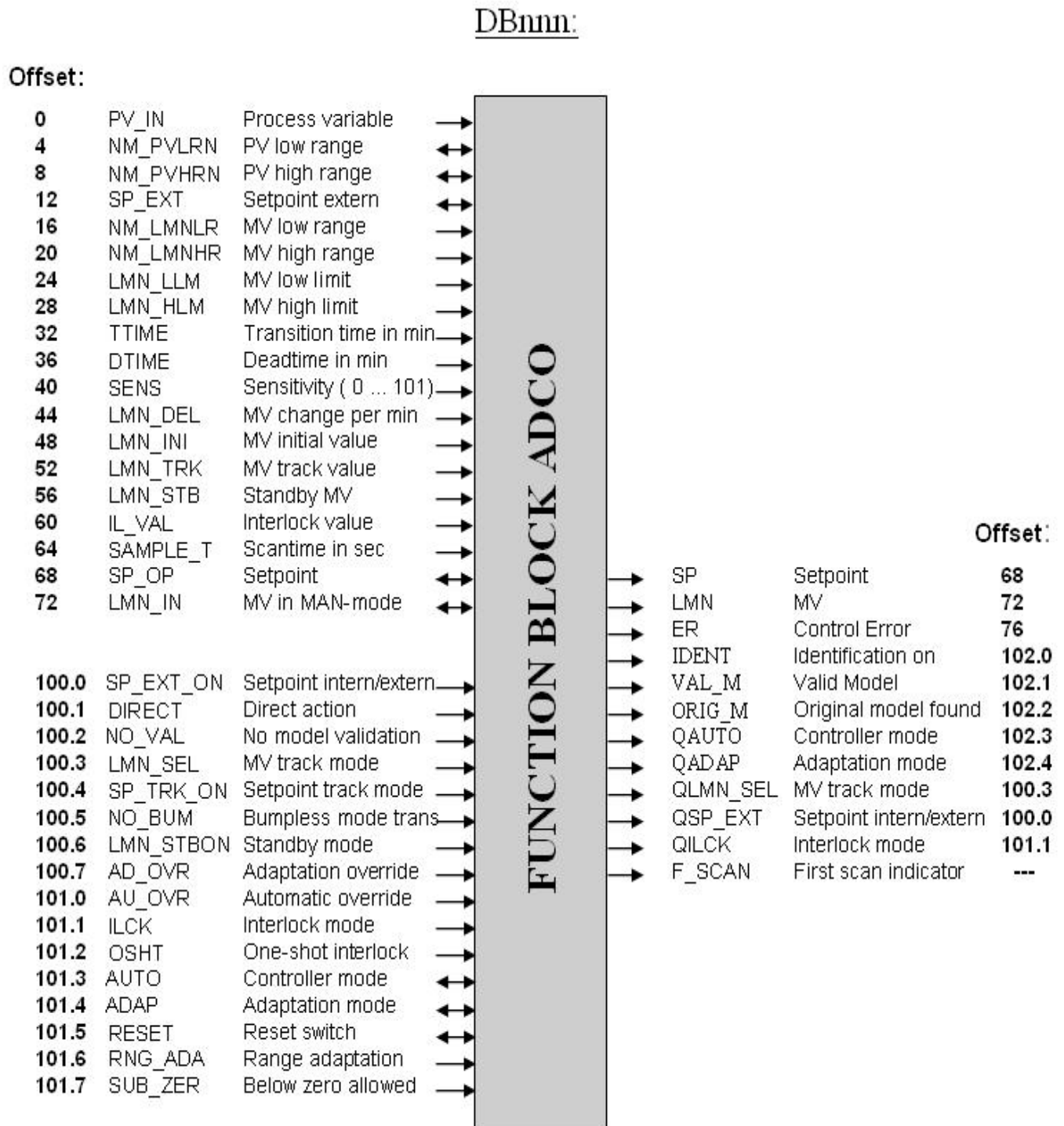


Bild 2.1: ADCO (FB50) Funktionsbaustein

Nachfolgend werden die Ein-/Ausgänge des Funktionsbausteins näher erläutert:

2.1. **Eingänge:**

- PV_IN (Offset 0):
Prozessvariable des Regelkreises in physikalischen Einheiten. Hier wird der zu regelnde Messwert (Istwert) angegeben.
- NM_PVLR (Offset 4):
Physikalische Bereichsuntergrenze der Prozessvariablen.
- NM_PVHR (Offset 8):
Physikalische Bereichsobergrenze der Prozessvariablen.
- SP_EXT (Offset 12):
Mit diesem Eingang wird der externe Sollwert (in physikalischen Einheiten) verbunden.
- SP_EXT_ON (Offset 100.0):
Mit diesem Eingang wird zwischen internem und externem Sollwert umgeschaltet (0: interner Sollwert; 1: externer Sollwert).
- NM_LMNLR (Offset 16):
Physikalische Bereichsuntergrenze der Stellgröße (Reglerausgang).
- NM_LMNHR (Offset 20):
Physikalische Bereichsobergrenze der Stellgröße (Reglerausgang).
- LMN_LLM (Offset 24):
Untere Begrenzung der Stellgröße innerhalb des physikalischen Stellgrößenbereichs.
- LMN_HLM (Offset 28):
Obere Begrenzung der Stellgröße innerhalb des physikalischen Stellgrößenbereichs.
- TTIME (Offset 32):
Die Übergangszeit (Definition in Kap. 3) muss während der Konfiguration hier oder später beim Anfahren des Reglers im entsprechenden Einzelbild eingegeben werden. Die Dimension des Eintrags ist [min]. Die Übergangszeit bestimmt, mit welcher Abtastrate der Regler intern arbeitet (interne Abtastrate = Übergangszeit/60). Die Abtastzeit des Funktionsbausteins sollte so gewählt werden, dass die mit obiger Formel errechnete Abtastrate überhaupt realisierbar ist. Der Grund für die Spezifikation einer überlagerten Abtastrate liegt darin, dass es keinen Sinn ergibt, einen trägen Prozess (z.B. Temperaturstrecken) im [msec]-Bereich zu regeln. Die sich aus dieser Abtastung ergebenden Signaldifferenzen liefern nämlich keine Informationen über das Prozessverhalten. Sie besitzen einen Rauschanteil von annähernd 100%. Wird in diesen Blockeingang eine 0 eingetragen, so wird die interne Abtastzeit mit der Abtastzeit des Funktionsbausteins gleichgesetzt.

- **DTIME (Offset 36):**
Der Zustandsregler eignet sich, wie bereits erwähnt, besonders zur Regelung von Totzeitprozessen. Die Prozesstotzeit (Definition in Kap. 3) wird jedoch nicht automatisch ermittelt, sondern muss vorgegeben werden (in [min]). Bei der Berechnung der Stellgröße wird dieser Eintrag ausgewertet. Mit anderen Worten, die Berechnung der Stellgröße bezieht sich bei Vorgabe einer Totzeit nicht auf die aktuelle Prozessvariable, sondern auf einen zukünftigen Prozesswert, der mit Hilfe des Prozessmodells und der eingegebenen Totzeit errechnet wird. Die Totzeit kann online an sich ändernde Prozesscharakteristiken angepasst werden.
- **SENS (Offset 40):**
Die Sensitivität des Reglers ist eigentlich der einzige „Tuning Parameter“, der vom Anwender eingestellt werden kann. Dieser Eingang sollte mit einem Standardwert (50; Bereich: 0 ... 101) vorbelegt werden und muss im Regelfall nicht geändert werden. Prinzipiell bedeutet eine Erhöhung der Sensitivität eine Verstärkung der Regleraktivität.
- **DIRECT (Offset 100.1):**
Die Eingabe spezifiziert, ob bei einer Erhöhung der Stellgröße die Prozessvariable auch mit einer Erhöhung reagiert (1: DIRECT) oder ob der Wert der Prozessvariablen abnimmt (0: UMGEKEHRT). In dem hier vorliegenden Algorithmus wird die Spezifizierung der Regleraktion mit dazu benutzt, das in der Identifikation geschätzte Prozessmodell zu validieren. „DIRECT“ bedeutet nämlich, dass das Prozessmodell einen positiven Verstärkungsfaktor haben muss. „UMGEKEHRT“ muss einen negativen Verstärkungsfaktor zur Folge haben. Das sich aus der Parameterschätzung ergebende Prozessmodell wird erst dann an das interne Modul „Regleroptimierung“ weitergegeben, wenn der Verstärkungsfaktor des geschätzten Modells mit der Spezifikation im „DIRECT“-Eingang übereinstimmt. Für Prozesse mit integrierendem Verhalten ist dieser Eintrag nicht relevant, da dort kein Prozessverstärkungsfaktor definiert ist.
- **NO_VAL (Offset 100.2):**
Das geschätzte Prozessmodell wird, bevor es zur Reglerparameteroptimierung herangezogen wird, umfangreichen Validierungsschritten unterzogen. Erst wenn alle Überprüfungen ein positives Ergebnis zeigen, wird das Prozessmodell freigegeben. In diesem Feld kann die Prozessmodellüberprüfung ausgeschaltet werden. Dies sollte jedoch nur dann getan werden, wenn z.B. bedingt durch sehr stark verrauschte Signale kein gültiges Modell gefunden werden kann. Dies sollte aber nur äußerst selten der Fall sein (0: Prozessmodellüberprüfung; 1: keine Prozessmodellüberprüfung).
- **LMN_DEL (Offset 44):**
Der Wert in diesem Feld beschränkt die Änderungsgeschwindigkeit der Stellgröße (physikalische Änderung - i.a. % - pro [min]). Dies kann z.B. für Ventile sinnvoll sein, deren Öffnungs- und Schließgeschwindigkeiten prozessbedingt limitiert werden müssen. Im Ausgangs-Track-Modus und im manuellen Modus ist diese Limitierung nicht relevant.

- **LMN_INI (Offset 48):**
Hier wird festgelegt, mit welchem Stellgrößenwert ein neu konfigurierter Regler bei seiner ersten Aktivierung auf den Prozess wirken soll. Ein von 0 abweichender Wert muss meistens bei sogenannten Split-Range-Regelungen eingegeben werden, um beide vom Regler beeinflussten Ventile in der sicheren (stromlosen) Stellung hochzufahren.
- **LMN_TRK, LMN_SEL (Offset 52, 100.3):**
In diesem Modus ist es möglich, einen definierbaren Wert (LMN_TRK) auf den Reglerausgang (Stellgröße) zu schalten, d.h den Regler in den Track-Modus (LMN_SEL; 0: kein Track-Modus; 1: Track-Modus) zu schalten. Diese Nachfolgeschaltung (Track) kann z.B. dazu benutzt werden, die Modus-Umschaltung in Kaskadenkreisen zu vereinfachen (siehe auch Kap. 6).
- **SP_TRK_ON (Offset 100.4):**
Um einen stoßfreien Übergang des Reglers vom manuellen in den automatischen Modus zu ermöglichen, kann der Sollwert im manuellen Modus dem Wert der Prozessvariablen gleichgesetzt werden. Beim Umschalten in den automatischen Modus ist der Regelfehler (Sollwert - Istwert) gleich 0, was bedeutet, dass keine sprungförmigen Änderungen am Reglerausgang (Stellgröße) generiert werden (0: kein Sollwert-Track-Modus; 1: Sollwert-Track-Modus).
- **NO BUM (Offset 100.5):**
Wird, wie in Kaskadenkreisen üblich, der Sollwert eines Reglers von extern in den Baustein übertragen, so ist es mit dieser Einstellung dennoch möglich, einen stoßfreien Übergang vom manuellen in den automatischen Modus sicherzustellen, indem der Reglerausgang temporär einer Tiefpassfilterung unterworfen wird (0: keine Tiefpassfilterung; 1: Tiefpassfilterung).
- **LMN_STB, LMN_STBON (Offset 56, 100.6):**
Der Algorithmus stellt einen Modus (LMN_STBON; 0: Aktivbetrieb; 1: Parallelbetrieb) zur Verfügung, in dem der Regler parallel zu einem existierenden Regelkonzept betrieben werden kann. Über den Eingang „LMN_STB“ wird dann die Stellgröße des aktiven Reglers in den adaptiven Algorithmus eingespeist. Mit Hilfe dieses Signals kann zusammen mit der Prozessvariablen ein Prozessmodell aufgebaut und anschließend der Zustandsregler optimiert werden. Die vom adaptiven Regler generierte Stellgröße sollte in diesem Modus die Ventilstellung im Feld nicht beeinflussen (dies ist durch entsprechende Verschaltung des Reglerausgangs sicherzustellen). Die Stellgröße sollte lediglich mitgeschrieben werden und kann dann qualitativ mit dem Stellgrößenverlauf des aktiven Regelkonzeptes verglichen werden, was i.a. eine Aussage über die Güte des adaptiven Algorithmus' erlaubt. So kann der adaptive Regler getestet werden, ohne dass man prozesseitig ein Risiko eingeht.
- **AD_OVR (Offset 100.7):**
Ein Aktivieren dieses Eingangs (0 -> 1) zwingt den Regler in den nicht-adaptiven Modus. Dies ist z.B. dann sinnvoll, wenn die reale Prozessvariable durch einen Sensorfehler ausfällt. Falsche Zusammenhänge (Stellgröße <-> Istwert) würden ansonsten bei weiterlaufender Adaption in das Modell projiziert werden. Ist der Grund für diese Umschaltung (Override) nicht länger existent, so geht der Regler nicht von selbst in den adaptiven Modus zurück. Dies kann nur durch einen expliziten Eingriff des Anwenders (Anlagenfahrers) erfolgen.

- **AU_OVR (Offset 101.0):**
Ein Aktivieren dieses Eingangs (0 -> 1) zwingt den Regler in den manuellen Modus. Dies kann z.B. bei gewissen Not-Aus-Szenarien sinnvoll sein. Ist der Grund für diese Umschaltung (Override) nicht länger existent, so geht der Regler nicht von selbst in den automatischen Modus zurück. Dies kann nur durch einen expliziten Eingriff des Anwenders (Anlagenfahrers) erfolgen.
- **IL_VAL, ILCK (Offset 60, 101.1):**
Ein Aktivieren des Eingangs „ILCK“ schaltet den Regler in den Verriegelungsmodus, d.h. die Adaption wird deaktiviert, der Regler geht in den manuellen Modus und die Stellgröße nimmt den durch „IL_VAL“ spezifizierten Wert an.
- **OSHT (Offset 101.2):**
In diesem Auswahlfeld wird definiert, ob der Regler nach der Umschaltung in den Verriegelungsmodus sofort wieder bedienbar sein soll oder ob der geschaltete Zustand eingefroren wird, solange der Verriegelungseingang gesetzt ist (0: Verriegelungszustand eingefroren; 1: Regler sofort wieder bedienbar).
- **SAMPLE_T (Offset 64):**
Abtastzeit des Funktionsbausteins in [sec].
- **SP_OP (Offset 68):**
Interner Sollwert des Regelkreises in physikalischen Einheiten. Dieser Blockeingang ist als sogenannter „IN-OUT“-Eingang definiert, d.h. er kann sowohl von außen als auch von innen, vom Regelalgorithmus, verändert werden. Dies wiederum bedeutet, dass dieser Eingang nicht mit dem Ausgang eines anderen Funktionsbausteins verschaltet werden sollte.
- **LMN_IN (Offset 72):**
Physikalischer Stellgrößenwert im manuellen Reglermodus. Dieser Blockeingang ist als sogenannter „IN-OUT“-Eingang definiert, d.h. er kann sowohl von außen als auch von innen, vom Regelalgorithmus, verändert werden. Dies wiederum bedeutet, dass dieser Eingang nicht mit dem Ausgang eines anderen Funktionsbausteins verschaltet werden sollte.
- **AUTO (Offset 101.3):**
Mit diesem Eingang wird der Reglermodus eingestellt (0: manuell; 1: automatisch). Dieser Blockeingang ist als sogenannter „IN-OUT“-Eingang definiert, d.h. er kann sowohl von außen als auch von innen, vom Regelalgorithmus, verändert werden. Dies wiederum bedeutet, dass dieser Eingang nicht mit dem Ausgang eines anderen Funktionsbausteins verschaltet werden sollte.
- **ADAP (Offset 101.4):**
Mit diesem Eingang wird die Adaption des Reglers aktiviert (0: Adaption aus; 1: Adaption an). Dieser Blockeingang ist als sogenannter „IN-OUT“-Eingang definiert, d.h. er kann sowohl von außen als auch von innen, vom Regelalgorithmus, verändert werden. Dies wiederum bedeutet, dass dieser Eingang nicht mit dem Ausgang eines anderen Funktionsbausteins verschaltet werden sollte.

- **RESET (Offset 101.5):**
Mit diesem Eingang wird bestimmt, ob der adaptive Regler zurückgesetzt werden soll. Rücksetzen bedeutet, dass der Regler alle evtl. vorher gesammelten Informationen über das Prozessverhalten verliert und demzufolge neu optimiert werden muss (0: kein Rücksetzen; 1: Rücksetzen). Dieser Blockeingang ist als sogenannter „IN-OUT“-Eingang definiert, d.h. er kann sowohl von außen als auch von innen, vom Regelalgorithmus, verändert werden. Dies wiederum bedeutet, dass dieser Eingang nicht mit dem Ausgang eines anderen Funktionsbausteins verschaltet werden sollte.
- **RNG_ADA (Offset 101.6):** In diesem Auswahlfeld wird definiert, ob der Wertebereich der Prozessvariablen (Istwert) angepasst werden soll. Ist dieses Feld aktiviert, so wird der Wertebereich, wenn sich die Prozessvariable der Obergrenze nähert, um 50% nach oben verschoben. Analog wird der Wertebereich, wenn sich die Prozessvariable der Untergrenze nähert, um 50% nach unten verschoben (0: keine Anpassung des Wertebereichs; 1: Anpassung des Wertebereichs).
- **SUB_ZER (Offset 101.7):** Ist dieses Feld aktiviert, so werden bei einem variablen Wertebereich (RNG_ADA = 1) für die Untergrenze keine Werte unter 0 zugelassen (0: keine Werte unter 0 erlaubt; 1: Werte unter 0 erlaubt).

2.2. **Ausgänge:**

- **LMN (Offset 72):**
Stellgrößen Ausgang (LMN: Manipulated Variable).
- **SP (Offset 68):**
Sollwert des Regelkreises.
- **IDENT (Offset 102.0):**
Ist die Adaption eingeschaltet und erkennt der Algorithmus eine „genügende“ Anregung der Prozessvariablen, so wird die Identifikation, d.h. die Berechnung bzw. Anpassung des Prozessmodells aktiviert. Dies wird durch Setzen dieses Ausgangs angezeigt.
- **VAL_M (Offset 102.1):**
Das geschätzte Prozessmodell wird, bevor es zur Reglerparameteroptimierung herangezogen wird, umfangreichen Validierungsschritten unterzogen. Erst wenn alle Überprüfungen ein positives Ergebnis zeigen, wird das Prozessmodell freigegeben. Dies wird durch Setzen dieses Ausgangs angezeigt.
- **ORIG_M (Offset 102.2):**
Dieser Ausgang zeigt an, dass bereits ein erstes gültiges Prozessmodell gefunden wurde und der Regler demzufolge auch in den automatischen Modus geschaltet werden kann.
- **QAUTO (Offset 101.3):**
Dieser Ausgang zeigt den Reglermodus an (0: manuell; 1: automatisch).

- QADAP (Offset 101.4):
Dieser Ausgang zeigt an, ob der Regler gerade adaptiv arbeitet (0: Adaption aus; 1: Adaption an).
- QLMN_SEL (Offset 100.3):
Mit diesem Ausgang steht der Track-Modus (LMN_SEL) auch für die Bearbeitung in anderen Funktionsbausteinen zur Verfügung.
- QSP_EXT (Offset 100.0):
Mit diesem Ausgang steht der Zustand der Sollwertbeschaltung (SP_EXT_ON) auch für die Bearbeitung in anderen Funktionsbausteinen zur Verfügung.
- QILCK (Offset 101.1):
Mit diesem Ausgang steht der Verriegelungs-Zustand (ILCK) auch für die Bearbeitung in anderen Funktionsbausteinen zur Verfügung.
- ER (Offset 76):
Mit diesem Ausgang steht der Regelfehler, d.h. die Abweichung des Istwertes vom Sollwert (SP-PV), auch für die Bearbeitung in anderen Funktionsbausteinen zur Verfügung.
- F_SCAN:
Im ersten aktiven Abtastschritt nach der Konfiguration / Programmierung (F_SCAN = 0) eines Reglers müssen gewisse interne Instanzvariablen initialisiert werden. Der Wert „F_SCAN“ wird im Zuge dieser Initialisierung auf 1 gesetzt. Dies hat zur Folge, dass mit diesem Indikator in den weiteren Abtastschritten die Initialisierungsschritte übersprungen werden können.
- VERSIO (Offset 84):
Versionsnummer des adaptiven Reglers (z.B. „Rev. 2.0“).

LMN_IN und LMN, SP_OP und SP, AUTO und QAUTO, ADAP und QADAP, LMN_SEL und QLMN_SEL, SP_EXT_ON und QSP_EXT sowie ILCK und QILCK haben jeweils dieselben Offsets (72, 68, 101.3, 101.4, 100.3, 100.0, 101.1). Mit diesem Konstrukt wird u.a. erreicht, daß der adaptive Regler Blockeingangswerte korrigieren, d.h. überschreiben kann. Sind z.B. noch keine Einstellparameter ermittelt, so macht es keinen Sinn, den Regler in den Automatik-Modus zu schalten. Tut der Bediener (Anlagenfahrer) dies dennoch (AUTO, 101.3), so kann der Regelalgorithmus über den entsprechenden Blockausgang (QAUTO, 101.3) den betroffenen Merker zurücksetzen.

3. Einstellen der Reglerparameter

Wird ein neuer Regler konfiguriert oder ein existierender initialisiert, so besitzt der Regelalgorithmus zunächst keine Information über das Prozeßverhalten. Die Regleroptimierung, die auf einem geschätzten Prozeßmodell aufsetzt, kann demzufolge nicht durchgeführt werden. In dieser Situation verhindert der Algorithmus ein Umschalten in den Automatik-Modus. Durch manuelle Prozeßanregungen müssen der Modellschätzung Informationen über die Prozeßcharakteristika vermittelt werden.

Zunächst benötigt der Regler aber grundlegende Informationen über das dynamische Verhalten des Prozesses (Übergangszeit) und über eventuelle Totzeitanteile. Die Übergangszeit (siehe Bild 3.1) ist definiert sowohl für Ausgleichsprozesse als auch für Prozesse mit integrierendem Verhalten. Bei Prozessen mit Ausgleich ist die Übergangszeit die Zeit, die der Prozeß benötigt, um nach einem Stellgrößenprung von einem stationären Zustand in den nächsten zu gelangen.

Bei integrierenden Prozessen ist die Übergangszeit die Zeit, die der Prozeß benötigt, um aus einem stationären Zustand (Beharrungszustand) bei einer n-prozentigen Stellgrößenänderung n/2 Prozent des Prozeßvariablenbereiches zu überwinden (z.B. 20%-ige Stellgrößenänderung -> 10%-ige Änderung der Prozeßvariablen). Die Übergangszeit muß nur als

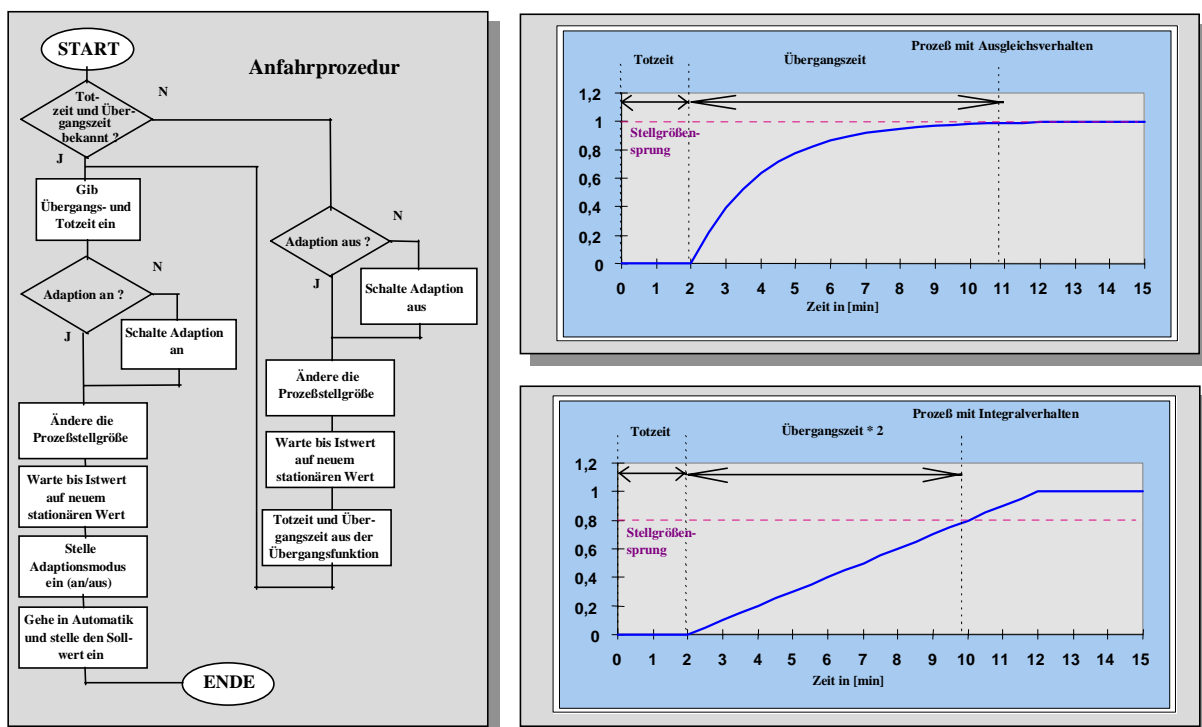


Bild 3.1: Ablaufschema zum Anfahren des Reglers

ungefährer Wert in [min] eingegeben werden. Der Regelalgorithmus ist so robust, daß der eingegebene Wert fünfmal größer oder fünfmal kleiner als die reale Übergangszeit sein kann, ohne daß die resultierende Regelqualität beeinträchtigt wird. Die Totzeit [min] sollte dagegen mit einem höheren Genauigkeitsgrad eingegeben werden.

i.p.a.s.-systeme

Sind diese Zeiten nicht bekannt, so müssen sie zuerst durch einen Stellgrößensprung bei abgeschalteter Adaption und anschließendem Ablesen der Kennwerte aus dem Prozeßvariablenverlauf ermittelt werden. Während der eigentlichen Lernphase (Adaption an!!) kann eine klassische Übergangsfunktion als Folge eines Stellgrößensprungs aufgenommen werden. Die Stellgröße kann in der Lernphase aber auch beliebig oft angepaßt werden. So ist es z.B. denkbar, die Prozeßvariable durch mehrfach manuelles Ändern der Stellgröße zum Sollwert zu führen (manuelles Einregeln). Sobald der Algorithmus sein erstes gültiges Prozeßmodell gefunden hat, kann der Regler in den Automatik-Modus geschaltet werden, d.h. die Verriegelung, die den Regler im manuellen Modus hält, wird aufgehoben. Bei den meisten Prozessen ist es nicht notwendig, den Regler im kontinuierlich-adaptiven Modus zu betreiben. Der Regler kann dort mit einem festen Reglerparametersatz (nach Abschalten der Adaption) arbeiten.

4. Sichern der Reglerparameter

Nachdem die Parameter der Regler nach obiger Vorgehensweise optimiert sind, sollten diese gesichert werden, um die Optimierprozeduren nicht nach jedem S7-Neustart durchführen zu müssen. Dies geschieht, indem die entsprechenden Instanzdatenbausteine vom STEP-7 „on-line-Fenster“ in das „off-line-Fenster“ kopiert werden und das Gesamtprojekt gesichert wird.

5. Kontinuierliche Adaption

Bei Prozessen mit nichtlinearen Anteilen oder bei Prozessen mit zeitveränderlichen Verhaltensmustern kann es notwendig sein, den Regler im kontinuierlich-adaptiven Modus zu betreiben. Hierdurch wird der Regler über die Prozeßmodellanpassung und das anschließende Nachfahren der Reglerparameter den sich ändernden Prozeßeigenschaften angepaßt.

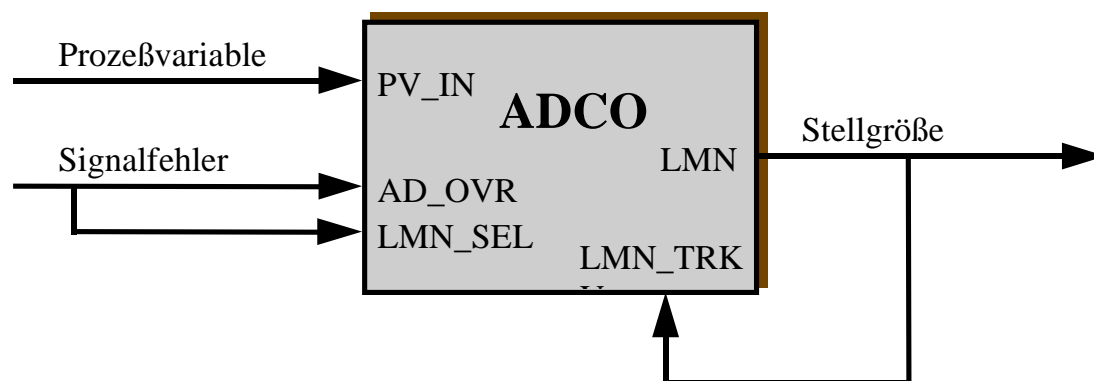


Bild 5.1: Blockschaltbild bei kontinuierlicher Adaption

Da die Prozeßidentifikation des Reglers alle Prozeßein-/ausgangswerte in das Prozeßmodell einbringt, sollten Vorkehrungen gegen fehlerhafte Prozeßsignale getroffen werden. In Bild 5.1 schaltet ein Prozeßsignalfehler (sofern erfaßbar) die Adaption aus und den Regler gleichzeitig in den Ausgangs-Track Modus. Da die Stellgröße auf den Trackeingang geführt wird, wird der Reglerausgang bei einem Signalfehler auf dem letzten gültigen Wert gehalten.

6. Kaskadenregelkreis

Der adaptive Zustandsregler eignet sich auch für den Einsatz im Kaskadenregelkreis. Die in Bild 6.1 gezeigte Verschaltung vereinfacht die Bedienung des Gesamtregelkreises.

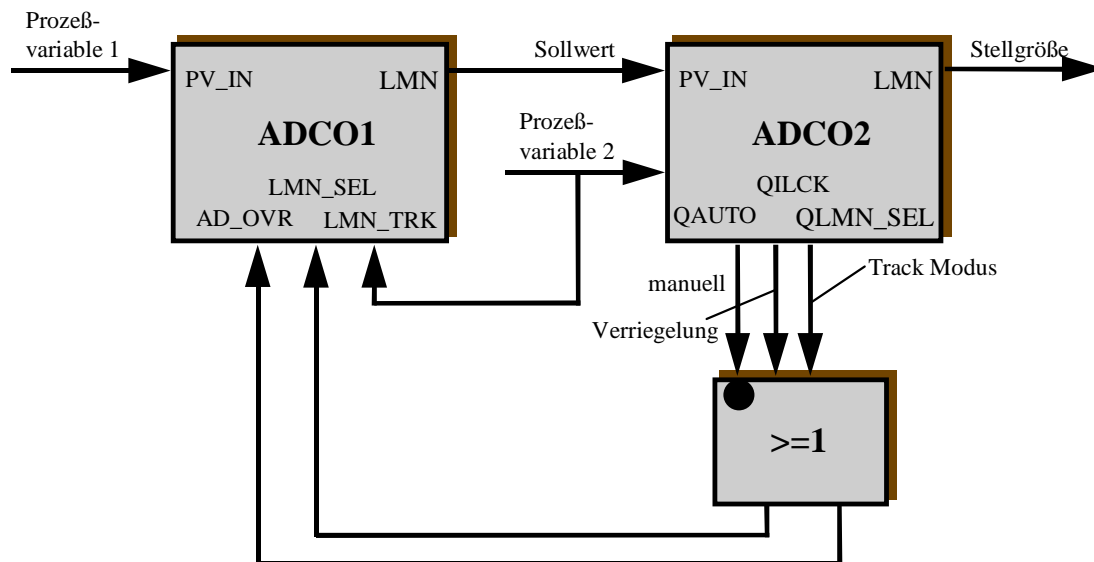


Bild 6.1: Blockschaltbild im Kaskadenkreis

Wird der innere Regler (ADCO2) in den manuellen Modus, den Track Modus, den Verriegelungsmodus oder auf lokalen Sollwert gestellt, so hat dies zur Folge, daß die Adaption des äußeren Reglers (ADCO1, Führungsregler) abgeschaltet wird und daß dieser Regler in den Ausgangs-Track Modus gesetzt wird. Der Reglerausgang von ADCO1 (Sollwert für ADCO2) folgt nun dem Wert der Prozeßvariablen des inneren Reglers. Somit kann ADCO2 jederzeit stoßfrei in den regulären Betriebsmodus zurückgeschaltet werden, was die Gesamtfunktionalität der Kaskade wiederherstellt.

Bei Bedarf muß dann lediglich die Adaption des Führungsreglers (ADCO1) wieder eingeschaltet werden.

In einem Kaskadenregelkreis sollten nie beide Regler im kontinuierlich-adaptiven Modus betrieben werden, da gegenseitige Rückkopplungen die Qualität der Regelung stark beeinträchtigen können.

7. Mehrbereichs-Regler

Neben dem einfach adaptiven Regler enthält die Software auch den sogenannten adaptiven Mehrbereichs-Regler. Die Besonderheit dieses Reglers ist, daß er in bis zu 8 Zonen aufgeteilt werden kann und daß diese Bereiche individuell optimiert werden können. Die Umschaltung zwischen den Zonen kann durch den Anwender oder aber auch bedingt durch beliebige, externe Ereignisse erfolgen.

Bei stark nichtlinearen Prozessen kann z.B. der gesamte Regelbereich in mehrere Abschnitte (bis zu 8) unterteilt werden. Hierdurch wird der Prozeß abschnittsweise „linearisiert“. Dadurch daß verschiedene, jeweils optimale Reglerparametersätze den „linearen“ Abschnitten zugewiesen werden, sollte eine wesentliche Verbesserung der Regelgüte zu erzielen sein. Ein weiteres Anwendungsgebiet liegt bei Batchprozessen, deren Charakteristik sich im Laufe einer Produktionscharge ändert. Hier können die verschiedenen Parametersätze des Mehrbereichs-Reglers abhängig vom Prozeßfortschritt aktiviert werden.

Übergangszeit, Totzeit und Sensitivität sind im Mehrbereichs-Algorithmus nur einmal definiert und gelten gleichermaßen für alle Reglerabschnitte.

Die Vorgehensweise zum Einstellen des Mehrbereichs-Reglers stimmt prinzipiell mit der in Kap. 3 beschriebenen Vorgehensweise überein. Der einzige Unterschied besteht darin, daß die Anfahrprozedur mit Ausnahme der Übergangs- und Totzeitdefinition für jeden Bereich individuell durchgeführt werden muß.

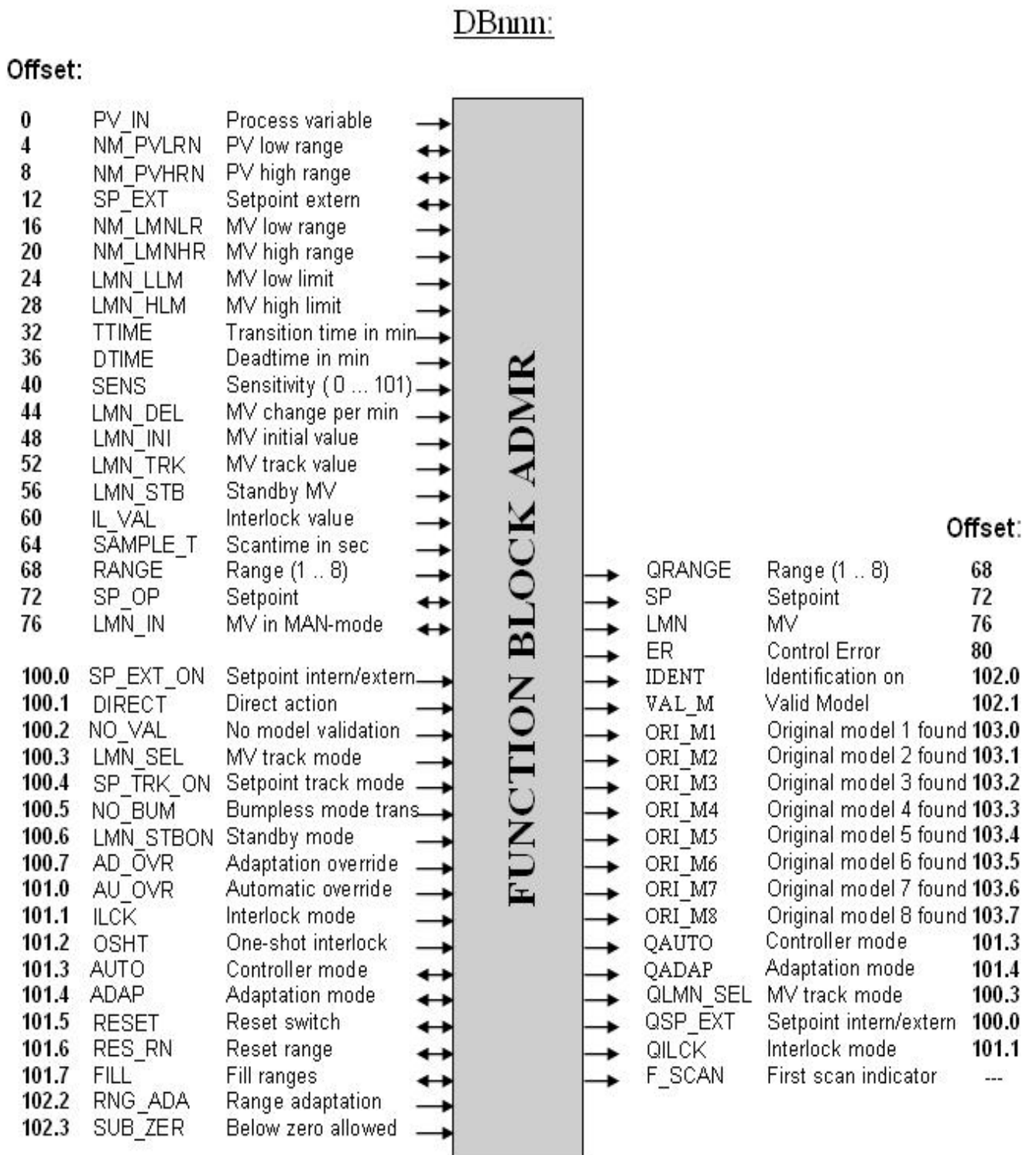


Bild 7.1: ADCO-Mehrbereichsregler (FB80)

In Bild 7.1 sind die Ein-/Ausgänge des Funktionsbausteins ADCO-Mehrbereichsregler dargestellt. Die Unterschiede zum einfachen Regler sollen nachfolgend erläutert werden:

7.1. **Eingänge:**

- RANGE (Offset 68):
Die Definition des Reglerbereichs legt fest, welcher Reglerparametersatz (1 ... 8) geladen werden und somit aktiv sein soll.
- RES_RN (Offset 101.6):
Der Mehrbereichs-Algorithmus sieht 2 Möglichkeiten der Regler-Initialisierung vor. Es können entweder der momentan angewählte Bereich (RES_RN) oder alle Reglerbereiche zusammen (RESET) initialisiert werden.
- FILL (Offset 101.7):
Ist ein bereits optimierter Bereich angewählt und wird der „FILL“-Eingang aktiviert, so wird der aktuelle Reglerparametersatz in alle noch nicht optimierten Bereiche kopiert.

7.2. **Ausgänge:**

- ORI_M1 ... ORI_M8 (Offset 103.0 ... 103.7):
Diese Ausgänge zeigen bereichsspezifisch (1 ... 8) an, ob bereits ein erstes gültiges Prozeßmodell gefunden wurde und der Regler demzufolge auch in den Automatik-Modus geschaltet werden kann.
- QRANGE (Offset 68):
Dieser Ausgang enthält die Nummer (1 ... 8) des momentan angewählten Bereichs.

Neben den für den einfachen Regler geltenden Offsets für Ein-/Ausgänge gibt es beim Mehrbereichsregler noch eine weiteres Ein-/Ausgangspaar (RANGE, QRANGE; 68). Diese Einstellung dient dazu, den Eingabebereich für „RANGE“ auf den Wertebereich 1 ... 8 zu begrenzen. Gibt der Bediener (Anlagenfahrer) hier z.B. den Wert 10 ein, so kann der Algorithmus über den entsprechenden Blockausgang diese Fehleingabe durch Überschreiben des Merkeroffsets 68 korrigieren, d.h. der Algorithmus kann in diesem Beispiel den Merkerinhalt auf den Wert 8 setzen.

8. Besonderheiten zum Betrieb auf S7-Steuerungen

8.1. Systemvoraussetzungen

- CPU 314 (6ES7314-1AG14-0AB0)
- CPU 317 und 318
- CPU 4xx
- SAIA xx7
- Step7 Basis V4.02.1 oder höher
- Step7 SCL V4.01 oder höher. Nur in Verbindung mit den SCL-Quellcodes der Regler notwendig, im Normalfall also nicht erforderlich.

8.2. Installation der Regler-Bibliothek

- Deinstallieren Sie die Bibliothek mit den Funktionsbausteinen S7_ADCO_LIB.ZIP in das Bibliotheks-Verzeichnis innerhalb des SIMATIC-Managers auf Ihrer Festplatte. Das Verzeichnis heißt üblicherweise C:\SIEMENS\TEP7\S7LIBS.
- Nach dem Entpacken befindet sich die Bibliothek in dem neuen Verzeichnis C:\SIEMENS\TEP7\S7LIBS\ADCO.
- Jetzt können Sie Ihr Step7-Projekt öffnen.
- Rufen Sie z.B. den OB36 in den KOP/FUP/AWL-Editor auf.
- Öffnen Sie die Liste der verfügbaren Programmelemente (Button auf der Button-Leiste oben).
- Öffnen Sie den Baum „Bibliotheken“
- Hier finden Sie jetzt die beiden Funktionsbausteine FB50 (ADCO) und FB80 (ADMR).

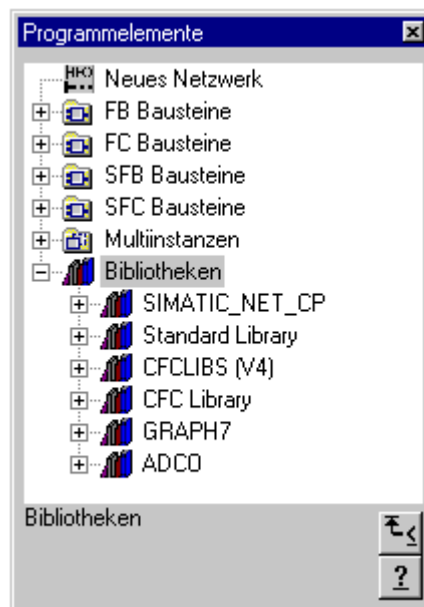


Bild 8.1: Regler-Bibliothek

8.3. *Installation der Beispiel-Applikation*

- Dearchivieren Sie das Projekt S7_ADCO.ZIP
- Öffnen Sie das Projekt. Es hat den Namen S7_ADCO
- Ändern Sie ggf. die konfigurierte Hardware ab, sodass sie auf Ihre Testinstallation paßt.
- Laden Sie das AG mit allen Bausteinen.

8.4. *Erklärung der Bausteine*

- FB50 Adaptiver Einbereichsregler ADCO
- FB60 LAG-Baustein zur Streckensimulation
- FB80 Adaptiver Mehrbereichsregler ADMR

- OB36 Weckalarm-OB zur zyklischen Abarbeitung der beiden Reglerbausteine und der Streckensimulation mit der Zykluszeit 50 msec.
- OB100 Wiederanlauf-OB (Grundeinstellung der Regler)
- OB1 Nicht benutzt
- OB80 Nicht benutzt

- DB10 Datenbaustein für ADCO zur Kommunikation mit dem Regler über VARIABLEN STEUERN oder über ein Visualisierungssystem
- DB30 Datenbaustein für ADMR zur Kommunikation mit dem Regler über VARIABLEN STEUERN oder über ein Visualisierungssystem
- DB50 Instanz-DB für ADCO
- DB60 Instanz-DB für LAG für ADCO
- DB80 Instanz-DB für ADMR
- DB90 Instanz-DB für LAG für ADMR

- VAT10 Variablen-tabelle zur Beobachtung / Bedienung von ADCO
- VAT30 Variablen-tabelle zur Beobachtung / Bedienung von ADMR

8.5. Arbeiten mit der Test-Applikation

Die Test-Applikation umfaßt zwei Regelkreise.

- Ein Regelkreis besteht aus dem Regler ADCO und einer Streckensimulation über einen LAG-Baustein.
- Der zweite Regelkreis besteht aus dem Regler ADMR und einer Streckensimulation über einen LAG-Baustein.

Die Regler laufen ca. 2 Tage nach dem 1. Abarbeitungszyklus. D.h., dass nach dem Ablauf der Demo-Zeit von ca. 2 Tagen ein Neustart der S7 stattfinden muss, um weitere 2 Tage regeln zu können.

8.6. Nach dem Start des AG

... wird der OB100 durchlaufen. Hier werden zur Erleichterung des Tests sinnvolle Grundparameter an den einzelnen Reglern eingestellt.

8.7. Zyklischer Aufruf der Regler

Die Regler werden im OB36 zyklisch, und zwar mit der Zykluszeit von 0,05 sec aufgerufen. Es ist notwendig, die Regler in einem Weckalarm-OB abzuarbeiten, um eine definierte Zykluszeit zu gewährleisten. Die Wahl des Weckalarm-OB obliegt dem Projektteur und ist abhängig von der benötigten Zykluszeit der Regelkreise.

8.8. Bedienen der Regler

Die Bedienung der Regler kann über

- mitgelieferte Variablen tabellen VAT10 (ADCO) bzw. VAT30 (ADMR).
- ein Visualisierungssystem (FIX, WinCC o.ä.) erfolgen.

8.9. Bedienung des ADCO (FB50, DB50 und DB10)

1. Rufen Sie die VAT10 in *Variable beobachten und steuern* auf.
2. Vergrößern Sie die Tabelle auf das Maximum, damit Sie möglichst alle Daten beobachten können.
3. Gehen Sie online und beobachten Sie die Daten.
4. Die Tabelle ist in verschiedene Abschnitte gegliedert. Die beiden ersten Abschnitte beinhalten die wichtigeren Daten.
5. Der Symbolische Name stimmt mit der Dokumentation zu den Reglern (WinWord-Datei) überein.
6. Nach dem Laden des AG sollte der Regler im Zustand MANUELL sein ("ADCO-Daten".AUTO_or_QAUTO = 0) und kein Prozeßmodell aufweisen ("ADCO-Daten".ORIG_M = 0).
7. Schalten Sie zunächst die Adaption an (über Button *Variablen steuern*) und kontrollieren Sie, ob die Übergangszeit ("ADCO-Daten".TTIME) = 1.0 ist.

8.10. Um den Regler adaptieren zu lassen, verfahren Sie wie folgt :

1. Tragen Sie in "ADCO-Daten".LMN_IN_or_LMN einen Wert ein (z.B. 80.0). hierdurch verändern Sie den Stellausgang und erzeugen somit einen Sprung auf der Regelstrecke.
2. "ADCO-Daten".PV_IN muss sich jetzt verändern.
3. "ADCO-Daten".IDENT muß jetzt auf 1 wechseln (war vorher = 0).
4. "ADCO-Daten".VAL_M und "ADCO-Daten".ORIG_M müssen = 0 sein, da beim ersten Start des AG nach der Installation des Demo-Projektes ein RESET auf den Regler ausgeführt wurde und er bis dahin noch kein gültiges Prozeßmodell hatte.
5. Nachdem "ADCO-Daten".PV_IN eingeschwungen ist, sollten "ADCO-Daten".VAL_M und "ADCO-Daten".ORIG_M = 1 sein.
6. Jetzt sind die Voraussetzungen erfüllt, um den Regler über *Variablen steuern* in die Betriebsart AUTO zu schalten ("ADCO-Daten".AUTO_or_QAUTO = 1).
7. Jetzt kann der Sollwert über *Variable steuern* geändert werden. Der "ADCO-Daten".PV_IN folgt danach dem Sollwert.

8.11. Bedienung des ADMR (FB80, DB80, DB30)

Die Vorgehensweise ist analog zu ADCO.

Die Bedienung erfolgt über VAT30.

Es ist allerdings zu beachten, daß es sich hierbei um den Mehrbereichsregler handelt. Er kann bis zu 8 Prozeßmodelle identifizieren. Bevor der Regler adaptieren soll, ist ein Prozeßmodell über RANGE einzustellen. Wenn sodann die Adaption eingeleitet wird, wird das gefundene Prozeßmodell diesem RANGE zu geordnet.

Auf diese Art und weise können durch Einstellen des RANGE bis zu 8 verschiedene Prozeßmodelle identifiziert und gespeichert werden.

8.12. Vorschläge zum testen und probieren

Die folgenden Testszzenarien sind mit Sicherheit interessant :

- Betrieb des Reglers unter einer höheren Zykluszeit (z.B. 100 ms)
- Verändern verschiedener Parameter (TTIME, SENS, ...) zur Erkennung der Robustheit des Reglers
- Programmierung zusätzlicher Strecken
- Betrieb an einem realen oder Laborprozeß
- Hinzufügen von zusätzlichen Regelkreisen, um z.B. die Rechenleistung und den Ressourcenverbrauch im AG herauszufinden.
- Anbindung an WinCC (bitte in diesem Falle Rücksprache mit uns halten, da wir schon unter PCS7 eine solche Anbindung mit einem Regler-Einzelbild realisiert haben und dieses gerne zur Verfügung stellen).

9. Tipps und Tricks

- Prinzipiell kann die Lernphase zur Ermittlung eines Prozeßmodells mit anschließender Optimierung des Zustandsregler zu jedem beliebigen Zeitpunkt gestartet werden. In der ersten Lernphase (d.h. nach der Konfiguration des Reglers oder nach einer Initialisierung) soll aber darauf geachtet werden, daß die Suche nach einem Prozeßmodell in einem (annähernd) statischen Arbeitspunkt beginnt und in einem anderen, wiederum (annähernd) statischen Arbeitspunkt endet. Der Grund hierfür ist, daß beim Übergang aus einer statischen in eine dynamische Phase und beim Übergang aus einer dynamischen in eine statische Phase die jeweils „beste Prozeßinformation“ in das Prozeßmodell transferiert werden kann (siehe Bild 8.1). Diese Vorgehensweise macht sich später, im geschlossenen Regelkreis in einer sehr guten Regelgüte bemerkbar. Eine Nachoptimierung (aufbauend auf einem bereits existierenden Prozeßmodell) kann jedoch auch während eines dynamischen Übergangs gestartet werden.

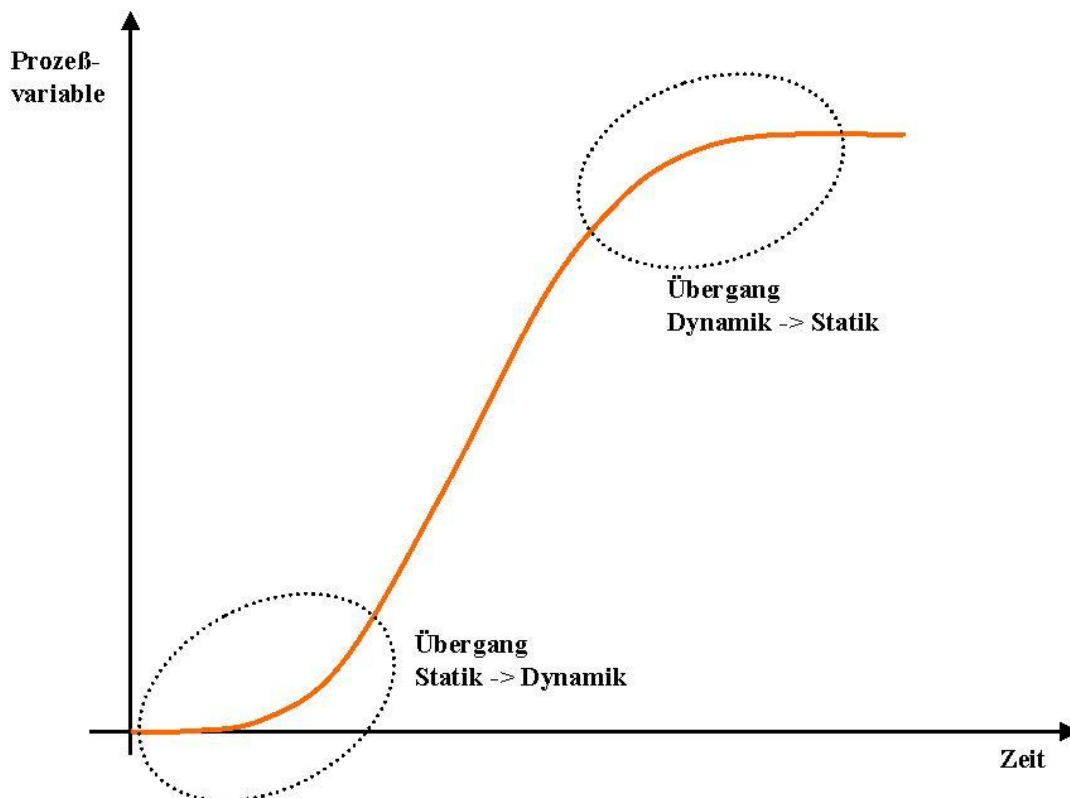


Bild 9.1: Übergangsphasen mit informationstechnisch hohem Gehalt

- Wird der Regler mit kontinuierlicher Adaption betrieben, so ist es sinnvoll, die Änderungsgeschwindigkeit der Stellgröße (MV_DEL) zu begrenzen. Diese Begrenzung führt dazu, daß der Regelalgorithmus bei einem falsch geschätzten Prozeßmodell, das die Validierungsroutinen nicht bemerken (dies sollte nur äußerst selten vorkommen), so stark oszilliert, daß die Prozeßvariable einen annähernd konstanten Wert annimmt und das Prozeßmodell aus diesem Grund nicht mehr korrigiert werden kann. Auf welchen Wert die Änderungsgeschwindigkeit begrenzt werden sollte, hängt von dem zu regelnden Prozeß und den Anforderungen an die Regelgüte ab, so daß hier kein allgemein gültiger Wert angegeben werden kann.
- Beinhaltet das Prozeßverhalten einen signifikanten Totzeitanteil, so sollte die bei der Regleroptimierung einzugebende Totzeit (DTIME) immer kleiner als oder höchstens so groß wie die reale Totzeit sein. Der Grund hierfür ist, daß die Regelgüte nur relativ langsam sinkt, wenn eine zu kleine Totzeit eingegeben wird. Wird die Totzeit jedoch zu groß gewählt, so verschlechtert sich das Regelverhalten sehr schnell.
- Um eine optimale Regelgüte zu erreichen, ist eine möglichst genaue zyklische Abtastung (Äquidistanz!!) notwendig. Dies bedeutet, daß der adaptive Regler von einem zyklischen OB (Organisationsbaustein) getriggert werden muß. Ein Betreiben des Reglers im SPS-Modus (Abarbeitung des SPS-Programms in einer Endlosschleife, d.h. Triggerung durch OB1) würde die Qualität der Regelung negativ beeinflussen. Darüberhinaus ist darauf zu achten, daß die Zykluszeit, mit der der adaptive Algorithmus bearbeitet wird, nicht kürzer ist als die Zykluszeit, mit der die Prozeßvariable als analoger Eingang aufgefrischt wird. Wird diese Forderung nicht beachtet, d.h. wird der Algorithmus bei einem dynamischen Übergang mehrfach mit demselben Prozeßvariablenwert bearbeitet, so kann die Qualität der Regelung ebenfalls negativ beeinflußt werden.
- Wenn der Prozess bei verschiedenen Übergängen unterschiedliche dynamische Charakteristiken besitzt (z.B. kann ein Temperaturprozess beim Aufheizen und Abkühlen vollkommen unterschiedliche Charakteristiken haben), dann sollte der schnellere Übergang benutzt werden, um den Regler zu optimieren. Unter der Voraussetzung, dass das Abkühlen im Vergleich zum Aufheizen die schnellere Dynamik besitzt, sollte man bei obigem Beispiel den Prozess zunächst mit abgeschalteter Adaption aufheizen. Anschließend sollte die Adaption eingeschaltet und die Kühlung aktiviert werden. Der resultierende Regler (mit konstanten Reglerparametern) beherrscht nun beide Übergänge mit einer guten Regelqualität.

- Wenn der Regler zu stark auf den Prozess einwirkt, so dass sich ein schwingendes Stellgrößenverhalten und demzufolge eine schwingende Regelgröße (bis in den instabilen Bereich) ergibt, so können folgende Eingriffe (in der unten beschriebenen Reihenfolge) durchgeführt werden:
 - Reduzierung des Sensitivitätsfaktors (SENS) – wenn nötig auf 0.
 - Begrenzung der maximalen Stellgrößenänderung (LMN_DEL). Der Wert, der hier einzustellen ist, hängt im wesentlichen von der Prozessdynamik ab. Als Ausgangspunkt kann dieser Wert so eingestellt werden, dass eine maximale Änderung der Stellgröße von 100% innerhalb eines Schwingungszyklus‘ möglich wird. Schwingt die Regelgröße mit einer Zykluszeit von z.B. 0,5 min, dann sollte LMN_DEL zunächst auf 200 eingestellt werden

- Wenn der Regler zu träge auf den Prozess einwirkt, so können folgende Eingriffe (in der unten beschriebenen Reihenfolge) durchgeführt werden:
 - Erhöhung der maximalen Stellgrößenänderung (LMN_DEL), um die Eingriffsmöglichkeiten des Reglers zu verbessern. Der Wert kann auch auf 0 gesetzt werden, um eine Begrenzung der Stellgrößenänderung zu eliminieren.
 - Erhöhung des Sensitivitätsfaktors (SENS) – wenn nötig auf 101.