

SIMATIC

S7-1200, S7-1500 Regolazione PID

Manuale di guida alle funzioni

Introduzione

1

Avvertenze di sicurezza

2

Principi di regolazione

3

Progettazione di un
regolatore software

4

Impiego di PID_Compact

5

Impiego di PID_3Step

6

Impiego di PID_Temp

7

Applicare le funzioni di base
PID

8

Funzioni ausiliarie

9

Istruzioni

10

Avvertenze di legge

Concetto di segnaletica di avvertimento

Questo manuale contiene delle norme di sicurezza che devono essere rispettate per salvaguardare l'incolumità personale e per evitare danni materiali. Le indicazioni da rispettare per garantire la sicurezza personale sono evidenziate da un simbolo a forma di triangolo mentre quelle per evitare danni materiali non sono precedute dal triangolo. Gli avvisi di pericolo sono rappresentati come segue e segnalano in ordine decrescente i diversi livelli di rischio.

 PERICOLO
questo simbolo indica che la mancata osservanza delle opportune misure di sicurezza provoca la morte o gravi lesioni fisiche.
 AVVERTENZA
il simbolo indica che la mancata osservanza delle relative misure di sicurezza può causare la morte o gravi lesioni fisiche.
 CAUTELA
indica che la mancata osservanza delle relative misure di sicurezza può causare lesioni fisiche non gravi.
ATTENZIONE
indica che la mancata osservanza delle relative misure di sicurezza può causare danni materiali.

Nel caso in cui ci siano più livelli di rischio l'avviso di pericolo segnala sempre quello più elevato. Se in un avviso di pericolo si richiama l'attenzione con il triangolo sul rischio di lesioni alle persone, può anche essere contemporaneamente segnalato il rischio di possibili danni materiali.

Personale qualificato

Il prodotto/sistema oggetto di questa documentazione può essere adoperato solo da **personale qualificato** per il rispettivo compito assegnato nel rispetto della documentazione relativa al compito, specialmente delle avvertenze di sicurezza e delle precauzioni in essa contenute. Il personale qualificato, in virtù della sua formazione ed esperienza, è in grado di riconoscere i rischi legati all'impiego di questi prodotti/sistemi e di evitare possibili pericoli.

Uso conforme alle prescrizioni di prodotti Siemens

Si prega di tener presente quanto segue:

 AVVERTENZA
I prodotti Siemens devono essere utilizzati solo per i casi d'impiego previsti nel catalogo e nella rispettiva documentazione tecnica. Qualora vengano impiegati prodotti o componenti di terzi, questi devono essere consigliati oppure approvati da Siemens. Il funzionamento corretto e sicuro dei prodotti presuppone un trasporto, un magazzinaggio, un'installazione, un montaggio, una messa in servizio, un utilizzo e una manutenzione appropriati e a regola d'arte. Devono essere rispettate le condizioni ambientali consentite. Devono essere osservate le avvertenze contenute nella rispettiva documentazione.

Marchio di prodotto

Tutti i nomi di prodotto contrassegnati con ® sono marchi registrati della Siemens Aktiengesellschaft. Gli altri nomi di prodotto citati in questo manuale possono essere dei marchi il cui utilizzo da parte di terzi per i propri scopi può violare i diritti dei proprietari.

Esclusione di responsabilità

Abbiamo controllato che il contenuto di questa documentazione corrisponda all'hardware e al software descritti. Non potendo comunque escludere eventuali differenze, non possiamo garantire una concordanza perfetta. Il contenuto di questa documentazione viene tuttavia verificato periodicamente e le eventuali correzioni o modifiche vengono inserite nelle successive edizioni.

Indice del contenuto

1	Introduzione.....	13
1.1	Scopo, convenzioni e informazioni supplementari.....	13
1.2	Guida alla consultazione della documentazione Manuali di guida alle funzioni.....	14
1.2.1	Classi di informazione Manuali di guida alle funzioni.....	14
1.2.2	Utensili di base.....	16
1.2.3	Documentazione tecnica SIMATIC.....	19
2	Avvertenze di sicurezza.....	21
2.1	Avvertenze di Cybersecurity.....	21
3	Principi di regolazione.....	22
3.1	Circuito di regolazione e attuatori.....	22
3.2	Sistemi regolati.....	24
3.3	Valori caratteristici del sistema regolato.....	25
3.4	Regolatore a impulsi.....	27
3.5	Comportamento pilota e di disturbo.....	29
3.6	Comportamento della regolazione in presenza di strutture di retroazione diverse.....	30
3.7	Scelta della struttura di regolazione nel sistema regolato dato.....	33
3.8	Impostazione dei parametri PID.....	34
4	Progettazione di un regolatore software.....	35
4.1	Panoramica dei regolatori software.....	35
4.2	Fasi di progettazione di un regolatore software.....	37
4.3	Inserire oggetti tecnologici.....	37
4.4	Configurare oggetti tecnologici.....	38
4.5	Richiamare l'istruzione nel programma utente.....	39
4.6	Caricamento degli oggetti tecnologici del dispositivo.....	40
4.7	Mettere in funzione il regolatore software.....	41
4.8	Salvare nel progetto i parametri PID ottimizzati.....	41
4.9	Uso degli oggetti delle multiistanze.....	42
4.10	Confronto di valori.....	45
4.10.1	Visualizzazione del confronto e condizioni generali.....	45
4.10.2	Confronto di valori.....	46
4.11	Vista parametri.....	48
4.11.1	Introduzione alla vista parametri.....	48

4.11.2	Struttura della vista parametri.....	50
4.11.2.1	Barra degli strumenti.....	50
4.11.2.2	Navigazione.....	51
4.11.2.3	Tabella dei parametri.....	51
4.11.3	Apertura della vista parametri.....	53
4.11.4	Preimpostazione della vista parametri.....	53
4.11.5	Utilizzo della vista parametri.....	56
4.11.5.1	Panoramica.....	56
4.11.5.2	Filtri della tabella dei parametri.....	56
4.11.5.3	Ordinamento della tabella dei parametri.....	57
4.11.5.4	Applicazione dei dati dei parametri in altri editor.....	58
4.11.5.5	Segnalazione degli errori.....	58
4.11.5.6	Modifica dei valori di avvio nel progetto.....	59
4.11.5.7	Stato della configurazione (offline).....	60
4.11.5.8	Controllo dei valori nella vista parametri.....	61
4.11.5.9	Conversione del formato di visualizzazione del valore.....	62
4.11.5.10	Crea istantanea dei valori di controllo.....	63
4.11.5.11	Comando dei valori.....	64
4.11.5.12	Confrontare i valori.....	65
4.11.5.13	Applicazione di valori del programma online come valori di avvio.....	67
4.11.5.14	Inizializzazione dei valori di impostazione nel programma online.....	68
4.12	Visualizzare il DB di istanza di un oggetto tecnologico.....	69
5	Impiego di PID_Compact.....	70
5.1	Oggetto tecnologico PID_Compact.....	70
5.2	PID_Compact dalla V2.....	71
5.2.1	Configurazione di PID_Compact dalla V2.....	71
5.2.1.1	Impostazioni di base dalla V2.....	71
5.2.1.1.1	Introduzione dalla V2.....	71
5.2.1.1.2	Tipo di regolazione dalla V2.....	72
5.2.1.1.3	Setpoint dalla V2.....	73
5.2.1.1.4	Valore istantaneo dalla V2.....	73
5.2.1.1.5	Valore di uscita dalla V2.....	74
5.2.1.2	Impostazioni del valore istantaneo dalla V2.....	75
5.2.1.2.1	Messa in scala del valore istantaneo dalla V2.....	75
5.2.1.2.2	Limiti del valore istantaneo dalla V2.....	75
5.2.1.3	Impostazioni avanzate dalla V2.....	75
5.2.1.3.1	Controllo del valore istantaneo dalla V2.....	75
5.2.1.3.2	Limitazioni PWM dalla V2.....	77
5.2.1.3.3	Valore di uscita dalla V2.....	79
5.2.1.3.4	Parametri PID V2.....	81
5.2.1.3.5	Parametri PID V3.....	84
5.2.2	Messa in servizio di PID_Compact dalla V2.....	89
5.2.2.1	Ottimizzazione iniziale dalla V2.....	89
5.2.2.2	Ottimizzazione fine dalla V2.....	90
5.2.2.3	Modo di funzionamento manuale dalla V2.....	92
5.2.3	Regolazione override con PID_Compact dalla V2.....	93
5.2.4	Simulazione di PID_Compact dalla V2 con PLCSIM.....	97

5.3	PID_Compact V1.....	98
5.3.1	Configurazione di PID_Compact V1.....	98
5.3.1.1	Impostazione di base V1.....	98
5.3.1.1.1	Introduzione V1.....	98
5.3.1.1.2	Tipo di regolazione V1.....	99
5.3.1.1.3	Setpoint V1.....	99
5.3.1.1.4	Valore istantaneo V1.....	100
5.3.1.1.5	Valore di uscita V1.....	100
5.3.1.2	Impostazioni del valore istantaneo V1.....	101
5.3.1.3	Impostazioni avanzate V1.....	102
5.3.1.3.1	Controllo valore istantaneo V1.....	102
5.3.1.3.2	Limitazioni PWM V1.....	104
5.3.1.3.3	Valore di uscita V1.....	106
5.3.1.3.4	Parametri PID V1.....	106
5.3.2	Messa in servizio di PID_Compact V1.....	109
5.3.2.1	Messa in servizio V1.....	109
5.3.2.2	Ottimizzazione iniziale V1.....	110
5.3.2.3	Ottimizzazione fine V1.....	112
5.3.2.4	Modo di funzionamento "Funzionamento manuale" V1.....	114
5.3.3	Simulazione di PID_Compact V1 con PLCSIM.....	115
6	Impiego di PID_3Step.....	116
6.1	Oggetto tecnologico PID_3Step.....	116
6.2	PID_3Step V2.....	117
6.2.1	Configurazione di PID_3Step V2.....	117
6.2.1.1	Impostazione di base V2.....	117
6.2.1.1.1	Introduzione V2.....	117
6.2.1.1.2	Tipo di regolazione V2.....	118
6.2.1.1.3	Setpoint V2.....	119
6.2.1.1.4	Valore istantaneo V2.....	119
6.2.1.1.5	Risposta di posizione V2.....	120
6.2.1.1.6	Valore di uscita V2.....	121
6.2.1.2	Impostazioni del valore istantaneo V2.....	122
6.2.1.2.1	Riporta in scala valore istantaneo V2.....	122
6.2.1.2.2	Limiti valore istantaneo V2.....	123
6.2.1.3	Impostazioni attuatore V2.....	123
6.2.1.3.1	Attuatore V2.....	123
6.2.1.3.2	Riporta in scala risposta di posizione V2.....	125
6.2.1.3.3	Valori limite di uscita V2.....	126
6.2.1.4	Impostazioni avanzate V2.....	126
6.2.1.4.1	Controllo valore istantaneo V2.....	126
6.2.1.4.2	Parametri PID V2.....	127
6.2.2	Messa in servizio di PID_3Step V2.....	130
6.2.2.1	Ottimizzazione iniziale V2.....	130
6.2.2.2	Ottimizzazione fine V2.....	131
6.2.2.3	Messa in servizio con parametri PID manuali V2.....	133
6.2.2.4	Misura del tempo di regolazione motore V2.....	134
6.2.3	Simulazione di PID_3Step V2 con PLCSIM.....	137

6.3	PID_3Step V1.....	137
6.3.1	Configurazione di PID_3Step V1.....	137
6.3.1.1	Impostazione di base V1.....	137
6.3.1.1.1	Introduzione V1.....	137
6.3.1.1.2	Tipo di regolazione V1.....	138
6.3.1.1.3	Setpoint V1.....	139
6.3.1.1.4	Valore istantaneo V1.....	139
6.3.1.1.5	Risposta di posizione V1.....	140
6.3.1.1.6	Valore di uscita V1.....	141
6.3.1.2	Impostazioni del valore istantaneo V1.....	142
6.3.1.3	Impostazioni attuatore V1.....	143
6.3.1.4	Impostazioni avanzate V1.....	146
6.3.1.4.1	Controllo valore istantaneo V1.....	146
6.3.1.4.2	Parametri PID V1.....	147
6.3.2	Messa in servizio di PID_3Step V1.....	150
6.3.2.1	Messa in servizio V1.....	150
6.3.2.2	Ottimizzazione iniziale V1.....	151
6.3.2.3	Ottimizzazione fine V1.....	152
6.3.2.4	Messa in servizio con parametri PID manuali V1.....	154
6.3.2.5	Misura del tempo di regolazione motore V1.....	154
6.3.3	Simulazione di PID_3Step V1 con PLCSIM.....	157
7	Impiego di PID_Temp.....	158
7.1	Oggetto tecnologico PID_Temp.....	158
7.2	Configurazione di PID_Temp.....	159
7.2.1	Impostazioni di base.....	159
7.2.1.1	Introduzione.....	159
7.2.1.2	Modo di regolazione.....	160
7.2.1.3	Setpoint.....	161
7.2.1.4	Valore istantaneo.....	161
7.2.1.5	Valore di uscita riscaldamento e raffreddamento.....	162
7.2.1.6	Cascata.....	164
7.2.2	Impostazioni del valore iniziale.....	165
7.2.2.1	Limiti del valore istantaneo.....	165
7.2.2.2	Riporta in scala valore istantaneo.....	165
7.2.3	Impostazioni dell'uscita.....	166
7.2.3.1	Impostazioni di base dell'uscita.....	166
7.2.3.2	Limiti e graduazione di scala del valore di uscita.....	169
7.2.4	Impostazioni avanzate.....	172
7.2.4.1	Controllo del valore istantaneo.....	172
7.2.4.2	Limitazioni PWM.....	173
7.2.4.3	Parametri PID.....	176
7.3	Messa in servizio di PID_Temp.....	183
7.3.1	Messa in servizio.....	183
7.3.2	Ottimizzazione iniziale.....	184
7.3.3	Ottimizzazione fine.....	187
7.3.4	Modo "Funzionamento manuale".....	191
7.3.5	Setpoint sostitutivo.....	192

7.3.6	Messa in servizio in cascata.....	193
7.4	Regolazione in cascata con PID_Temp.....	193
7.4.1	Introduzione.....	193
7.4.2	Creazione del programma.....	195
7.4.3	Configurazione.....	196
7.4.4	Messa in servizio.....	197
7.4.5	Setpoint sostitutivo.....	198
7.4.6	Modi di funzionamento e reazione agli errori.....	199
7.5	Regolazione multizona con PID_Temp.....	199
7.6	Regolazione override con PID_Temp.....	202
7.7	Simulazione di PID_Temp con PLCSIM.....	207
8	Applicare le funzioni di base PID.....	208
8.1	CONT_C.....	208
8.1.1	Oggetto tecnologico CONT_C.....	208
8.1.2	Configurare la differenza di regolazione CONT_C.....	208
8.1.3	Configurazione dell'algoritmo di regolazione CONT_C.....	209
8.1.4	Configurazione del valore regolante CONT_C.....	210
8.1.5	Programmazione del regolatore a impulsi.....	211
8.1.6	Messa in funzione di CONT_C.....	211
8.2	CONT_S.....	212
8.2.1	Oggetto tecnologico CONT_S.....	212
8.2.2	Configurare la differenza di regolazione CONT_S.....	213
8.2.3	Configurazione dell'algoritmo di regolazione CONT_S.....	214
8.2.4	Configurazione del valore regolante CONT_S.....	214
8.2.5	Messa in funzione di CONT_S.....	215
8.3	TCONT_CP.....	215
8.3.1	Oggetto tecnologico TCONT_CP.....	215
8.3.2	Configurare TCONT_CP.....	216
8.3.2.1	Differenza di regolazione.....	216
8.3.2.2	Algoritmo di regolazione.....	217
8.3.2.3	Valore regolante regolatore continuo.....	219
8.3.2.4	Valore regolante regolatore ad impulsi.....	220
8.3.3	Messa in funzione di TCONT_CP.....	222
8.3.3.1	Ottimizzazione TCONT_CP.....	222
8.3.3.2	Requisiti per l'ottimizzazione.....	225
8.3.3.3	Possibilità dell'ottimizzazione.....	227
8.3.3.4	Risultato dell'ottimizzazione.....	229
8.3.3.5	Ottimizzazione parallela dei canali del regolatore.....	230
8.3.3.6	Immagini degli errori e rimedi.....	231
8.3.3.7	Esecuzione dell'ottimizzazione iniziale.....	234
8.3.3.8	Esecuzione dell'ottimizzazione fine.....	235
8.3.3.9	Interruzione dell'ottimizzazione iniziale o fine.....	235
8.3.3.10	Riottimizzazione manuale nel funzionamento di regolazione.....	236
8.3.3.11	Esecuzione manuale dell'ottimizzazione fine.....	237
8.4	TCONT_S.....	238

8.4.1	Oggetto tecnologico TCONT_S.....	238
8.4.2	Configurare la differenza di regolazione TCONT_S.....	238
8.4.3	Configurare l'algoritmo di regolazione TCONT_S.....	239
8.4.4	Configurare il valore regolante TCONT_S.....	240
8.4.5	Messa in funzione di TCONT_S.....	240
9	Funzioni ausiliarie.....	241
9.1	Polyline.....	241
9.2	SplitRange.....	241
9.3	RampFunction.....	242
9.4	RampSoak.....	242
9.5	Filter_PT1.....	243
9.6	Filter_PT2.....	243
9.7	Filter_DT1.....	244
9.8	Filter_Universal.....	244
10	Istruzioni.....	245
10.1	PID_Compact.....	245
10.1.1	Novità di PID_Compact.....	245
10.1.2	Compatibilità con CPU e FW.....	249
10.1.3	Tempo di elaborazione della CPU e memoria necessaria di PID_Compact dalla V2.....	249
10.1.4	PID_Compact dalla V2.....	251
10.1.4.1	Descrizione di PID_Compact V3.....	251
10.1.4.2	Descrizione di PID_Compact V2.....	255
10.1.4.3	Modo di funzionamento di PID_Compact dalla V2.....	258
10.1.4.4	Parametri di ingresso di PID_Compact dalla V2.....	260
10.1.4.5	Parametri di uscita di PID_Compact dalla V2.....	261
10.1.4.6	Parametri di transito di PID_Compact dalla V2.....	262
10.1.4.7	Variabili statiche di PID_Compact dalla V2.....	263
10.1.4.8	Modifiche dell'interfaccia PID_Compact a partire da V2.....	273
10.1.4.9	Parametri State e Mode a partire da V2.....	275
10.1.4.10	Parametro ErrorBits dalla V2.....	278
10.1.4.11	Variabile ActivateRecoverMode dalla V2.....	280
10.1.4.12	Variabile Warning dalla V2.....	282
10.1.4.13	Variabile IntegralResetMode dalla V2.....	283
10.1.4.14	Esempio di programma per PID_Compact V2.....	284
10.1.5	PID_Compact V1.....	292
10.1.5.1	Descrizione di PID_Compact V1.....	292
10.1.5.2	Parametri di ingresso PID_Compact V1.....	295
10.1.5.3	Parametri di uscita PID_Compact V1.....	296
10.1.5.4	Variabili statiche di PID_Compact V1.....	297
10.1.5.5	Parametri State e sRet.i_Mode V1.....	302
10.1.5.6	Parametro Error V1.....	304
10.1.5.7	Parametro Reset V1.....	306
10.1.5.8	Variabile sd_warning V1.....	307
10.1.5.9	Variabile i_Event_SUT V1.....	308
10.1.5.10	Variabile i_Event_TIR V1.....	308
10.2	PID_3Step.....	309

10.2.1	Novità di PID_3Step.....	309
10.2.2	Compatibilità con CPU e FW	311
10.2.3	Tempo di elaborazione della CPU e memoria necessaria di PID_3Step V2.x.....	312
10.2.4	PID_3Step V2.....	314
10.2.4.1	Descrizione di PID_3Step V2.....	314
10.2.4.2	Funzionamento di PID_3Step V2.....	319
10.2.4.3	Modifiche dell'interfaccia PID_3Step V2.....	322
10.2.4.4	Parametri di ingresso di PID_3Step V2.....	323
10.2.4.5	Parametri di uscita PID_3Step V2.....	325
10.2.4.6	Parametri di transito di PID_3Step V2.....	326
10.2.4.7	Variabili statiche di PID_3Step V2.....	327
10.2.4.8	Parametri State e Mode V2.....	336
10.2.4.9	Parametro ErrorBits V2.....	341
10.2.4.10	Variabile ActivateRecoverMode V2.....	343
10.2.4.11	Variabile Warning V2.....	345
10.2.5	PID_3Step V1.....	346
10.2.5.1	Descrizione PID_3Step V1.....	346
10.2.5.2	Funzionamento PID_3Step V1.....	351
10.2.5.3	Parametro di ingresso PID_3Step V1.....	354
10.2.5.4	Parametro di uscita PID_3Step V1.....	356
10.2.5.5	Variabile statica PID_3Step V1.....	357
10.2.5.6	Parametri State e Retain.Mode V1.....	364
10.2.5.7	Parametro ErrorBits V1.....	371
10.2.5.8	Parametro Reset V1.....	372
10.2.5.9	Variabile ActivateRecoverMode V1.....	373
10.2.5.10	Variabile Warning V1.....	374
10.2.5.11	Variabile SUT.State V1.....	375
10.2.5.12	Variabile TIR.State V1.....	375
10.3	PID_Temp.....	376
10.3.1	Novità PID_Temp.....	376
10.3.2	Compatibilità con CPU e FW.....	376
10.3.3	Tempo di elaborazione della CPU e memoria necessaria di PID_Temp V1.....	377
10.3.4	PID_Temp.....	379
10.3.4.1	Descrizione di PID_Temp.....	379
10.3.4.2	Funzionamento di PID_Temp.....	383
10.3.4.3	Parametri di ingresso di PID_Temp.....	389
10.3.4.4	Parametri di uscita di PID_Temp.....	390
10.3.4.5	Parametri di transito di PID_Temp.....	392
10.3.4.6	Variabili statiche di PID_Temp.....	393
10.3.4.7	Parametri State e Mode PID_Temp.....	420
10.3.4.8	Parametri ErrorBits PID_Temp.....	427
10.3.4.9	Variabile ActivateRecoverMode PID_Temp.....	430
10.3.4.10	Variabile Warning PID_Temp.....	431
10.3.4.11	Variabile PwmPeriode.....	433
10.3.4.12	Variabile IntegralResetMode.....	435
10.4	Funzioni di base PID.....	436
10.4.1	CONT_C.....	436
10.4.1.1	Descrizione di CONT_C.....	436
10.4.1.2	Funzionamento di CONT_C.....	437
10.4.1.3	Schema a blocchi di CONT_C.....	439
10.4.1.4	Parametri di ingresso di CONT_C.....	440

10.4.1.5	Parametri di uscita di CONT_C.....	441
10.4.2	CONT_S.....	442
10.4.2.1	Descrizione di CONT_S.....	442
10.4.2.2	Funzionamento di CONT_S.....	443
10.4.2.3	Schema a blocchi di CONT_S.....	444
10.4.2.4	Parametri di ingresso di CONT_S.....	445
10.4.2.5	Parametri di uscita di CONT_S.....	446
10.4.3	PULSEGEN.....	447
10.4.3.1	Descrizione di PULSEGEN.....	447
10.4.3.2	Funzionamento di PULSEGEN.....	448
10.4.3.3	Modi di funzionamento di PULSEGEN.....	451
10.4.3.4	Regolazione a tre punti.....	451
10.4.3.5	Regolazione a due punti.....	454
10.4.3.6	Parametri di ingresso di PULSEGEN.....	455
10.4.3.7	Parametri di uscita di PULSEGEN.....	456
10.4.4	TCONT_CP.....	456
10.4.4.1	Descrizione di TCONT_CP.....	456
10.4.4.2	Funzionamento di TCONT_CP.....	457
10.4.4.3	Funzionamento del generatore di impulsi.....	466
10.4.4.4	Schema a blocchi TCONT_CP.....	469
10.4.4.5	Parametri di ingresso di TCONT_CP.....	470
10.4.4.6	Parametri di uscita TCONT_CP.....	471
10.4.4.7	Parametri di transito di TCONT_CP.....	472
10.4.4.8	Variabili statiche di TCONT_CP.....	472
10.4.4.9	Parametro STATUS_H.....	477
10.4.4.10	Parametro STATUS_D.....	478
10.4.5	TCONT_S.....	478
10.4.5.1	Descrizione di TCONT_S.....	478
10.4.5.2	Funzionamento di TCONT_S.....	480
10.4.5.3	Schema a blocchi di TCONT_S.....	484
10.4.5.4	Parametri di ingresso di TCONT_S.....	485
10.4.5.5	Parametri di uscita di TCONT_S.....	486
10.4.5.6	Parametri di transito di TCONT_S.....	486
10.4.5.7	Variabili statiche di TCONT_S.....	487
10.4.6	Funzioni di sistema integrate.....	488
10.4.6.1	CONT_C_SF.....	488
10.4.6.2	CONT_S_SF.....	488
10.4.6.3	PULSEGEN_SF.....	489
10.5	Polyline.....	489
10.5.1	Compatibilità con CPU e FW.....	489
10.5.2	Descrizione Polyline.....	490
10.5.3	Funzionamento Polyline.....	493
10.5.4	Parametro di ingresso Polyline.....	496
10.5.5	Parametro di uscita Polyline.....	497
10.5.6	Variabili statiche Polyline.....	497
10.5.7	Parametro ErrorBits.....	498
10.6	SplitRange.....	501
10.6.1	Compatibilità con CPU e FW.....	501
10.6.2	Descrizione di SplitRange.....	502
10.6.3	Parametro di ingresso SplitRange.....	504
10.6.4	Parametro di uscita SplitRange.....	505

10.6.5	Variabili statiche SplitRange.....	505
10.6.6	Parametro ErrorBits.....	506
10.7	RampFunction.....	508
10.7.1	Compatibilità con CPU e FW.....	508
10.7.2	Descrizione di RampFunction.....	508
10.7.3	Modo di funzionamento di RampFunction.....	513
10.7.4	Parametro di ingresso RampFunction.....	516
10.7.5	Parametro di uscita RampFunction.....	516
10.7.6	Variabili statiche RampFunction.....	517
10.7.7	Parametro ErrorBits.....	518
10.8	RampSoak.....	521
10.8.1	Compatibilità con CPU e FW.....	521
10.8.2	Descrizione di RampSoak.....	522
10.8.3	Modo di funzionamento RampSoak.....	524
10.8.3.1	Configurazione e controllo dei dati del profilo.....	524
10.8.3.2	Esecuzione del profilo.....	527
10.8.3.3	Configurazione del comportamento all'avvio - variabile statica StartMode.....	533
10.8.3.4	Configurazione del comportamento all'arresto - variabile statica StopMode.....	536
10.8.3.5	Misura del tempo di ciclo.....	538
10.8.3.6	Comportamento di abilitazione EN/ENO.....	539
10.8.4	Parametro di ingresso RampSoak.....	539
10.8.5	Parametro di uscita RampSoak.....	540
10.8.6	Parametro di transito RampSoak.....	540
10.8.7	Variabili statiche RampSoak.....	541
10.8.8	Parametro ErrorBits.....	542
10.9	Filter_PT1.....	546
10.9.1	Compatibilità con CPU e FW.....	546
10.9.2	Descrizione del filtro_PT1.....	547
10.9.3	Modo di funzionamento Filter_PT1.....	553
10.9.4	Parametro di ingresso Filter_PT1.....	555
10.9.5	Parametro di uscita Filter_PT1.....	555
10.9.6	Variabili Filter_PT1 statiche.....	555
10.9.7	Parametro ErrorBits.....	556
10.10	Filter_PT2.....	559
10.10.1	Compatibilità con CPU e FW.....	559
10.10.2	Descrizione del filtro_PT2.....	559
10.10.3	Modo di funzionamento Filter_PT2.....	566
10.10.4	Parametro di ingresso Filter_PT2.....	568
10.10.5	Parametro di uscita Filter_PT2.....	568
10.10.6	Variabili statiche Filter_PT2.....	568
10.10.7	Parametro ErrorBits.....	569
10.11	Filter_DT1.....	572
10.11.1	Compatibilità con CPU e FW.....	572
10.11.2	Descrizione del filtro_PT1.....	573
10.11.3	Modo di funzionamento Filter_DT1.....	580
10.11.4	Parametro di ingresso Filter_DT1.....	581
10.11.5	Parametro di uscita Filter_DT1.....	582
10.11.6	Variabili Filter_DT1 statiche.....	582
10.11.7	Parametro ErrorBits.....	583

10.12	Filter_Universal.....	586
10.12.1	Compatibilità con CPU e FW.....	586
10.12.2	Descrizione di Filter_Universal.....	586
10.12.3	Modo di funzionamento Filter_Universal.....	588
10.12.3.1	Parametri del filtro.....	588
10.12.3.2	Inizializzazione dei valori di uscita.....	594
10.12.3.3	Valore finale nello stato stazionario.....	595
10.12.3.4	Impiego nelle applicazioni sensibili al tempo.....	596
10.12.3.5	Ambiente di richiamo e riconoscimento automatico del tempo di ciclo.....	597
10.12.3.6	Comportamento in caso di reset.....	598
10.12.3.7	Comportamento di abilitazione EN/ENO.....	598
10.12.4	Parametro di ingresso Filter_Universal.....	599
10.12.5	Parametro di uscita Filter_Universal.....	599
10.12.6	Variabili statiche Filter_Universal.....	600
10.12.7	Parametro ErrorBits.....	601
Indice analitico.....		605

Introduzione

1.1 Scopo, convenzioni e informazioni supplementari

Scopo della documentazione

La presente documentazione offre supporto alla progettazione e alla programmazione di compiti di regolazione con i sistemi di automazione S7-1200 e S7-1500.

Nozioni di base necessarie

La comprensione della documentazione presuppone le seguenti conoscenze:

- Conoscenze generali nel campo della tecnica di automazione
- Conoscenze del sistema di automazione industriale SIMATIC
- Esperienza nell'uso di STEP 7 (TIA Portal)

Campo di validità della documentazione

La presente documentazione vale per l'impiego dei regolatori SW nelle CPU dei sistemi di automazione S7-1200, S7-1200G2 e S7-1500 in combinazione con STEP 7 (TIA Portal). Per l'utilizzo di S7-300 e S7-400 con STEP 7 (TIA Portal) sono disponibili ulteriori regolatori SW che non sono compresi nella presente documentazione. Il capitolo Panoramica dei regolatori software [\(Pagina 35\)](#) fornisce una panoramica di tutti i regolatori SW in STEP 7 (TIA Portal) e relative possibilità di impiego.

Convenzioni

Osservare le avvertenze contrassegnate nel modo seguente:

NOTA

Un'avvertenza contiene importanti informazioni sul prodotto descritto nella documentazione, sulla sua manipolazione o su una parte di documentazione alla quale occorre prestare particolare attenzione.

Industry Mall

Industry Mall è il catalogo online e il sistema di ordinazione di Siemens AG per le soluzioni di automazione e azionamento basate su Totally Integrated Automation (TIA) e Totally Integrated Power (TIP).

I cataloghi di tutti i prodotti inerenti la tecnica di automazione e azionamento sono disponibili in Internet (<https://mall.industry.siemens.com>).

Vedere anche



Pagina degli argomenti "Tecnologia SIMATIC - PID Control: Panoramica e link importanti "
<https://support.industry.siemens.com/cs/ww/de/view/109751051>
(<https://support.industry.siemens.com/cs/ww/it/view/109751051>)

1.2 Guida alla consultazione della documentazione Manuali di guida alle funzioni

1.2.1 Classi di informazione Manuali di guida alle funzioni



La documentazione per il sistema di automazione SIMATIC S7-1500 per le CPU 1513/1516pro-2 PN, SIMATIC Drive Controller per SIMATIC S7-1500 e i sistemi di periferia decentrata SIMATIC ET 200MP, ET 200SP, ET 200AL e ET 200eco PN è suddivisa in tre parti.

Questa suddivisione consente di accedere in maniera mirata ai contenuti di interesse.

La documentazione può essere scaricata gratuitamente da Internet.

(<https://support.industry.siemens.com/cs/ww/it/view/109742705>)

Informazioni di base



I manuali di sistema e il Getting Started descrivono dettagliatamente la progettazione, il montaggio, il cablaggio e la messa in servizio dei sistemi SIMATIC S7-1500, SIMATIC Drive Controller, ET 200MP, ET 200SP, ET 200AL e ET 200eco PN. Per le CPU 1513/1516pro-2 PN utilizzare le istruzioni operative corrispondenti.

La Guida in linea di STEP 7 supporta l'utente nelle fasi di progettazione e programmazione.

Esempi:

- Getting Started S7-1500
- Manuali di sistema
- Istruzioni operative ET 200pro e CPU 1516pro-2 PN
- Guida in linea TIA Portal

Informazioni sul dispositivo



I manuali di prodotto contengono una descrizione compatta delle informazioni specifiche del modulo, come proprietà, schemi dei collegamenti, curve caratteristiche e dati tecnici.

Esempi:

- Manuali del prodotto delle CPU
- Manuali del prodotto dei moduli di interfaccia
- Manuali del prodotto dei moduli digitali
- Manuali del prodotto dei moduli analogici
- Manuali del prodotto dei moduli di comunicazione
- Manuale del prodotto dei moduli tecnologici
- Manuale del prodotto delle unità di alimentazione
- Manuale del prodotto delle BaseUnit

Informazioni generali



I manuali di guida alle funzioni contengono descrizioni dettagliate su argomenti generali inerenti a SIMATIC Drive Controller e al sistema di automazione S7-1500.

Esempi:

- Manuale di guida alle funzioni Diagnostica
- Manuale di guida alle funzioni Comunicazione
- Manuali di guida alle funzioni Motion Control
- Manuale di guida alle funzioni Server web
- Manuale di guida alle funzioni Tempi di ciclo e di reazione
- Manuale di guida alle funzioni PROFINET
- Manuale di guida alle funzioni PROFIBUS

Informazioni sul prodotto

Eventuali modifiche e integrazioni dei manuali vengono descritte in un file di informazioni sul prodotto. Le informazioni sul prodotto sono da considerarsi prioritarie rispetto a quelle contenute nel manuale del prodotto e nel manuale di sistema.

Le informazioni più aggiornate sul prodotto sono disponibili in Internet:

- S7-1500/ET 200MP (<https://support.industry.siemens.com/cs/it/it/view/68052815>)
- SIMATIC Drive Controller (<https://support.industry.siemens.com/cs/de/it/view/109772684/it>)
- Motion Control (<https://support.industry.siemens.com/cs/de/it/view/109794046/it>)
- ET 200SP (<https://support.industry.siemens.com/cs/it/it/view/73021864>)
- ET 200eco PN (<https://support.industry.siemens.com/cs/ww/it/view/109765611>)

Manual Collection

Le Manual Collection raggruppano in un unico file l'intera documentazione relativa ai diversi sistemi.

Le Manual Collection sono disponibili in Internet.

- S7-1500/ET 200MP/SIMATIC Drive Controller (<https://support.industry.siemens.com/cs/ww/it/view/86140384>)
- ET 200SP (<https://support.industry.siemens.com/cs/ww/it/view/84133942>)
- ET 200AL (<https://support.industry.siemens.com/cs/ww/it/view/95242965>)
- ET 200eco PN (<https://support.industry.siemens.com/cs/ww/it/view/109781058>)

1.2.2 Utensili di base

Strumenti

Gli strumenti descritti di seguito supportano durante tutte le fasi, dalla pianificazione e la messa in servizio all'analisi del sistema.

TIA Selection Tool

TIA Selection Tool supportano durante la selezione, la configurazione e l'ordinazione di dispositivi per Totally Integrated Automation (TIA).

Come successore del SIMATIC Selection Tools , il TIA Selection Tool raggruppa in un solo strumento i configuratori già noti per la tecnica di automazione.

TIA Selection Tool consente di generare un elenco di ordinazione completo dalla selezione del prodotto o dalla configurazione del prodotto.

TIA Selection Tool è disponibile in Internet.

(<https://support.industry.siemens.com/cs/ww/it/view/109767888/en>)

SIMATIC Automation Tool

Con SIMATIC Automation Tool è possibile eseguire operazioni di massa per interventi di service e di messa in servizio su diverse stazioni SIMATIC S7, indipendentemente da TIA Portal.

SIMATIC Automation Tool mette a disposizione numerose funzioni:

- Scansione di una rete di impianto PROFINET/Ethernet e identificazione di tutte le CPU collegate
- Assegnazione di indirizzi (IP, sottorete, Gateway) e nome del dispositivo (PROFINET Device) relativo ad una CPU
- Trasmissione della data e dell'ora del PG/PC convertita in formato UTC all'unità
- Download del programma sulla CPU
- Commutazione del modo di funzionamento RUN/STOP
- Localizzazione della CPU tramite lampeggio dei LED
- Lettura delle informazioni di errore della CPU

- Lettura del buffer di diagnostica della CPU
- Reset alle impostazioni di fabbrica
- Aggiornamento del firmware della CPU e dei moduli collegati

SIMATIC Automation Tool è disponibile in Internet.

(<https://support.industry.siemens.com/cs/ww/it/view/98161300>)

PRONETA

SIEMENS PRONETA (analisi della rete PROFINET) è uno strumento di messa in servizio e di diagnostica per reti PROFINET. PRONETA Basic dispone di 2 funzioni fondamentali:

- Nell'analisi della rete si ottiene una panoramica della topologia PROFINET. È possibile confrontare una configurazione reale con un impianto di riferimento o effettuare semplici modifiche di parametri, ad es. dei nomi e degli indirizzi IP dei dispositivi.
- Il "Test IO" consente un test semplice e rapido del cablaggio e della configurazione modulare di un impianto, inclusa una documentazione del risultato del test.

SIEMENS PRONETA Basic è disponibile in Internet:

(<https://support.industry.siemens.com/cs/ww/it/view/67460624>)

SIEMENS PRONETA Professional offre ulteriori funzioni come prodotto concesso in licenza. Consente il semplice Asset Management nelle reti PROFINET e supporta l'operatore di impianti di automazione nel rilevamento automatico dei dati dei componenti impiegati attraverso una vasta gamma di funzioni:

- L'interfaccia utente (API) offre un punto di accesso alla cella di automazione per automatizzare le funzioni di scansione tramite MQTT o una cella di comando.
- La diagnostica PROFlenergy consente di rilevare e in caso di necessità modificare molto rapidamente la modalità di pausa o la disponibilità al funzionamento per i dispositivi che supportano PROFlenergy.
- L'assistente per i set di dati consente allo sviluppatore PROFINET di leggere e scrivere in modo semplice e rapido i set di dati PROFINET aciclici, senza PLC ed engineering.

SIEMENS PRONETA Professional è disponibile in Internet: (<https://www.siemens.com/proneta-professional>)

SINETPLAN

SINETPLAN, il Siemens Network Planner, supporta l'utente nella pianificazione dei sistemi e nelle reti di automazione sulla base di PROFINET. Questo tool facilita il dimensionamento professionale e predittivo dell'installazione PROFINET già nella fase di pianificazione. Inoltre SINETPLAN fornisce all'utente strumenti utili per ottimizzare la rete, sfruttare al meglio le risorse di rete e pianificare le riserve. In questo modo, già prima dell'impiego pianificato si evitano problemi durante la messa in servizio e interruzioni nella fase produttiva. Questo aumenta la disponibilità dell'impianto produttivo e contribuisce a migliorare la sicurezza operativa.

I vantaggi in sintesi

- ottimizzazione della rete grazie al calcolo del carico di rete per le singole porte
- maggiore disponibilità della produzione grazie alla scansione online e alla verifica degli impianti esistenti
- trasparenza prima della messa in servizio mediante importazione e simulazione di progetti STEP 7 esistenti
- efficienza grazie alla protezione degli investimenti nel lungo periodo e allo sfruttamento ottimale delle risorse

SINETPLAN è disponibile in Internet.

(<https://new.siemens.com/global/en/products/automation/industrial-communication/profinet/sinetplan.html>)

1.2.3 Documentazione tecnica SIMATIC

Ulteriori documenti SIMATIC che integrano le informazioni. Questi documenti e la relativa utilità si trovano ai seguenti link e codici QR.

Industry Online Support completa le possibilità di ottenere informazioni su tutti gli argomenti. Gli esempi applicativi forniscono un supporto nella soluzione dei compiti di automazione.

Panoramica della documentazione tecnica SIMATIC

Qui si trova una panoramica della documentazione relativa a SIMATIC disponibile in Siemens Industry Online Support:



Industry Online Support International

(<https://support.industry.siemens.com/cs/ww/it/view/109742705>)

Un breve video illustra dove reperire la panoramica della documentazione direttamente in Siemens Industry Online Support e come utilizzare Siemens Industry Online Support su un dispositivo mobile:



Accesso rapido alla documentazione tecnica dei prodotti di automazione tramite video (<https://support.industry.siemens.com/cs/it/it/view/109780491/>)



Video YouTube: Siemens Automation Products - Technical Documentation at a Glance (<https://youtu.be/TwLSxxRQsA>)

Conservazione della documentazione

Conservare la documentazione per poterla utilizzare in seguito.

Documentazione allegata in formato digitale:

1. Scaricarle la documentazione pertinente dopo aver ricevuto il prodotto, al più tardi prima del primo montaggio/della prima messa in servizio. Per il download, utilizzare le seguenti opzioni:

- Industry Online Support International: (<https://support.industry.siemens.com>)

Il prodotto è abbinato alla documentazione in base al numero di articolo. Il numero di articolo si trova sul prodotto e sull'etichetta dell'imballo. Ai prodotti con nuove funzioni non compatibili viene assegnato un nuovo numero di articolo e una nuova documentazione.

- ID Link:

Se il prodotto è contrassegnato con un ID Link, questo è riconoscibile come codice QR con una cornice e angoli neri in basso a destra. L'ID Link collega alla targhetta identificativa digitale del prodotto. Scansionare il codice QR sul prodotto o sull'etichetta dell'imballo con la fotocamera dello smartphone, con un lettore di codici a barre o un'app di lettura. Richiamare l'ID Link.

2. Conservare questa versione della documentazione.

Aggiornamento della documentazione

La documentazione del prodotto viene aggiornata digitalmente. In particolare se si effettua un ampliamento delle funzioni vengono fornite nuove caratteristiche funzionali nella versione più aggiornata.

1. Scaricare la versione aggiornata come descritto più sopra tramite l'Industry Online Support o l'ID Link.
2. Conservare anche questa versione della documentazione.

mySupport

mySupport consente di sfruttare al meglio il servizio Industry Online Support.

Registrazione	Per poter usufruire della funzionalità completa di mySupport è necessaria una registrazione iniziale. Dopo la registrazione è possibile creare filtri, preferiti e schede nell'area di lavoro personale.
Richiesta di assistenza	I dati personali dell'utente sono già preimpostati nelle richieste di assistenza e l'utente ha modo di controllare in qualsiasi momento lo stato di elaborazione delle richieste che ha presentato.
Documentazione	Nell'area Documentazione è possibile raggruppare la biblioteca personale.
Preferiti	Utilizzare il pulsante "Aggiungi ai preferiti di mySupport" per inserire nei preferiti contenuti particolarmente interessanti o utilizzati di frequente. Al punto "Preferiti" si trova un elenco delle voci preimpostate.
Ultimi messaggi visualizzati	Le pagine richiamate per ultime nel mySupport si trovano in "Ultimi messaggi visualizzati".
Dati CAx	L'area dei dati CAx consente di accedere ai dati attuali del prodotto per il sistema CAx o CAe. Con pochi clic è possibile configurare il proprio pacchetto di download: • immagini del prodotto, disegni quotati in 2D, modelli in 3D, schemi elettrici dell'apparecchio, file macro EPLAN • manuali, curve caratteristiche, istruzioni operative, certificati • dati di base del prodotto

mySupport è disponibile in Internet. (<https://support.industry.siemens.com/My/ww/it>)

Esempi applicativi

Gli esempi applicativi forniscono diversi strumenti ed esempi utili nella soluzione dei problemi di automazione. In questa sezione vengono illustrate soluzioni che prevedono l'interazione di più componenti del sistema, senza soffermarsi sui singoli prodotti.

Gli esempi pratici sono disponibili in Internet.

(<https://support.industry.siemens.com/cs/ww/it/ps/ae>)

Avvertenze di sicurezza

2.1 Avvertenze di Cybersecurity

Siemens commercializza prodotti e soluzioni dotati di funzioni di cybersecurity industriale che contribuiscono al funzionamento sicuro di impianti, soluzioni, macchine e reti.

Al fine di proteggere impianti, sistemi, macchine e reti da minacce cibernetiche, è necessario implementare - e mantenere continuamente - un concetto di cybersecurity industriale globale ed all'avanguardia. I prodotti e le soluzioni Siemens costituiscono soltanto una componente di questo concetto.

È responsabilità dei clienti prevenire accessi non autorizzati ai propri impianti, sistemi, macchine e reti. Tali sistemi, macchine e componenti dovrebbero essere connessi unicamente a una rete aziendale o a Internet se e nella misura in cui detta connessione sia necessaria e solo quando siano attive appropriate misure di sicurezza (ad es. impiego di firewall e segmentazione della rete).

Per ulteriori informazioni inerenti alle misure di cybersecurity industriale che possono essere implementabili potete visitare il sito
<https://www.siemens.com/cybersecurity-industry>.

I prodotti e le soluzioni Siemens vengono costantemente perfezionati per incrementarne la sicurezza. Siemens raccomanda espressamente che gli aggiornamenti dei prodotti siano effettuati non appena disponibili e che siano utilizzate le versioni più aggiornate. L'utilizzo di versioni di prodotti non più supportate ed il mancato aggiornamento degli stessi incrementa il rischio di attacchi cibernetici.

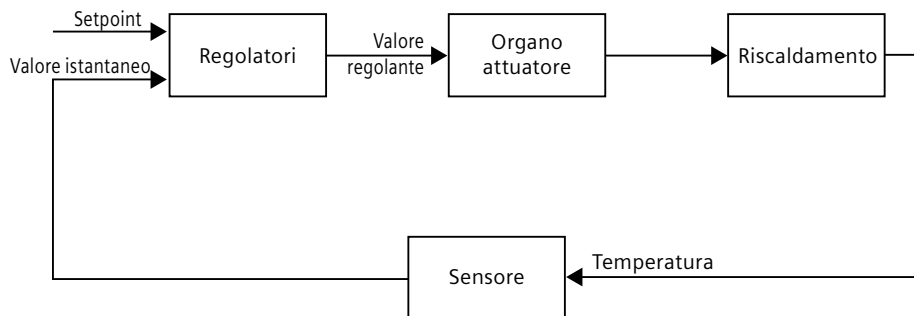
Per essere informati riguardo agli aggiornamenti dei prodotti, potete iscrivervi a Siemens Industrial Cybersecurity RSS Feed al sito
<https://www.siemens.com/cert>.

Principi di regolazione

3.1 Circuito di regolazione e attuatori

Circuito di regolazione

Un semplice esempio di un circuito di regolazione è la regolazione della temperatura ambiente tramite riscaldamento. La temperatura ambiente viene misurata con un sensore e trasmessa a un regolatore. Questo regolatore confronta la temperatura ambiente attuale con un setpoint e calcola un valore di uscita (valore regolante) per il controllo del riscaldamento.



Un regolatore PID impostato correttamente raggiunge il setpoint il più rapidamente possibile e lo mantiene costante. La variazione del valore di uscita determina la variazione del valore istantaneo, spesso con ritardo temporale. Questo comportamento deve essere compensato dal regolatore.

Attuatori

L'attuatore è parte integrante del circuito di regolazione e viene influenzato dal regolatore. Ciò determina una variazione della portata di massa ed energia.

La seguente tabella fornisce una panoramica dell'impiego degli attuatori

Impiego per	Organo attuatore
Portata di massa liquida e gassosa	Valvola, valvola a farfalla, valvola a saracinesca
Portata di massa fissa, ad es. materiali sfusi	Valvola di scarico, nastro trasportatore, canale a vibrazione
Portata di energia elettrica	Contatto di commutazione, contattore, relé, tiristore Resistenza di regolazione, trasformatore variabile, transistor

Si distinguono i seguenti attuatori:

- Attuatori proporzionali con segnale di regolazione costante

In modo proporzionale rispetto al valore di uscita vengono acquisiti il grado di apertura, l'angolo di rotazione o le posizioni. All'interno del campo di regolazione il valore di uscita influenza in modo analogo il processo.

Fanno parte degli attuatori di questo gruppo azionamenti pneumatici a molla ma anche azionamenti a motore con risposta di posizione nei quali viene creato un circuito di regolazione della posizione.

Il valore di uscita viene generato da un regolatore continuo, ad es. PID_Compact.

- Attuatori proporzionali con segnale PWM

In questi attuatori viene generato un impulso in base alla cadenza del tempo di campionamento la cui lunghezza è proporzionale al valore di uscita. L'attuatore, ad es. una resistenza di riscaldamento o un'unità di raffreddamento, viene attivato in sincronismo di clock per durate diverse a seconda del valore di uscita.

Il segnale di regolazione può assumere gli stati "On" o "Off" in modo unipolare o rappresentare in modo bipolare ad es. i valori "Aperto/Chiuso", "Avanti/Indietro", "Accelera/Frena" ecc.

Il valore di uscita viene generato da un regolatore a due punti, ad es. PID_Compact con modulazione di larghezza di impulso.

- Attuatori ad effetto integrale con segnale di regolazione a tre punti

Spesso gli attuatori vengono azionati da motori la cui durata di attivazione è proporzionale al movimento di regolazione dell'organo di regolazione di portata. Tra questi figurano ad es. valvole, valvole a farfalla e valvole a saracinesca. Nonostante le diverse forme costruttive tutti questi attuatori hanno il medesimo effetto di un componente I sull'ingresso del circuito.

Il valore di uscita viene generato da un regolatore a passi, ad es. PID_3Step.

3.2 Sistemi regolati

Le proprietà di un sistema regolato sono definite dalle condizioni della tecnica di processo e della macchina e non subiscono praticamente alcuna influenza. Per ottenere una buona regolazione è indispensabile scegliere un tipo di regolatore idoneo per questo sistema regolato e adeguarlo al comportamento temporale di quest'ultimo. È pertanto essenziale disporre di conoscenze precise sul tipo e i dati caratteristici del sistema regolato per la configurazione dei componenti P, I e D del regolatore.

Tipi di sistemi regolati

I sistemi regolati vengono classificati sulla base del loro comportamento temporale in seguito a una variazione brusca del valore di uscita.

Si distinguono i seguenti sistemi regolati:

- Sistemi regolati con compensazione
 - Sistemi regolati P
 - Sistemi regolati PT1
 - Sistemi regolati PT2
- Sistemi regolati senza compensazione
- Sistemi regolati con e senza tempo morto

Sistemi regolati con compensazione

Sistemi regolati P

Nei circuiti proporzionali il valore istantaneo segue quasi senza ritardo il valore di uscita. Il comportamento del valore istantaneo rispetto al valore di uscita viene indicato dal coefficiente proporzionale Gain del sistema regolato.

Esempi:

- Valvola a saracinesca in un sistema di tubi
- Ripartitore di tensione
- Scansione nei sistemi idraulici

Sistemi regolati PT1

In un sistema regolato PT1 il valore istantaneo varia dapprima in modo proporzionale rispetto alla variazione del valore di uscita. Col tempo la variazione del valore istantaneo rallenta, quindi ritarda, fino a raggiungere un valore finale.

Esempi:

- Sistema a molle e ammortizzatori
- Carica di componenti RC
- Serbatoio d'acqua riscaldato a vapore.

Spesso valgono le stesse costanti temporali per i processi di riscaldamento e raffreddamento o per le curve di scarica e carica. Con costanti temporali diverse la regolazione diventa notevolmente più difficile.

Sistemi regolati PT2

In un sistema regolato PT2 in presenza di un gradino del valore di uscita il valore istantaneo dapprima non varia, successivamente aumenta con passo crescente e infine tende al setpoint con passo decrescente. Il sistema regolato mostra un comportamento di trasmissione proporzionale con ritardo di 2° ordine.

Esempi:

- Regolazione della pressione
- Regolazione della portata
- Regolazione della temperatura

Sistemi regolati senza compensazione

I sistemi regolati senza compensazione mostrano un comportamento integrale. Il valore istantaneo tende a un valore infinitamente grande.

Esempio:

- Inserimento di un liquido in un recipiente

Sistemi regolati con tempo morto

Il tempo morto è sempre il tempo di esecuzione o di trasporto fino alla misurazione nell'uscita del sistema di una variazione all'ingresso dello stesso.

Nei sistemi regolati con tempo morto il valore istantaneo varia ritardato del tempo morto.

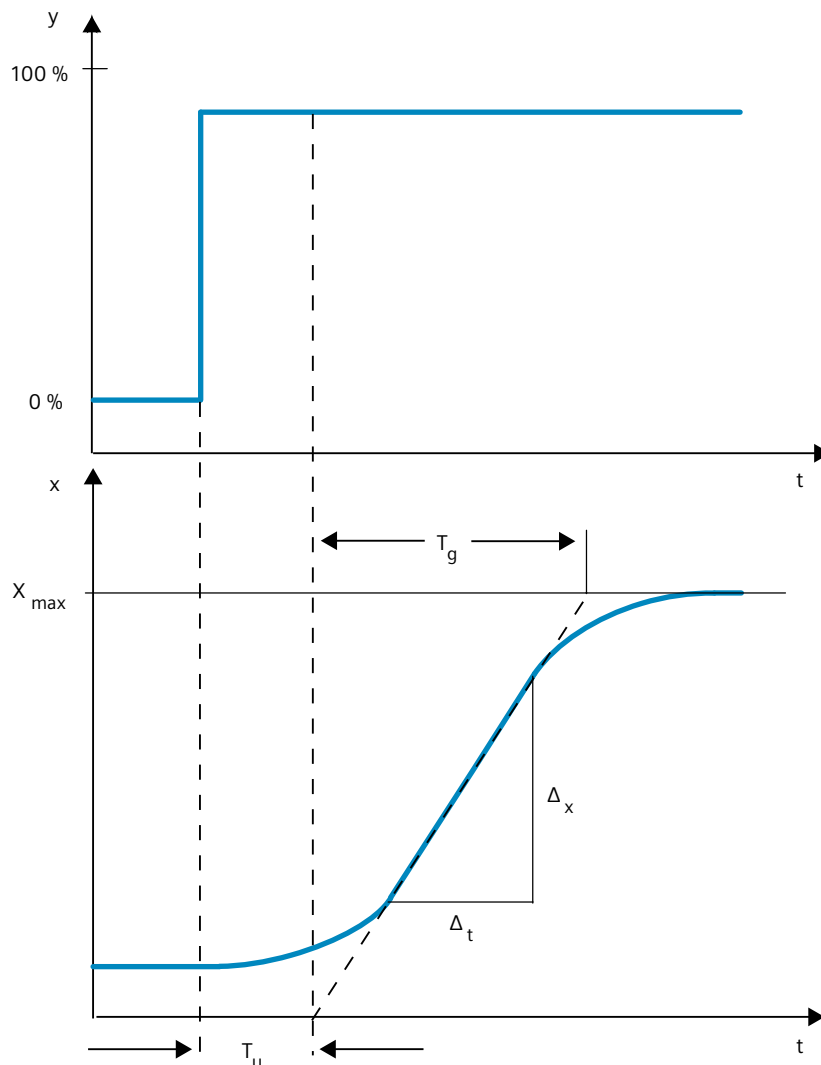
Esempio:

- Nastro trasportatore

3.3 Valori caratteristici del sistema regolato

Rilevamento del comportamento temporale in base alla risposta al gradino

Il comportamento temporale del sistema regolato può essere determinato in base all'andamento temporale del valore istantaneo x in seguito a una variazione brusca del valore di uscita y . La maggior parte dei sistemi regolati sono sistemi regolati con compensazione.



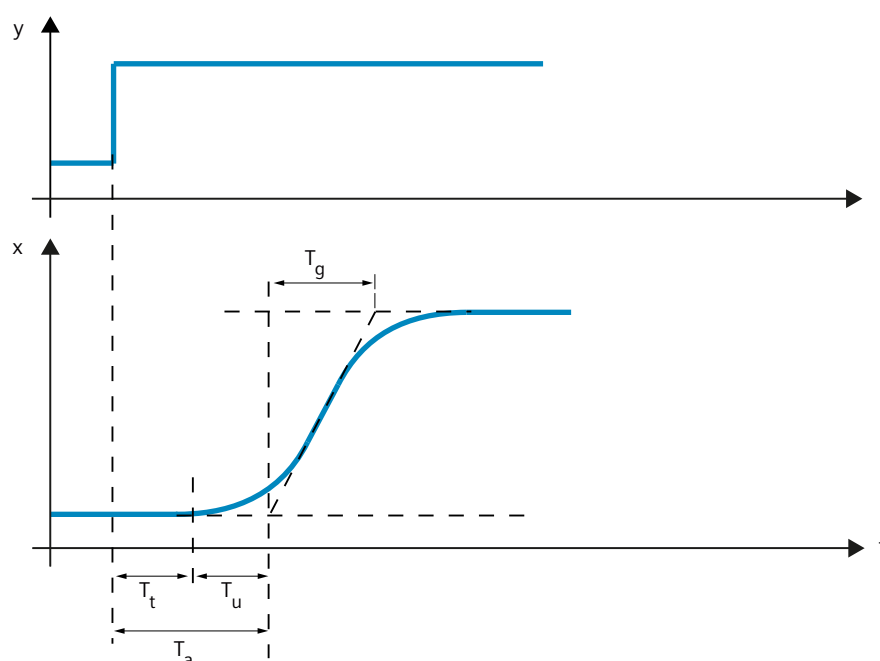
Il comportamento temporale può essere determinato, in via approssimativa, con le grandezze Tempo di ritardo T_u , Tempo di compensazione T_g e Valore massimo $X_{\max}$. Tali grandezze vengono determinate creando tangenti sul valore massimo e sul punto di flesso della risposta al gradino. In molti casi non è possibile utilizzare la funzione di trasferimento fino al valore massimo perché il valore istantaneo non deve superare determinati valori. In questi casi, per l'identificazione del sistema regolato si utilizza la velocità di incremento $v_{\max}$ ($v_{\max} = \Delta_x / \Delta_t$).

È possibile effettuare una stima della regolabilità del sistema regolato in base al rapporto T_u/T_g o $T_u \times v_{\max}/X_{\max}$. Vale quanto segue:

Tipo di circuito	T_u / T_g	Regolabilità del sistema regolato
I	$< 0,1$	buona
II	da 0,1 a 0,3	appena possibile
III	$> 0,3$	difficilmente regolabile

Effetto del tempo morto sulla regolabilità di un sistema regolato

Un sistema regolato con tempo morto e compensazione mostra il seguente comportamento in seguito a un gradino del valore di uscita.



T_t Tempo morto

T_u Ritardo

T_g Tempo di compensazione

y Valore di uscita

x Valore istantaneo

La regolabilità di un sistema regolato con tempo morto e compensazione viene determinata dal rapporto tra T_t e T_g . T_t deve essere inferiore a T_g . Deve valere:

$$T_t/T_g \leq 1$$

Velocità di reazione dei sistemi regolati

La valutazione dei sistemi regolati viene effettuata sulla base dei seguenti valori:

$T_u < 0,5 \text{ min}$, $T_g < 5 \text{ min}$ = sistema regolato veloce

$T_u > 0,5 \text{ min}$, $T_g > 5 \text{ min}$ = sistema regolato lento

Valori caratteristici di alcuni sistemi regolati

Grandezza fisica	Sistema regolato	Ritardo T_u	Tempo di compensazione T_g	Velocità di incremento $v_{\max}$
Temperatura	Forno elettrico piccolo	0,5 ... 1 min	5 ... 15 min	fino a 60 K/min
	Forno di ricottura elettrico grande	1 ... 5 min	10 ... 20 min	fino a 20 K/min
	Forno di ricottura a gas grande	0,2 ... 5 min	3 ... 60 min	1 ... 30 K/min
	Torre di distillazione	1 ... 7 min	40 ... 60 min	0,1 ... 0,5 °C/s
	Autoclavi (2,5 m ³)	0,5 ... 0,7 min	10 ... 20 min	nessuna indicazione
	Autoclavi ad alta pressione	12 ... 15 min	200 ... 300 min	nessuna indicazione
	Surriscaldatore a vapore	30 s ... 2,5 min	1 ... 4 min	2 °C/s
	Macchine per pressofusione	0,5 ... 3 min	3 ... 30 min	5 ... 20 K/min
	Estrusori	1 ... 6 min	5 ... 60 min	
	Macchine da imballaggio	0,5 ... 4 min	3 ... 40 min	2 ... 35 K/min
	Riscaldamento ambiente	1 ... 5 min	10 ... 60 min	1 °C/min
Portata	Condotta con gas	0 ... 5 s	0,2 ... 10 s	irrilevante
	Condotta con liquido	nessuna	nessuna	
Pressione	Condotta con gas	nessuna	0,1 s	irrilevante
	Caldaia a corpi cilindrici con combustione ad olio o a gas	nessuna	150 s	irrilevante
	Caldaia a corpi cilindrici con mole a battimento	1 ... 2 min	2 ... 5 min	irrilevante
Stato del serbatoio (livello)	Caldaia a corpi cilindrici	0,6 ... 1 min	nessuna indicazione	0,1 ... 0,3 cm/s
Numero di giri	Motore elettrico piccolo	nessuna	0,2 ... 10 s	irrilevante
	Motore elettrico grande	nessuna	5 ... 40 s	irrilevante
	Turbina a vapore	nessuna	nessuna indicazione	50 min ⁻¹
Tensione elettrica	Generatori piccoli	nessuna	1 ... 5 s	irrilevante
	Generatori grandi	nessuna	5 ... 10 s	irrilevante

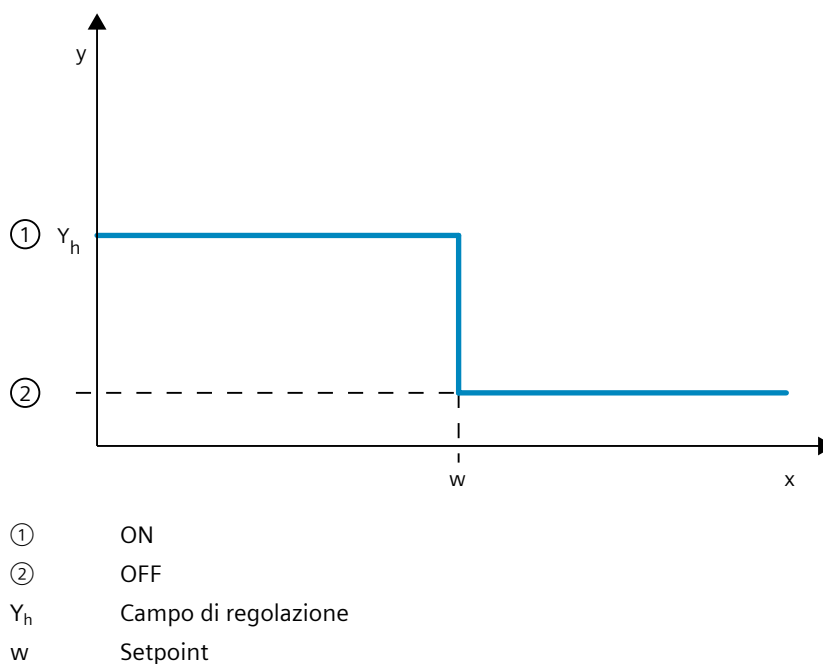
3.4 Regolatore a impulsi

Regolatore a due punti senza retroazione

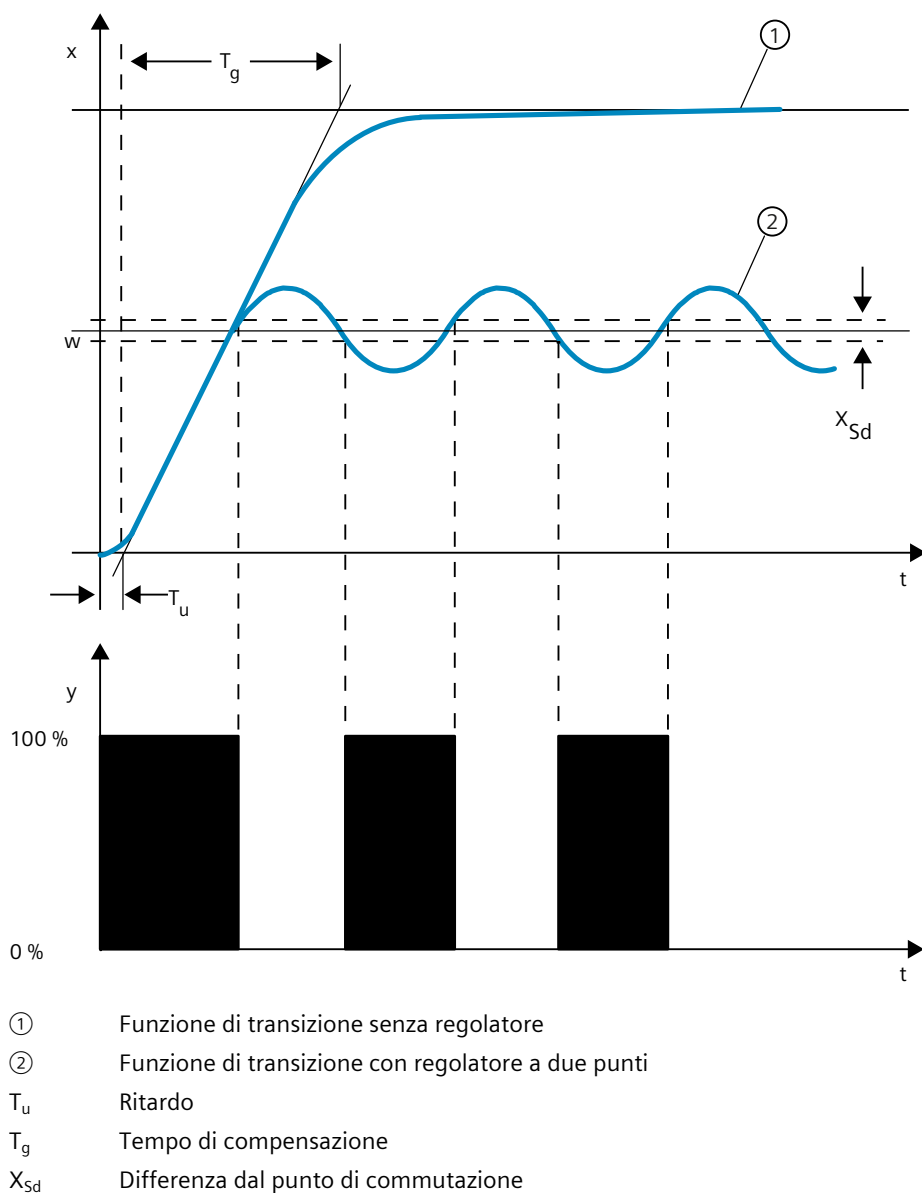
I regolatori a due punti hanno come funzione di commutazione gli stati "ON" e "OFF" che corrispondono a una potenza del 100 % o dello 0 %. Questo comportamento determina un'oscillazione continua del valore istantaneo x intorno al setpoint w .

L'ampiezza e la durata dell'oscillazione aumentano in funzione del rapporto tra il tempo di ritardo T_u e quello di compensazione T_g del sistema regolato. Questi regolatori vengono impiegati prevalentemente per semplici regolazioni della temperatura (ad es. per i forni elettrici a riscaldamento diretto) oppure come segnalatori di valore limite.

Il seguente grafico mostra la curva caratteristica di un regolatore a due punti



Il seguente grafico mostra la funzione di regolazione di un regolatore a due punti



Regolatore a due punti con retroazione

Il comportamento dei regolatori a due punti sui sistemi regolati con elevati tempi di ritardo, ad es. nei forni in cui l'area di utilizzo è separata dal riscaldamento, può essere migliorato tramite retroazioni elettroniche.

La retroazione consente di incrementare la frequenza di commutazione del regolatore e quindi di ridurre l'ampiezza del valore istantaneo. Essa consente inoltre di migliorare sensibilmente i risultati della regolazione nel funzionamento dinamico. Il limite della frequenza di commutazione viene determinato dal gradino di uscita. Nel caso di attuatori meccanici quali relè e contattori si raccomanda di non superare da 1 a 5 commutazioni al minuto. Per le uscite binarie di tensione e di corrente a monte di commutatori tipo tiristore o

triac si possono scegliere frequenze di commutazione elevate di gran lunga superiori alla frequenza limite del sistema regolato.

Poiché gli impulsi di commutazione nell'uscita del sistema regolato non possono più essere determinati, i risultati ottenuti sono paragonabili a quelli dei regolatori continui.

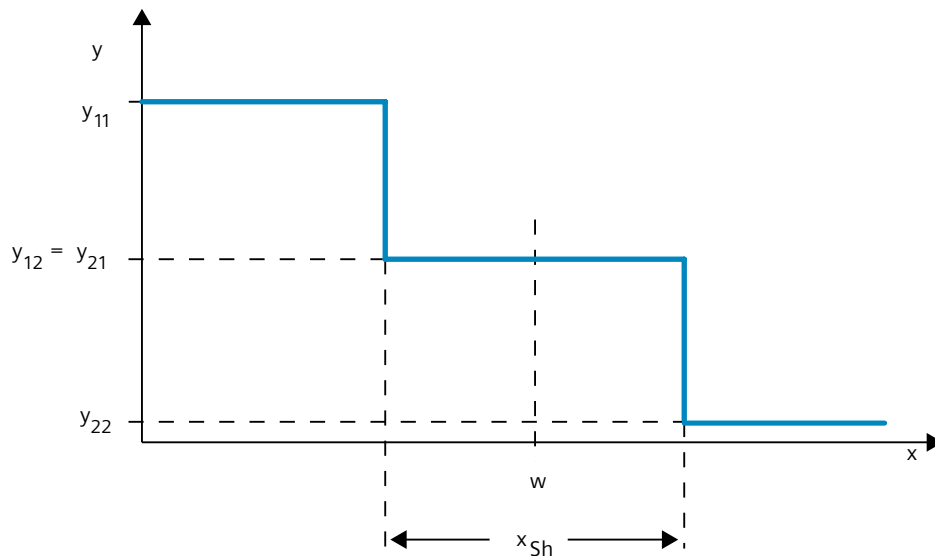
Il valore di uscita viene generato dalla modulazione di larghezza di impulso del valore di uscita di un regolatore continuo.

I regolatori a due punti con retroazione vengono impiegati per la regolazione della temperatura dei forni, nelle macchine utensili utilizzate nell'industria tessile, alimentare, della carta, della gomma e delle materie plastiche e nelle apparecchiature di riscaldamento e raffreddamento.

Regolatore a tre punti

I regolatori a tre punti vengono impiegati per il riscaldamento/raffreddamento. Questi regolatori hanno come uscita due punti di commutazione. I risultati della regolazione vengono ottimizzati mediante di strutture di retroazione elettroniche. Questi regolatori trovano impiego in camere climatiche, frigorifere e termiche nonché nel riscaldamento degli utensili delle macchine per la lavorazione delle materie plastiche.

Il seguente grafico mostra la curva caratteristica di un regolatore a tre punti

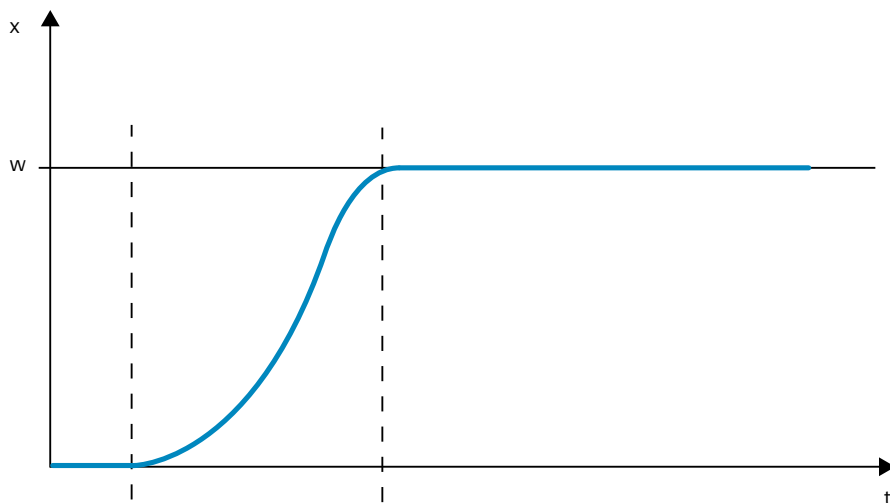


y	Valore di uscita, ad es. $y_{11} = 100\%$ riscaldamento $y_{12} = 0\%$ riscaldamento $y_{21} = 0\%$ raffreddamento $y_{22} = 100\%$ raffreddamento
x	Grandezza fisica del valore istantaneo, ad es. temperatura in $^{\circ}\text{C}$
w	Setpoint
x_{Sh}	Distanza tra i punti di commutazione 1 e 2

3.5 Comportamento pilota e di disturbo

Comportamento pilota

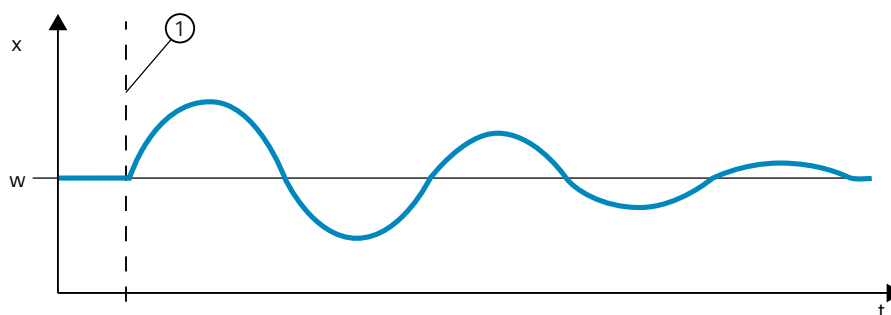
Il valore istantaneo deve seguire il più rapidamente possibile una variazione del setpoint. Il comportamento pilota produce risultati migliori quanto più breve è il tempo impiegato per il raggiungimento del nuovo setpoint e minore l'oscillazione del valore istantaneo.



x Valore istantaneo
w Setpoint

Comportamento di disturbo

Il setpoint viene influenzato da grandezze di disturbo: Il regolatore deve eliminare il più rapidamente possibile la differenza di regolazione risultante. Il comportamento di disturbo produce risultati migliori quanto più breve è il tempo impiegato per il raggiungimento del setpoint e minore l'oscillazione del valore istantaneo.



x Valore istantaneo
w Setpoint
① Effetto di una grandezza di disturbo

Una grandezza di disturbo viene compensata da regolatori con componente I. La qualità della regolazione non viene compromessa da una grandezza di disturbo ad azione costante perché la differenza di regolazione è relativamente costante. Una grandezza di disturbo dinamica

invece influenza maggiormente la qualità della regolazione perché la differenza di regolazione oscilla. La differenza di regolazione viene ridotta nuovamente solo mediante la lenta attivazione del componente I.

Una grandezza di disturbo misurabile può essere inserita nel circuito di regolazione accelerando notevolmente la reazione del regolatore.

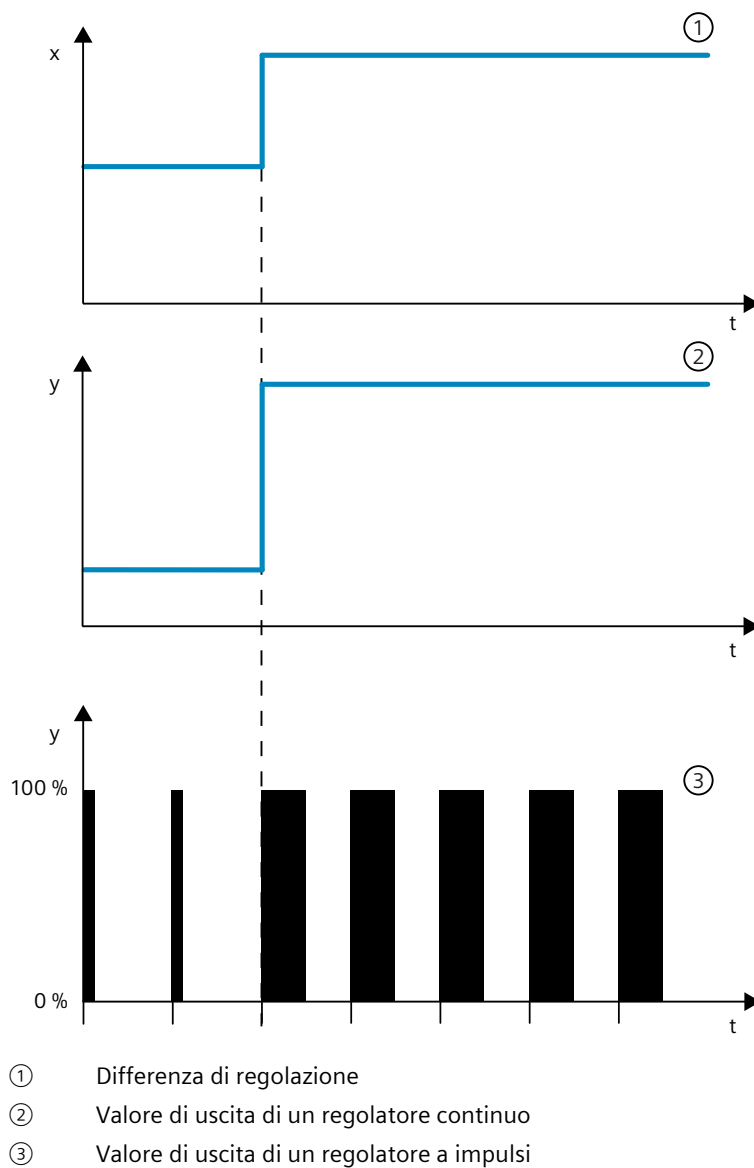
3.6 Comportamento della regolazione in presenza di strutture di retroazione diverse

Comportamenti del regolatore

Maggiore è la precisione di adattamento del regolatore al comportamento temporale del sistema regolato, maggiore sarà la precisione con cui esso imposta il setpoint e reagisce in modo ottimale alle grandezze di disturbo.

L'attivazione della retroazione può avere un'azione proporzionale (P), proporzionale-derivativa (PD), proporzionale-integrale (PI) o proporzionale-integrale-derivativa (PID).

Se alla differenza di regolazione viene indicata una funzione a gradino, risultano diverse risposte a gradino a seconda del regolatore.

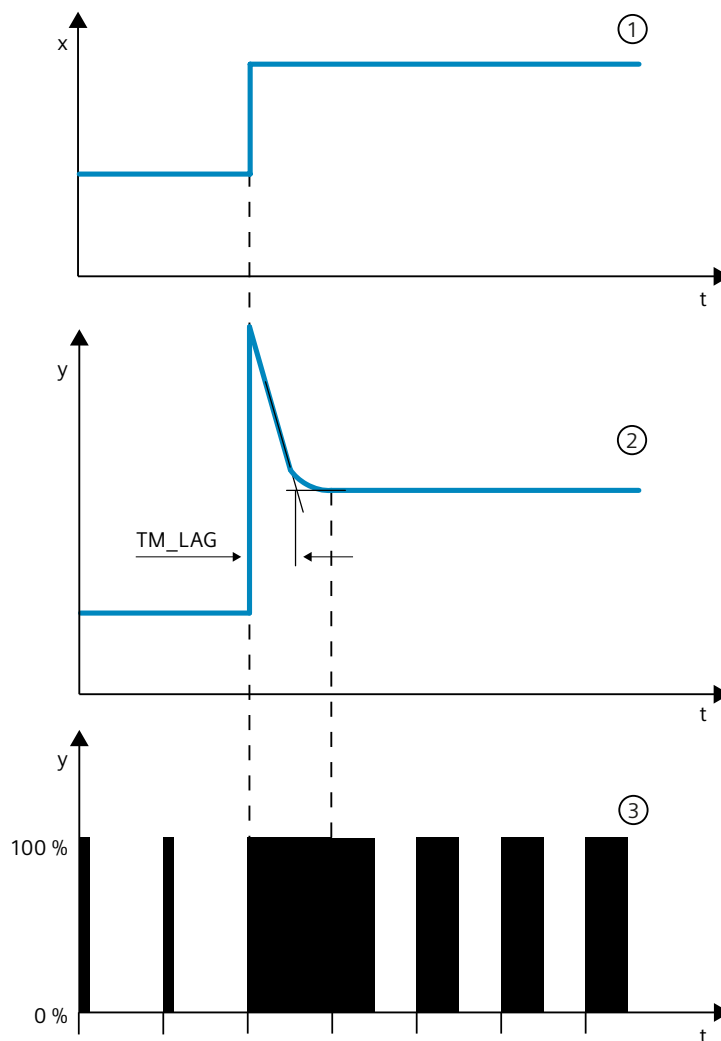
Risposta a gradino di un regolatore P**Formula del regolatore P**

Il valore di uscita e la differenza di regolazione sono direttamente proporzionali, ovvero:

valore di uscita = coefficiente proporzionale $\times$ differenza di regolazione

$$y = \text{GAIN} \times x$$

Risposta a gradino di un regolatore PD



- ① Differenza di regolazione
- ② Valore di uscita di un regolatore continuo
- ③ Valore di uscita di un regolatore a impulsi
- TM_LAG Ritardo della componente D

Formula del regolatore PD

Per la risposta a gradino del regolatore PD nell'intervallo di tempo vale quanto segue:

$$y = \text{GAIN} \cdot X_W \cdot \left(1 + \frac{TD}{TM\_LAG} \cdot e^{\frac{-t}{TM\_LAG}} \right)$$

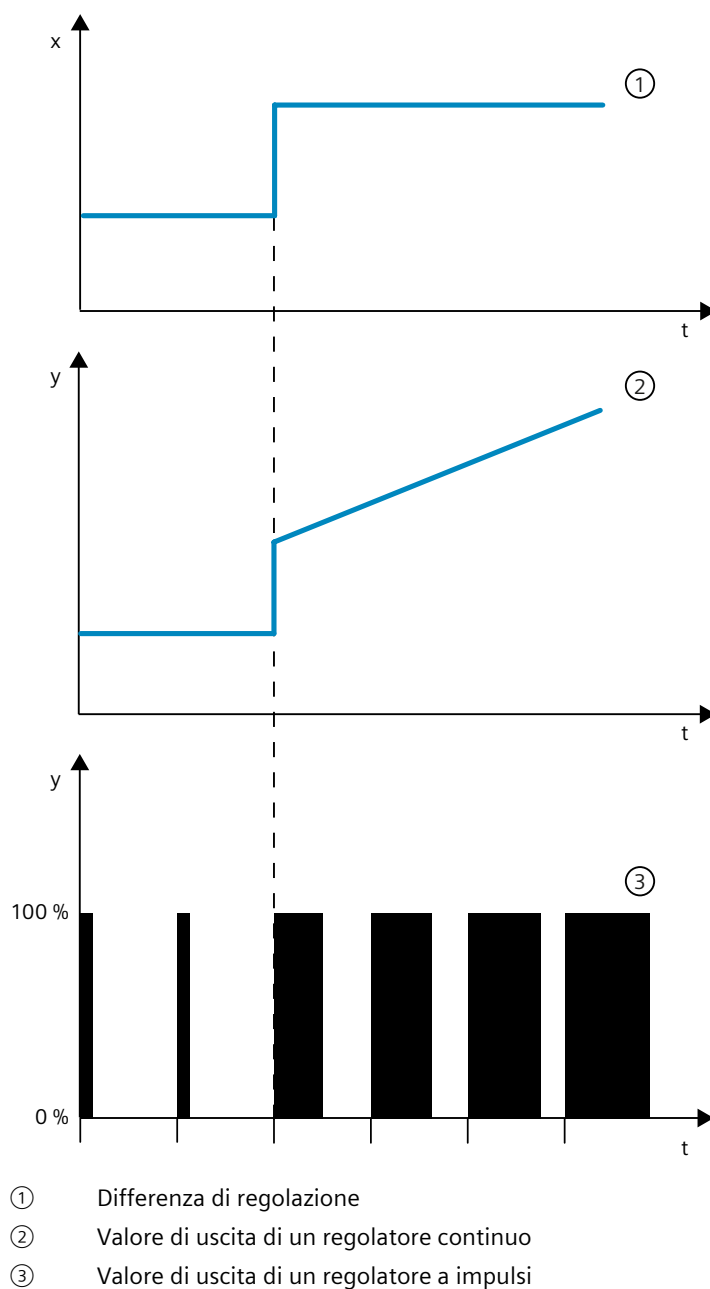
t = intervallo di tempo dal gradino della differenza di regolazione

La componente D genera un valore di uscita in funzione della velocità di variazione del valore istantaneo. La componente D non è adatta per la regolazione perché solo una variazione del valore istantaneo determina una variazione del valore di uscita. Se il valore istantaneo rimane costante, anche il valore di uscita non subisce più variazioni.

In combinazione con una componente P la componente D migliora il comportamento di disturbo. Gli eventuali disturbi non vengono completamente compensati. Il vantaggio consiste nel buon comportamento dinamico. Essa consente di realizzare, durante l'approssimazione e la variazione dell'asetpoint, una transizione smorzata e priva di oscillazioni.

I regolatori con componente D non sono tuttavia adatti se il sistema regolato ha grandezze di misura pulsanti, ad esempio in caso di regolazione della pressione o della portata.

Risposta a gradino di un regolatore PI



La componente I nel regolatore somma la differenza di regolazione nel tempo ed effettua l'integrazione fino ad eliminare la differenza di regolazione. Nei regolatori P la differenza di regolazione è costante. E può essere in genere eliminata da una componente I nel regolatore.

In pratica l'ideale è una combinazione delle diverse componenti P, I e D in funzione del tipo di regolazione richiesto. Il comportamento temporale delle singole componenti può essere descritto mediante i parametri di regolazione coefficiente proporzionale GAIN, tempo dell'azione integrale TI (componente I) e tempo dell'azione derivata TD (componente D).

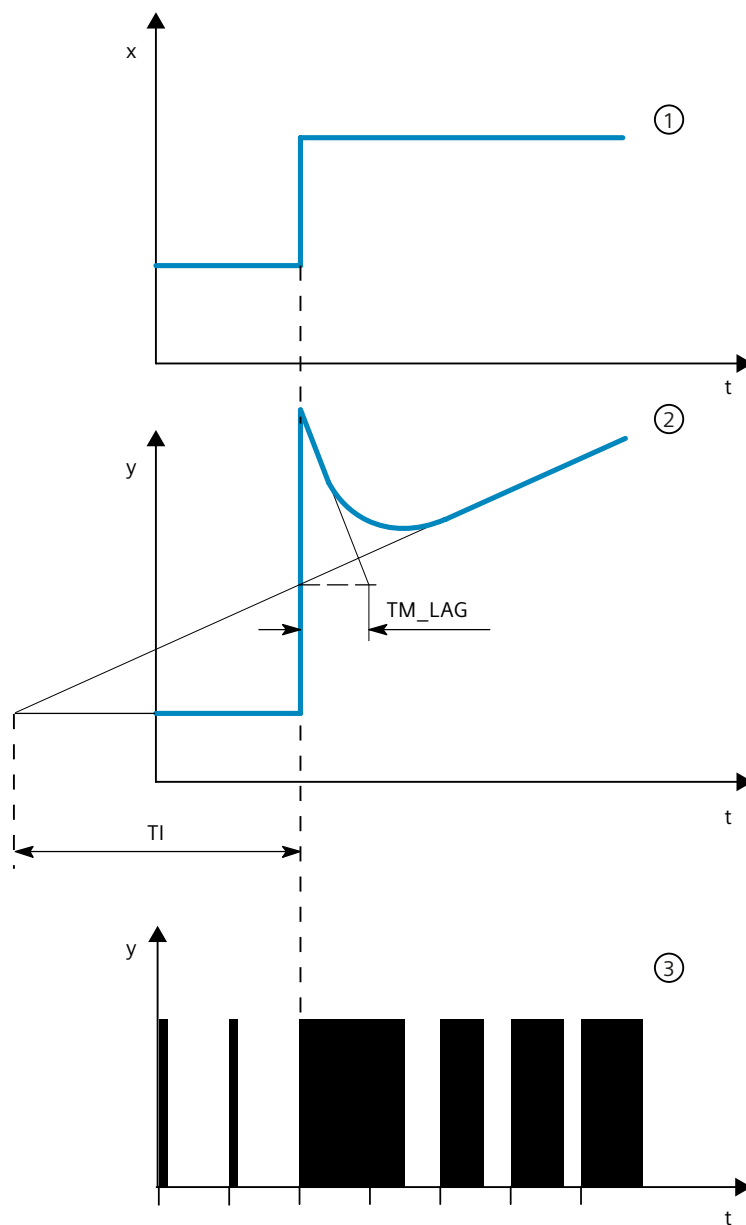
Formula del regolatore PI

Per la risposta a gradino del regolatore PI nel campo dei tempi vale quanto segue:

$$y = \text{GAIN} \cdot X_W \cdot \left(1 + \frac{1}{\text{TI} \cdot t} \right)$$

t = intervallo di tempo dal gradino della differenza di regolazione

Risposta a gradino di un regolatore PID



- ① Differenza di regolazione
- ② Valore di uscita di un regolatore continuo
- ③ Valore di uscita di un regolatore a impulsi
- TM_LAG Ritardo della componente D
- T_i Tempo di integrazione

Formula del regolatore PID

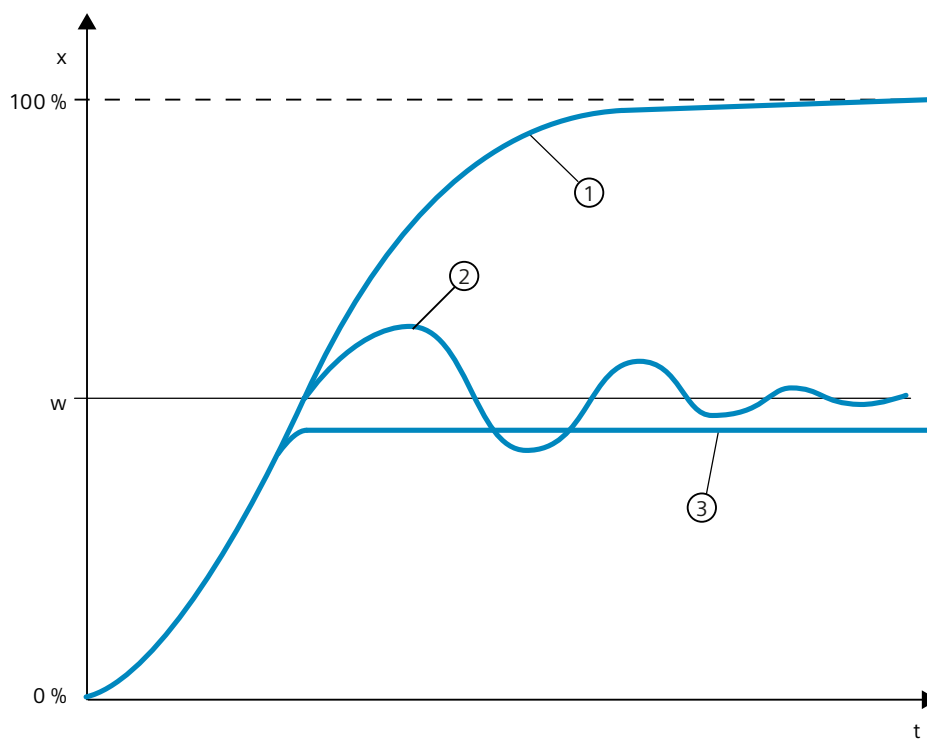
Per la risposta a gradino del regolatore PID nell'intervallo di tempo vale quanto segue:

$$y = \text{GAIN} \cdot X_w \cdot \left(1 + \frac{1}{\text{TI} \cdot t} + \frac{\text{TD}}{\text{TM\_LAG}} \cdot e^{-\frac{t}{\text{TM\_LAG}}} \right)$$

t = intervallo di tempo dal gradino della differenza di regolazione

Comportamento di un sistema regolato in diverse strutture di regolazione

La maggior parte delle regolazioni nell'ambito della tecnica di processo possono essere effettuate tramite un regolatore PI. Nel caso di sistemi regolati lenti con tempo di ritardo elevato, ad esempio nella regolazione della temperatura, il risultato della regolazione può essere migliorato tramite un regolatore PID.



- ① Nessun regolatore
- ② Regolatore PID
- ③ Regolatore PD
- w Setpoint
- x Valore istantaneo

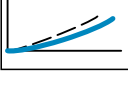
I regolatori PI e PID hanno il vantaggio che, dopo l'assestamento, il valore istantaneo non presenta più deviazioni rispetto al setpoint. Durante l'approssimazione il valore istantaneo oscilla intorno al setpoint.

3.7 Scelta della struttura di regolazione nel sistema regolato dato

Scelta delle strutture di regolazione adeguate

Per raggiungere una regolazione ottimale occorre selezionare una struttura di regolazione adatta al sistema regolato e che possa essere adeguata ad esso entro determinati limiti.

La seguente tabella fornisce una panoramica della struttura di regolazione adatta per un determinato tipo di sistema regolato.

Sistema regolato		Struttura di regolazione			
		P	PD	PI	PID
	solo con tempo morto	Non idoneo	Non idoneo	Idoneo	Non idoneo
	PT1 con tempo morto	Non idoneo	Non idoneo	Idoneo	Idoneo
	PT2 con tempo morto	Non idoneo	Idoneo limitatam.	Idoneo	Idoneo
	Ordine superiore	Non idoneo	Non idoneo	Idoneo limitatam.	Idoneo
	Senza compensazione	Idoneo	Idoneo	Idoneo	Idoneo

La seguente tabella fornisce una panoramica sulla struttura di regolazione adatta per una determinata grandezza fisica.

Grandezza fisica	Struttura di regolazione			
	P	PD	PI	PID
	Differenza di regolazione residua		Nessuna differenza di regolazione residua	
Temperatura	Per esigenze moderate e circuiti P con $T_u/T_g < 0,1$	Idoneo	Per strutture di regolazione adatte a esigenze superiori (esclusi i regolatori speciali adattati in modo specifico)	
Pressione	Idoneo in presenza di tempi di ritardo trascurabili	Non idoneo	Per strutture di regolazione adatte a esigenze superiori (esclusi i regolatori speciali adattati in modo specifico)	
Portata	Non idoneo, perché generalmente il campo GAIN necessario è troppo grande	Non idoneo	Utilizzabile ma spesso è meglio solo il regolatore I	Praticamente non necessario

3.8 Impostazione dei parametri PID

Regola generale per l'impostazione dei parametri

Struttura di regolazione	Impostazione
P	$GAIN \approx v_{max} \times T_u [^{\circ}C]$
PI	$GAIN \approx 1,2 \times v_{max} \times T_u [^{\circ}C]$ $TI \approx 4 \times T_u [min]$
PD	$GAIN \approx 0,83 \times v_{max} \times T_u [^{\circ}C]$ $TD \approx 0,25 \times v_{max} \times T_u [min]$ $TM\_LAG \approx 0,5 \times TD [min]$
PID	$GAIN \approx 0,83 \times v_{max} \times T_u [^{\circ}C]$ $TI \approx 2 \times T_u [min]$ $TD \approx 0,4 \times T_u [min]$ $TM\_LAG \approx 0,5 \times TD [min]$
PD/PID	$GAIN \approx 0,4 \times v_{max} \times T_u [^{\circ}C]$ $TI \approx 2 \times T_u [min]$ $TD \approx 0,4 \times T_u [min]$ $TM\_LAG \approx 0,5 \times TD [min]$

Invece di $v_{max} = \Delta_x / \Delta_t$ è possibile impiegare X_{max} / T_g .

Nei regolatori PID le impostazioni del tempo dell'azione integrale e di quello dell'azione derivata sono generalmente collegate.

Il rapporto TI / TD è compreso tra 4 e 5 ed è ideale per la maggior parte dei sistemi regolati.

Nei regolatori PD la mancata osservanza del tempo dell'azione derivativa TD non è critica.

Nei regolatori PI o PID si verificano oscillazioni se il tempo dell'azione integrale TI selezionato è meno della metà del valore previsto.

Un tempo dell'azione integrale troppo elevato rallenta l'eliminazione dei disturbi. Tenere quindi presente che dopo la prima parametrizzazione i circuiti di regolazione non possono operare in modo "ottimale". Per esperienza la regolazione a posteriori è sempre necessaria in presenza di un circuito "difficile da regolare" con $T_u / T_g > 0,3$.

Progettazione di un regolatore software

4.1 Panoramica dei regolatori software

Per la progettazione di un regolatore software è necessaria un'istruzione con l'algoritmo di regolazione e un oggetto tecnologico. L'oggetto tecnologico di un regolatore software corrisponde al DB di istanza dell'istruzione. Nell'oggetto tecnologico viene salvata la configurazione del regolatore. A differenza dei DB di istanza di altre istruzioni, gli oggetti tecnologici non vengono salvati nelle risorse dei programmi, ma in CPU > Oggetti tecnologici.

Oggetti tecnologici e istruzioni

CPU	Biblioteca	Istruzione	Oggetto tecnologico	Descrizione
S7-1200	Compact PID	PID_Compact V1.x	PID_Compact V1.x	Regolatore PID universale con ottimizzazione integrata
S7-1200		PID_3Step V1.x	PID_3Step V1.x	Regolatore PID con ottimizzazione integrata per valvole
S7-1500 S7-1200 V4.x		PID_Compact V2.x	PID_Compact V2.x	Regolatore PID universale con ottimizzazione integrata
S7-1500 S7-1200 V4.x S7-1200 G2		PID_3Step V2.x	PID_3Step V2.x	Regolatore PID con ottimizzazione integrata per valvole
S7-1500 ≥ V1.7 S7-1200 ≥ V4.1 S7-1200 G2		PID_Temp V1.x	PID_Temp V1.x	Regolatore di temperatura PID universale con ottimizzazione integrata
S7-1500 ≥ V3.1 S7-1200 G2		PID_Compact V3.x	PID_Compact V3.x	Regolatore PID universale con ottimizzazione integrata
S7-1500/300/40-0	Funzioni di base PID	CONT_C	CONT_C	Regolatore continuo
S7-1500/300/40-0		CONT_S	CONT_S	Regolatore a passi per organi attuatori integranti
S7-1500/300/40-0		PULSEGEN	-	Generatore di impulsi per organi attuatori ad azione proporzionale
S7-1500/300/40-0		TCONT_CP	TCONT_CP	Regolatore di temperatura continuo con generatore di impulsi
S7-1500/300/40-0		TCONT_S	TCONT_S	Regolatore di temperatura per organi attuatori integrati
S7-300/400	PID Self Tuner	TUN_EC	TUN_EC	Ottimizzazione di un regolatore continuo
S7-300/400		TUN_ES	TUN_ES	Ottimizzazione di un regolatore passo a passo
S7-300/400	Standard PID Control (pacchetto opzionale PID Professional)	PID_CP	PID_CP	Regolatore continuo con generatore di impulsi
S7-300/400		PID_ES	PID_ES	Regolatore a passi per organi attuatori integranti
S7-300/400		LP_SCHED	-	Distribuisce richiami regolatore

CPU	Biblioteca	Istruzione	Oggetto tecnologico	Descrizione
S7-300/400	Modular PID Control (pacchetto opzionale PID Professional)	A_DEAD_B	-	Filtra segnali di disturbo della differenza di regolazione
S7-300/400		CRP_IN	-	Riporta in scala segnale di ingresso analogico
S7-300/400		CRP_OUT	-	Riporta in scala segnale di uscita analogico
S7-300/400		DEAD_T	-	Emissione ritardata del segnale di ingresso
S7-300/400		DEADBAND	-	Soppressione di lievi oscillazioni del valore istantaneo
S7-300/400		DIF	-	Differenziazione ritardata del segnale di ingresso
S7-300/400		ERR_MON	-	Sorveglianza differenza di regolazione
S7-300/400		INTEG	-	Integrazione ritardata del segnale di ingresso
S7-300/400		LAG1ST	-	Stadio di ritardo di 1° ordine
S7-300/400		LAG2ND	-	Stadio di ritardo di 2° ordine
S7-300/400		LIMALARM	-	Segnalazione valori limite
S7-300/400		LIMITER	-	Limitazione del valore regolante
S7-300/400		LMNGEN_C	-	Rileva valore regolante per regolatore continuo
S7-300/400		LMNGEN_S	-	Rilevazione valore regolante del regolatore a passi
S7-300/400		NONLIN	-	Linearizzazione del segnale encoder
S7-300/400		NORM	-	Normazione fisica del valore istantaneo
S7-300/400		OVERRIDE	-	Attiva valori regolanti di 2 regolatori PID su 1 attuatore
S7-300/400		PARA_CTL	-	Commutazione set di parametri
S7-300/400		PID	-	Algoritmo PID
S7-300/400		PUSLEGEN_M	-	Genera impulsi per attuatori ad azione proporzionale
S7-300/400		RMP_SOAK	-	Predefinizione dei setpoint secondo la schedulazione
S7-300/400		ROC_LIM	-	Limita velocità di variazione
S7-300/400		SCALE_M	-	Riporta in scala valore istantaneo
S7-300/400		SP_GEN	-	Inserimento manuale del setpoint
S7-300/400		SPLT_RAN	-	Distribuzione campi del valore regolante
S7-300/400		SWITCH	-	Attiva valori analogici
S7-300/400		LP_SCHED_M	-	Distribuisce richiami regolatore

4.2 Fasi di progettazione di un regolatore software

Progettare tutti i regolatori SW secondo il medesimo schema:

Fase	Descrizione
1	Aggiungere un oggetto tecnologico (Pagina 37)
2	Configurare l'oggetto tecnologico (Pagina 38)
3	Richiamare l'istruzione nel programma utente (Pagina 39)
4	Caricare un oggetto tecnologico nel dispositivo (Pagina 40)
5	Mettere in funzione il regolatore software (Pagina 41)
6	Salvare nel progetto i parametri PID ottimizzati (Pagina 41)
7	Confrontare i valori (Pagina 46)
8	Visualizzare le istanze di un oggetto tecnologico (Pagina 69)

4.3 Inserire oggetti tecnologici

Inserire l'oggetto tecnologico nel navigatore del progetto

Quando si inserisce un oggetto tecnologico viene creato un DB di istanza dell'istruzione relativa all'oggetto, che contiene la configurazione dell'oggetto.

Premessa

È stato creato un progetto con una CPU.

Procedimento

Per inserire un oggetto tecnologico, procedere nel modo seguente:

1. Aprire la cartella della CPU nella navigazione del progetto.
2. Aprire la cartella "Oggetti tecnologici".
3. Fare doppio clic su "Inserisci nuovo oggetto".
Si apre la finestra di dialogo "Inserisci nuovo oggetto".
4. Fare clic sul pulsante "Regolatore PID".
Vengono visualizzati tutti i regolatori PID disponibili per questa CPU.
5. Scegliere l'istruzione per l'oggetto tecnologico, per esempio PID_Compact.
6. Specificare un nome univoco per l'oggetto tecnologico nel campo di immissione "Nome".
7. Se si vuole modificare il numero proposto per il DB di istanza selezionare l'opzione "Manuale".
8. Fare clic su "Ulteriori informazioni" per specificare altre informazioni sull'oggetto tecnologico.
9. Confermare con "OK".

Risultato

Il nuovo oggetto tecnologico viene creato e inserito nella navigazione del progetto all'interno della cartella "Oggetti tecnologici". L'oggetto tecnologico viene utilizzato quando l'istruzione impartita a questo oggetto tecnologico viene richiamata in un OB di schedulazione orologio.

NOTA

Se si attiva la casella di opzione "Aggiungi nuovo e apri" nell'area inferiore della finestra di dialogo una volta inserito l'oggetto tecnologico, si apre la relativa configurazione.

4.4 Configurare oggetti tecnologici

Le proprietà degli oggetti tecnologici PID_Compact, PID_3Step e PID_Temp possono essere configurate in due modi.

- Nella finestra di ispezione dell'editor di programma
- Nella finestra di configurazione

Le proprietà degli altri oggetti tecnologici per la regolazione PID possono essere configurate solo nella finestra di configurazione.

Nella finestra di ispezione dell'editor di programma

Nella finestra di ispezione dell'editor di programma si possono configurare solo i parametri necessari per il funzionamento.

I valori offline dei parametri vengono visualizzati anche nella modalità online. I valori online si possono modificare solo nella finestra di messa in funzione.

Per aprire la finestra di ispezione di un oggetto tecnologico, procedere nel modo seguente:

1. Aprire la cartella "Blocchi di programma" nella navigazione del progetto.
2. Fare doppio clic sul blocco (OB di schedulazione orologio) in cui viene richiamata l'istruzione del regolatore SW.
Il blocco si apre nell'area di lavoro.
3. Fare clic sull'istruzione del regolatore SW.
4. Selezionare le schede "Proprietà" e "Configurazione" nella finestra di ispezione.

Finestra di configurazione

Per ciascun oggetto tecnologico vi è una finestra di configurazione specifica in cui poter configurare tutte le proprietà.

Per aprire la finestra di configurazione di un oggetto tecnologico, procedere nel modo seguente:

1. Aprire la cartella "Oggetti tecnologici" nella navigazione del progetto.
2. Aprire l'oggetto tecnologico nella navigazione del progetto.
3. Fare doppio clic sull'oggetto "Configurazione".

Icone

Le icone che compaiono nella navigazione di settore delle finestre di configurazione e di ispezione forniscono ulteriori informazioni sulla completezza della configurazione:

✓	La configurazione contiene valori preimpostati ed è completa. La configurazione contiene solo valori preimpostati che consentono di utilizzare l'oggetto tecnologico senza ulteriori modifiche.
✓	La configurazione contiene valori definiti dall'utente o adeguati automaticamente ed è completa. Tutti i campi di immissione della configurazione contengono valori validi e almeno un valore preimpostato è stato modificato.
✗	La configurazione è incompleta o errata. Almeno un campo di immissione o un elenco a cascata è vuoto o contiene un valore non valido. Il campo o l'elenco sono evidenziati in rosso. Facendo clic con il mouse compare un messaggio di errore a tendina che spiega la causa.

Le proprietà di un oggetto tecnologico sono descritte in modo dettagliato nel capitolo dell'oggetto tecnologico.

4.5 Richiamare l'istruzione nel programma utente

L'istruzione del regolatore software deve essere richiamata in un OB di schedulazione orologio. Il tempo di campionamento del regolatore software viene definito dall'intervallo temporale tra i richiami nell'OB di schedulazione orologio.

Requisiti

L'OB di schedulazione orologio è creato e la durata del ciclo dell'OB è configurata correttamente.

Procedura

Per richiamare l'istruzione nel programma utente, procedere nel modo seguente:

1. Aprire la cartella della CPU nella navigazione del progetto.
2. Aprire la cartella "Blocchi di programma".
3. Fare doppio clic sull'OB di schedulazione orologio.
Il blocco si apre nell'area di lavoro.
4. Nella finestra "Istruzioni" aprire il gruppo "Tecnologia" e la cartella "PID Control".
La cartella contiene tutte le istruzioni del regolatore software che possono essere progettate sulla CPU.
5. Selezionare un'istruzione e trascinarla mediante drag & drop nell'OB di schedulazione orologio.
Si apre la finestra di dialogo "Opzioni di richiamo".
6. Selezionare nell'elenco "Nome" un oggetto tecnologico o immettere il nome per un nuovo oggetto tecnologico.

Risultato

Se l'oggetto tecnologico non esiste ancora, viene creato. L'istruzione viene inserita nell'OB di schedulazione orologio. L'oggetto tecnologico è assegnato a questo richiamo dell'istruzione.

4.6 Caricamento degli oggetti tecnologici del dispositivo

Per il funzionamento online è necessario caricare una configurazione dell'oggetto tecnologico nella CPU. Nel caricamento dei dati a ritenzione valgono le seguenti particolarità:

- **Software (solo modifiche)**
 - S7-1200, S7-1500:
I dati a ritenzione vengono mantenuti.
 - S7-300/400:
I dati a ritenzione vengono subito aggiornati. La CPU non va in Stop.
- **Carica e resetta programma del PLC nel dispositivo**
 - S7-1200, S7-1500:
I dati a ritenzione vengono aggiornati alla successiva commutazione da Stop a RUN. Il programma del PLC può essere caricato esclusivamente per intero.
 - S7-300/400:
I dati a ritenzione vengono aggiornati alla successiva commutazione da Stop a RUN.

Caricamento dei dati a ritenzione in una CPU S7-1200 o S7-1500

NOTA

Il caricamento ed il resettaggio del programma del PLC durante il funzionamento dell'impianto può causare, in caso di guasti di funzionamento o di errori di programmazione, gravi lesioni personali e danni materiali!

Escludere il verificarsi di situazioni di pericolo prima di eseguire il caricamento e resettaggio del programma del PLC.

Per caricare i dati a ritenzione, procedere nel modo seguente:

1. Selezionare la CPU desiderata nella navigazione del progetto.
2. Nel menu "Online" selezionare il comando "Carica e resetta programma del PLC nel dispositivo".
 - Se non è stato ancora creato il collegamento online, si apre la finestra di dialogo "Caricamento avanzato". Impostare tutti i parametri necessari al collegamento, quindi fare clic su "Carica".
 - Una volta creato il collegamento online, i dati di progetto vengono, se necessario, compilati e viene aperta la finestra di dialogo "Anteprima di carica". Questa finestra di dialogo visualizza i messaggi e le operazioni necessarie al caricamento.
3. Controllare i messaggi.
Non appena il caricamento è possibile, viene attivato il pulsante "Carica".

4. Fare clic su "Carica".

Viene caricato il programma del PLC completo e aperta la finestra di dialogo "Carica risultati". Questa finestra di dialogo visualizza lo stato e le operazioni successive al caricamento.

5. Se le unità devono essere direttamente riavviate dopo il caricamento, attivare la casella di scelta "Avvia tutto".
6. Chiudere la finestra di dialogo "Carica risultati" premendo il pulsante "Fine".

Risultato

Il programma del PLC completo viene caricato nel dispositivo. I blocchi presenti soltanto online nel dispositivo, vengono cancellati. Con il caricamento nel dispositivo di tutti i blocchi interessati e la cancellazione dei blocchi non necessari, vengono evitate eventuali incoerenze tra i blocchi nel programma utente.

I messaggi alla voce "Informazioni su ... > Generale" forniscono informazioni circa la riuscita del caricamento.

4.7 Mettere in funzione il regolatore software

Procedimento

Per aprire l'area di lavoro "Messa in servizio" di un oggetto tecnologico procedere nel modo seguente:

1. Aprire la cartella "Oggetti tecnologici" nella navigazione del progetto.
2. Aprire l'oggetto tecnologico nella navigazione del progetto.
3. Fare doppio clic sull'oggetto "Messa in servizio".

Qui sono descritte le funzioni per la messa in servizio, che sono specifiche per ogni regolatore.

4.8 Salvare nel progetto i parametri PID ottimizzati

Il regolatore software viene ottimizzato nella CPU. I valori nel DB di istanza sulla CPU non coincidono più con quelli del progetto.

Per aggiornare i parametri PID nel progetto con quelli ottimizzati nella CPU, procedere nel modo seguente:

Presupposti

- È stato stabilito un collegamento online con la CPU, la quale si trova nello stato di funzionamento "RUN".
- Le funzioni della finestra di messa in servizio sono state abilitate con il pulsante "Avvio".

Procedimento

1. Aprire la cartella della CPU nella navigazione del progetto.
2. Aprire la cartella "Oggetti tecnologici".
3. Aprire l'oggetto tecnologico.
4. Fare doppio clic su "Messa in funzione".
5. Fare clic sul simbolo  "Carica il parametro PID".
6. Salvare il progetto.

Risultato

I parametri PID attivi vengono salvati nei dati del progetto. In caso di nuovo caricamento dei dati del progetto nella CPU verranno utilizzati i parametri ottimizzati.

4.9 Uso degli oggetti delle multiistanze

Se un blocco funzionale (FB) richiama un altro blocco funzionale, i relativi dati di istanza possono essere salvati anche nel DB di istanza dell'FB richiamante. Questo tipo di richiamo del blocco viene definito multiistanza. I regolatori software PID supportano questo tipo di richiamo e possono essere impiegati come multiistanza.

Vantaggi

L'impiego di multiistanze offre i seguenti vantaggi:

- Buone possibilità di strutturazione
- Numero ridotto di DB di istanza
- FB con configurazione personalizzata, instanziabili con la frequenza desiderata, come modelli per un regolatore software

Limitazioni

Rispetto alle istanze singole, l'impiego di multiistanze per regolatori software PID è soggetto alle seguenti limitazioni:

- Nessuno supporto di Openness per gli oggetti delle multiistanze PID
- Nessuno confronto di oggetti delle multiistanze PID in un editor di confronto. Il confronto è possibile soltanto tramite il blocco che contiene gli oggetti della multiistanza
- Nessuna finestra di ispezione dell'editor di programma, specifica dell'oggetto tecnologico per il richiamo delle istruzioni PID_Compact, PID_3Step e PID_Temp

Configurazione di oggetti delle multiistanze

La configurazione e la messa in servizio di oggetti di multiistanze PID non viene aperta dalla voce "Oggetti tecnologici" nella navigazione di progetto come invece avviene per gli oggetti di istanze singole. La configurazione e la messa in servizio degli oggetti di multiistanze sono disponibili nella scheda "Oggetti tecnologici" della vista dettagli.

Per aprire la configurazione degli oggetti di multiistanze, procedere nel seguente modo:

1. Selezionare l'FB o il DB di istanza con l'oggetto multiistanza nella navigazione del progetto.
2. Nella vista dettagli fare clic sulla scheda "Oggetti tecnologici".
3. Navigare fino all'oggetto della multiistanza desiderato.
4. Aprire la configurazione dell'oggetto della multiistanza.

NOTA

L'Editor di configurazione non offre funzionalità online per gli oggetti di multiistanze in un FB. Nei DB di istanza invece, questa limitazione non esiste.

Messa in servizio degli oggetti delle multiistanze

Per aprire la messa in servizio degli oggetti di multiistanze, procedere nel seguente modo:

1. Selezionare il DB di istanza con l'oggetto multiistanza nella navigazione del progetto.
2. Nella vista dettagli fare clic sulla scheda "Oggetti tecnologici".
3. Navigare fino all'oggetto della multiistanza desiderato.
4. Aprire la Messa in servizio dell'oggetto della multiistanza.

Questa funzionalità non è disponibile negli FB degli oggetti delle multiistanze PID.

NOTA

Se gli oggetti delle multiistanze PID si trovano in un array, è possibile accedere a tali oggetti nella vista dettagli solo se l'array non contiene più di 100 elementi. In caso di array con più di 100 elementi, i singoli oggetti delle multiistanze PID non vengono visualizzati e gli editor di configurazione e messa in servizio non sono disponibili.

Esempio di utilizzo degli oggetti delle multiistanze PID

Per utilizzare gli oggetti delle multiistanze PID nella propria applicazione è possibile procedere come segue:

1. Inserire un nuovo blocco funzionale nel programma.
2. Richiamare uno o più regolatori PID idonei con l'opzione di richiamo "Multiistanza" all'interno dell'FB.
3. Inserire nello stesso FB una funzionalità specifica per l'applicazione, ad es. Pre-elaborazione del setpoint.
4. Selezionare l'FB nella navigazione di progetto, quindi aprire gli editor di configurazione degli oggetti della multiistanza PID nella scheda "Oggetti tecnologici" della vista dettagli.
5. Effettuare la configurazione negli editor di configurazione. La configurazione deve essere identica per tutte le istanze dell'FB.
6. Chiudere gli editor di configurazione.
7. Istanziare l'FB secondo necessità nel programma utente per creare i DB di istanza.
8. Selezionare uno dei DB di istanza nella navigazione di progetto, quindi aprire gli editor di configurazione degli oggetti di multiistanza PID nella scheda "Oggetti tecnologici" della vista dettagli.
9. Effettuare la configurazione del DB di istanza interessato negli editor di configurazione.
10. Chiudere gli editor di configurazione.
11. Ripetere i passi da 8 a 10 anche per gli altri DB di istanza con oggetti di multiistanza PID.
12. Compilare il programma, caricarlo nel dispositivo e stabilire la connessione online.
13. Selezionare un DB di istanza contenente oggetti di multiistanza PID nella navigazione di progetto, quindi aprire gli editor di messa in servizio degli oggetti di multiistanza PID nella scheda "Oggetti tecnologici" della vista dettagli.
14. Eseguire la messa in servizio degli oggetti di multiistanza PID.
15. Chiudere gli editor di messa in servizio.
16. Ripetere i passi da 13 a 15 anche per gli altri DB di istanza con oggetti di multiistanza PID.

4.10 Confronto di valori

4.10.1 Visualizzazione del confronto e condizioni generali

La funzione "Confronta valori" offre le seguenti opzioni:

- Confronto dei valori di avvio del progetto configurati con i valori di avvio nella CPU e i valori attuali
- Elaborazione diretta dei valori attuali e dei valori di avvio del progetto
- Riconoscimento immediato e visualizzazione degli errori di inserimento dati con supporto alla correzione
- Salvataggio dei valori attuali nel progetto
- Trasferimento dei valori attuali del progetto nella CPU come valori attuali

Simboli ed elementi di comando

Sono disponibili i seguenti simboli ed elementi di comando:

Simbolo	Funzione
	Il valore di avvio nella CPU è uguale al valore di avvio nel progetto
	Il valore di avvio nella CPU è diverso dal valore di avvio nel progetto
	Il confronto tra il valore di avvio nella CPU e il valore di avvio nel progetto non può essere eseguito
	Almeno uno dei due valori di confronto è tecnologicamente o sintatticamente falso
	Crea istantanea dei valori di controllo e applica i relativi valori come valori di avvio
	Carica valori iniziali di impostazione come valori attuali (inizializzazione dei valori di impostazione)
	Si apre la finestra di dialogo "Confronta valori"

Condizioni generali

La funzione "Confronta valori" è disponibile per S7-1200 e S7-1500 senza limitazioni.

Per S7-300 e S7-400 vale la seguente limitazione:

In modo di supervisione un S7-300/S7-400 non può trasferire i valori di avvio nella CPU. Questi valori non possono essere visualizzati online con "Confronta valori".

I valori attuali dell'oggetto tecnologico vengono visualizzati e possono essere modificati direttamente.

4.10.2 Confronto di valori

Nel seguito viene descritto il procedimento utilizzando come esempio i "Parametri PID".

Presupposti

- È stato configurato un progetto con un regolatore software.
- Il progetto è caricato nella CPU.
- Nella navigazione del progetto è aperta la finestra di configurazione.

Procedimento

1. Aprire il regolatore software nella navigazione del progetto.
2. Fare doppio clic sull'oggetto "Configurazione".
3. Nella finestra di configurazione, spostarsi nella finestra di dialogo "Parametri PID".
4. Fare clic sul simbolo  per attivare il modo di supervisione.
I simboli e gli elementi di comando (Pagina 53) della funzione "Confronta valori" vengono visualizzati dopo i parametri.
5. Fare clic nel campo di immissione del parametro desiderato e modificarne manualmente i valori inserendoli direttamente.
 - Se lo sfondo del campo di immissione è grigio, i valori possono essere solo letti e non modificati.
 - Per modificare i valori nella finestra di dialogo "Parametri PID", attivare prima l'immissione manuale facendo clic sulla casella di riepilogo "Attiva immissione manuale".
6. Fare clic sul simbolo  per aprire la finestra di dialogo con i valori di avvio.
Questa finestra riporta due valori del parametro:
 - Valore di avvio nella CPU: nella parte superiore viene visualizzato il valore di avvio nella CPU.
 - Valore di avvio nel progetto: nella parte inferiore viene visualizzato il valore di avvio configurato nel progetto.
7. Nel campo di immissione del progetto inserire il valore desiderato.

Riconoscimento degli errori

L'immissione di valori errati viene riconosciuta e viene fornito supporto alla correzione.

Se si inserisce un valore con una sintassi errata, sotto il parametro si apre un rollout contenente il rispettivo messaggio di errore. Il valore errato non viene applicato.

Se viene inserito un valore con tecnologia errata, si apre una finestra di dialogo in cui viene segnalato l'errore e visualizzata un'informazione di correzione:

- Facendo clic su "No" è possibile applicare la correzione e correggere i dati immessi.
- Facendo clic su "OK" si applica il valore errato.

ATTENZIONE

Funzionamento errato del regolatore

Valori con tecnologia errata possono provocare un funzionamento errato del regolatore.

Salvataggio dei valori attuali

Facendo clic sul simbolo  si trasferiscono i valori attuali del regolatore nei valori di avvio del progetto configurato.

Trasferimento dei valori del progetto nella CPU

Facendo clic sul simbolo  si trasferiscono i valori configurati del progetto nella CPU.

CAUTELA

Evitare danni materiali e lesioni personali

Il caricamento ed il resettaggio del programma utente durante il funzionamento dell'impianto può causare, in caso di guasti di funzionamento o di errori del programma, gravi lesioni personali e danni materiali!

Prima di eseguire il caricamento e il resettaggio del programma utente escludere che possano verificarsi situazioni di pericolo.

4.11 Vista parametri

4.11.1 Introduzione alla vista parametri

La Vista parametri contiene una panoramica generale di tutti i parametri rilevanti di un oggetto tecnologico. Viene visualizzata una panoramica delle impostazioni dei parametri, che possono essere modificate comodamente offline e online.

Nome nella vista della funzione	Nome nel DB	Valore di avvio	Tipo di dati	Commento
Valore di uscita sostitutivo	SavePosition	0.0	% Real	Immissione del valore di uscita sostitutivo
Limite superiore valore di uscita	./OutputUpperLimit	100.0	% Real	Immissione del limite superiore del valore di uscita
Limite inferiore valore di uscita	./OutputLowerLimit	0.0	% Real	Immissione del limite inferiore del valore di uscita
Riscontro superiore	./UpperPointOut	100.0	% Real	Immissione del valore di uscita al punto superiore
Riscontro inferiore	./LowerPointOut	0.0	% Real	Immissione del valore di uscita al punto inferiore
Feedback_PER giù	./LowerPointIn	0	Real	Immissione del valore inferiore di feedback
Feedback_PER su	./UpperPointIn	27648	Real	Immissione del valore superiore di feedback
Limite di avviso superiore	./InputUpperWarning	3.402822e...	% Real	Immissione del limite di avviso superiore
Limite di avviso inferiore	./InputLowerWarning	-3.402822e...	% Real	Immissione del limite di avviso inferiore
Attiva immissione manuale		FALSE		Attiva l'immissione manuale dei parametri
Guadagno proporzionale	./Gain	1.0	Real	Immissione del guadagno proporzionale
Tempo di integrazione	./Ti	20.0	s Real	Immissione del tempo di integrazione
Tempo derivativo	./Td	0.0	Real	Immissione del tempo derivativo
Coefficiente ritardo derivativo	./TdFiltRatio	0.2	Real	Immissione del coefficiente per il ritardo derivativo
Ponderazione del component...	./PWeighting	1.0	Real	Immissione del fattore di ponderazione
Ponderazione del component...	./DWeighting	1.0	Real	Immissione del fattore di ponderazione
Tempo di campionamento de...	./Cycle	1.0	s Real	Immissione del tempo di campionamento
Ampiezza banda morta	./InputDeadBand	0.0	Real	Immissione dell'ampiezza della banda morta

- ① Scheda "Vista parametri"
- ② Barra degli strumenti (Pagina 50)
- ③ Navigazione (Pagina 51)
- ④ Tabella dei parametri (Pagina 51)

Funzioni

Per l'analisi, il controllo e il comando mirati dei parametri degli oggetti tecnologici sono disponibili le funzioni seguenti.

Funzioni di visualizzazione:

- Visualizzazione dei parametri in modalità offline e online
- Visualizzazione di informazioni di stato dei parametri
- Visualizzazione delle differenze di valore e possibilità di correggerle direttamente
- Visualizzazione di errori di configurazione
- Visualizzazione di variazioni dei valori in seguito a relazioni di dipendenza dei parametri

- Visualizzazione di tutti i valori di memoria di un parametro: valore di avvio nella CPU, valore di avvio nel progetto, valore di controllo
- Visualizzazione del confronto tra parametri dei valori di memoria di un parametro

Funzioni operative:

- Navigazione, per spostarsi rapidamente tra i parametri e le loro strutture.
- Filtro di testo per trovare più rapidamente determinati parametri.
- Funzione di ordinamento per adeguare l'ordine di parametri e gruppi di parametri.
- Funzione di memoria per salvare le impostazioni della Vista parametri.
- Controllo e comando dei valori dei parametri online.
- Conversione del formato di visualizzazione del valore.
- Salvataggio di un'Istantanea dei valori dei parametri della CPU per poter riprodurre situazioni di breve durata e reagire.
- Applicazione di un'Istantanea dei valori dei parametri come valori di avvio.
- Caricamento di Valori di avvio nella CPU.
- Funzioni di confronto per confrontare tra loro i valori dei parametri.

Validità

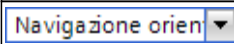
La Vista parametri qui descritta è disponibile per i seguenti oggetti tecnologici:

- PID_Compact
- PID_3Step
- PID_Temp
- CONT_C (solo S7-1500)
- CONT_S (solo S7-1500)
- TCONT_CP (solo S7-1500)
- TCONT_S (solo S7-1500)
- TO_Axis_PTO (S7-1200 Motion Control)
- TO_Positioning_Axis (S7-1200 Motion Control)
- TO_CommandTable_PTO (S7-1200 Motion Control)
- TO_CommandTable (S7-1200 Motion Control)

4.11.2 Struttura della vista parametri

4.11.2.1 Barra degli strumenti

Nella barra degli strumenti della vista parametri si possono selezionare le funzioni seguenti:

Simbolo	Funzione	Descrizione
	Controlla tutto	Avvia il controllo dei parametri visibili nella Vista parametri attiva (funzionamento online).
	Crea istantanea dei valori di controllo e applica i relativi valori come valori di avvio	Applica i valori di controllo attuali alla colonna "Istantanea" e aggiorna i valori di avvio nel progetto. Soltanto nel funzionamento online con PID_Compact, PID_3Step e PID_Temp.
	Carica valori iniziali di impostazione come valori attuali (inizializzazione dei valori di impostazione)	Trasferisce nella CPU i valori di avvio aggiornati nel progetto. Soltanto nel funzionamento online con PID_Compact, PID_3Step e PID_Temp.
	Crea istantanea dei valori di controllo	Applica i valori di controllo attuali alla colonna "Istantanea". Solo online.
	Comanda i parametri selezionati direttamente per una volta	Questo comando viene eseguito una volta il più rapidamente possibile senza relazione con un punto preciso del programma utente. Solo online.
	Seleziona struttura di navigazione	Commuta tra Navigazione orientata alle funzioni e Navigazione orientata ai dati.
	Filtro di testo...	Dopo aver inserito una stringa di caratteri vengono visualizzati tutti i parametri che contengono la stessa stringa in una delle colonne visibili al momento.
	Seleziona valori di confronto	Selezione dei valori dei parametri da confrontare online (Valore di avvio nel progetto, Valore di avvio nella CPU, Istantanea) Solo online.
	Memorizzare la disposizione	Salva le impostazioni di visualizzazione definite per la Vista parametri (ad es. struttura di navigazione selezionata, colonne delle tabelle attive ecc.)

4.11.2.2 Navigazione

All'interno della scheda "Vista parametri" è possibile selezionare alternativamente le seguenti struttura di navigazione:

Navigazione		Descrizione
Navigazione orientata alle funzioni	Tutti i parametri Parametri di configurazione Parametri di messa in servizio Altri parametri	Nella Navigazione orientata alle funzioni la struttura dei parametri si basa sulla struttura nella finestra di dialogo di configurazione (scheda "Vista delle funzioni"), di messa in servizio e di diagnostica. L'ultimo gruppo "Altri parametri" contiene tutti i restanti parametri dell'oggetto tecnologico.
Navigazione orientata ai dati	Tutti i parametri Input Output InOut Static Altri parametri	Nella Navigazione orientata ai dati la struttura dei parametri si basa sulla struttura nel DB di istanza /DB tecnologico. L'ultimo gruppo "Altri parametri" contiene quei parametri che non sono contenuti nel DB di istanza /DB tecnologico.

Dalla casella di riepilogo "Seleziona struttura di navigazione" è possibile cambiare struttura di navigazione.

4.11.2.3 Tabella dei parametri

La tabella seguente mostra il significato delle singole colonne della tabella dei parametri. Le colonne si possono visualizzare e nascondere secondo necessità.

- Colonna "Offline" = X: la colonna è visibile nel funzionamento offline.
- Colonna "Online" = X: la colonna è visibile online (collegamento online con la CPU).

Colonna	Descrizione	Offline	Online
Nome nella vista della funzione	Nome del parametro nella vista della funzione. Il campo di visualizzazione è vuoto nei parametri che non sono stati configurati con l'oggetto tecnologico.	X	X
Nome completo nel DB	Percorso completo del parametro nel DB di istanza /DB tecnologico. Il campo di visualizzazione è vuoto nei parametri che non sono contenuti nel DB di istanza /DB tecnologico.	X	X
Nome nel DB	Nome del parametro nel DB di istanza /DB tecnologico. Se il parametro è parte di una struttura o di un UDT viene aggiunto il prefisso ". ./" Il campo di visualizzazione è vuoto nei parametri che non sono contenuti nel DB di istanza /DB tecnologico.	X	X
Stato della configurazione	Visualizzazione dello stato di completezza della configurazione attraverso simboli di stato. vedere Stato della configurazione (offline) (Pagina 60)	X	
Risultato del confronto	Risultato della funzione "Confronta valori". Questa colonna viene visualizzata quando è attivo un collegamento online ed è selezionato il pulsante  "Controlla tutto".		X
Valore di avvio nel progetto	Valore di avvio configurato nel progetto. Segnalazione di errore quando i valori immessi sono sintatticamente o tecnologicamente falsi.	X	X

4.11 Vista parametri

Colonna	Descrizione	Offline	Online
Valore di default	Valore preassegnato al parametro. Il campo di visualizzazione è vuoto nei parametri che non sono contenuti nel DB di istanza /DB tecnologico.	X	X
Istantanea	Istantanea dei valori attuali nella CPU (valori di controllo). Segnalazione di errore in caso di valori tecnologicamente falsi.	X	X
Valore di avvio nella CPU	Valore di avvio nella CPU. Questa colonna viene visualizzata quando è attivo un collegamento online ed è selezionato il pulsante  "Controlla tutto". Segnalazione di errore in caso di valori tecnologicamente falsi.		X
Valore di controllo	Valore attuale nella CPU. Questa colonna viene visualizzata quando è attivo un collegamento online ed è selezionato il pulsante  "Controlla tutto". Segnalazione di errore in caso di valori tecnologicamente falsi.		X
Valore di comando	Valore con il quale deve essere modificato il Valore di controllo. Questa colonna viene visualizzata quando è attivo un collegamento online ed è selezionato il pulsante  "Controlla tutto". Segnalazione di errore quando i valori immessi sono sintatticamente o tecnologicamente falsi.		X
Selezione del valore di comando 	Selezione dei valori di comando da trasferire per mezzo del pulsante "Comanda i parametri selezionati direttamente per una volta". Questa colonna viene visualizzata insieme alla colonna "Valore di comando".		X
Valore minimo	Valore tecnologico più piccolo del parametro. Se il valore minimo dipende da altri parametri viene determinato: - Offline: dai valori di avvio nel progetto. - Online: dai valori di controllo.	X	X
Valore massimo	Valore tecnologico più alto del parametro. Se il valore massimo dipende da altri parametri viene determinato: - Offline: dai valori di avvio nel progetto. - Online: dai valori di controllo.	X	X
Valore di impostazione	Identifica il parametro come Valore di impostazione. Questi parametri possono essere inizializzati online.	X	X
Tipo di dati	Tipo di dati del parametro. Il campo di visualizzazione è vuoto nei parametri che non sono contenuti nel DB di istanza /DB tecnologico.	X	X
Ritenzione	Identifica il valore come valore a ritenzione. I valori dei parametri a ritenzione vengono conservati anche quando si disinserisce la tensione di alimentazione.	X	X
Accessibile da HMI	Indica se il sistema HMI può accedere a questo parametro in runtime.	X	X
Visibile in HMI	Indica se il parametro è visibile per default nella casella di riepilogo del sistema HMI.	X	X
Commento	Breve descrizione del parametro.	X	X

Vedere anche

[Confronto di valori \(Pagina 53\)](#)

4.11.3 Apertura della vista parametri**Presupposti**

L'oggetto tecnologico è stato aggiunto nella navigazione del progetto, ovvero il corrispondente DB di istanza /DB tecnologico dell'istruzione è già stato creato.

Procedimento

1. Aprire la cartella "Oggetti tecnologici" nella navigazione del progetto.
2. Aprire l'oggetto tecnologico nella navigazione del progetto.
3. Fare doppio clic sull'oggetto "Configurazione".
4. Selezionare nell'angolo in alto a destra la scheda "Vista parametri".

Risultato

Si apre la Vista parametri. Nella tabella dei parametri ogni parametro visualizzato è rappresentato da una riga.

Le proprietà dei parametri visualizzabili (colonne delle tabelle) variano a seconda che la Vista parametri funzioni offline oppure online.

Inoltre è possibile visualizzare o nascondere singole colonne precise.

Vedere anche

[Preimpostazione della vista parametri \(Pagina 53\)](#)

4.11.4 Preimpostazione della vista parametri**Preimpostazioni**

Per un utilizzo efficiente della Vista parametri è possibile adeguare la rappresentazione dei parametri e salvare le impostazioni definite.

È possibile eseguire e salvare i seguenti adeguamenti:

- Visualizzare e nascondere le colonne
- Modificare la larghezza delle colonne
- Modificare l'ordine delle colonne
- Commutare la navigazione
- Selezionare un gruppo di parametri nella navigazione
- Selezionare valori di confronto

Visualizzare e nascondere le colonne

Per visualizzare o nascondere le colonne nella tabella dei parametri procedere come segue:

1. Posizionare il puntatore del mouse nella riga di intestazione della tabella dei parametri.
2. Selezionare nel menu di scelta rapida il comando "Visualizza/Nascondi".
Vengono visualizzate le colonne disponibili.
3. Per visualizzare una colonna attivare la casella di scelta corrispondente.
4. Per nascondere una colonna disattivare la casella di scelta corrispondente.

Oppure

1. Posizionare il puntatore del mouse nella riga di intestazione della tabella dei parametri.
2. Selezionare nel menu di scelta rapida il comando "Visualizza tutte le colonne" per visualizzare tutte le colonne in modalità offline oppure online.

Alcune colonne possono essere visualizzate solo online: Vedere Tabella dei parametri ([Pagina 59](#)).

Modificare la larghezza delle colonne

Per adeguare la larghezza di una colonna al contenuto in modo che tutti i testi nelle righe siano leggibili procedere come segue:

1. Posizionare il puntatore del mouse sulla riga di intestazione della tabella dei parametri a destra accanto alla colonna da ottimizzare finché il puntatore si trasforma in una crocetta.
2. Fare doppio clic su questo punto.

Oppure

1. Aprire il menu di scelta rapida sulla riga di intestazione della tabella dei parametri.
2. Fare clic su
 - "Adatta larghezza colonne" oppure
 - "Ottimizza la larghezza di tutte le colonne".

Se le colonne sono troppo strette è possibile visualizzare il contenuto dei campi per intero passando brevemente il mouse sul singolo campo.

Modificare l'ordine delle colonne

Le colonne della tabella dei parametri si possono posizionare liberamente.

Per modificare l'ordine delle colonne procedere come segue:

1. Fare clic sull'intestazione della colonna e trascinarla nella posizione desiderata.
Quando si rilascia il tasto del mouse la colonna viene ancorata nella nuova posizione.

Commutare la navigazione

Per commutare la struttura di visualizzazione dei parametri procedere come segue:

1. Selezionare la navigazione desiderata nella casella di riepilogo "Seleziona struttura di navigazione":
 - Navigazione orientata ai dati
 - Navigazione orientata alle funzioni

Vedere anche Navigazione [\(Pagina 59\)](#).

Selezionare un gruppo di parametri nella navigazione

All'interno della navigazione selezionata è possibile scegliere tra la visualizzazione "Tutti i parametri" o la visualizzazione di un gruppo di parametri subordinato.

1. Nella navigazione fare clic sul gruppo di parametri desiderato.
Nella tabella dei parametri vengono visualizzati solo i parametri del gruppo.

Selezionare valori di confronto (online)

Per impostare i valori di confronto per la funzione "Confronta valori" procedere come segue:

1. Selezionare i valori di confronto desiderati nella casella di riepilogo "Seleziona valori di confronto":
 - Valore di avvio nel progetto / Valore di avvio nella CPU
 - Valore di avvio nel progetto / Istantanea
 - Valore di avvio nella CPU / Istantanea

Per default è impostata l'opzione "Valore di avvio nel progetto / Valore di avvio nella CPU".

Salvataggio della preimpostazione della Vista parametri

Per salvare gli adeguamenti della Vista parametri sopra descritti procedere come segue:

1. Adeguare la Vista parametri in base alle proprie esigenze.
2. Fare clic sul pulsante  "Memorizzare la disposizione" in alto a destra nella Vista parametri.

4.11.5 Utilizzo della vista parametri

4.11.5.1 Panoramica

La tabella seguente contiene una panoramica delle funzioni della Vista parametri in modalità online e offline descritte nel seguito.

- Colonna "Offline" = X: questa funzione è possibile in modalità offline.
- Colonna "Online" = X: questa funzione è possibile in modalità online.

Funzione/operazione	Offline	Online
Filtri della tabella dei parametri (Pagina 56)	X	X
Ordinamento della tabella dei parametri (Pagina 57)	X	X
Applicazione dei dati dei parametri in altri editor (Pagina 58)	X	X
Segnalazione degli errori (Pagina 58)	X	X
Modifica dei valori di avvio nel progetto (Pagina 59)	X	X
Stato della configurazione (offline) (Pagina 60)	X	
Controllo dei valori nella vista parametri (Pagina 61)		X
Crea istantanea dei valori di controllo (Pagina 63)		X
Comando dei valori (Pagina 64)		X
Confrontare i valori (Pagina 65)		X
Applicazione di valori del programma online come valori di avvio (Pagina 67)		X
Inizializzazione dei valori di impostazione nel programma online (Pagina 68)		X

4.11.5.2 Filtri della tabella dei parametri

I parametri nella tabella possono essere filtrati nei modi seguenti:

- Con il filtro di testo
- Con i sottogruppi nella navigazione

I metodi si possono applicare entrambi contemporaneamente.

Con il filtro di testo

È possibile applicare un filtro in base ai testi visibili nella tabella dei parametri. In altri termini è possibile limitare il filtro ai testi contenuti nelle righe e nelle colonne visualizzate.

1. Immettere nel campo di immissione "Filtro di testo..." la stringa di testo desiderata come criterio di filtro.

Nella tabella dei parametri compariranno solo i parametri che contengono la stringa specificata.

Il filtro di testo viene resettato:

- Quando si seleziona un altro gruppo di parametri nella navigazione.
- Quando si commuta tra navigazione orientata ai dati e orientata alle funzioni.

Con i sottogruppi nella navigazione

1. Nella navigazione fare clic sul gruppo di parametri desiderato, ad es. "Static".
Nella tabella dei parametri vengono visualizzati solo i parametri del gruppo Static. In alcuni gruppi della navigazione è possibile selezionare ulteriori sottogruppi.
2. Per visualizzare nuovamente tutti i parametri fare clic su "Tutti i parametri" nella navigazione.

4.11.5.3 Ordinamento della tabella dei parametri

I valori dei parametri sono ordinati per righe. La tabella dei parametri può essere ordinata in base a qualsiasi colonna visualizzata.

- Nelle colonne con valori numerici il contenuto è disposto in ordine numerico.
- Nelle colonne di testo il contenuto è disposto in ordine alfabetico.

Ordinamento per colonne

1. Posizionare il puntatore del mouse nella riga di intestazione della colonna desiderata.
Lo sfondo della cella viene evidenziato in blu.
2. Fare clic sull'intestazione della colonna.

Risultato

L'intera tabella dei parametri viene ordinata in base alla colonna selezionata. Nell'intestazione della colonna viene visualizzato un triangolino con il vertice verso l'alto.

Facendo nuovamente clic sull'intestazione della colonna si modifica l'ordinamento come segue:

- Simbolo "▲": tabella dei parametri in ordine ascendente.
- Simbolo "▼": tabella dei parametri in ordine discendente.
- Nessun simbolo: l'ordinamento viene annullato. Viene ripristinata la visualizzazione di default della tabella dei parametri.

Con l'ordinamento il prefisso "../" nella colonna "Nome nel DB" viene ignorato.

4.11.5.4 Applicazione dei dati dei parametri in altri editor

Dopo aver selezionato un'intera riga della tabella dei parametri è possibile utilizzare

- drag&drop
- <Ctrl+C>/<Ctrl+V>
- i comandi del menu di scelta rapida Copia/Incolla

per applicare i parametri ai seguenti editor di TIA Portal:

- Editor di programma
- Tabella di controllo
- Tabella dei segnali per Trace

Il parametro viene inserito con il nome completo: cfr. indicazione nella colonna "Nome completo nel DB".

4.11.5.5 Segnalazione degli errori

Segnalazione di errore

Gli errori di parametrizzazione che causano errori di compilazione (ad es. superamento di un valore limite) vengono visualizzati nella Vista parametri.

Ogni volta che viene immesso un valore nella Vista parametri viene direttamente verificata e visualizzata la correttezza tecnologica e sintattica.

I valori scorretti sono segnalati da:

- Simbolo di errore rosso nelle colonne "Stato della configurazione" (offline) o "Risultato del confronto" (online, in funzione del tipo di confronto scelto)

e/o

- Campo della tabella con sfondo rosso

Facendo clic sul campo con l'errore: messaggio di errore a tendina con indicazione del campo di valori ammesso o della sintassi necessaria (formato)

Errori di compilazione

Dal messaggio di errore del compiler è possibile aprire direttamente la Vista parametri (Navigazione orientata alle funzioni) con il parametro che ha causato l'errore per quei parametri che non sono visualizzati nella finestra di configurazione.

4.11.5.6 Modifica dei valori di avvio nel progetto

Con la Vista parametri è possibile modificare sia offline che online i Valori di avvio nel progetto:

- Per modificare i valori si utilizza la colonna "Valore di avvio nel progetto" della tabella dei parametri.
- Nella colonna "Stato della configurazione" della tabella dei parametri viene visualizzato l'avanzamento della configurazione attraverso i simboli della finestra di configurazione dell'oggetto tecnologico.

Condizioni generali

- Se dal parametro di cui è stato modificato il valore di avvio dipendono altri parametri, verrà adeguato anche il valore di avvio di questi parametri.
- Se un parametro di un oggetto tecnologico non è modificabile, non sarà modificabile nemmeno nella vista parametri. La modificabilità di un parametro può dipendere anche dai valori di altri parametri.

Definizione di nuovi valori di avvio

Per definire i valori di avvio per i parametri nella Vista parametri procedere come segue:

1. Aprire la Vista parametri dell'oggetto tecnologico.
2. Immettere nella colonna "Valore di avvio nel progetto" i Valori di avvio desiderati. Il valore deve corrispondere al tipo di dati del parametro e non deve superare il campo di valori del parametro.
I limiti del campo di valori sono visibili nelle colonne "Valore massimo" e "Valore minimo".

Nella colonna "Stato della configurazione" viene visualizzato l'"Avanzamento" della configurazione attraverso simboli colorati.

Vedere anche Stato della configurazione (offline) [\(Pagina 60\)](#)

Una volta adeguati i Valori di avvio e caricato l'oggetto tecnologico nella CPU, i parametri assumono il valore definito purché non siano stati dichiarati a ritenzione (colonna "Ritenzione").

Segnalazione di errore

Quando si immette un valore di avvio viene verificata e visualizzata direttamente la correttezza tecnologica e sintattica:

I valori di avvio scorretti sono segnalati da

- Simbolo di errore rosso nelle colonne "Stato della configurazione" (offline) o "Risultato del confronto" (online, in funzione del tipo di confronto scelto)

e/o

- Sfondo rosso nel campo "Valore di avvio nel progetto"
Facendo clic sul campo con l'errore: messaggio di errore a tendina con indicazione del campo di valori ammesso o della sintassi necessaria (formato)

Correzione Valori di avvio errati

1. Correggere i valori di avvio scorretti con l'aiuto delle informazioni del messaggio di errore a tendina.

Il simbolo di errore rosso, lo sfondo rosso del campo e il messaggio di errore a tendina non vengono più visualizzati.

Il progetto può essere compilato correttamente solo se i valori di avvio sono corretti.

4.11.5.7 Stato della configurazione (offline)

Lo stato della configurazione viene segnalato per mezzo di simboli:

- Nella colonna "Stato della configurazione" nella tabella dei parametri
- Nella struttura della Navigazione orientata alle funzioni o della Navigazione orientata ai dati

Simbolo nella colonna "Stato della configurazione"

Simbolo	Significato
	Il valore di avvio del parametro corrisponde al valore di default ed è valido. L'utente non ha ancora definito un valore di avvio.
	Il valore di avvio del parametro contiene un valore definito dall'utente o adeguato automaticamente. Il valore di avvio è diverso dal valore di default. Il valore di avvio è corretto e valido.
	Il valore di avvio del parametro non è valido (errore di sintassi o di tecnologia). Il campo di immissione ha lo sfondo rosso. Facendo clic con il mouse compare un messaggio di errore a tendina che spiega la causa.
	Solo per S7-1200 Motion Control: Il valore di controllo del parametro è valido ma contiene degli avvisi. Il campo di immissione ha lo sfondo giallo.
	Il parametro non è rilevante nella configurazione attuale.

Simbolo nella navigazione

I simboli nella navigazione mostrano l'"Avanzamento" della configurazione come nella finestra di configurazione dell'oggetto tecnologico.

Vedere anche

[Configurare oggetti tecnologici \(Pagina 46\)](#)

4.11.5.8 Controllo dei valori nella vista parametri

I valori che i parametri dell'oggetto tecnologico assumo attualmente nella CPU (Valori di controllo) possono essere controllati direttamente nella Vista parametri.

Presupposti

- Deve essere attivo un collegamento online.
- L'oggetto tecnologico è caricato nella CPU.
- L'elaborazione del programma è attiva (CPU in "RUN").
- La Vista parametri dell'oggetto tecnologico è aperta.

Procedura

1. Avviare il controllo facendo clic sul simbolo .

Non appena la Vista parametri è online vengono visualizzate anche le colonne seguenti:

- Risultato del confronto
- Valore di avvio nella CPU
- Valore di controllo
- Valore di comando
- Selezione del valore di comando

La colonna "Valore di controllo" mostra i valori dei parametri attuali sulla CPU.

Significato delle ulteriori colonne: vedere Tabella dei parametri (Pagina 59)

2. Per terminare la supervisione fare nuovamente clic sul simbolo .

Visualizzazione

Tutte le colonne disponibili esclusivamente online hanno lo sfondo arancione:

- I valori nelle celle con sfondo arancione chiaro  sono modificabili.
- I valori nelle celle con sfondo arancione scuro  non sono modificabili.

4.11.5.9 Conversione del formato di visualizzazione del valore

Il formato di visualizzazione del valore può essere selezionato dal menu di scelta rapida di una riga della tabella nella Vista parametri dell'oggetto tecnologico.

Il formato di visualizzazione dei seguenti valori può essere convertito sia nel funzionamento online che in quello offline:

- Valore di avvio nel progetto
- Valore di avvio nella CPU
- Valore massimo
- Valore minimo
- Istantanea
- Valore di controllo
- Valore di default
- Valore di comando

Il formato di visualizzazione impostato vale per tutti i valori della riga della tabella.

I seguenti formati di visualizzazione del valore possono essere modificati:

- Preimpostazione
- Esadec
- Ottale
- Bin
- Dec (+/-)
- DEC

A seconda del parametro selezionato nella Vista parametri, sono disponibili soltanto i formati di visualizzazione supportati.

Presupposti

- La Vista parametri dell'oggetto tecnologico è aperta

Procedura

Per convertire il formato di visualizzazione del valore, procedere nel seguente modo:

1. Selezionare una o più righe della tabella in cui si intende convertire il formato di visualizzazione.
2. Nel menu di scelta rapida selezionare il comando "Formato di visualizzazione".
3. Selezionare il formato di visualizzazione desiderato.

NOTA

Per convertire il formato di visualizzazione di un determinato tipo di dati in diverse righe della tabella, disporre la vista parametri secondo questo tipo di dati. Tenendo premuto il tasto <Maius>, evidenziare la prima e l'ultima riga della tabella con questo tipo di dati, quindi modificare il formato di visualizzazione per le righe selezionate della tabella.

4.11.5.10 Crea istantanea dei valori di controllo

I valori attuali dell'oggetto tecnologico sulla CPU (Valori di controllo) possono essere salvati e visualizzati nella Vista parametri.

Presupposti

- Deve essere attivo un collegamento online.
- L'oggetto tecnologico è caricato nella CPU.
- L'elaborazione del programma è attiva (CPU in "RUN").
- La Vista parametri dell'oggetto tecnologico è aperta.
- Il pulsante "Controlla tutto"  è attivato.

Procedura

Per visualizzare i valori dei parametri attuali procedere come segue:

1. Nella Vista parametri fare clic sul simbolo  "Crea istantanea dei valori di controllo".

Risultato

I Valori di controllo attuali vengono applicati una volta nella colonna "Istantanea" della tabella dei parametri.

I valori così "congelati" possono essere analizzati mentre i valori di controllo nella colonna "Valori di controllo" continuano a essere aggiornati.

4.11.5.11 Comando dei valori

Con la Vista parametri è possibile comandare l'oggetto tecnologico nella CPU.

Al parametro si possono assegnare una volta dei valori (Valore di comando) da comandare immediatamente. L'ordine di comando viene eseguito il più rapidamente possibile senza relazione con un punto preciso del programma utente.

PERICOLO

Pericoli del comando:

Modificare i valori dei parametri durante il funzionamento dell'impianto può causare gravi lesioni personali e danni materiali in caso di malfunzionamento o di errori del programma! Prima di eseguire la funzione "Comando" escludere che si possa verificare qualsiasi situazione di pericolo.

Requisiti

- Deve essere attivo un collegamento online.
- L'oggetto tecnologico è caricato nella CPU.
- L'elaborazione del programma è attiva (CPU in "RUN").
- La Vista parametri dell'oggetto tecnologico è aperta.
- Il pulsante "Controlla tutto"  è attivato.
- Il parametro può essere comandato (il campo corrispondente nella colonna "Valore di comando" ha lo sfondo arancione chiaro).

Procedura

Per comandare un parametro immediatamente procedere come segue:

1. Immettere i valori di comando desiderati nella colonna "Valori di comando" della tabella dei parametri.
2. Verificare se la casella di scelta per il comando nella colonna "Selezione del valore di comando" è attivata.

I valori di comando e le relative caselle di scelta dei parametri dipendenti vengono adeguati automaticamente.

3. Fare clic sul simbolo  "Comanda i parametri selezionati direttamente per una volta".

I parametri selezionati vengono comandati immediatamente per una volta con i valori predefiniti e possono essere controllati nella colonna "Valori di controllo". Le caselle di scelta per il comando nella colonna "Selezione del valore di comando" vengono disattivate automaticamente dopo l'esecuzione dell'ordine di comando.

Segnalazione di errore

Quando si immette un valore di comando viene verificata e visualizzata direttamente la correttezza tecnologica e sintattica:

I valori di comando scorretti sono segnalati da

- sfondo rosso nel campo "Valore di comando"

e

- Facendo clic sul campo con l'errore: messaggio di errore a tendina con indicazione del campo di valori ammesso o della sintassi necessaria (formato)

Valori di comando con errori

- I valori di comando tecnologicamente errati possono essere trasferiti.
- I valori di comando con errori di sintassi **non** possono essere trasferiti.

4.11.5.12 Confrontare i valori

Con le funzioni di confronto è possibile confrontare i seguenti valori di memoria di un parametro:

- Valore di avvio nel progetto
- Valore di avvio nella CPU
- Istantanea

Presupposti

- Deve essere attivo un collegamento online.
- L'oggetto tecnologico è caricato nella CPU.
- L'elaborazione del programma è attiva (CPU in "RUN").
- La Vista parametri dell'oggetto tecnologico è aperta.
- Il pulsante "Controlla tutto"  è attivato.

Procedimento

Per confrontare i Valori di avvio sui diversi sistemi di destinazione procedere come segue:

1. Fare clic sul simbolo  "Seleziona valori di confronto".

Si apre una casella di selezione con le opzioni di confronto:

- Valore di avvio nel progetto - Valore di avvio nella CPU (impostazione di default)
- Valore di avvio nel progetto - Istantanea
- Valore di avvio nella CPU - Istantanea

2. Selezionare l'opzione di confronto.

L'opzione di confronto scelta viene eseguita nel modo seguente:

- Nelle celle di intestazione di entrambe le colonne scelte per il confronto viene visualizzato il segno di una bilancia.
- Nella colonna "Risultato del confronto" viene visualizzato per mezzo di simboli il risultato del confronto delle colonne selezionate.

Simbolo nella colonna "Risultato del confronto"

Simbolo	Significato
	I valori di confronto sono uguali e corretti.
	I valori di confronto sono diversi e corretti.
	Almeno uno dei due valori di confronto è tecnologicamente o sintatticamente falso.
	Il confronto non può essere eseguito. Almeno uno dei due valori di confronto non è disponibile (ad es. Istantanea).
	Il confronto del valore non è utile perché non è rilevante in una delle configurazioni.

Simbolo nella navigazione

I simboli vengono visualizzati nello stesso modo nella navigazione se il risultato del confronto vale almeno per uno dei parametri sotto la struttura di navigazione visualizzata.

4.11.5.13 Applicazione di valori del programma online come valori di avvio

Per applicare tutti insieme i valori ottimizzati della CPU come Valori di avvio nel progetto creare un'Istantanea dei valori di controllo. I valori contrassegnati come "Valore di impostazione" dell'Istantanea verranno applicati nel progetto come Valori di avvio.

Presupposti

- L'oggetto tecnologico è di tipo "PID_Compact", "PID_3Step" oppure "PID_Temp".
- Deve essere attivo un collegamento online.
- L'oggetto tecnologico è caricato nella CPU.
- L'elaborazione del programma è attiva (CPU in "RUN").
- La Vista parametri dell'oggetto tecnologico è aperta.
- Il pulsante "Controlla tutto"  è attivato.

Procedimento

Per applicare valori ottimizzati della CPU procedere come segue:

1. Fare clic sul simbolo  "Crea istantanea dei valori di controllo e applica i relativi valori come valori di avvio".

Risultato

I valori di controllo attuali vengono applicati nella colonna "Istantanea" e i relativi valori di impostazione copiati come nuovi valori di avvio nella colonna "Valore di avvio nel progetto".

NOTA

Applicazione dei valori di singoli parametri

Anche i valori di singoli parametri che non sono contrassegnati come Valore di impostazione possono essere applicati dalla colonna "Istantanea" alla colonna "Valore di avvio nel progetto". Copiare i valori utilizzando i comandi "Copia" e "Incolla" del menu di scelta rapida e inserirli nella colonna "Valore di avvio nel progetto".

4.11.5.14 Inizializzazione dei valori di impostazione nel programma online

Tutti i parametri che nella Vista parametri sono contrassegnati come "Valore di impostazione" possono essere inizializzati tutti insieme con un nuovo valore nella CPU. I valori di comando vengono caricati dal progetto alla CPU. La CPU è in stato di funzionamento "RUN".

Per evitare la perdita di dati sulla CPU in caso di avvio a freddo o di nuovo avvio (a caldo) è necessario caricare anche l'oggetto tecnologico nella CPU.

PERICOLO

Pericolo nel modificare i valori dei parametri

Modificare i valori dei parametri durante il funzionamento dell'impianto può causare gravi lesioni personali e danni materiali in caso di malfunzionamento o di errori del programma!

Prima di inizializzare i valori di impostazione escludere che si possa verificare qualsiasi situazione di pericolo.

Presupposti

- L'oggetto tecnologico è di tipo "PID_Compact", "PID_3Step" oppure "PID_Temp".
- Deve essere attivo un collegamento online.
- L'oggetto tecnologico è caricato nella CPU.
- L'elaborazione del programma è attiva (CPU in "RUN").
- La Vista parametri dell'oggetto tecnologico è aperta.
- Il pulsante "Controlla tutto"  è attivato.
- I parametri contrassegnati come "dispongono di un 'Valore di avvio nel progetto'" senza errori tecnologici o di sintassi.

Procedimento

Per inizializzare tutti i valori di impostazione procedere come segue:

1. Immettere i valori desiderati nella colonna "Valore di avvio nel progetto".
Assicurarsi che i Valori di avvio siano tecnologicamente e sintatticamente corretti.
2. Fare clic sul simbolo  "Carica valori iniziali di impostazione come valori attuali".

Risultato

I valori di impostazione nella CPU vengono inizializzati con i valori di avvio del progetto.

4.12 Visualizzare il DB di istanza di un oggetto tecnologico

Per ciascun oggetto tecnologico viene creato un DB di istanza, in cui vengono salvati i parametri e le variabili statiche.

Procedimento

Per visualizzare il DB di istanza di un oggetto tecnologico, procedere nel modo seguente:

1. Aprire la cartella della CPU nella navigazione del progetto.
2. Aprire la cartella "Oggetti tecnologici".
3. Contrassegnare un oggetto tecnologico.
4. Scegliere nel menu di contesto il comando "Apri nell'editor"

Impiego di PID_Compact

5.1 Oggetto tecnologico PID_Compact

L'oggetto tecnologico PID_Compact è un regolatore PID continuo con ottimizzazione integrata. In alternativa si può configurare un regolatore a impulsi. Sono possibili sia il funzionamento manuale che quello automatico.

PID-Compact rileva ininterrottamente il valore istantaneo misurato all'interno di un circuito di regolazione e lo confronta con il setpoint desiderato. Partendo dalla differenza di regolazione risultante, l'istruzione PID_Compact calcola un valore di uscita mediante il quale il valore istantaneo viene adeguato al setpoint in modo rapido e stabile. Il valore di uscita è formato dai tre seguenti componenti del regolatore PID:

- **Componente P**
Il componente P del valore di uscita aumenta in modo proporzionale alla differenza di regolazione.
- **Componente I**
Il componente I del valore di uscita aumenta fintanto che la differenza di regolazione è compensata.
- **Componente D**
La componente D cresce all'aumentare della velocità di modifica della differenza di regolazione. Il valore istantaneo viene adeguato il più rapidamente possibile al setpoint. Se la velocità alla quale la differenza di regolazione varia diminuisce nuovamente, il componente D si riduce.

L'istruzione PID_Compact calcola autonomamente i parametri P, I e D del sistema regolato durante l'ottimizzazione iniziale. I parametri possono essere ulteriormente ottimizzati mediante un'ottimizzazione fine. Non è necessario, quindi, determinare manualmente i parametri.

Ulteriori informazioni

- Panoramica dei regolatori software (Pagina 43)
- Inserire oggetti tecnologici (Pagina 45)
- Configurare oggetti tecnologici (Pagina 46)
- Configurazione di PID_Compact dalla V2 (Pagina 71)
- Configurazione di PID_Compact V1 (Pagina 98)

FAQ

Ulteriori informazioni sono disponibili nelle seguenti FAQ alla sezione Siemens Industry Online Support.

- ID articolo: 79047707 (<https://support.industry.siemens.com/cs/ww/en/view/79047707>)

5.2 PID_Compact dalla V2

5.2.1 Configurazione di PID_Compact dalla V2

5.2.1.1 Impostazioni di base dalla V2

Introduzione dalla V2

In "Impostazioni di base", nella finestra di ispezione o di configurazione, si possono configurare le seguenti proprietà dell'oggetto tecnologico PID_Compact:

- Grandezza fisica
- Senso di regolazione
- Comportamento all'avviamento dopo il reset
- Setpoint (solo nella finestra di ispezione)
- Valore istantaneo (solo nella finestra di ispezione)
- Valore di uscita (solo nella finestra di ispezione)

Setpoint, valore istantaneo e valore di uscita

Setpoint, valore istantaneo e valore di uscita si possono configurare solo nella finestra di ispezione dell'editor di programma. Per ogni valore selezionare la sorgente:

- DB di istanza

Viene utilizzato il valore salvato nel DB di istanza.

Il valore deve essere aggiornato dal programma utente nel DB di istanza.

Nell'istruzione non può essere inserito alcun valore.

È possibile effettuare le modifiche tramite HMI.

- Istruzione

Viene utilizzato il valore interconnesso con l'istruzione.

A ogni richiamo dell'istruzione il valore viene scritto nel DB di istanza.

Non è possibile effettuare le modifiche tramite HMI.

Tipo di regolazione dalla V2

Grandezza fisica

Nel gruppo "Modo di regolazione" selezionare la grandezza fisica e l'unità per il setpoint e il valore istantaneo. I valori verranno quindi visualizzati in questa unità.

Senso di regolazione

Aumentando il valore di uscita si dovrebbe ottenere un aumento del valore istantaneo. In questo caso si parla di un senso di regolazione normale.

PID_Compact non funziona con un guadagno proporzionale negativo. Per ridurre il valore istantaneo aumentando il valore di uscita, attivare la casella di opzione "Inversione del senso di regolazione".

Esempi

- Aprendo una valvola di scarico il livello di riempimento di un contenitore diminuisce.
- Aumentando la capacità di raffreddamento la temperatura diminuisce.

Comportamento all'avviamento

1. Per passare al modo di funzionamento "Inattivo" dopo il nuovo avvio della CPU disattivare la casella di scelta "Attiva Mode dopo il riavvio della CPU".

Per passare al modo di funzionamento salvato in "Mode" dopo il nuovo avvio della CPU attivare la casella di scelta "Attiva Mode dopo il riavvio della CPU".

2. Nella casella di riepilogo "Imposta Mode su", selezionare il modo di funzionamento da attivare dopo il caricamento completo nel dispositivo.

Al termine del caricamento completo nel dispositivo, PID_Compact si avvia nel modo di funzionamento selezionato. Ad ogni riavvio PID_Compact si avvia nell'ultimo modo di funzionamento salvato alla voce "Mode".

Esempio

E' stata attivata la casella di scelta "Attiva Mode dopo il riavvio della CPU" e nell'elenco "Imposta Mode su" è stata selezionata la voce "Ottimizzazione iniziale". Al termine del caricamento completo nel dispositivo, PID_Compact si avvia nel modo di funzionamento "Ottimizzazione iniziale". Se l'ottimizzazione iniziale è ancora attiva, PID_Compact si avvia, dopo il riavvio della CPU, nuovamente in questo modo di funzionamento. Se l'ottimizzazione iniziale si è conclusa correttamente e il funzionamento automatico è attivo, PID_Compact si avvia, dopo il riavvio della CPU, nuovamente in questo modo di funzionamento.

Setpoint dalla V2

Procedimento

Per predefinire un setpoint fisso procedere nel modo seguente:

1. Selezionare "DB di istanza".
2. Inserire un setpoint, ad es. 80 °C.
3. Se necessario, cancellare una voce dell'istruzione.

Per predefinire un setpoint variabile procedere nel modo seguente:

1. Selezionare "Istruzione".
2. Inserire il nome della variabile REAL in cui è memorizzato il setpoint.

Alla variabile REAL è possibile assegnare diversi valori tramite comandi del programma, ad es. per modificare il setpoint con comandi a tempo.

Valore istantaneo dalla V2

Se si utilizza direttamente il valore dell'ingresso analogico, PID_Compact lo riporta in scala nella grandezza fisica.

Se si desidera prima trattare il valore dell'ingresso analogico è necessario scrivere un proprio programma per il trattamento. Supponiamo che il valore istantaneo non sia direttamente proporzionale al valore nell'ingresso analogico. Il valore istantaneo trattato dovrà essere nel formato in virgola mobile.

Procedimento

Per utilizzare direttamente il valore dell'ingresso analogico procedere nel modo seguente:

1. Selezionare la voce "Input_PER" nella casella di riepilogo "Input".
2. Selezionare come sorgente "Istruzione".
3. Inserire l'indirizzo dell'ingresso analogico.

Per utilizzare il valore istantaneo trattato nel formato in virgola mobile procedere nel modo seguente:

1. Selezionare la voce "Input" nella casella di riepilogo "Input".
2. Selezionare come sorgente "Istruzione".
3. Inserire il nome della variabile in cui è memorizzato il valore istantaneo trattato.

Valore di uscita dalla V2

PID_Compact fornisce tre valori di uscita. L'utilizzo di un determinato valore di uscita dipende dall'organo attuatore.

- Output_PER

Si accede all'organo attuatore tramite un ingresso analogico e lo si comanda con un segnale continuo, ad es. 0...10V, 4...20mA.

- Output

Il valore di uscita deve essere trattato con il programma utente, ad es. perché l'organo attuatore mostra un comportamento non lineare.

- Output_PWM

L'organo attuatore viene comandato mediante un'uscita digitale. Tramite una modulazione di larghezza di impulso vengono calcolati i tempi di attivazione e disattivazione variabili.

Procedura

Per utilizzare il valore di uscita analogico procedere nel modo seguente:

1. Selezionare la voce "Output_PER (analog)" nella casella di riepilogo "Output".
2. Selezionare "Istruzione".
3. Inserire l'indirizzo dell'uscita analogica.

Per trattare il valore di uscita con il programma utente procedere nel modo seguente:

1. Selezionare la voce "Output" nella casella di riepilogo "Output".
2. Selezionare "DB di istanza".
Il valore di uscita calcolato viene salvato nell DB di istanza.
3. Per il trattamento del valore di uscita utilizzare il parametro di uscita Output.
4. Trasferire il valore di uscita trattato all'organo attuatore mediante un'uscita digitale o analogica della CPU.

Per utilizzare il valore di uscita digitale procedere nel modo seguente:

1. Selezionare la voce "Output_PWM" nella casella di riepilogo "Output".
2. Selezionare "Istruzione".
3. Inserire l'indirizzo dell'uscita digitale.

5.2.1.2 Impostazioni del valore istantaneo dalla V2

Messa in scala del valore istantaneo dalla V2

Se è stato configurato l'utilizzo di Input_PER nelle impostazioni di base è necessario convertire il valore dell'ingresso analogico nella grandezza fisica del valore istantaneo. Nel campo di visualizzazione Input_PER viene visualizzata la configurazione attuale.

Se il valore istantaneo è direttamente proporzionale al valore dell'ingresso analogico, Input_PER viene riportato in scala in base a una coppia di valori inferiore e una superiore.

Procedimento

Per riportare in scala il valore istantaneo procedere nel modo seguente:

1. Immettere nei campi "Valore istantaneo inferiore riportato in scala" e "Inferiore" la coppia di valori inferiore.
2. Immettere nei campi "Valore istantaneo superiore riportato in scala" e "Superiore" la coppia di valori superiore.

Nella configurazione HW sono memorizzate coppie di valori predefinite. Per utilizzare le coppie di valori della configurazione HW procedere nel modo seguente:

1. Selezionare l'istruzione PID_Compact nell'editor di programma.
2. Nelle impostazioni di base collegare Input_PER con un ingresso analogico.
3. Nelle impostazioni del valore istantaneo fare clic sul pulsante "Impostazione automatica".

I valori esistenti vengono sovrascritti dai valori della configurazione HW.

Limiti del valore istantaneo dalla V2

Come valori limite è necessario definire un limite assoluto superiore e inferiore del valore istantaneo adeguato al sistema regolato. Non appena questi valori vengono superati in negativo o in positivo si verifica un errore (ErrorBits = 0001h). L'ottimizzazione viene interrotta se i limiti del valore istantaneo vengono superati. Il comportamento di PID_Compact in caso di errore in funzionamento automatico si definisce nelle impostazioni del valore di uscita.

5.2.1.3 Impostazioni avanzate dalla V2

Controllo del valore istantaneo dalla V2

La finestra "Controllo del valore istantaneo" consente di configurare il limite di avviso inferiore e superiore del valore istantaneo. Se durante il funzionamento viene superato uno dei limiti, nell'istruzione "PID_Compact" viene visualizzato un avviso:

- Nel parametro di uscita InputWarning_H, se è stato superato il limite di avviso superiore
- Nel parametro di uscita InputWarning_L, se non è stato raggiunto il limite di avviso inferiore

I limiti di avviso devono essere compresi entro il limite superiore e inferiore del valore istantaneo.

Se non si inserisce alcun valore, vengono utilizzati il limite superiore e inferiore del valore istantaneo.

Esempio

Limite superiore del valore istantaneo = 98 °C ; limite di avviso superiore = 90 °C

Limite di avviso inferiore = 10 °C ; limite inferiore del valore istantaneo = 0 °C

PID_Compact si comporta come descritto di seguito:

Valore istantaneo	InputWarning_H	InputWarning_L	ErrorBits	Modo di funzionamento
> 98 °C	TRUE	FALSE	0001h	Inattivo o Valore di uscita sostitutivo con controllo errori
≤ 98 °C e > 90 °C	TRUE	FALSE	0000h	Funzionamento automatico
≤ 90 °C e ≥ 10 °C	FALSE	FALSE	0000h	Funzionamento automatico
< 10°C e ≥ 0 °C	FALSE	TRUE	0000h	Funzionamento automatico
< 0 °C	FALSE	TRUE	0001h	Inattivo o Valore di uscita sostitutivo con controllo errori

Nelle impostazioni dei valori di uscita si configura come reagisce PID_Compact in caso di vengano superati o non vengano raggiunti il limite superiore o inferiore del valore istantaneo.

Vedere anche

[Parametri State e Mode a partire da V2 \(Pagina 275\)](#)

Limitazioni PWM dalla V2

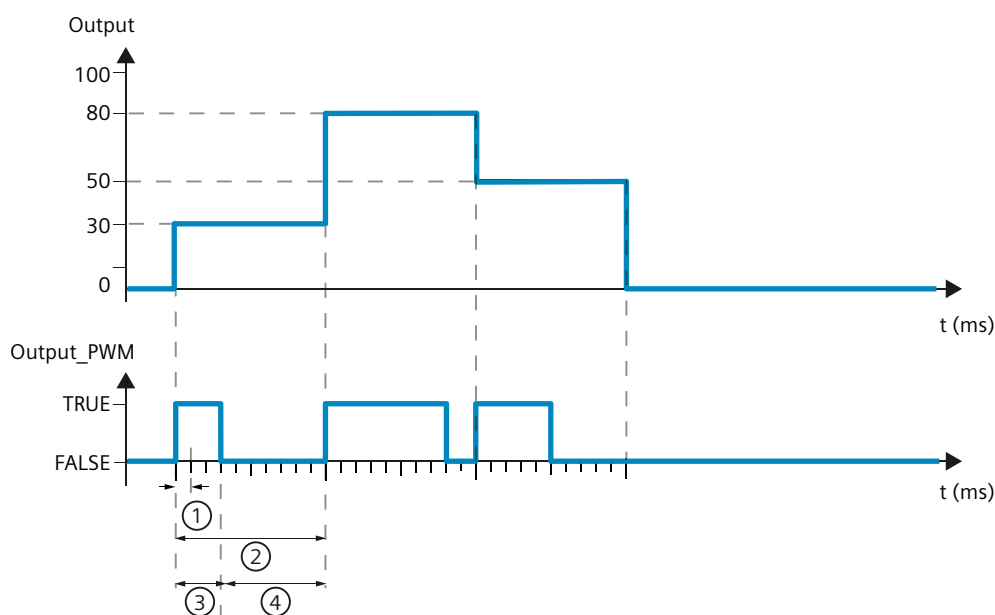
La modulazione di larghezza di impulso trasforma il valore del parametro di uscita Output in un treno di impulsi che viene emesso nel parametro di uscita Output_PWM.

Output viene calcolato durante il tempo di campionamento dell'algoritmo PID. Il tempo di campionamento viene utilizzato come durata periodo della modulazione di larghezza di impulso.

Il tempo di campionamento dell'algoritmo PID viene determinato durante la prima ottimizzazione o durante l'ottimizzazione fine. Se si impostano manualmente i parametri PID, si deve configurare lì anche il tempo di campionamento dell'algoritmo PID.

Output_PWM viene emesso durante il tempo di campionamento PID_Compact. Il tempo di campionamento PID_Compact corrisponde al tempo di ciclo dell'OB richiamante.

La durata dell'impulso è proporzionale al valore di Output e ammonta sempre a un multiplo intero del tempo di campionamento PID_Compact.



- ① Tempo di campionamento PID_Compact
- ② Tempo di campionamento algoritmo PID
- ③ Durata dell'impulso
- ④ Durata pausa

Il "Tempo ON minimo" e il "Tempo OFF minimo" vengono arrotondati a un multiplo intero del tempo di campionamento PID_Compact.

Un impulso o una pausa non sono mai inferiori ai Tempi ON minimo o OFF minimo. Le imprecisioni che risultano da ciò vengono sommate e compensate nel ciclo successivo.

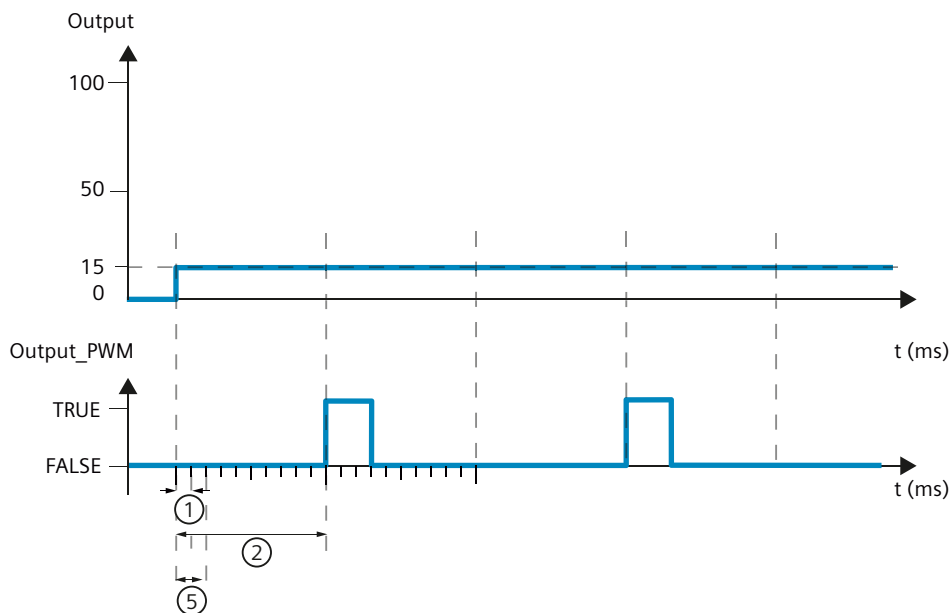
Esempio

Il tempo di campionamento PID_Compact (corrispondente al tempo di ciclo dell'OB richiamante) è di 100 ms.

Il tempo di campionamento dell'algoritmo PID (corrispondente alla durata del periodo) è di 1000 ms.

Tempo ON minimo = 200 ms

Output è del 15% costante. Il minimo impulso che PID_Compact può emettere corrisponde al 20%. Nel primo ciclo non viene emesso alcun impulso. Nel secondo ciclo l'impulso non emesso del primo ciclo viene aggiunto all'impulso del secondo ciclo.



- ① Tempo di campionamento PID_Compact
- ② Tempo di campionamento algoritmo PID
- ⑤ Tempo minimo di attivazione

Per ridurre la frequenza di commutazione e non sovraccaricare l'organo attuatore, prolungare i tempi di ON e OFF minimi.

Se si utilizzano "Output" oppure "Output_PER" è necessario configurare il tempo di attivazione e di disattivazione minimo sul valore 0.0.

NOTA

I tempi di attivazione e di disattivazione minimi hanno effetto solo sul parametro di uscita Output_PWM e non vengono utilizzati per generatori di impulsi eventualmente integrati nella CPU.

Valore di uscita dalla V2

Limiti valore di uscita

La finestra di configurazione "Limiti del valore di uscita" consente di impostare i limiti assoluti del valore di uscita in percentuale. Questi non vengono superati né nel funzionamento manuale, né in quello automatico. Se durante il funzionamento manuale viene impostato un valore di uscita non compreso entro i limiti, nella CPU il valore effettivo viene riportato entro i limiti configurati.

I limiti del valore di uscita devono essere adeguati al senso di regolazione.

I valori validi per i limiti del valore di uscita dipendono dal valore Output utilizzato.

Output	-100.0 ... 100.0 %
Output_PER	-100.0 ... 100.0 %
Output_PWM	0.0 ... 100.0 %

Comportamento in caso di errori

ATTENZIONE

L'impianto può subire danneggiamenti.

Se in caso di errore vengono emessi "Valore attuale per la durata dell'errore" o "Valore di uscita sostitutivo per la durata dell'errore", PID_Compact rimane in funzionamento automatico. In questo modo i limiti del valore istantaneo possono essere superati e l'impianto può essere danneggiato.

Configurare per il sistema regolato un comportamento da adottare in caso di errore che protegga l'impianto da danneggiamenti.

PID_Compact è preimpostato in modo tale per cui in caso di errore la regolazione resta attiva nella maggior parte dei casi. Se in funzionamento di regolazione si verificano errori frequenti, questa preimpostazione peggiora il comportamento del regolatore. In questo caso controllare il parametro Errorbits ed eliminare la causa dell'errore.

In caso di errori PID_Compact emette un valore di uscita configurabile:

- Zero (Inattivo)
emette per tutti gli errori 0.0 come valore di uscita e passa in modo di funzionamento "Inattivo". Il regolatore viene riattivato solo in presenza di un fronte di discesa Reset o di un fronte di salita ModeActivate.
- Valore attuale per la durata dell'errore
Se in **funzionamento automatico** si sono verificati i seguenti errori, PID_Compact ritorna a tale funzionamento non appena gli errori vengono eliminati.
Se si verificano uno o più dei seguenti errori PID_Compact resta in funzionamento automatico:
 - 0001h: Il parametro "Input" non rientra nei limiti del valore istantaneo.
 - 0800h: Errore del tempo di campionamento
 - 40000h: Valore del parametro Disturbance non valido.

Se in **funzionamento automatico** si verificano uno o più dei seguenti errori, PID_Compact commuta nel modo di funzionamento "Valore di uscita sostitutivo con controllo errori" ed emette l'ultimo valore di uscita valido:

- 0002h: Valore del parametro Input_PER non valido.
- 0200h: Valore del parametro Input non valido.
- 0400h: Calcolo del valore di uscita non riuscito.
- 1000h: Valore del parametro Setpoint non valido.

Se si verifica un errore in **funzionamento manuale** PID_Compact continua a utilizzare il valore manuale come valore di uscita. Se il valore manuale non è valido viene utilizzato il valore di uscita sostitutivo. Se non sono validi né il valore manuale né il valore di uscita sostitutivo viene utilizzato il limite inferiore del valore di uscita.

Se durante un'**ottimizzazione iniziale o un'ottimizzazione fine** si verifica il seguente errore, PID_Compact resta nel modo di funzionamento attivo:

- 0020h: L'ottimizzazione iniziale non è consentita durante l'ottimizzazione fine.

Con tutti gli altri errori PID_Compact interrompe l'ottimizzazione e passa al modo di funzionamento dal quale è stata avviata l'ottimizzazione.

Non appena l'errore viene eliminato PID_Compact torna al funzionamento automatico.

- Valore di uscita sostitutivo per la durata dell'errore
PID_Compact emette il valore di uscita sostitutivo.

Al verificarsi del seguente errore, PID_Compact rimane nel modo di funzionamento "Valore di uscita sostitutivo con controllo errori" ed emette il valore di uscita del limite inferiore:

- 20000h: Valore della variabile SubstituteOutput non valido.

Per tutti gli altri errori PID_Compact si comporta come descritto per "Valore attuale per la durata dell'errore".

Vedere anche

[Parametri State e Mode a partire da V2 \(Pagina 275\)](#)

Parametri PID V2

Nella finestra di configurazione "Parametri PID" vengono visualizzati i parametri PID. Durante l'ottimizzazione i parametri PID vengono adattati al proprio sistema regolato. Non è necessario, quindi, inserirli manualmente.

NOTA

In PID_Compact V1 i parametri PID attivi si trovano nella struttura sRet e a partire da PID_Compact V2 nella struttura Retain.CtrlParams.

Per evitare un comportamento errato del regolatore PID, modificare online i parametri PID attualmente effettivi solo nel modo di funzionamento "Inattivo".

Se si vogliono modificare i parametri PID online nei modi di funzionamento "Automatico" o "Manuale", modificarli nel modo seguente:

- PID_Compact V1: Modificare i parametri PID nella struttura sBackUp e applicare queste modifiche con sPid_Cmpt.b_LoadBackUp = TRUE nella struttura sRet.
- PID_Compact dalla V2: Modificare i parametri PID nella struttura CtrlParamsBackUp e applicare queste modifiche con LoadBackUp = TRUE nella struttura Retain.CtrlParams.

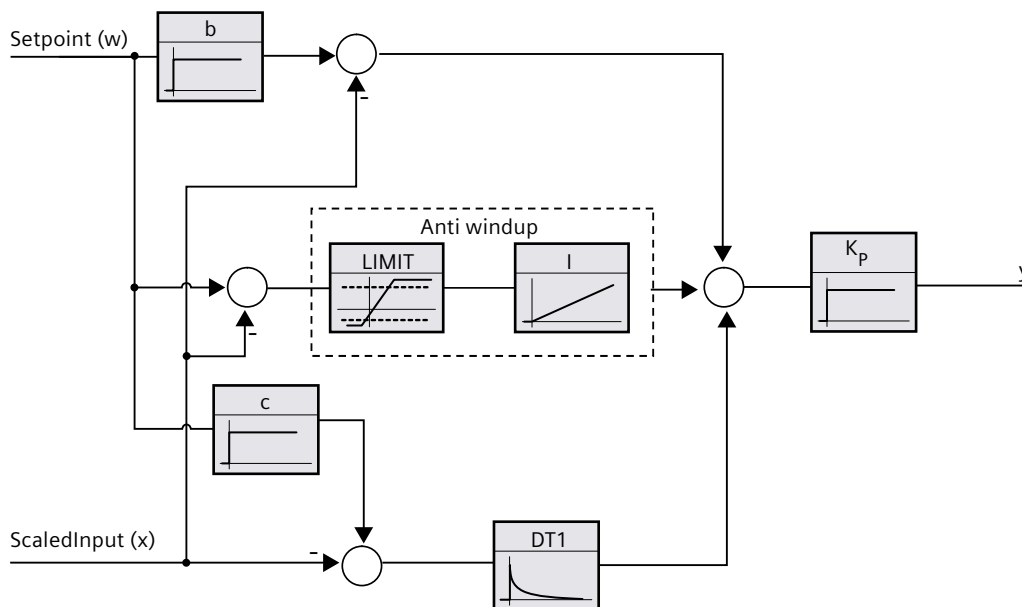
Le modifiche dei parametri PID eseguite online in modalità "Funzionamento automatico" possono causare gradini nel valore di uscita.

L'algoritmo PID utilizza la seguente formula:

$$y = K_p \left[(b \cdot w - x) + \frac{1}{T_I \cdot s} (w - x) + \frac{T_D \cdot s}{a \cdot T_D \cdot s + 1} (c \cdot w - x) \right]$$

Simbolo	Descrizione
y	Valore di uscita dell'algoritmo PID
K _p	Guadagno proporzionale
s	Operatore di Laplace
b	Ponderazione del componente P
w	Setpoint
x	Valore istantaneo
T _I	Tempo di integrazione
a	Coefficiente ritardo derivativo (ritardo derivativo T ₁ = a × T _D)
T _D	Tempo derivativo
c	Ponderazione del componente D

Il seguente grafico mostra i parametri nell'algoritmo PID:



Tutti i parametri PID sono a ritenzione. Per l'inserimento manuale dei parametri PID, eseguire il caricamento completo di PID_Compact.

Caricamento degli oggetti tecnologici del dispositivo [\(Pagina 48\)](#)

Guadagno proporzionale

Questo valore indica il guadagno proporzionale del regolatore. PID_Compact non opera con un guadagno proporzionale negativo. Il senso di regolazione si inverte in Impostazioni di base > Modo di regolazione.

Tempo di integrazione

Il tempo di integrazione determina il comportamento temporale del componente I. La disattivazione del componente I avviene impostando il tempo di integrazione = 0,0. Se il tempo di integrazione viene modificato online da un altro valore a 0,0 in modalità "Funzionamento automatico", il componente I precedente viene cancellato e si verifica un gradino del valore di uscita.

Tempo derivativo

Il tempo derivativo determina il comportamento temporale del componente D. La disattivazione del componente D avviene impostando il tempo derivativo = 0,0.

Coefficiente ritardo derivativo

L'effetto del componente D viene ritardato dal coefficiente per il ritardo derivativo.

Ritardo derivativo = Tempo derivativo × Coefficiente ritardo derivativo

- 0.0: Il componente D ha effetto solo per un ciclo e quindi è quasi inattivo.
- 0.5: Questo valore ha dato buoni risultati nelle operazioni con i sistemi regolati con **una** costante temporale dominante.
- > 1.0: Maggiore è il coefficiente, maggiore sarà il ritardo dell'azione del componente D.

Ponderazione del componente P

In caso di variazioni del setpoint è possibile ridurre il componente P.

Sono adeguati valori da 0.0 a 1.0.

- 1.0: componente P pienamente attivo in caso di variazione del setpoint
- 0.0: componente P non attivo in caso di variazione del setpoint

In caso di variazioni del valore istantaneo il componente P è sempre pienamente attivo.

Ponderazione del componente D

In caso di variazioni del setpoint è possibile ridurre il componente D.

Sono adeguati valori da 0.0 a 1.0.

- 1.0: componente D pienamente attivo in caso di variazione del setpoint
- 0.0: componente D non attivo in caso di variazione del setpoint

In caso di variazioni del valore istantaneo il componente D è sempre pienamente attivo.

Tempo di campionamento algoritmo PID

Poiché il sistema regolato richiede un certo tempo per reagire a una modifica del valore di uscita, è utile non calcolare questo valore in ogni ciclo. Il tempo di campionamento dell'algoritmo PID è il tempo che trascorre tra due calcoli del valore di uscita. Viene determinato durante l'ottimizzazione e arrotondato ad un multiplo del tempo di ciclo. Tutte le altre funzioni di PID_Compact vengono eseguite ad ogni richiamo.

In caso di impiego di Output_PWM, come durata periodo della modulazione di larghezza di impulso viene utilizzato il tempo di campionamento dell'algoritmo PID. La precisione del segnale di uscita viene determinata dal rapporto tra tempo di campionamento dell'algoritmo PID e tempo di ciclo dell'OB. Si raccomanda pertanto che il tempo di ciclo non superi un decimo del tempo di campionamento dell'algoritmo PID.

Regole per l'ottimizzazione

Nella casella di riepilogo "Struttura del regolatore" selezionare se debbano essere calcolati i parametri PI o PID.

- **PID**

Calcola i parametri PID durante l'ottimizzazione iniziale e finale

- **PI**

Calcola i parametri PI durante l'ottimizzazione iniziale e finale

- **Personalizzato**

Se, da un programma utente, sono state impostate diverse strutture del regolatore per l'ottimizzazione iniziale e finale, nella casella di riepilogo verrà visualizzata la voce "Personalizzato".

Parametri PID V3

Nella finestra di configurazione "Parametri PID" vengono visualizzati i parametri PID. Durante l'ottimizzazione i parametri PID vengono adeguati al sistema regolato ad eccezione dell'ampiezza della banda morta che deve essere configurata manualmente.

NOTA

I parametri PID attualmente attivi si trovano nella struttura Retain.CtrlParams.

Per evitare un comportamento errato del regolatore PID, modificare online i parametri PID attualmente effettivi solo nel modo di funzionamento "Inattivo".

Se si vogliono modificare i parametri PID online nei modi di funzionamento "Automatico" o "Manuale", modificarli nella struttura CtrlParamsBackUp e applicare le modifiche con LoadBackUp = TRUE nella struttura Retain.CtrlParams.

Le modifiche dei parametri PID eseguite online in modalità "Funzionamento automatico" possono causare gradini nel valore di uscita.

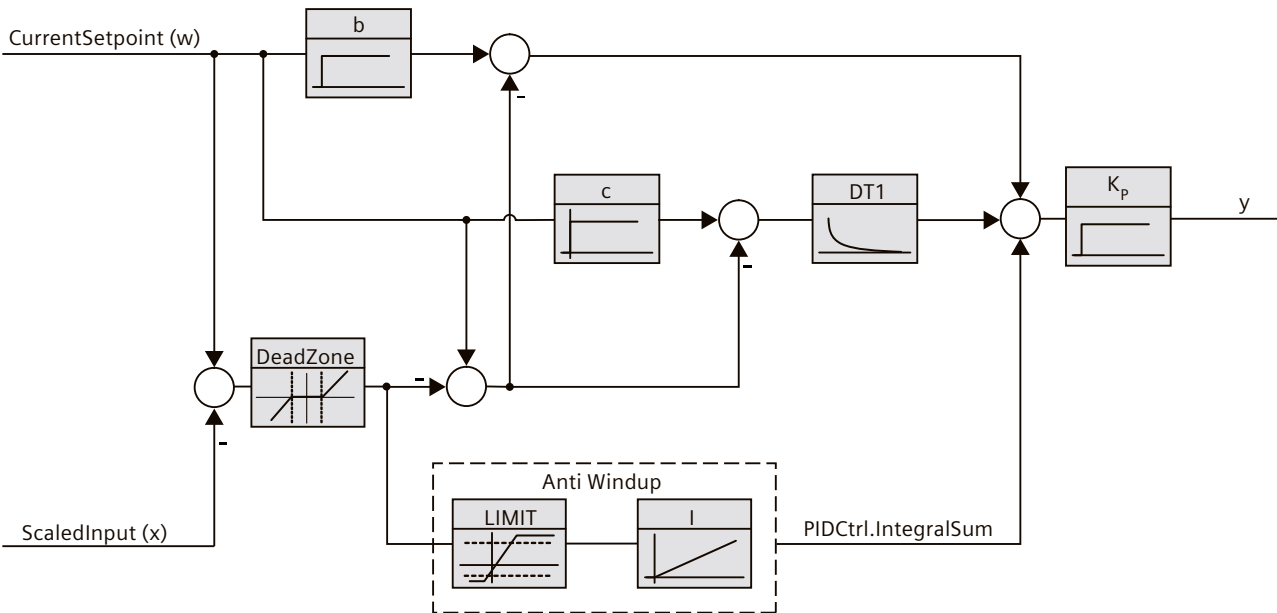
PID_Compact è un regolatore PIDT1 con Anti-Windup e ponderazione del componente P e D.

L'algorithmo PID utilizza la seguente formula (banda morta disattivata):

$$y = K_p \left[(b \cdot w - x) + \frac{1}{T_i \cdot s} (w - x) + \frac{T_d \cdot s}{a \cdot T_d \cdot s + 1} (c \cdot w - x) \right]$$

Simbolo	Descrizione	Parametri dell'istruzione PID_Compact
y	Valore di uscita dell'algorithmo PID	-
K _p	Guadagno proporzionale	Retain.CtrlParams.Gain
s	Operatore di Laplace	-
b	Ponderazione del componente P	Retain.CtrlParams.PWeighting
w	Setpoint	CurrentSetpoint
x	Valore istantaneo	ScaledInput
T _i	Tempo di integrazione	Retain.CtrlParams.Ti
a	Coefficiente ritardo derivativo (ritardo derivativo T ₁ = a × T _D)	Retain.CtrlParams.TdFiltRatio
T _D	Tempo derivativo	Retain.CtrlParams.Td
c	Ponderazione del componente D	Retain.CtrlParams.DWeighting
DeadZone	Ampiezza banda morta	Retain.CtrlParams.DeadZone

Il seguente grafico mostra i parametri nell'algorithmo PID:



Tutti i parametri PID sono a ritenzione. Per l'inserimento manuale dei parametri PID, eseguire il caricamento completo di PID_Compact.

Caricamento degli oggetti tecnologici del dispositivo

Guadagno proporzionale

Questo valore indica il guadagno proporzionale del regolatore. PID_Compact non opera con un guadagno proporzionale negativo. Il senso di regolazione si inverte in Impostazioni di base > Modo di regolazione.

Tempo di integrazione

Il tempo di integrazione determina il comportamento temporale del componente I. La disattivazione del componente I avviene impostando il tempo di integrazione = 0,0. Se il tempo di integrazione viene modificato online da un altro valore a 0,0 in modalità "Funzionamento automatico", il componente I precedente viene cancellato e si verifica un gradino del valore di uscita.

Se nel funzionamento automatico il valore di uscita raggiunge un suo limite, il componente I viene arrestato in funzione della direzione (Anti-Windup). A partire dalla versione 3.0 di PID_Compact il componente I viene limitato anche attivamente per evitare un comportamento di regolazione ritardato, ad esempio in caso di modifica dei limiti del valore di uscita. Le modifiche delle seguenti variabili possono causare un adeguamento del componente I in modalità automatica:

- Limiti del valore di uscita (variabili Config.OutputLowerLimit e Config.OutputUpperLimit)
- Setpoint (variabile Setpoint)
- Guadagno proporzionale (variabile Retain.CtrlParams.Gain)
- Ponderazione del componente P (Variabile Retain.CtrlParams.PWeighting)
- Grandezza di disturbo (variabile Disturbance)

Tempo derivativo

Il tempo derivativo determina il comportamento temporale del componente D. La disattivazione del componente D avviene impostando il tempo derivativo = 0,0.

Coefficiente ritardo derivativo

L'effetto del componente D viene ritardato dal coefficiente per il ritardo derivativo.

Ritardo derivativo = Tempo derivativo × Coefficiente ritardo derivativo

- 0,0: Il componente D ha effetto solo per un ciclo e quindi è quasi inattivo.
- 0,5: Questo valore ha dato buoni risultati nelle operazioni con i sistemi regolati con **una** costante temporale dominante.
- > 1,0: Maggiore è il coefficiente, maggiore sarà il ritardo dell'azione del componente D.

Ponderazione del componente P

In caso di variazioni del setpoint è possibile ridurre il componente P.

Sono adeguati valori da 0,0 a 1,0.

- 1,0: componente P pienamente attivo in caso di variazione del setpoint
- 0,0: componente P non attivo in caso di variazione del setpoint

In caso di variazioni del valore istantaneo il componente P è sempre pienamente attivo.

Ponderazione del componente D

In caso di variazioni del setpoint è possibile ridurre il componente D.

Sono adeguati valori da 0.0 a 1.0.

- 1.0: componente D pienamente attivo in caso di variazione del setpoint
- 0.0: componente D non attivo in caso di variazione del setpoint

In caso di variazioni del valore istantaneo il componente D è sempre pienamente attivo.

Tempo di campionamento algoritmo PID

Poiché il sistema regolato richiede un certo tempo per reagire a una modifica del valore di uscita, è utile non calcolare questo valore in ogni ciclo. Il tempo di campionamento dell'algoritmo PID è il tempo che trascorre tra due calcoli del valore di uscita. Viene determinato durante l'ottimizzazione e arrotondato ad un multiplo del tempo di ciclo. Tutte le altre funzioni di PID_Compact vengono eseguite ad ogni richiamo.

In caso di impiego di Output_PWM, come durata periodo della modulazione di larghezza di impulso viene utilizzato il tempo di campionamento dell'algoritmo PID. La precisione del segnale di uscita viene determinata dal rapporto tra tempo di campionamento dell'algoritmo PID e tempo di ciclo dell'OB. Si raccomanda pertanto che il tempo di ciclo non superi un decimo del tempo di campionamento dell'algoritmo PID.

Ampiezza banda morta

Se il valore istantaneo è disturbato, la componente di disturbo si ripercuote anche sul valore di uscita. Se il guadagno del regolatore è elevato e il componente D attivo, il valore di uscita può oscillare notevolmente. Se il valore istantaneo all'interno della banda morta si colloca intorno al setpoint, la differenza di regolazione viene soppressa, evitando così la reazione dell'algoritmo PID e consentendo di ridurre inutili oscillazioni del valore di uscita.

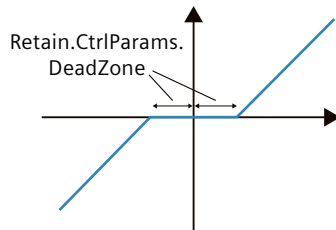
L'ampiezza della banda morta non viene impostata automaticamente durante l'ottimizzazione. L'ampiezza della banda morta deve essere configurata manualmente in modo corretto. La banda morta viene disattivata se l'ampiezza è = 0.0.

Con la banda morta attivata può verificarsi una differenza di regolazione permanente (differenza tra setpoint e valore istantaneo). Questo può avere conseguenze negative sull'esecuzione dell'ottimizzazione fine.

Se sono stati configurati valori diversi da 1.0 per la ponderazione del componente P o del componente D, le modifiche dei setpoint influenzano il valore di uscita anche all'interno della banda morta.

Indipendentemente dalla ponderazione, le modifiche del valore istantaneo all'interno della banda morta non influenzano il valore di uscita.

Il seguente grafico mostra l'effetto della banda morta: l'asse x / orizzontale rappresenta la differenza di regolazione = setpoint - valore istantaneo. L'asse y / verticale rappresenta il segnale di uscita della banda morta trasmesso all'algoritmo PID.



Regole per l'ottimizzazione

Nella casella di riepilogo "Struttura del regolatore" selezionare se debbano essere calcolati i parametri PI o PID.

- **PID**

Calcola i parametri PID durante l'ottimizzazione iniziale e finale

- **PI**

Calcola i parametri PI durante l'ottimizzazione iniziale e finale

- **Personalizzato**

Se, da un programma utente, sono state impostate diverse strutture del regolatore per l'ottimizzazione iniziale e finale, nella casella di riepilogo verrà visualizzata la voce "Personalizzato".

Vedere anche

[Scelta della struttura di regolazione nel sistema regolato dato \(Pagina 41\)](#)

5.2.2 Messa in servizio di PID_Compact dalla V2

5.2.2.1 Ottimizzazione iniziale dalla V2

L'ottimizzazione iniziale rileva la risposta del processo a un gradino del valore di uscita e cerca il punto di flesso. Dalla pendenza max. e dal tempo morto del sistema regolato vengono calcolati i parametri PID. I parametri PID migliori si ottengono eseguendo l'ottimizzazione iniziale e l'ottimizzazione fine.

Più stabile è il valore istantaneo, più facilmente e con maggiore precisione si possono determinare i parametri PID. Un rumore del valore istantaneo è accettabile fintanto che l'incremento del valore è significativamente maggiore rispetto al rumore. Ciò si verifica con la maggiore probabilità nei modi "Inattivo" o "Funzionamento manuale". I parametri PID vengono salvati prima di essere nuovamente calcolati.

Premessa

- L'istruzione "PID_Compact" viene richiamata in un OB di schedulazione orologio.
- ManualEnable = FALSE
- Reset = FALSE
- PID_Compact si trova in "Funzionamento manuale", nel modo "Inattivo" o in funzionamento automatico.
- Il setpoint e il valore istantaneo sono compresi entro i limiti configurati (vedere la configurazione di "Controllo del valore istantaneo").
- La differenza tra setpoint e valore istantaneo supera il 30 % della differenza tra limite superiore e limite inferiore del valore istantaneo.
- La distanza tra setpoint e valore istantaneo supera il 50% del setpoint.

Procedimento

Per eseguire l'ottimizzazione iniziale procedere nel seguente modo:

1. Fare doppio clic nella navigazione del progetto sulla voce "PID_Compact > Messa in servizio".
2. Selezionare la voce "Ottimizzazione iniziale" nella casella di riepilogo "Tipo di ottimizzazione".
3. Fare clic sul simbolo "Start".
 - Viene creato un collegamento online.
 - Viene avviata la registrazione dei valori.
 - Viene avviata l'ottimizzazione iniziale.

- Il campo "Stato" indica le operazioni in corso e gli eventuali errori. La barra di avanzamento indica lo stato di avanzamento dell'operazione in corso.

NOTA

Se la barra di avanzamento ha raggiunto il 100% e si deve presupporre un blocco dell'ottimizzazione, fare clic sul simbolo "Stop". Verificare la configurazione dell'oggetto tecnologico ed eventualmente riavviare l'ottimizzazione.

Risultato

Se l'ottimizzazione iniziale è stata eseguita senza messaggi di errore significa che i parametri PID sono stati ottimizzati. PID_Compact commuta sul funzionamento automatico e utilizza i parametri ottimizzati. Questi ultimi vengono mantenuti in caso di alimentazione OFF e nuovo avviamento della CPU.

Se un'ottimizzazione iniziale non è possibile, PID_Compact reagisce come configurato nel comportamento in caso di errore.

Vedere anche

[Parametri State e Mode a partire da V2 \(Pagina 275\)](#)

5.2.2.2 Ottimizzazione fine dalla V2

L'ottimizzazione fine genera un'oscillazione costante e limitata del valore istantaneo. I parametri PID per il punto di lavoro vengono ottimizzati partendo dall'ampiezza e dalla frequenza dell'oscillazione. A partire dai risultati vengono ricalcolati tutti i parametri PID. I parametri PID risultanti dall'ottimizzazione fine mostrano in genere un comportamento pilota e di disturbo migliore rispetto ai parametri PID dell'ottimizzazione iniziale. I parametri PID migliori si ottengono eseguendo l'ottimizzazione iniziale e l'ottimizzazione fine.

PID_Compact tenta di generare automaticamente un'oscillazione che sia maggiore rispetto al fruscio del valore istantaneo. L'ottimizzazione fine viene influenzata solo minimamente dalla stabilità del valore istantaneo. I parametri PID vengono salvati prima di essere nuovamente calcolati.

Requisiti

- L'istruzione PID_Compact viene richiamata in un OB di schedulazione orologio.
- ManualEnable = FALSE
- Reset = FALSE
- Il setpoint e il valore istantaneo sono compresi entro i limiti configurati.
- Il circuito di regolazione si è assestato nel punto di lavoro. Il punto di lavoro è raggiunto quando il valore istantaneo corrisponde al setpoint.
- Non sono previste anomalie.
- PID_Compact si trova nel modo di funzionamento inattivo, automatico o manuale.

Procedura a seconda della situazione di partenza

L'ottimizzazione fine può essere avviata dai modi di funzionamento "Inattivo", "Funzionamento automatico" o "Funzionamento manuale". All'avvio, l'ottimizzazione fine si svolge nel modo seguente:

- Funzionamento automatico

Per migliorare i parametri PID esistenti mediante un'ottimizzazione, avviare l'ottimizzazione fine dal modo di funzionamento automatico.

PID_Compact esegue la regolazione con i parametri PID esistenti finché il circuito di regolazione non si assesta e finché non sono soddisfatti i requisiti per un'ottimizzazione fine. Solo in seguito viene avviata l'ottimizzazione fine.

- Inattivo o Funzionamento manuale

Se sono soddisfatti i requisiti dell'ottimizzazione iniziale, viene avviata un'ottimizzazione iniziale. Con i parametri PID rilevati viene quindi effettuata una regolazione finché il circuito di regolazione è assestato e sono soddisfatti i requisiti per l'ottimizzazione fine. Solo in seguito viene avviata l'ottimizzazione fine. Se un'ottimizzazione iniziale non è possibile, PID_Compact reagisce come configurato nel comportamento in caso di errore.

Se il valore istantaneo per un'ottimizzazione iniziale si trova già troppo vicino al setpoint, si tenta di raggiungere il setpoint con il valore di uscita minimo o massimo. Ciò può causare una maggiore sovraelongazione.

Procedura

Per eseguire l'ottimizzazione fine procedere nel seguente modo:

1. Selezionare la voce "ottimizzazione fine" nella casella di riepilogo "Tipo di ottimizzazione".
2. Fare clic sul simbolo "Start".
 - Viene creato un collegamento online.
 - Viene avviata la registrazione dei valori.
 - Viene avviata l'ottimizzazione fine.
 - Il campo "Stato" indica le operazioni in corso e gli eventuali errori. La barra di avanzamento indica lo stato di avanzamento dell'operazione in corso.

NOTA

Se la barra di avanzamento ha raggiunto il 100% e si deve presumere un blocco dell'ottimizzazione, fare clic sul simbolo "Stop" nel gruppo "Tipo di ottimizzazione". Verificare la configurazione dell'oggetto tecnologico ed eventualmente riavviare l'ottimizzazione.

Risultato

Se durante l'ottimizzazione fine non si verificano errori, i parametri PID sono stati ottimizzati. PID_Compact commuta sul funzionamento automatico e utilizza i parametri ottimizzati. Questi ultimi vengono mantenuti in caso di alimentazione OFF e nuovo avviamento della CPU.

Se durante l'ottimizzazione fine si sono verificati errori, PID_Compact reagisce come configurato nel comportamento in caso di errore.

Vedere anche

[Parametri State e Mode a partire da V2 \(Pagina 275\)](#)

5.2.2.3 Modo di funzionamento manuale dalla V2

Qui di seguito viene spiegato come utilizzare il modo "Funzionamento manuale" nella finestra di messa in servizio dell'oggetto tecnologico "PID_Compact". Il funzionamento manuale è possibile anche in presenza di un errore.

Requisiti

- L'istruzione "PID_Compact" viene richiamata in un OB di schedulazione orologio.
- È stato stabilito un collegamento online con la CPU, la quale si trova nello stato di funzionamento "RUN".

Procedura

L'opzione "Funzionamento manuale" della finestra di messa in servizio può essere utilizzata per testare il sistema regolato impostando un valore manuale. Per predefinire un valore manuale procedere nel modo seguente:

1. Fare clic sul simbolo "Start".
2. Attivare la casella di controllo "Funzionamento manuale" nell'area "Stato online del regolatore".
PID_Compact lavora in modalità manuale. L'ultimo valore di uscita attuale rimane attivo.
3. Nel campo "Output" specificare il valore manuale espresso nell'unità %.
4. Fare clic sul simbolo .

Risultato

Il valore manuale viene scritto nella CPU e applicato immediatamente.

Per fare in modo che il valore di uscita venga nuovamente impostato dal regolatore PID disattivare la casella di controllo "Valore manuale". La commutazione nel funzionamento automatico è bumpless.

Vedere anche

[Parametri State e Mode a partire da V2 \(Pagina 275\)](#)

5.2.3 Regolazione override con PID_Compact dalla V2

Regolazione override

Nella regolazione override due o più regolatori si dividono un attuatore comune. In qualsiasi momento soltanto un regolatore accede all'attuatore e è attivo nel processo.

La scelta del regolatore preposto ad accedere all'attuatore avviene sulla base di un criterio logico. La decisione avviene spesso sulla base di un confronto dei valori di uscita di tutti i regolatori, partendo ad es. dalla selezione del massimo, sarà il regolatore con il valore di uscita più elevato ad avere accesso all'attuatore.

La scelta basata sul valore di uscita richiede che tutti i regolatori si operino nel funzionamento automatico. I regolatori che non esercitano alcuna azione sull'attuatore vengono adeguati tramite inseguimento. Questa procedura è necessaria al fine di evitare effetti windup e le conseguenti ripercussioni negative sul comportamento di regolazione, nonché per evitare la commutazione tra i regolatori.

Dalla versione V. 2.3 PID_Compact supporta la regolazione override. In questa versione il regolatore si avvale di una semplice procedura di inseguimento dei regolatori inattivi:

- Con le variabili `OverwriteInitialOutputValue` e `PIDCtrl.PIDInit` è possibile far preimpostare il componente I del regolatore in funzionamento automatico come se nell'ultimo ciclo l'algoritmo PID avesse calcolato $\text{Output} = \text{OverwriteInitialOutputValue}$ per il valore di uscita.
- `OverwriteInitialOutputValue` viene collegato al valore di uscita del regolatore che ha momentaneamente accesso all'organo attuatore.
- Impostando il bit `PIDCtrl.PIDInit` viene avviata la preassegnazione dei valori al componente I, e viene attivato il riavvio del clock del regolatore e del periodo PWM.
- Il calcolo successivo del valore di uscita nel ciclo attuale avviene sulla base dei valori preassegnati (e adeguati in tutti i regolatori), al componente I, nonché sulla base dei componenti I e P della differenza di regolazione attuale.
- Durante il richiamo di `PIDCtrl.PIDInit = TRUE` il componente D non è attivo e non viene quindi incluso nel calcolo del valore di uscita.

Questo comportamento è volto a garantire che il calcolo del valore di uscita attuale, e pertanto la scelta del regolatore preposto ad accedere all'attuatore, avvenga esclusivamente sulla base dello stato attuale del processo e dei parametri PI. Gli effetti windup nei regolatori inattivi, con conseguenti scelte errate nella logica di commutazione, vengono così evitati.

Presupposti

- PIDCtrl.PIDInit è valido soltanto se il componente I è attivo (variabile Retain.CtrlParams.Ti > 0.0).
- I valori a PIDCtrl.PIDInit e OverwriteInitialOutputValue nel programma utente devono essere assegnati dall'operatore (vedere l'esempio in basso). PID_Compact non apporta automaticamente alcuna modifica a queste variabili.
- PIDCtrl.PIDInit è valido soltanto se PID_Compact si trova nel funzionamento automatico (parametroState = 3).
- Se possibile, predisporre durante la selezione, un tempo di campionamento dell'algoritmo PID (variabile Retain.CtrlParams.Cycle) identico per tutti i regolatori e richiamare questi ultimi nello stesso OB di schedulazione orologio. Ciò garantisce che la commutazione non avvenga durante il clock del regolatore oppure durante il periodo PWM.

NOTA

Adattamento continuo dei limiti del valore di uscita

In luogo dell'inseguimento attivo qui descritto per i regolatori senza accesso all'attuatore, in altri sistemi di regolazione si realizza, in via alternativa, un adeguamento continuo dei limiti del valore di uscita.

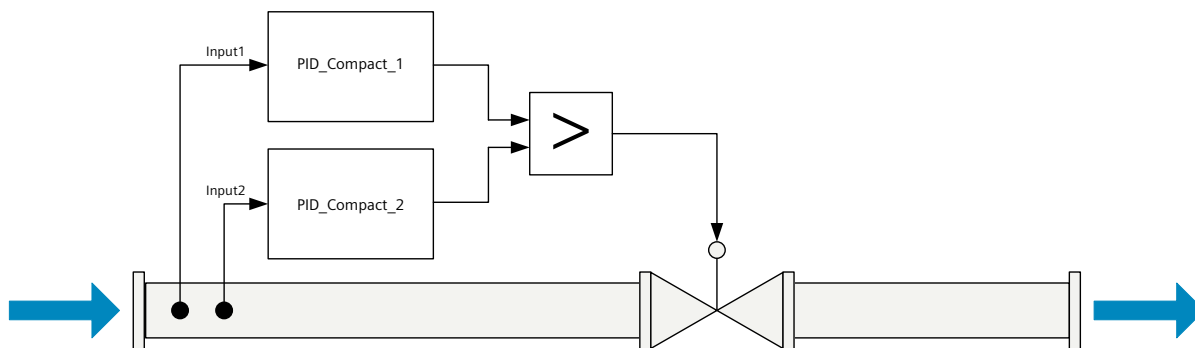
In PID_Compact questa procedura non è possibile in quanto il funzionamento automatico non supporta la modifica dei limiti del valore di uscita.

Esempio: Regolazione di un gasdotto

PID_Compact viene impiegato nella regolazione di un gasdotto.

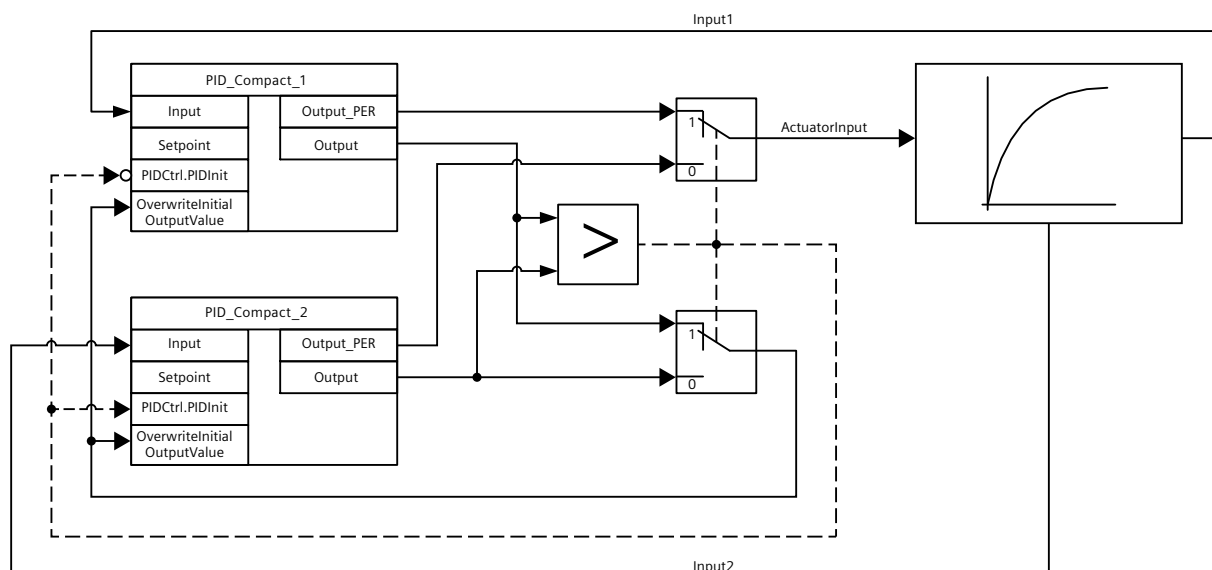
Il compito principale consiste nella regolazione della portata Input1. Per questo scopo trova impiego il regolatore PID_Compact_1. Inoltre, con l'ausilio del regolatore di limite PID_Compact_2, la pressioneInput2 (misurata nella direzione del flusso a monte della valvola) deve essere mantenuta al di sotto del limite superiore.

Portata e pressione vengono influenzate da un'unica valvola magnetica. Il valore di uscita del regolatore corrisponde all'apertura della valvola: con l'incremento del valore di uscita la valvola viene aperta con conseguente incremento della portata (senso di regolazione normale) e diminuzione della pressione (senso di regolazione inverso).



La valvola viene comandata tramite il valore di uscita PID_Compact nel formato periferia (parametro Output_PER) descrivendo la variabile di programma ActuatorInput.

La predefinitone del setpoint per la portata avviene nel parametro PID_Compact_1.Setpoint.
Il limite superiore della pressione viene predefinito come setpoint nel parametro PID_Compact_2.Setpoint.



Entrambi i regolatori dispongono della stessa, unica valvola che funge da attuatori comune. La logica di scelta del regolatore preposto ad accedere all'attuatori avviene in questo caso tramite selezione del massimo del valore di uscita (nel formato reale, parametro Output). Poiché il valore di uscita corrisponde all'apertura della valvola, il controllo viene assegnato al regolatore che richiede l'apertura più grande della valvola.

NOTA

Attivazione dell'inversione del senso di regolazione

Poiché un incremento del valore di uscita (apertura della valvola) nel regolatore di pressione PID_Compact_2 è finalizzato alla riduzione del valore istantaneo (pressione), è necessario attivare l'inversione del senso di regolazione: PID_Compact_2.Config.InvertControl = TRUE.

Nel funzionamento normale dell'impianto, il valore istantaneo della portata corrisponde al setpoint. Il regolatore di flusso PID_Compact_1 si è assestato sul valore di uscita stazionario PID_Compact_1.Output. Il valore istantaneo della pressione si colloca, nel funzionamento normale, nettamente al di sotto del limite superiore, predefinito come setpoint per PID_Compact_2. Per chiudere ulteriormente la valvola al fine di incrementare la pressione, il regolatore di pressione calcolerà un valore di uscita PID_Compact_2.Output inferiore a quello del regolatore di flusso PID_Compact_1.Output. La selezione del massimo prevista dalla logica di commutazione, consente pertanto al regolatore di flusso PID_Compact_1 di continuare ad accedere all'attuatori. Inoltre con le assegnazioni
PID_Compact_2.OverwriteInitialOutputValue = PID_Compact_1.Output e
PID_Compact_2.PIDCtrl.PIDInit = TRUE si garantisce l'adeguamento tramite inseguimento di PID_Compact_2.

Se la pressione si avvicina al limite superiore o lo supera ad es. a causa di un guasto, il regolatore di pressione PID_Compact_2 calcola un valore di uscita più elevato per aprire ulteriormente la valvola e ridurre in questo modo la pressione. Se PID_Compact_2.Output supera PID_Compact_1.Output, tramite selezione del massimo, il regolatore di pressione PID_Compact_2 accede all'attuatori e lo apre. Con le assegnazioni

PID_Compact_1.OverwriteInitialOutputValue = PID_Compact_2.Output e
PID_Compact_1.PIDCtrl.PIDInit = TRUE si garantisce l'adeguamento tramite inseguimento di
PID_Compact_1.

La pressione viene ridotta mentre la portata viene incrementata e non può più essere
mantenuta sul valore del setpoint.

Una volta eliminato il guasto, la pressione scende ulteriormente e l'apertura della valvola
viene ridotta dal regolatore di pressione. Se il regolatore di flusso calcola come valore di
uscita un'apertura superiore, l'impianto torna al funzionamento normale così da consentire al
regolatore di flusso PID_Compact_1 di accedere nuovamente all'attuatore.

Quest'esempio può essere realizzato con il seguente codice di programma SCL:

```
"PID_Compact_1"(Input := "Input1");  
"PID_Compact_2"(Input := "Input2");  
IF "PID_Compact_1".Output >= "PID_Compact_2".Output THEN  
  "ActuatorInput" := "PID_Compact_1".Output_PER;  
  "PID_Compact_1".PIDCtrl.PIDInit := FALSE;  
  "PID_Compact_2".PIDCtrl.PIDInit := TRUE;  
  "PID_Compact_2".OverwriteInitialOutputValue :=  
  "PID_Compact_1".Output;  
ELSE  
  "ActuatorInput" := "PID_Compact_2".Output_PER;  
  "PID_Compact_1".PIDCtrl.PIDInit := TRUE;  
  "PID_Compact_2".PIDCtrl.PIDInit := FALSE;  
  "PID_Compact_1".OverwriteInitialOutputValue :=  
  "PID_Compact_2".Output;  
END_IF;
```

5.2.4 Simulazione di PID_Compact dalla V2 con PLCSIM

NOTA

Simulazione con PLCSIM

La simulazione di PID_Compact V2.x con PLCSIM per la CPU S7-1200 non viene supportata.

PID_Compact dalla versione V2 può essere simulata con PLCSIM solo per la CPU S7-1500 e la CPU S7-1200 G2.

Nella simulazione con PLCSIM il comportamento temporale del PLC simulato non è esattamente identico a quello di un PLC "reale". In un PLC simulato, il clock di tempo effettivo di un OB di schedulazione orologio può presentare oscillazioni maggiori rispetto ai PLC "reali".

Nella configurazione standard, PID_Compact rileva automaticamente gli intervalli tra i richiami e ne controlla le oscillazione.

Nella simulazione di PID_Compact con PLCSIM può essere pertanto identificato un errore del tempo di campionamento (ErrorBits = DW#16#00000800).

Ciò comporta l'annullamento di ottimizzazioni in corso.

La reazione nel funzionamento automatico dipende dal valore della variabile ActivateRecoverMode.

Per evitare questo comportamento, si consiglia, nella simulazione con il PLCSIM, di configurare PID_Compact come indicato nel seguito:

- CycleTime.EnEstimation = FALSE
 - CycleTime.EnMonitoring = FALSE
 - CycleTime.Value: Assegnare a questa variabile il clock di tempo dell'OB di schedulazione orologio richiamante espresso in secondi.
-

5.3 PID_Compact V1

5.3.1 Configurazione di PID_Compact V1

5.3.1.1 Impostazione di base V1

Introduzione V1

In "Impostazioni di base", nella finestra di ispezione o di configurazione, si possono configurare le seguenti proprietà dell'oggetto tecnologico PID_Compact:

- Grandezza fisica
- Senso di regolazione
- Comportamento all'avviamento dopo il reset
- Setpoint (solo nella finestra di ispezione)
- Valore istantaneo (solo nella finestra di ispezione)
- Valore di uscita (solo nella finestra di ispezione)

Setpoint, valore istantaneo e valore di uscita

Setpoint, valore istantaneo e valore di uscita si possono configurare solo nella finestra di ispezione dell'editor di programma. Per ogni valore selezionare la sorgente:

- DB di istanza
Viene utilizzato il valore salvato nel DB di istanza.
Il valore deve essere aggiornato dal programma utente nel DB di istanza.
Nell'istruzione non può essere inserito alcun valore.
È possibile effettuare le modifiche tramite HMI.
- Istruzione
Viene utilizzato il valore interconnesso con l'istruzione.
A ogni richiamo dell'istruzione il valore viene scritto nel DB di istanza.
Non è possibile effettuare le modifiche tramite HMI.

Tipo di regolazione V1

Grandezza fisica

Nel gruppo "Modo di regolazione" selezionare la grandezza fisica e l'unità per il setpoint e il valore istantaneo. I valori verranno quindi visualizzati in questa unità.

Senso di regolazione

Aumentando il valore di uscita si dovrebbe ottenere un aumento del valore istantaneo. In questo caso si parla di un senso di regolazione normale.

PID_Compact non funziona con un guadagno proporzionale negativo. Per ridurre il valore istantaneo aumentando il valore di uscita, attivare la casella di opzione "Inversione del senso di regolazione".

Esempi

- Aprendo una valvola di scarico il livello di riempimento di un contenitore diminuisce.
- Aumentando la capacità di raffreddamento la temperatura diminuisce.

Comportamento all'avviamento dopo il reset

Per passare immediatamente, dopo il riavvio della CPU, all'ultimo modo di funzionamento attivo, attivare la casella di opzione "Dopo il nuovo avviamento della CPU attivare l'ultimo modo di funzionamento".

Se la casella è disattivata, PID_Compact rimane nel modo di funzionamento "Inattivo".

Setpoint V1

Procedimento

Per predefinire un setpoint fisso procedere nel modo seguente:

1. Selezionare "DB di istanza".
2. Inserire un setpoint, ad es. 80 °C.
3. Se necessario, cancellare una voce dell'istruzione.

Per predefinire un setpoint variabile procedere nel modo seguente:

1. Selezionare "Istruzione".
2. Inserire il nome della variabile REAL in cui è memorizzato il setpoint.

Alla variabile REAL è possibile assegnare diversi valori tramite comandi del programma, ad es. per modificare il setpoint con comandi a tempo.

Valore istantaneo V1

Se si utilizza direttamente il valore dell'ingresso analogico, PID_Compact lo riporta in scala nella grandezza fisica.

Se si desidera prima trattare il valore dell'ingresso analogico è necessario scrivere un proprio programma per il trattamento. Supponiamo che il valore istantaneo non sia direttamente proporzionale al valore nell'ingresso analogico. Il valore istantaneo trattato dovrà essere nel formato in virgola mobile.

Procedimento

Per utilizzare direttamente il valore dell'ingresso analogico procedere nel modo seguente:

1. Selezionare la voce "Input_PER" nella casella di riepilogo "Input".
2. Selezionare come sorgente "Istruzione".
3. Inserire l'indirizzo dell'ingresso analogico.

Per utilizzare il valore istantaneo trattato nel formato in virgola mobile procedere nel modo seguente:

1. Selezionare la voce "Input" nella casella di riepilogo "Input".
2. Selezionare come sorgente "Istruzione".
3. Inserire il nome della variabile in cui è memorizzato il valore istantaneo trattato.

Valore di uscita V1

PID_Compact fornisce tre valori di uscita. L'utilizzo di un determinato valore di uscita dipende dall'organo attuatore.

- Output_PER

Si accede all'organo attuatore tramite un ingresso analogico e lo si comanda con un segnale continuo, ad es. 0...10V, 4...20mA.

- Output

Il valore di uscita deve essere trattato con il programma utente, ad es. perché l'organo attuatore mostra un comportamento non lineare.

- Output_PWM

L'organo attuatore viene comandato mediante un'uscita digitale. Tramite una modulazione di larghezza di impulso vengono calcolati i tempi di attivazione e disattivazione variabili.

Procedura

Per utilizzare il valore di uscita analogico procedere nel modo seguente:

1. Selezionare la voce "Output_PER (analog)" nella casella di riepilogo "Output".
2. Selezionare "Istruzione".
3. Inserire l'indirizzo dell'uscita analogica.

Per trattare il valore di uscita con il programma utente procedere nel modo seguente:

1. Selezionare la voce "Output" nella casella di riepilogo "Output".
2. Selezionare "DB di istanza".
Il valore di uscita calcolato viene salvato nell DB di istanza.
3. Per il trattamento del valore di uscita utilizzare il parametro di uscita Output.
4. Trasferire il valore di uscita trattato all'organo attuatore mediante un'uscita digitale o analogica della CPU.

Per utilizzare il valore di uscita digitale procedere nel modo seguente:

1. Selezionare la voce "Output_PWM" nella casella di riepilogo "Output".
2. Selezionare "Istruzione".
3. Inserire l'indirizzo dell'uscita digitale.

5.3.1.2 Impostazioni del valore istantaneo V1

La finestra di configurazione "Impostazioni del valore istantaneo" consente di impostare i limiti assoluti del valore istantaneo per configurarne la normalizzazione.

Scala del valore istantaneo

Se nelle impostazioni di base è stato configurato l'utilizzo di Input_PER è necessario convertire il valore dell'ingresso analogico nella grandezza fisica del valore istantaneo. Nel campo di visualizzazione Input_PER viene visualizzata la configurazione attuale.

Se il valore istantaneo è direttamente proporzionale al valore dell'ingresso analogico, Input_PER viene riportato in scala in base ad una coppia di valori inferiore e una superiore.

1. Inserire nei campi di immissione "Valore istantaneo inferiore riportato in scala" e "Inferiore" la coppia di valori inferiore.
2. Inserire nei campi di immissione "Valore istantaneo superiore riportato in scala" e "Superiore" la coppia di valori superiore.

Nella configurazione HW sono memorizzate le preimpostazioni per le coppie di valori. Per utilizzare le coppie di valori nella configurazione HW procedere nel modo seguente:

1. Selezionare l'istruzione PID_Compact nell'editor di programma.
2. Nelle impostazioni di base collegare Input_PER con un ingresso analogico.
3. Fare clic nelle impostazioni del valore istantaneo sul pulsante "Impostazione automatica".

I valori esistenti vengono sovrascritti dai valori nella configurazione HW.

Controllo del valore istantaneo

Definire il limite superiore e inferiore del valore istantaneo. Quando, durante il funzionamento, viene superato uno dei limiti, la regolazione si disattiva e il valore di uscita viene impostato a 0%. Come valori limite si devono indicare valori sensati per il sistema regolato. È importante scegliere i valori opportuni durante l'ottimizzazione per ottenere parametri PID ottimali.

Il "Limite superiore valore istantaneo" preimpostato è il 120%. Nell'ingresso di periferia il valore istantaneo può superare al massimo del 18% il campo normale (campo di sovracomando). Non vengono più segnalati errori di superamento del "Limite superiore valore istantaneo". Vengono riconosciuti solo la rottura conduttore e il cortocircuito e PID_Compact passa nel modo di funzionamento "Inattivo".

AVVERTENZA

Se come valori limite del valore istantaneo si impostano valori molto elevati (per es. $-3,4 \cdot 10^{38} \dots +3,4 \cdot 10^{38}$), il controllo del valore istantaneo viene disattivato. Così, a causa di un errore si possono verificare danni all'impianto.

Vedere anche

[Controllo valore istantaneo V1 \(Pagina 102\)](#)

[Limitazioni PWM V1 \(Pagina 104\)](#)

[Valore di uscita V1 \(Pagina 106\)](#)

[Parametri PID V1 \(Pagina 106\)](#)

5.3.1.3 Impostazioni avanzate V1

Controllo valore istantaneo V1

La finestra "Controllo del valore istantaneo" consente di configurare il limite di avviso inferiore e superiore del valore istantaneo. Se durante il funzionamento viene superato uno dei limiti, nell'istruzione "PID_Compact" viene visualizzato un avviso:

- Nel parametro di uscita InputWarning_H, se è stato superato il limite di avviso superiore
- Nel parametro di uscita InputWarning_L, se non è stato raggiunto il limite di avviso inferiore

I limiti di avviso devono essere compresi entro il limite superiore e inferiore del valore istantaneo.

Se non si inserisce alcun valore, vengono utilizzati il limite superiore e inferiore del valore istantaneo.

Esempio

Limite superiore del valore istantaneo = 98 °C ; limite di avviso superiore = 90 °C

Limite di avviso inferiore = 10 °C ; limite inferiore del valore istantaneo = 0 °C

PID_Compact si comporta come descritto di seguito:

Valore istantaneo	InputWarning_H	InputWarning_L	Modo di funzionamento
> 98 °C	TRUE	FALSE	Inattivo
≤ 98 °C e > 90 °C	TRUE	FALSE	Funzionamento automatico
≤ 90 °C e ≥ 10 °C	FALSE	FALSE	Funzionamento automatico
< 10°C e ≥ 0 °C	FALSE	TRUE	Funzionamento automatico
< 0 °C	FALSE	TRUE	Inattivo

Vedere anche

[Impostazioni del valore istantaneo V1 \(Pagina 109\)](#)

[Limitazioni PWM V1 \(Pagina 104\)](#)

[Valore di uscita V1 \(Pagina 106\)](#)

[Parametri PID V1 \(Pagina 106\)](#)

Limitazioni PWM V1

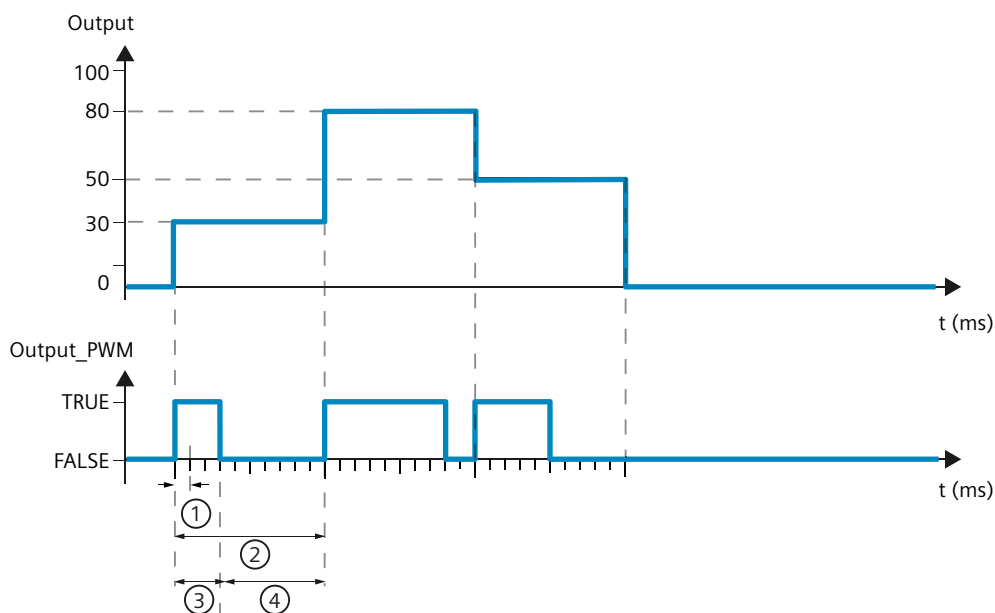
La modulazione di larghezza di impulso trasforma il valore del parametro di uscita Output in un treno di impulsi che viene emesso nel parametro di uscita Output_PWM.

Output viene calcolato durante il tempo di campionamento dell'algoritmo PID. Il tempo di campionamento viene utilizzato come durata periodo della modulazione di larghezza di impulso.

Il tempo di campionamento dell'algoritmo PID viene determinato durante la prima ottimizzazione o durante l'ottimizzazione fine. Se si impostano manualmente i parametri PID, si deve configurare lì anche il tempo di campionamento dell'algoritmo PID.

Output_PWM viene emesso durante il tempo di campionamento PID_Compact. Il tempo di campionamento PID_Compact corrisponde al tempo di ciclo dell'OB richiamante.

La durata dell'impulso è proporzionale al valore di Output e ammonta sempre a un multiplo intero del tempo di campionamento PID_Compact.



- ① Tempo di campionamento PID_Compact
- ② Tempo di campionamento algoritmo PID
- ③ Durata dell'impulso
- ④ Durata pausa

Il "Tempo ON minimo" e il "Tempo OFF minimo" vengono arrotondati a un multiplo intero del tempo di campionamento PID_Compact.

Un impulso o una pausa non sono mai inferiori ai Tempi ON minimo o OFF minimo. Le imprecisioni che risultano da ciò vengono sommate e compensate nel ciclo successivo.

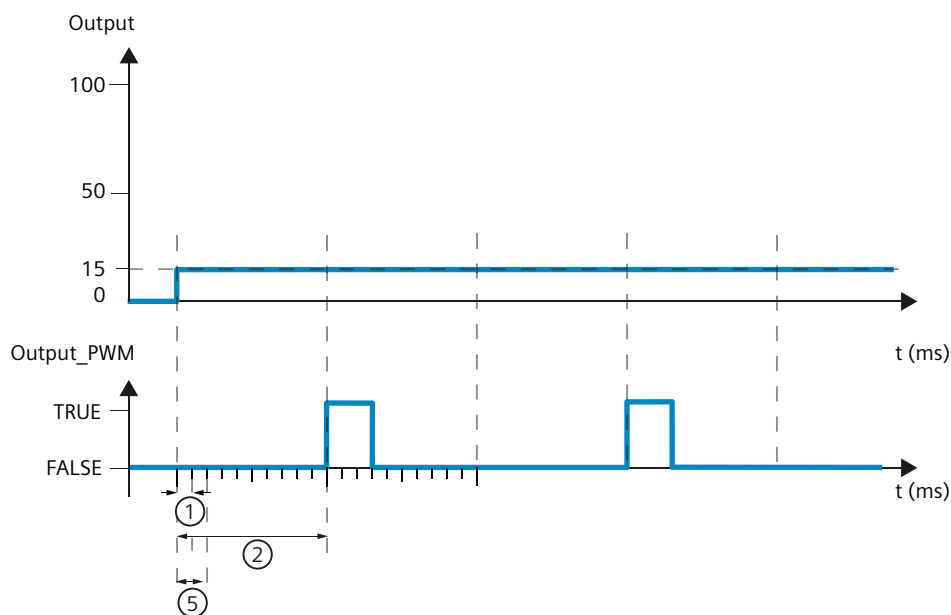
Esempio

Il tempo di campionamento PID_Compact (corrispondente al tempo di ciclo dell'OB richiamante) è di 100 ms.

Il tempo di campionamento dell'algoritmo PID (corrispondente alla durata del periodo) è di 1000 ms.

Tempo ON minimo = 200 ms

Output è del 15% costante. Il minimo impulso che PID_Compact può emettere corrisponde al 20%. Nel primo ciclo non viene emesso alcun impulso. Nel secondo ciclo l'impulso non emesso del primo ciclo viene aggiunto all'impulso del secondo ciclo.



- ① Tempo di campionamento PID_Compact
- ② Tempo di campionamento algoritmo PID
- ⑤ Tempo minimo di attivazione

Per ridurre la frequenza di commutazione e non sovraccaricare l'organo attuatore, prolungare i tempi di ON e OFF minimi.

Se si utilizzano "Output" oppure "Output_PER" è necessario configurare il tempo di attivazione e di disattivazione minimo sul valore 0.0.

NOTA

I tempi di attivazione e di disattivazione minimi hanno effetto solo sul parametro di uscita Output_PWM e non vengono utilizzati per generatori di impulsi eventualmente integrati nella CPU.

Vedere anche

[Impostazioni del valore istantaneo V1 \(Pagina 109\)](#)

[Controllo valore istantaneo V1 \(Pagina 110\)](#)

[Valore di uscita V1 \(Pagina 106\)](#)

[Parametri PID V1 \(Pagina 106\)](#)

Valore di uscita V1

La finestra di configurazione "Limiti del valore di uscita" consente di impostare i limiti assoluti del valore di uscita in percentuale. Questi non vengono superati né nel funzionamento manuale, né in quello automatico. Se durante il funzionamento manuale viene impostato un valore di uscita non compreso entro i limiti, nella CPU il valore effettivo viene riportato entro i limiti configurati.

I valori validi per i limiti del valore di uscita dipendono dal valore Output utilizzato.

Output	da -100,0 a 100,0
Output_PER	da -100,0 a 100,0
Output_PWM	da 0,0 a 100,0

In caso di errore PID_Compact imposta il valore di uscita a 0,0. Pertanto 0,0 deve sempre essere compreso entro i limiti del valore di uscita. Per ottenere un limite inferiore del valore di uscita maggiore di 0,0 è necessario sommare un offset a Output e Output_PER nel programma utente.

Vedere anche

[Impostazioni del valore istantaneo V1 \(Pagina 109\)](#)

[Controllo valore istantaneo V1 \(Pagina 110\)](#)

[Limitazioni PWM V1 \(Pagina 112\)](#)

[Parametri PID V1 \(Pagina 106\)](#)

Parametri PID V1

Nella finestra di configurazione "Parametri PID" vengono visualizzati i parametri PID. Durante l'ottimizzazione i parametri PID vengono adattati al proprio sistema regolato. Non è necessario, quindi, inserirli manualmente.

NOTA

In PID_Compact V1 i parametri PID attivi si trovano nella struttura sRet e a partire da PID_Compact V2 nella struttura Retain.CtrlParams.

Per evitare un comportamento errato del regolatore PID, modificare online i parametri PID attualmente effettivi solo nel modo di funzionamento "Inattivo".

Se si vogliono modificare i parametri PID online nei modi di funzionamento "Automatico" o "Manuale", modificarli nel modo seguente:

- PID_Compact V1: Modificare i parametri PID nella struttura sBackUp e applicare queste modifiche con sPid_Cmpt.b_LoadBackUp = TRUE nella struttura sRet.
- PID_Compact dalla V2: Modificare i parametri PID nella struttura CtrlParamsBackUp e applicare queste modifiche con LoadBackUp = TRUE nella struttura Retain.CtrlParams.

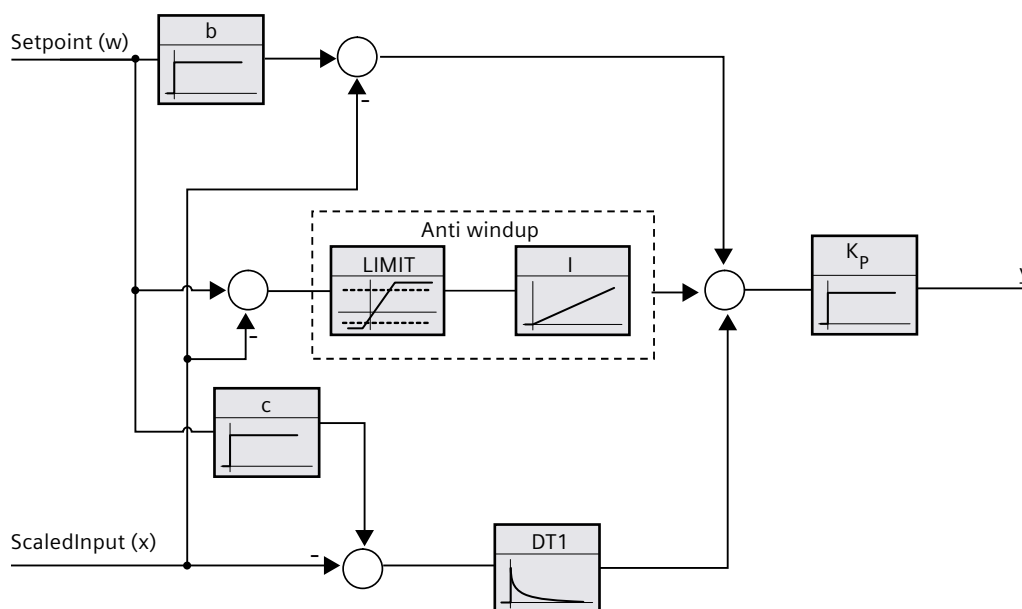
Le modifiche dei parametri PID eseguite online in modalità "Funzionamento automatico" possono causare gradini nel valore di uscita.

L'algoritmo PID utilizza la seguente formula:

$$y = K_p \left[(b \cdot w - x) + \frac{1}{T_i \cdot s} (w - x) + \frac{T_d \cdot s}{a \cdot T_d \cdot s + 1} (c \cdot w - x) \right]$$

Simbolo	Descrizione
y	Valore di uscita dell'algoritmo PID
K _p	Guadagno proporzionale
s	Operatore di Laplace
b	Ponderazione del componente P
w	Setpoint
x	Valore istantaneo
T _i	Tempo di integrazione
a	Coefficiente ritardo derivativo (ritardo derivativo T1 = a × T _D)
T _D	Tempo derivativo
c	Ponderazione del componente D

Il seguente grafico mostra i parametri nell'algoritmo PID:



Tutti i parametri PID sono a ritenzione. Per l'inserimento manuale dei parametri PID, eseguire il caricamento completo di PID_Compact.

Caricamento degli oggetti tecnologici del dispositivo [\(Pagina 48\)](#)

Guadagno proporzionale

Questo valore indica il guadagno proporzionale del regolatore. PID_Compact non opera con un guadagno proporzionale negativo. Il senso di regolazione si inverte in Impostazioni di base > Modo di regolazione.

Tempo di integrazione

Il tempo di integrazione determina il comportamento temporale del componente I. La disattivazione del componente I avviene impostando il tempo di integrazione = 0,0. Se il tempo di integrazione viene modificato online da un altro valore a 0.0 in modalità "Funzionamento automatico", il componente I precedente viene cancellato e si verifica un gradino del valore di uscita.

Tempo derivativo

Il tempo derivativo determina il comportamento temporale del componente D. La disattivazione del componente D avviene impostando il tempo derivativo = 0,0.

Coefficiente ritardo derivativo

L'effetto del componente D viene ritardato dal coefficiente per il ritardo derivativo.

Ritardo derivativo = Tempo derivativo × Coefficiente ritardo derivativo

- 0.0: Il componente D ha effetto solo per un ciclo e quindi è quasi inattivo.
- 0.5: Questo valore ha dato buoni risultati nelle operazioni con i sistemi regolati con **una** costante temporale dominante.
- > 1.0: Maggiore è il coefficiente, maggiore sarà il ritardo dell'azione del componente D.

Ponderazione del componente P

In caso di variazioni del setpoint è possibile ridurre il componente P.

Sono adeguati valori da 0.0 a 1.0.

- 1.0: componente P pienamente attivo in caso di variazione del setpoint
- 0.0: componente P non attivo in caso di variazione del setpoint

In caso di variazioni del valore istantaneo il componente P è sempre pienamente attivo.

Ponderazione del componente D

In caso di variazioni del setpoint è possibile ridurre il componente D.

Sono adeguati valori da 0.0 a 1.0.

- 1.0: componente D pienamente attivo in caso di variazione del setpoint
- 0.0: componente D non attivo in caso di variazione del setpoint

In caso di variazioni del valore istantaneo il componente D è sempre pienamente attivo.

Tempo di campionamento algoritmo PID

Poiché il sistema regolato richiede un certo tempo per reagire a una modifica del valore di uscita, è utile non calcolare questo valore in ogni ciclo. Il tempo di campionamento dell'algoritmo PID è il tempo che trascorre tra due calcoli del valore di uscita. Viene determinato durante l'ottimizzazione e arrotondato ad un multiplo del tempo di ciclo. Tutte le altre funzioni di PID_Compact vengono eseguite ad ogni richiamo.

In caso di impiego di Output_PWM, come durata periodo della modulazione di larghezza di impulso viene utilizzato il tempo di campionamento dell'algoritmo PID. La precisione del segnale di uscita viene determinata dal rapporto tra tempo di campionamento dell'algoritmo PID e tempo di ciclo dell'OB. Si raccomanda pertanto che il tempo di ciclo non superi un decimo del tempo di campionamento dell'algoritmo PID.

Regole per l'ottimizzazione

Nella casella di riepilogo "Struttura del regolatore" selezionare se debbano essere calcolati i parametri PI o PID.

- **PID**

Calcola i parametri PID durante l'ottimizzazione iniziale e finale

- **PI**

Calcola i parametri PI durante l'ottimizzazione iniziale e finale

- **Personalizzato**

Se, da un programma utente, sono state impostate diverse strutture del regolatore per l'ottimizzazione iniziale e finale, nella casella di riepilogo verrà visualizzata la voce "Personalizzato".

5.3.2 Messa in servizio di PID_Compact V1

5.3.2.1 Messa in servizio V1

La finestra di messa in servizio facilita la messa in servizio del regolatore PID. La vista delle curve permette di controllare il valore del setpoint, il valore istantaneo e il valore di uscita lungo l'asse temporale. La finestra di messa in servizio supporta le seguenti funzioni:

- Ottimizzazione iniziale del regolatore
- Ottimizzazione fine del regolatore
Questa opzione consente una regolazione precisa dei parametri PID.
- Controllo della regolazione in corso nella finestra delle curve
- Test del sistema regolato mediante impostazione di un valore di uscita manuale

Per tutte le funzione deve essere stabilito un collegamento online alla CPU.

Operazioni principali

- Selezionare il tempo di aggiornamento desiderato nella casella di riepilogo "Tempo di aggiornamento".
Tutti i valori nella finestra di messa in servizio vengono aggiornati nel momento indicato.
- Fare clic sul simbolo "Avvio" nel gruppo Misurazione se si vogliono utilizzare le funzioni per la messa in servizio.
Viene avviata la registrazione dei valori. Nella vista delle curve vengono immessi i valori attuali per il setpoint, il valore istantaneo e il valore di uscita. Viene abilitato l'utilizzo della finestra di messa in servizio.
- Fare clic sul simbolo "Stop" per disattivare le funzioni di messa in servizio.
I valori registrati nella vista delle curve possono essere ulteriormente analizzati.

Chiudendo la finestra di messa in servizio si interrompe la registrazione nella vista delle curve e i valori registrati vengono cancellati.

Vedere anche

[Ottimizzazione iniziale V1 \(Pagina 110\)](#)

[Ottimizzazione fine V1 \(Pagina 112\)](#)

[Modo di funzionamento "Funzionamento manuale" V1 \(Pagina 114\)](#)

5.3.2.2 Ottimizzazione iniziale V1

L'ottimizzazione iniziale rileva la risposta del processo a un gradino del valore di uscita e cerca il punto di flesso. Dall'incremento massimo e dal tempo morto del sistema regolato vengono calcolati i parametri PID ottimali.

Più stabile è il valore istantaneo, più facilmente e con maggiore precisione si possono determinare i parametri PID. Un disturbo del valore istantaneo è accettabile fintanto che l'incremento dello stesso è significativamente maggiore rispetto al disturbo. I parametri PID vengono salvati prima di essere nuovamente calcolati.

Premessa

- L'istruzione "PID_Compact" viene richiamata in un OB di schedulazione orologio.
- ManualEnable = FALSE
- PID_Compact si trova nel modo di funzionamento "Inattivo" o "Funzionamento manuale".
- Durante l'ottimizzazione il setpoint non può essere modificato altrimenti PID_Compact viene disattivato.
- Il setpoint e il valore istantaneo sono compresi entro i limiti configurati (vedere la configurazione di "Controllo del valore istantaneo").
- La differenza tra setpoint e valore istantaneo supera il 30 % della differenza tra limite superiore e limite inferiore del setpoint.
- La distanza tra setpoint e valore istantaneo supera il 50% del setpoint.

Procedimento

Per eseguire l'ottimizzazione iniziale procedere nel seguente modo:

1. Fare doppio clic nella navigazione del progetto sulla voce "PID_Compact > Messa in servizio".
2. Selezionare la voce "Ottimizzazione iniziale" nella casella di riepilogo "Tipo di ottimizzazione".
3. Fare clic sul simbolo "Start".
 - Viene creato un collegamento online.
 - Viene avviata la registrazione dei valori.
 - Viene avviata l'ottimizzazione iniziale.
 - Il campo "Stato" indica le operazioni in corso e gli eventuali errori. La barra di avanzamento indica lo stato di avanzamento dell'operazione in corso.

NOTA

Se la barra di avanzamento ha raggiunto il 100% e si deve presupporre un blocco dell'ottimizzazione, fare clic sul simbolo "Stop". Verificare la configurazione dell'oggetto tecnologico ed eventualmente riavviare l'ottimizzazione.

Risultato

Se l'ottimizzazione iniziale è stata eseguita senza messaggi di errore significa che i parametri PID sono stati ottimizzati. PID_Compact commuta sul funzionamento automatico e utilizza i parametri ottimizzati. Questi ultimi vengono mantenuti in caso di alimentazione OFF e nuovo avviamento della CPU.

Se non è possibile eseguire l'ottimizzazione iniziale, PID_Compact passa al modo di funzionamento "Inattivo".

Vedere anche

[Parametri State e sRet.i_Mode V1 \(Pagina 302\)](#)

[Messa in servizio V1 \(Pagina 117\)](#)

[Ottimizzazione fine V1 \(Pagina 112\)](#)

[Modo di funzionamento "Funzionamento manuale" V1 \(Pagina 114\)](#)

5.3.2.3 Ottimizzazione fine V1

L'ottimizzazione fine genera un'oscillazione costante e limitata del valore istantaneo. I parametri PID per il punto di lavoro vengono ottimizzati partendo dall'ampiezza e dalla frequenza dell'oscillazione. A partire dai risultati vengono ricalcolati tutti i parametri PID. I parametri PID risultanti dall'ottimizzazione fine mostrano in genere un comportamento pilota e di disturbo migliore rispetto ai parametri PID dell'ottimizzazione iniziale.

PID_Compact cerca di generare automaticamente un'oscillazione che è maggiore rispetto al disturbo del valore istantaneo. L'ottimizzazione fine viene influenzata solo minimamente dalla stabilità del valore istantaneo. I parametri PID vengono salvati prima di essere nuovamente calcolati.

Presupposti

- L'istruzione PID_Compact viene richiamata in un OB di schedulazione orologio.
- ManualEnable = FALSE
- Il setpoint e il valore istantaneo sono compresi entro i limiti configurati (vedere la configurazione di "Controllo del valore istantaneo").
- Il circuito di regolazione si è assestato nel punto di lavoro. Il punto di lavoro è raggiunto quando il valore istantaneo corrisponde al setpoint.
- Non si attendono anomalie.
- Durante l'ottimizzazione il setpoint non può essere modificato
- PID_Compact si trova nel modo di funzionamento inattivo, automatico o manuale.

Procedimento a seconda della situazione di partenza

L'ottimizzazione fine può essere avviata dai modi di funzionamento "Inattivo", "Funzionamento automatico" o "Funzionamento manuale", rispettivamente nelle modalità descritte di seguito:

- Funzionamento automatico

Per migliorare i parametri PID esistenti mediante un'ottimizzazione, avviare l'ottimizzazione fine dal modo di funzionamento automatico.

PID_Compact esegue quindi la regolazione con i parametri PID esistenti finché il circuito di regolazione è assestato e sono soddisfatti i requisiti per l'ottimizzazione fine. Solo in seguito viene avviata l'ottimizzazione fine.

- Inattivo o Funzionamento manuale

Se sono soddisfatti i presupposti dell'ottimizzazione iniziale, viene avviata un'ottimizzazione iniziale. Viene quindi effettuata la regolazione con i parametri PID rilevati finché il circuito di regolazione è assestato e sono soddisfatti i requisiti per l'ottimizzazione fine. Solo in seguito viene avviata l'ottimizzazione fine. Se non è possibile eseguire l'ottimizzazione iniziale, PID_Compact passa al modo di funzionamento "Inattivo".

Se il valore istantaneo per un'ottimizzazione iniziale si trova già troppo vicino al setpoint, si tenta di raggiungere il setpoint con il valore di uscita minimo o massimo. Ciò può causare un'elevata sovraoscillazione.

Procedimento

Per eseguire l'ottimizzazione fine procedere nel seguente modo:

1. Selezionare la voce "ottimizzazione fine" nella casella di riepilogo "Tipo di ottimizzazione".
2. Fare clic sul simbolo "Start".
 - Viene creato un collegamento online.
 - Viene avviata la registrazione dei valori.
 - Viene avviata l'ottimizzazione fine.
 - Il campo "Stato" indica le operazioni in corso e gli eventuali errori. La barra di avanzamento indica lo stato di avanzamento dell'operazione in corso.

NOTA

Se la barra di avanzamento ha raggiunto il 100% e si deve presumere un blocco dell'ottimizzazione, fare clic sul simbolo "Stop" nel gruppo "Tipo di ottimizzazione". Verificare la configurazione dell'oggetto tecnologico ed eventualmente riavviare l'ottimizzazione.

Risultato

Se l'ottimizzazione fine è stata eseguita senza messaggi di errore significa che i parametri PID sono stati ottimizzati. PID_Compact commuta sul funzionamento automatico e utilizza i parametri ottimizzati. Questi ultimi vengono mantenuti in caso di alimentazione OFF e nuovo avviamento della CPU.

Se durante l'ottimizzazione fine si verificano degli errori, PID_Compact passa alla modalità di funzionamento "Inattivo".

Vedere anche

[Parametri State e sRet.i_Mode V1 \(Pagina 302\)](#)

[Messa in servizio V1 \(Pagina 117\)](#)

[Ottimizzazione iniziale V1 \(Pagina 118\)](#)

[Modo di funzionamento "Funzionamento manuale" V1 \(Pagina 114\)](#)

5.3.2.4 Modo di funzionamento "Funzionamento manuale" V1

Di seguito viene spiegato come utilizzare il modo "Funzionamento manuale" nella finestra di messa in servizio dell'oggetto tecnologico "PID Compact".

Requisiti

- L'istruzione "PID_Compact" viene richiamata in un OB di schedulazione orologio.
- È stato stabilito un collegamento online con la CPU, la quale si trova nello stato di funzionamento "RUN".
- Le funzioni della finestra di messa in servizio sono state abilitate con l'icona "Misurazione attivata".

Procedura

L'opzione "Funzionamento manuale" della finestra di messa in servizio può essere utilizzata per testare il sistema regolato impostando un valore manuale. Per predefinire un valore manuale procedere nel modo seguente:

1. Attivare la casella di controllo "Funzionamento manuale" nell'area "Stato online del regolatore".
PID_Compact lavora in modalità manuale. L'ultimo valore di uscita attuale rimane attivo.
2. Nel campo "Output" specificare il valore manuale espresso nell'unità %.
3. Fare clic sull'icona di comando .

Risultato

Il valore manuale viene scritto nella CPU e applicato immediatamente.

NOTA

PID_Compact continua a sorvegliare il valore istantaneo. Se i limiti del valore istantaneo vengono superati, viene disattivato PID_Compact.

Per fare in modo che il valore di uscita venga nuovamente impostato dal regolatore PID disattivare la casella di controllo "Valore manuale". La commutazione nel funzionamento automatico è bumpless.

Vedere anche

[Parametri State e sRet.i_Mode V1 \(Pagina 302\)](#)

[Messa in servizio V1 \(Pagina 117\)](#)

[Ottimizzazione iniziale V1 \(Pagina 118\)](#)

[Ottimizzazione fine V1 \(Pagina 120\)](#)

5.3.3 Simulazione di PID_Compact V1 con PLCSIM

NOTA**Simulazione con PLCSIM**

Nella simulazione con PLCSIM il comportamento temporale del PLC simulato non è esattamente identico a quello di un PLC "reale". In un PLC simulato, il clock di tempo effettivo di un OB di schedulazione orologio può presentare oscillazioni maggiori rispetto ai PLC "reali".

Nella configurazione standard, PID_Compact rileva automaticamente gli intervalli tra i richiami e ne controlla le oscillazioni.

Nella simulazione di PID_Compact con PLCSIM può essere pertanto identificato un errore del tempo di campionamento (ErrorBits = DW#16#00000800)

PID_Compact passa in questo caso al modo di funzionamento "Inattivo" (State = 0)..

Per evitare questo comportamento, si consiglia, nella simulazione con il PLCSIM, di configurare PID_Compact come indicato nel seguito:

- sb_EnCyclEstimation = FALSE
 - sb_EnCyclMonitoring = FALSE
 - sPid_Calc.r_Cycle: Assegnare a questa variabile il clock di tempo dell'OB di schedulazione orologio richiamaente espresso in secondi.
-

Impiego di PID_3Step

6.1 Oggetto tecnologico PID_3Step

L'oggetto tecnologico PID_3Step mette a disposizione un regolatore PID con ottimizzazione delle valvole o degli attuatori con comportamento integrante.

È possibile configurare i seguenti regolatori:

- Regolatore passo-passo a tre punti con risposta di posizione
- Regolatore passo-passo a tre punti senza risposta di posizione
- Regolatore a valvola con valore di uscita analogico

All'interno di un circuito di regolazione, PID_3Step rileva ininterrottamente il valore istantaneo misurato e lo confronta con il setpoint. Partendo dalla differenza di regolazione risultante, PID_3Step calcola un valore di uscita mediante il quale il valore istantaneo raggiunge il setpoint nel modo più stabile e rapido possibile. Il valore di uscita è formato dai tre seguenti componenti del regolatore PID:

- **Componente P**
Il componente P del valore di uscita aumenta in modo proporzionale alla differenza di regolazione.
- **Componente I**
Il componente I del valore di uscita aumenta fintanto che la differenza di regolazione è compensata.
- **Componente D**
Il componente D cresce all'aumentare della velocità di modifica della differenza di regolazione. Il valore istantaneo viene adeguato il più rapidamente possibile al setpoint. Se la velocità alla quale la differenza di regolazione varia diminuisce nuovamente, il componente D si riduce.

L'istruzione PID_3Step calcola autonomamente i parametri P, I e D dei sistemi regolati durante l'ottimizzazione iniziale. I parametri possono essere ulteriormente ottimizzati mediante un'ottimizzazione fine. Non è necessario, quindi, determinare manualmente i parametri.

Ulteriori informazioni

- Panoramica dei regolatori software ([Pagina 43](#))
- Inserire oggetti tecnologici ([Pagina 45](#))
- Configurare oggetti tecnologici ([Pagina 46](#))
- Configurazione di PID_3Step V2 ([Pagina 117](#))
- Configurazione di PID_3Step V1 ([Pagina 137](#))

Principio

Ulteriori informazioni sono disponibili nelle seguenti FAQ alla sezione Siemens Industry Online Support.

- ID articolo: 68011827 (<https://support.industry.siemens.com/cs/ww/en/view/68011827>)

6.2 PID_3Step V2

6.2.1 Configurazione di PID_3Step V2

6.2.1.1 Impostazione di base V2

Introduzione V2

In "Impostazioni di base", nella finestra di ispezione o di configurazione, si possono configurare le seguenti proprietà dell'oggetto tecnologico PID_3Step:

- Grandezza fisica
- Senso di regolazione
- Comportamento all'avviamento dopo il reset
- Setpoint (solo nella finestra di ispezione)
- Valore istantaneo (solo nella finestra di ispezione)
- Valore di uscita (solo nella finestra di ispezione)
- Risposta di posizione (solo nella finestra di ispezione)

Setpoint, valore istantaneo, valore di uscita e risposta di posizione

Setpoint, valore istantaneo, valore di uscita e risposta di posizione si possono configurare solo nella finestra di ispezione dell'editor di programma. Per ogni valore selezionare la sorgente:

- DB di istanza

Viene utilizzato il valore salvato nel DB di istanza.

Il valore deve essere aggiornato dal programma utente nel DB di istanza.

Nell'istruzione non può essere inserito alcun valore.

È possibile effettuare le modifiche tramite HMI.

- Istruzione

Viene utilizzato il valore interconnesso con l'istruzione.

A ogni richiamo dell'istruzione il valore viene scritto nel DB di istanza.

Non è possibile effettuare le modifiche tramite HMI.

Tipo di regolazione V2

Grandezza fisica

Nel gruppo "Modo di regolazione" selezionare la grandezza fisica e l'unità per il setpoint e il valore istantaneo. I valori verranno quindi visualizzati in questa unità.

Senso di regolazione

Aumentando il valore di uscita si dovrebbe ottenere un aumento del valore istantaneo. In questo caso si parla di un senso di regolazione normale.

PID_3Step non funziona con un guadagno proporzionale negativo. Per ridurre il valore istantaneo aumentando il valore di uscita, attivare la casella di opzione "Inversione del senso di regolazione".

Esempi

- Aprendo una valvola di scarico il livello di riempimento di un contenitore diminuisce.
- Aumentando la capacità di raffreddamento la temperatura diminuisce.

Comportamento all'avviamento

1. Per passare al modo di funzionamento "Inattivo" dopo il nuovo avvio della CPU disattivare la casella di scelta "Attiva Mode dopo il riavvio della CPU".

Per passare al modo di funzionamento salvato in "Mode" dopo il nuovo avvio della CPU attivare la casella di scelta "Attiva Mode dopo il riavvio della CPU".

2. Nella casella di riepilogo "Imposta Mode su", selezionare il modo di funzionamento da attivare dopo il caricamento completo nel dispositivo.

Al termine del caricamento completo nel dispositivo, PID_3Step si avvia nel modo di funzionamento selezionato. Ad ogni riavvio PID_3Step si avvia nell'ultimo modo di funzionamento salvato alla voce "Mode".

Esempio

E' stata attivata la casella di scelta "Attiva Mode dopo il riavvio della CPU" e nell'elenco "Imposta Mode su" è stata selezionata la voce "Ottimizzazione iniziale". Al termine del caricamento completo nel dispositivo, PID_3Step si avvia nel modo di funzionamento "Ottimizzazione iniziale". Se l'ottimizzazione iniziale è ancora attiva, PID_3Step si avvia, dopo il riavvio della CPU, nuovamente in questo modo di funzionamento. Se l'ottimizzazione iniziale si è conclusa correttamente e il funzionamento automatico è attivo, PID_3Step si avvia, dopo il riavvio della CPU, nuovamente in questo modo di funzionamento.

Setpoint V2

Procedimento

Per predefinire un setpoint fisso procedere nel modo seguente:

1. Selezionare "DB di istanza".
2. Inserire un setpoint, ad es. 80 °C.
3. Se necessario, cancellare una voce dell'istruzione.

Per predefinire un setpoint variabile procedere nel modo seguente:

1. Selezionare "Istruzione".
2. Inserire il nome della variabile REAL in cui è memorizzato il setpoint.

Alla variabile REAL è possibile assegnare diversi valori tramite comandi del programma, ad es. per modificare il setpoint con comandi a tempo.

Valore istantaneo V2

Se si utilizza direttamente il valore dell'ingresso analogico, PID_3Step lo riporta in scala nella grandezza fisica.

Se si desidera prima trattare il valore dell'ingresso analogico è necessario scrivere un proprio programma per il trattamento. Supponiamo che il valore istantaneo non sia direttamente proporzionale al valore nell'ingresso analogico. Il valore istantaneo trattato dovrà essere nel formato in virgola mobile.

Procedimento

Per utilizzare direttamente il valore dell'ingresso analogico procedere nel modo seguente:

1. Selezionare la voce "Input_PER" nella casella di riepilogo "Input".
2. Selezionare come sorgente "Istruzione".
3. Inserire l'indirizzo dell'ingresso analogico.

Per utilizzare il valore istantaneo trattato nel formato in virgola mobile procedere nel modo seguente:

1. Selezionare la voce "Input" nella casella di riepilogo "Input".
2. Selezionare come sorgente "Istruzione".
3. Inserire il nome della variabile in cui è memorizzato il valore istantaneo trattato.

Risposta di posizione V2

La configurazione della risposta di posizione dipende dall'organo attuatore utilizzato.

- Organo attuatore senza risposta di posizione
- Organo attuatore con segnali di riscontro digitali
- Organo attuatore con risposta di posizione analogica
- Organo attuatore con risposta di posizione analogica e segnali di riscontro

Organo attuatore senza risposta di posizione

Per configurare PID_3Step per un organo attuatore senza risposta di posizione procedere nel seguente modo:

1. Selezionare la voce "No Feedback" nella casella di riepilogo "Feedback".

Organo attuatore con segnali di riscontro digitali

Per configurare PID_3Step per un organo attuatore con segnali di riscontro procedere nel seguente modo:

1. Selezionare la voce "No Feedback" nella casella di riepilogo "Feedback".
2. Attivare la casella di opzione "Segnali di riscontro organo attuatore".
3. Selezionare la voce "Istruzione" come sorgente per Actuator_H e Actuator_L.
4. Inserire gli indirizzi degli ingressi digitali per Actuator_H e Actuator_L.

Organo attuatore con risposta di posizione analogica

Per configurare PID_3Step per un organo attuatore con risposta di posizione analogica procedere nel seguente modo:

1. Selezionare la voce "Feedback" o "Feedback_PER" nella casella di riepilogo "Feedback".
 - Per Feedback_PER utilizzare direttamente il valore dell'ingresso analogico. La scala di Feedback_PER viene configurata nelle impostazioni dell'organo attuatore.
 - Per Feedback trattare il valore dell'ingresso analogico con il programma utente.
2. Selezionare come sorgente "Istruzione".
3. Inserire l'indirizzo dell'ingresso analogico o la variabile del programma utente.

Organo attuatore con risposta di posizione analogica e segnali di riscontro

Per configurare PID_3Step per un organo attuatore con risposta di posizione analogica e segnali di riscontro procedere nel seguente modo:

1. Selezionare la voce "Feedback" o "Feedback_PER" nella casella di riepilogo "Feedback".
2. Selezionare come sorgente "Istruzione".
3. Inserire l'indirizzo dell'ingresso analogico o la variabile del programma utente.
4. Attivare la casella di opzione "Segnali di riscontro organo attuatore".
5. Selezionare la voce "Istruzione" come sorgente per Actuator_H e Actuator_L.
6. Inserire gli indirizzi degli ingressi digitali per Actuator_H e Actuator_L.

Valore di uscita V2

PID_3Step fornisce un valore di uscita analogico (Output_PER) e valori di uscita digitali (Output_UP, Output_DN). L'utilizzo di un determinato valore di uscita dipende dall'organo attuatore.

- Output_PER

L'attuatore, che dispone di un tempo di regolazione motore considerevole, è accessibile da un'uscita analogica e viene comandato da un segnale continuo, ad es. 0...10 V, 4...20 mA. Il valore di Output_PER corrisponde alla destinazione della valvola ad es. Output_PER = 13824, se l'apertura della valvola deve essere del 50 %.

PID_3Step considera che, per quanto concerne ad es. l'ottimizzazione automatica e il comportamento anti-windup, a causa del tempo di regolazione motore, il valore di uscita analogico agisce con ritardo sul processo. Se nel processo non è attivo un tempo di regolazione del motore rilevante (ad es. con le valvole magnetiche) in modo da consentire al valore di uscita di agire direttamente e pienamente nel processo, utilizzare in suo luogo PID_Compact.

- Output_UP, Output_DN

L'attuatore dispone di un tempo di regolazione motore considerevole e viene comandato da due uscite digitali.

Output_UP sposta la valvola verso lo stato di apertura.

Output_DN sposta la valvola verso lo stato di chiusura.

Il tempo di regolazione del motore viene considerato sia nel calcolo del valore di uscita analogico, sia nel calcolo dei valori di uscita digitali. Questo tempo è necessario soprattutto nell'ottimizzazione automatica e nel comportamento anti-windup per un corretto funzionamento. Configurare pertanto il tempo di regolazione, alla voce "Impostazioni dell'organo attuatore", con un valore necessario al motore per spostare l'attuatore dallo stato chiuso a quello aperto.

Procedimento

Per utilizzare il valore di uscita analogico procedere nel modo seguente:

1. Selezionare la voce "Output (analog)" nella casella di riepilogo "Output".
2. Selezionare "Istruzione".
3. Inserire l'indirizzo dell'uscita analogica.

Per utilizzare il valore di uscita digitale procedere nel modo seguente:

1. Selezionare la voce "Output (digital)" nella casella di riepilogo "Output".
2. Selezionare la voce "Istruzione" per Output_UP e Output_DN.
3. Inserire gli indirizzi delle uscite digitali.

Per trattare il valore di uscita con il programma utente procedere nel modo seguente:

1. Selezionare la voce adatta all'organo attuatore nella casella di riepilogo "Output".
2. Selezionare "Istruzione".
3. Indicare il nome della variabile utilizzata per il trattamento del valore di uscita.
4. Trasferire il valore di uscita trattato all'organo attuatore mediante un'uscita analogica o digitale della CPU.

6.2.1.2 Impostazioni del valore istantaneo V2

Riporta in scala valore istantaneo V2

Se è stato configurato l'utilizzo di Input_PER nelle impostazioni di base è necessario convertire il valore dell'ingresso analogico nella grandezza fisica del valore istantaneo. Nel campo di visualizzazione Input_PER viene visualizzata la configurazione attuale.

Se il valore istantaneo è direttamente proporzionale al valore dell'ingresso analogico, Input_PER viene riportato in scala in base a una coppia di valori inferiore e una superiore.

Procedimento

Per riportare in scala il valore istantaneo procedere nel modo seguente:

1. Immettere nei campi "Valore istantaneo inferiore riportato in scala" e "Inferiore" la coppia di valori inferiore.
2. Immettere nei campi "Valore istantaneo superiore riportato in scala" e "Superiore" la coppia di valori superiore.

Nella configurazione HW sono memorizzate coppie di valori predefinite. Per utilizzare le coppie di valori della configurazione HW procedere nel modo seguente:

1. Selezionare l'istruzione PID_3Step nell'editor di programma.
2. Nelle impostazioni di base collegare Input_PER con un ingresso analogico.
3. Nelle impostazioni del valore istantaneo fare clic sul pulsante "Impostazione automatica".

I valori esistenti vengono sovrascritti dai valori della configurazione HW.

Limiti valore istantaneo V2

Come valori limite è necessario definire un limite assoluto superiore e inferiore del valore istantaneo adeguato al sistema regolato. Non appena questi valori vengono superati in negativo o in positivo si verifica un errore (ErrorBits = 0001h). L'ottimizzazione viene interrotta se i limiti del valore istantaneo vengono superati. Il comportamento di PID_3Step in caso di errore in funzionamento automatico si definisce nelle impostazioni dell'organo attuatore.

6.2.1.3 Impostazioni attuatore V2

Attuatore V2

Tempi specifici dell'organo attuatore

Per proteggere l'organo attuatore da eventuali danni occorre configurare il tempo di regolazione motore e i tempi minimi di attivazione e disattivazione. I dati sono disponibili nei dati tecnici dell'organo attuatore.

Il tempo di regolazione del motore è il tempo in secondi necessario al motore per spostare l'organo attuatore dallo stato chiuso a quello aperto. Il tempo di regolazione motore può essere rilevato durante la messa in servizio.

Il tempo di regolazione del motore viene considerato sia nel calcolo del valore di uscita analogico, sia nel calcolo dei valori di uscita digitali. Questo tempo è necessario soprattutto nell'ottimizzazione automatica e nel comportamento anti-windup per un corretto funzionamento.

Se nel processo non è attivo un tempo di regolazione del motore rilevante (ad es. con le valvole magnetiche) in modo da consentire al valore di uscita di agire direttamente e pienamente nel processo, utilizzare in suo luogo PID_Compact.

Il tempo di regolazione del motore è a ritenzione. Per la modifica manuale del tempo di regolazione del motore, eseguire il caricamento completo di PID_3Step.

Caricamento degli oggetti tecnologici del dispositivo [\(Pagina 48\)](#)

Se si utilizzano Output_UP e Output_DN ridurre la frequenza di commutazione con il tempo di attivazione e disattivazione minimo.

Nel funzionamento automatico i tempi di attivazione o disattivazione calcolati vengono sommati e diventano effettivi solo se la somma è maggiore o uguale al tempo di attivazione o disattivazione minimo.

Nel funzionamento manuale l'organo attuatore viene comandato tramite Manual_UP = TRUE o Manual_DN = TRUE almeno per il tempo di attivazione o disattivazione minimo.

Se nelle impostazioni di base è stato selezionato il valore di uscita analogicoOutput_PER, il tempo di inserzione e disinserzione minimo non viene analizzato e non è modificabile.

Comportamento in caso di errori

PID_3Step è preimpostato in modo tale per cui in caso di errore la regolazione resta attiva nella maggior parte dei casi. Se in funzionamento di regolazione si verificano errori frequenti, questa preimpostazione peggiora il comportamento del regolatore. In questo caso controllare il parametro Errorbits ed eliminare la causa dell'errore.

ATTENZIONE

L'impianto può subire danneggiamenti.

Se in caso di errore vengono emessi "Valore attuale per la durata dell'errore" o "Valore di uscita sostitutivo per la durata dell'errore", PID_3Step rimane in funzionamento automatico anche se vengono superati i limiti del valore istantaneo. In seguito a ciò l'impianto può subire danneggiamenti.

Configurare per il sistema regolato un comportamento da adottare in caso di errore che protegga l'impianto da danneggiamenti.

In caso di errori PID_3Step emette un valore di uscita configurabile:

- Valore attuale

PID_3Step è disattivato e non cambia più la posizione dell'organo attuatore.

- Valore attuale per la durata dell'errore

Le funzioni del regolatore PID_3Step sono disattivate e la posizione dell'organo attuatore non cambia più.

Se in funzionamento automatico si sono verificati i seguenti errori, PID_3Step ritorna a tale funzionamento non appena gli errori vengono eliminati.

- 0002h: Valore del parametro Input_PER non valido.
- 0200h: Valore del parametro Input non valido.
- 0400h: Calcolo del valore di uscita non riuscito.
- 1000h: Valore del parametro Setpoint non valido.
- 2000h: Valore del parametro Feedback_PER non valido.
- 4000h: Valore del parametro Feedback non valido.
- 8000h: Errore nella risposta di posizione digitale.
- 20000h: Valore della variabile SavePosition non valido.

Se si verificano uno o più dei seguenti errori PID_3Step resta in funzionamento automatico:

- 0001h: Il parametro Input non rientra nei limiti del valore istantaneo.
- 0800h: Errore del tempo di campionamento
- 40000h: Valore del parametro Disturbance non valido.

Se si verifica un errore nel funzionamento manuale, PID_3Step rimane in questo modo di funzionamento.

Se durante l'ottimizzazione o la misura del tempo di regolazione si verifica un errore, PID_3Step passa nel modo di funzionamento dal quale è stata avviata l'ottimizzazione o la misura del tempo di regolazione. Solo con i seguenti errori non viene arrestata l'ottimizzazione:

- 0020h: L'ottimizzazione iniziale non è consentita durante l'ottimizzazione fine.

- Valore di uscita sostitutivo
PID_3Step porta l'organo attuatore sul valore di uscita sostitutivo e si disattiva.
- Valore di uscita sostitutivo per la durata dell'errore
PID_3Step porta l'organo attuatore sul valore di uscita sostitutivo. Al raggiungimento del valore di uscita sostitutivo, PID_3Step si comporta come descritto al punto "Valore attuale per la durata dell'errore".

Il valore di uscita sostitutivo va inserito in "%".

In caso di utilizzo di Output_UP e Output_DN senza risposta di posizione analogica (Config.OutputPerOn = FALSE e Config.FeedbackOn = FALSE) possono essere configurati solo i valori di uscita sostitutivi 0 % e 100 %. Per permettere il raggiungimento del riscontro superiore o inferiore, l'organo attuatore si sposta nella direzione corrispondente. Se sono presenti segnali di riscontro (Config.ActuatorEndStopOn = TRUE), Output_UP e Output_DN vengono resettati con Actuator_H = TRUE o Actuator_L = TRUE. Se non sono presenti segnali di riscontro (Config.ActuatorEndStopOn = FALSE), Output_UP e Output_DN vengono resettati dopo un tempo di spostamento di Config.VirtualActuatorLimit * Retain.TransitTime/100.

Se si utilizza Output_PER o una risposta di posizione analogica (Config.OutputPerOn = TRUE o Config.FeedbackOn = TRUE) è possibile configurare qualsiasi valore di uscita sostitutivo compreso entro i limiti del valore di uscita.

Riporta in scala risposta di posizione V2

Riporta in scala risposta di posizione

Se nelle impostazioni di base è stato configurato l'utilizzo di Feedback_PER è necessario convertire il valore dell'ingresso analogico in %. Nel campo di visualizzazione "Feedback" viene visualizzata la configurazione attuale.

Feedback_PER viene riportato in scala sulla base di una coppia di valori inferiore e una superiore.

1. Inserire nei campi di immissione "Riscontro inferiore" e "Inferiore" la coppia di valori inferiore.
2. Inserire nei campi di immissione "Riscontro superiore" e "Superiore" la coppia di valori superiore.

Il "riscontro inferiore" deve essere minore del "riscontro superiore"; il valore per "inferiore" deve essere minore del valore di "superiore".

I valori validi per "Riscontro superiore" e "Riscontro inferiore" dipendono da:

- Nessun Feedback, Feedback, Feedback_PER
- Output (analogico), Output (digitale)

Output	Feedback	Riscontro inferiore	Riscontro superiore
Output (digitale)	Nessun Feedback	non impostabile (0.0 %)	non impostabile (100.0 %)
Output (digitale)	Feedback	-100.0 % o 0.0 %	0.0 % o +100.0 %
Output (digitale)	Feedback_PER	-100.0 % o 0.0 %	0.0 % o +100.0 %
Output (analogico)	Nessun Feedback	non impostabile (0.0 %)	non impostabile (100.0 %)
Output (analogico)	Feedback	-100.0 % o 0.0 %	0.0 % o +100.0 %
Output (analogico)	Feedback_PER	-100.0 % o 0.0 %	0.0 % o +100.0 %

Valori limite di uscita V2

Limitazione del valore di uscita

Durante la misura del tempo di regolazione e con modo = 10 i limiti dei valori di uscita possono essere superati in positivo o in negativo. Negli altri modi di funzionamento, il valore di uscita viene limitato su questi valori.

Nei campi di immissione "Valore di uscita limite superiore" e "Valore di uscita limite inferiore" inserire i limiti assoluti del valore di uscita. I limiti del valore di uscita devono essere compresi tra il "Riscontro inferiore" e il "Riscontro superiore".

Se manca un Feedback ed è impostato Output (digitale) non è possibile delimitare il valore di uscita. Output_UP e Output_DN vengono quindi resettati se Actuator_H = TRUE o Actuator_L = TRUE. Se non sono presenti neppure segnali di riscontro Output_UP e Output_DN vengono resettati dopo un tempo di spostamento pari a 150% del tempo di regolazione del motore.

Il valore di default 150% può essere adeguato con la variabile Config.VirtualActuatorLimit. Da PID_3Step versione 2.3 il controllo e la limitazione della durata corsa con Config.VirtualActuatorLimit = 0.0 possono essere disattivati.

6.2.1.4 Impostazioni avanzate V2

Controllo valore istantaneo V2

La finestra "Controllo del valore istantaneo" consente di configurare il limite di avviso inferiore e superiore del valore istantaneo. Se durante il funzionamento viene superato uno dei limiti, nell'istruzione "PID_3Step" viene visualizzato un avviso:

- Nel parametro di uscita InputWarning_H, se è stato superato il limite di avviso superiore
- Nel parametro di uscita InputWarning_L, se non è stato raggiunto il limite di avviso inferiore

I limiti di avviso devono essere compresi entro il limite superiore e inferiore del valore istantaneo.

Se non si inserisce alcun valore, vengono utilizzati il limite superiore e inferiore del valore istantaneo.

Esempio

Limite superiore del valore istantaneo = 98 °C ; limite di avviso superiore = 90 °C

Limite di avviso inferiore = 10 °C ; limite inferiore del valore istantaneo = 0 °C

PID_3Step si comporta come descritto di seguito:

Valore istantaneo	InputWarning_H	InputWarning_L	ErrorBits	Modo di funzionamento
> 98 °C	TRUE	FALSE	0001h	Come configurato
≤ 98 °C e > 90 °C	TRUE	FALSE	0000h	Funzionamento automatico

Valore istantaneo	InputWarning_H	InputWarning_L	ErrorBits	Modo di funzionamento
$\leq 90\text{ °C}$ e $\geq 10\text{ °C}$	FALSE	FALSE	0000h	Funzionamento automatico
$< 10\text{ °C}$ e $\geq 0\text{ °C}$	FALSE	TRUE	0000h	Funzionamento automatico
$< 0\text{ °C}$	FALSE	TRUE	0001h	Come configurato

Nelle impostazioni dell'organo attuatore si configura come reagisce PID_3Step nel caso in cui non vengano superati o non vengano raggiunti il limite superiore o il limite inferiore del valore istantaneo.

Parametri PID V2

Nella finestra di configurazione "Parametri PID" vengono visualizzati i parametri PID. Durante l'ottimizzazione i parametri PID vengono adattati al proprio sistema regolato. Non è necessario, quindi, inserirli manualmente.

NOTA

I parametri PID attualmente effettivi si trovano nella struttura Retain.CtrlParams.

Per evitare un comportamento errato del regolatore PID, modificare online i parametri PID attualmente effettivi solo nel modo di funzionamento "Inattivo".

Se si vogliono modificare i parametri PID online nei modi di funzionamento "Automatico" o "Manuale", modificarli nella struttura CtrlParamsBackUp e applicare le modifiche nella struttura Retain.CtrlParams nel modo seguente:

- PID_3Step V1: applicare le modifiche con Config.LoadBackUp = TRUE
- PID_3Step V2: applicare le modifiche con LoadBackUp = TRUE

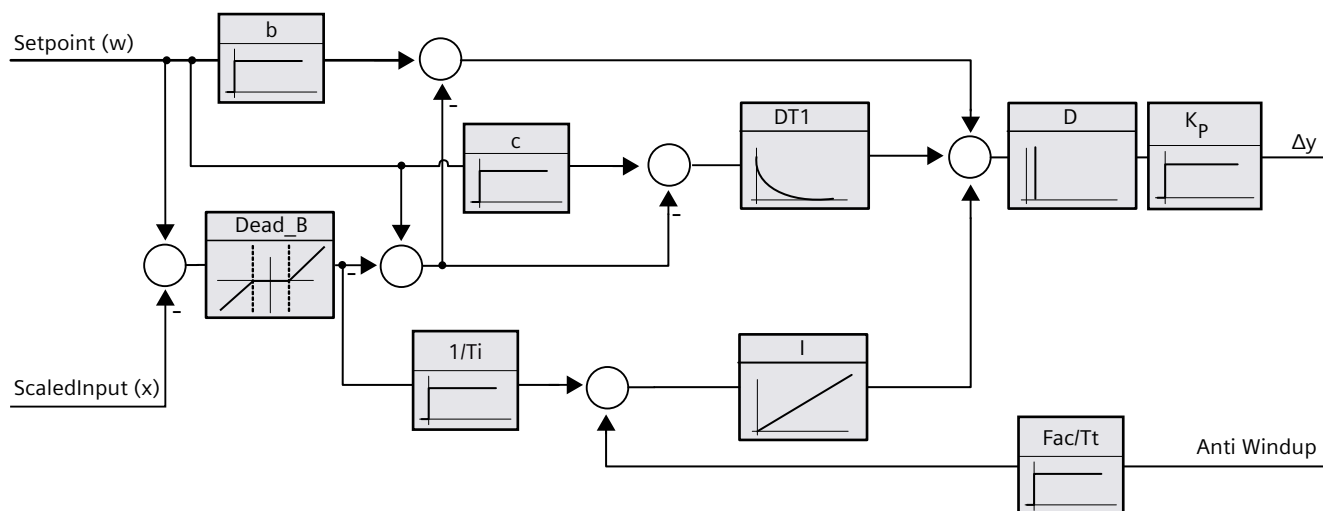
Le modifiche dei parametri PID eseguite online in modalità "Funzionamento automatico" possono causare gradini nel valore di uscita.

L'algoritmo PID utilizza la seguente formula:

$$\Delta y = K_p \cdot s \cdot \left[(b \cdot w - x) + \frac{1}{T_i \cdot s} (w - x) + \frac{T_d \cdot s}{a \cdot T_d \cdot s + 1} (c \cdot w - x) \right]$$

Δy	Valore di uscita dell'algoritmo PID
K_p	Guadagno proporzionale
s	Operatore di Laplace
b	Ponderazione del componente P
w	Setpoint
x	Valore istantaneo
T_i	Tempo di integrazione
a	Coefficiente per il ritardo derivativo (ritardo derivativo $T_1 = a \times T_D$)
T_D	Tempo derivativo
c	Ponderazione del componente D

Il seguente grafico mostra i parametri nell'algoritmo PID:



Tutti i parametri PID sono a ritenzione. Per l'inserimento manuale dei parametri PID, eseguire il caricamento completo di PID_3Step.

Caricamento degli oggetti tecnologici del dispositivo [\(Pagina 48\)](#)

Guadagno proporzionale

Questo valore indica il guadagno proporzionale del regolatore. PID_3Step non opera con un guadagno proporzionale negativo. Il senso di regolazione si inverte in Impostazioni di base > Modo di regolazione.

Tempo di integrazione

Il tempo di integrazione determina il comportamento temporale del componente I. La disattivazione del componente I avviene impostando il tempo di integrazione = 0,0. Se il tempo di integrazione viene modificato online da un altro valore a 0.0 in modalità "Funzionamento automatico", il componente I precedente viene cancellato e si verifica un gradino del valore di uscita.

Tempo derivativo

Il tempo derivativo determina il comportamento temporale del componente D. La disattivazione del componente D avviene impostando il tempo derivativo = 0,0.

Coefficiente ritardo derivativo

L'effetto del componente D viene ritardato dal coefficiente per il ritardo derivativo.

Ritardo derivativo = Tempo derivativo × Coefficiente ritardo derivativo

- 0.0: Il componente D ha effetto solo per un ciclo e quindi è quasi inattivo.
- 0.5: Questo valore ha dato buoni risultati nelle operazioni con i sistemi regolati con **una** costante temporale dominante.
- > 1.0: Maggiore è il coefficiente, maggiore sarà il ritardo dell'azione del componente D.

Ponderazione del componente P

In caso di variazioni del setpoint è possibile ridurre il componente P.

Sono adeguati valori da 0.0 a 1.0.

- 1.0: componente P pienamente attivo in caso di variazione del setpoint
- 0.0: componente P non attivo in caso di variazione del setpoint

In caso di variazioni del valore istantaneo il componente P è sempre pienamente attivo.

Ponderazione del componente D

In caso di variazioni del setpoint è possibile ridurre il componente D.

Sono adeguati valori da 0.0 a 1.0.

- 1.0: componente D pienamente attivo in caso di variazione del setpoint
- 0.0: componente D non attivo in caso di variazione del setpoint

In caso di variazioni del valore istantaneo il componente D è sempre pienamente attivo.

Tempo di campionamento algoritmo PID

Poiché il sistema regolato richiede un certo tempo per reagire a una modifica del valore di uscita, è utile non calcolare questo valore in ogni ciclo. Il tempo di campionamento dell'algoritmo PID è il tempo che trascorre tra due calcoli del valore di uscita. Viene determinato durante l'ottimizzazione e arrotondato a un multiplo del tempo di campionamento di PID_3Step. Tutte le altre funzioni di PID_3Step vengono eseguite ad ogni richiamo.

Ampiezza banda morta

La banda morta sopprime la componente di disturbo nello stato assestato del regolatore. L'ampiezza di banda morta indica la grandezza della banda morta. Con un'ampiezza di banda morta di 0,0, essa è disattivata.

Se sono stati configurati valori diversi da 1.0 per la ponderazione del componente P o per quella del componente D le modifiche dei setpoint influenzano il valore di uscita anche all'interno della banda morta.

Le modifiche del valore istantaneo all'interno della banda morta non influenzano il valore di uscita indipendentemente dalla ponderazione.

6.2.2 Messa in servizio di PID_3Step V2

6.2.2.1 Ottimizzazione iniziale V2

L'ottimizzazione iniziale rileva la risposta del processo a un impulso del valore di uscita e cerca il punto di flesso. Dall'incremento massimo e dal tempo morto del sistema regolato vengono calcolati i parametri PID ottimali. I parametri PID migliori si ottengono eseguendo l'ottimizzazione iniziale e l'ottimizzazione fine.

Più stabile è il valore istantaneo, più facilmente e con maggiore precisione si possono determinare i parametri PID. Un disturbo del valore istantaneo è accettabile fintanto che l'incremento dello stesso è significativamente maggiore rispetto al disturbo. Ciò si verifica con la maggiore probabilità nei modi "Inattivo" o "Funzionamento manuale". I parametri PID vengono salvati prima di essere nuovamente calcolati.

Il setpoint è congelato durante l'ottimizzazione iniziale.

Presupposto

- L'istruzione PID_3Step viene richiamata in un OB di schedulazione orologio.
- ManualEnable = FALSE
- Reset = FALSE
- Il tempo di regolazione del motore è stato configurato e misurato.
- PID_3Step si trova in modo "Inattivo", in "Funzionamento manuale" o in "Funzionamento automatico".
- Il setpoint e il valore istantaneo sono compresi entro i limiti configurati (vedere la configurazione di "Impostazione del valore istantaneo").

Procedimento

Per eseguire l'ottimizzazione iniziale procedere nel seguente modo:

1. Fare doppio clic nella navigazione del progetto sulla voce "PID_3Step > Messa in servizio".
2. Selezionare la voce "Ottimizzazione iniziale" nella casella di riepilogo "Tipo di ottimizzazione" del campo di lavoro "Ottimizzazione".
3. Fare clic sul simbolo "Start".
 - Viene creato un collegamento online.
 - Viene avviata la registrazione dei valori.
 - Viene avviata l'ottimizzazione iniziale.
 - Il campo "Stato" indica le operazioni in corso e gli eventuali errori. La barra di avanzamento indica lo stato di avanzamento dell'operazione in corso.

NOTA

Se la barra di avanzamento ha raggiunto il 100% e si deve presupporre un blocco dell'ottimizzazione, fare clic sul simbolo "Stop". Verificare la configurazione dell'oggetto tecnologico ed eventualmente riavviare l'ottimizzazione.

Risultato

Se l'ottimizzazione iniziale è stata eseguita senza messaggi di errore significa che i parametri PID sono stati ottimizzati. PID_3Step commuta sul funzionamento automatico e utilizza i parametri ottimizzati. Questi ultimi vengono mantenuti in caso di alimentazione OFF e nuovo avviamento della CPU.

Se un'ottimizzazione iniziale non è possibile, PID_3Step reagisce come configurato nel comportamento in caso di errore.

6.2.2.2 Ottimizzazione fine V2

L'ottimizzazione fine genera un'oscillazione costante e limitata del valore istantaneo. I parametri PID per il punto di lavoro vengono ottimizzati partendo dall'ampiezza e dalla frequenza dell'oscillazione. A partire dai risultati vengono ricalcolati tutti i parametri PID. I parametri PID risultanti dall'ottimizzazione fine mostrano in genere un comportamento pilota e di disturbo migliore rispetto ai parametri PID dell'ottimizzazione iniziale. I parametri PID migliori si ottengono eseguendo l'ottimizzazione iniziale e l'ottimizzazione fine.

PID_3Step cerca di generare automaticamente un'oscillazione che è maggiore rispetto al disturbo del valore istantaneo. L'ottimizzazione fine viene influenzata solo minimamente dalla stabilità del valore istantaneo. I parametri PID vengono salvati prima di essere nuovamente calcolati.

Il setpoint è congelato durante l'ottimizzazione fine.

Presupposti

- L'istruzione PID_3Step viene richiamata in un OB di schedulazione orologio.
- ManualEnable = FALSE
- Reset = FALSE
- Il tempo di regolazione del motore è stato configurato e misurato.
- Il setpoint e il valore istantaneo sono compresi entro i limiti configurati (vedere la configurazione di "Impostazione del valore istantaneo").
- Il circuito di regolazione si è assestato nel punto di lavoro. Il punto di lavoro è raggiunto quando il valore istantaneo corrisponde al setpoint.
- Non sono previste anomalie.
- PID_3Step si trova nel modo di funzionamento inattivo, automatico o manuale.

Procedimento a seconda della situazione di partenza

All'avvio, l'ottimizzazione fine si svolge nel modo seguente:

- Funzionamento automatico

Per migliorare i parametri PID esistenti mediante l'ottimizzazione, avviare l'ottimizzazione fine dal modo di funzionamento automatico.

PID_3Step esegue la regolazione con i parametri PID esistenti finché il circuito di regolazione è assestato e sono soddisfatti i requisiti per l'ottimizzazione fine. Solo in seguito viene avviata l'ottimizzazione fine.

- Inattivo o Funzionamento manuale

Viene sempre avviata per prima un'ottimizzazione iniziale. Viene quindi effettuata la regolazione con i parametri PID rilevati finché il circuito di regolazione è assestato e sono soddisfatti i requisiti per l'ottimizzazione fine. Solo in seguito viene avviata l'ottimizzazione fine.

Procedimento

Per eseguire l'ottimizzazione fine procedere nel seguente modo:

1. Selezionare la voce "ottimizzazione fine" nella casella di riepilogo "Tipo di ottimizzazione".
2. Fare clic sul simbolo "Start".
 - Viene creato un collegamento online.
 - Viene avviata la registrazione dei valori.
 - Viene avviata l'ottimizzazione fine.
 - Il campo "Stato" indica le operazioni in corso e gli eventuali errori. La barra di avanzamento indica lo stato di avanzamento dell'operazione in corso.

NOTA

Se la barra di avanzamento ha raggiunto il 100% e si deve presumere un blocco dell'ottimizzazione, fare clic sul simbolo "Stop" nel gruppo "Tipo di ottimizzazione". Verificare la configurazione dell'oggetto tecnologico ed eventualmente riavviare l'ottimizzazione.

Risultato

Se durante l'ottimizzazione fine non si verificano errori, i parametri PID sono stati ottimizzati. PID_3Step commuta sul funzionamento automatico e utilizza i parametri ottimizzati. Questi ultimi vengono mantenuti in caso di alimentazione OFF e nuovo avviamento della CPU.

Se durante l'ottimizzazione fine si sono verificati errori, PID_3Step reagisce come configurato nel comportamento in caso di errore.

6.2.2.3 Messa in servizio con parametri PID manuali V2

Presupposto

- L'istruzione PID_3Step viene richiamata in un OB di schedulazione orologio.
- ManualEnable = FALSE
- Reset = FALSE
- Il tempo di regolazione del motore è stato configurato e misurato.
- PID_3Step si trova nel modo di funzionamento "Inattivo".
- Il setpoint e il valore istantaneo sono compresi entro i limiti configurati (vedere la configurazione di "Impostazione del valore istantaneo").

Procedimento

Per mettere in funzione PID_3Step con parametri PID manuali, procedere nel modo seguente:

1. Fare doppio clic nella navigazione del progetto sulla voce "PID_3Step > Configurazione".
2. Nella finestra di configurazione fare clic su "Impostazioni avanzate > Parametri PID".
3. Attivare la casella di opzione "Attiva immissione manuale".
4. Immettere i parametri PID.
5. Fare doppio clic nella navigazione del progetto sulla voce "PID_3Step > Messa in servizio".
6. Creare un collegamento online alla CPU.
7. Caricare i parametri PID nella CPU.
8. Fare clic sul simbolo "Start PID_3Step".

Risultato

PID_3Step commuta nel funzionamento automatico ed esegue la regolazione con i parametri PID attuali.

Vedere anche

[Parametri PID V2 \(Pagina 135\)](#)

6.2.2.4 Misura del tempo di regolazione motore V2

Introduzione

Per ottenere un buon risultato di regolazione, PID_3Step richiede un tempo di regolazione del motore il più preciso possibile. I dati contenuti nella documentazione dell'organo attuatore sono valori medi per questo tipo di organi attuatori. Il valore può variare per l'organo attuatore effettivamente utilizzato.

Se si utilizzano organi attuatori con risposta di posizione o con segnali di riscontro, durante la messa in servizio è possibile misurare il tempo di regolazione del motore. Durante la misura del tempo di regolazione del motore, i limiti del valore di uscita non vengono considerati. L'organo attuatore può essere traslato fino al riscontro superiore o fino a quello inferiore.

Se non sono disponibili né una risposta di posizione né segnali di riscontro il tempo di regolazione del motore non può essere misurato.

Organi attuatori con risposta della posizione analogica

Per misurare il tempo di regolazione del motore con risposta di posizione procedere nel seguente modo:

Requisiti

- Nelle impostazioni di base deve essere selezionato Feedback o Feedback_PER e il segnale deve essere connesso.
- Deve esistere un collegamento online alla CPU.

1. Attivare la casella di scelta "Utilizza risposta di posizione".
2. Nel campo di immissione "Destinazione" inserire la destinazione del movimento dell'organo attuatore.

La risposta di posizione attuale (posizione di partenza) viene visualizzata. La differenza tra "Destinazione" e "Risposta di posizione" deve essere almeno pari al 50 % del campo del valore di uscita consentito.

3. Fare clic sul simbolo "Start".

Risultato

L'organo attuatore viene portato dalla posizione di partenza alla destinazione. La misura del tempo si avvia immediatamente e termina quando l'organo attuatore ha raggiunto la destinazione. Il tempo di regolazione del motore si calcola sulla base della formula:

Tempo di regolazione motore = (Valore di uscita limite superiore – Valore di uscita limite inferiore) × Tempo di misura / IMPORTO (Destinazione – Posizione di partenza).

L'avanzamento e lo stato della misura del tempo di regolazione vengono visualizzati. Il tempo di regolazione misurato viene memorizzato nel blocco dati di istanza della CPU e visualizzato nel campo "Tempo di regolazione misurato". Quando la misura del tempo di regolazione è terminata e ActivateRecoverMode = TRUE, PID_3Step passa al modo di funzionamento dal quale è stata avviata la misura del tempo di regolazione. Se la misura del tempo di regolazione è terminata e ActivateRecoverMode = FALSE, PID_3Step passa al modo di funzionamento "Inattivo".

NOTA

Per applicare il tempo di regolazione del motore misurato nel progetto, fare clic sul simbolo  "Carica tempo di regolazione misurato".

Organi attuatori con segnali di riscontro

Per misurare il tempo di regolazione degli organi attuatori con segnali di riscontro, procedere nel seguente modo:

Requisiti

- Nelle impostazioni di base deve essere attivata la casella di opzione "Segnali di riscontro" e Actuator_H e Actuator_L devono essere connessi.
- Deve esistere un collegamento online alla CPU.

Per misurare il tempo di regolazione del motore con segnali di riscontro, procedere nel seguente modo:

1. Attivare la casella di scelta "Utilizza i segnali di riscontro dell'organo attuatore".
2. Selezionare la direzione in cui spostare l'organo attuatore.
 - Apri - Chiudi - Apri
L'organo attuatore viene dapprima portato verso il riscontro superiore, quindi verso quello inferiore e di nuovo verso quello superiore.
 - Chiudi - Apri - Chiudi
L'organo attuatore viene dapprima portato verso il riscontro inferiore, quindi verso quello superiore e di nuovo verso quello inferiore.
3. Fare clic sul simbolo "Start".

Risultato

L'organo attuatore viene spostato nella direzione selezionata. La misura del tempo si avvia quando l'organo attuatore ha raggiunto il primo riscontro e termina quando lo raggiunge per la seconda volta. Dal tempo misurato diviso per due risulta il tempo di regolazione del motore.

L'avanzamento e lo stato della misura del tempo di regolazione vengono visualizzati. Il tempo di regolazione misurato viene memorizzato nel blocco dati di istanza della CPU e visualizzato nel campo "Tempo di regolazione misurato". Quando la misura del tempo di regolazione è terminata e ActivateRecoverMode = TRUE, PID_3Step passa al modo di funzionamento dal quale è stata avviata la misura del tempo di regolazione. Se la misura del tempo di regolazione è terminata e ActivateRecoverMode = FALSE, PID_3Step passa al modo di funzionamento "Inattivo".

Annullamento della misura del tempo di regolazione

Se la misura del tempo di regolazione viene annullata con il pulsante Stop, PID_3Step passa immediatamente al modo di funzionamento "Inattivo".

6.2.3 Simulazione di PID_3Step V2 con PLCSIM

NOTA

Simulazione con PLCSIM

La simulazione di PID_3Step V2.x con PLCSIM per la CPU S7-1200 non viene supportata.

PID_3Step V2.x può essere simulata con PLCSIM solo per la CPU S7-1500 e la CPU S7-1200 G2.

Nella simulazione con PLCSIM il comportamento temporale del PLC simulato non è esattamente identico a quello di un PLC "reale". In un PLC simulato, il clock di tempo effettivo di un OB di schedulazione orologio può presentare oscillazioni maggiori rispetto ai PLC "reali".

Nella configurazione standard, PID_3Step rileva automaticamente gli intervalli tra i richiami e ne controlla le oscillazioni.

Nella simulazione di PID_3Step con PLCSIM può essere pertanto identificato un errore del tempo di campionamento (ErrorBits = DW#16#00000800).

Ciò comporta l'annullamento di ottimizzazioni in corso.

La reazione nel funzionamento automatico dipende dal valore della variabile ActivateRecoverMode.

Per evitare questo comportamento, si consiglia, nella simulazione con il PLCSIM, di configurare PID_3Step come indicato nel seguito:

- CycleTime.EnEstimation = FALSE
 - CycleTime.EnMonitoring = FALSE
 - CycleTime.Value: Assegnare a questa variabile il clock di tempo dell'OB di schedulazione orologio richiamaente espresso in secondi.
-

6.3 PID_3Step V1

6.3.1 Configurazione di PID_3Step V1

6.3.1.1 Impostazione di base V1

Introduzione V1

In "Impostazioni di base", nella finestra di ispezione o di configurazione, si possono configurare le seguenti proprietà dell'oggetto tecnologico PID_3Step:

- Grandezza fisica
- Senso di regolazione
- Comportamento all'avviamento dopo il reset
- Setpoint (solo nella finestra di ispezione)
- Valore istantaneo (solo nella finestra di ispezione)
- Valore di uscita (solo nella finestra di ispezione)
- Risposta di posizione (solo nella finestra di ispezione)

Setpoint, valore istantaneo, valore di uscita e risposta di posizione

Setpoint, valore istantaneo, valore di uscita e risposta di posizione si possono configurare solo nella finestra di ispezione dell'editor di programma. Per ogni valore selezionare la sorgente:

- DB di istanza

Viene utilizzato il valore salvato nel DB di istanza.

Il valore deve essere aggiornato dal programma utente nel DB di istanza.

Nell'istruzione non può essere inserito alcun valore.

È possibile effettuare le modifiche tramite HMI.

- Istruzione

Viene utilizzato il valore interconnesso con l'istruzione.

A ogni richiamo dell'istruzione il valore viene scritto nel DB di istanza.

Non è possibile effettuare le modifiche tramite HMI.

Tipo di regolazione V1

Grandezza fisica

Nel gruppo "Modo di regolazione" selezionare la grandezza fisica e l'unità per il setpoint e il valore istantaneo. I valori verranno quindi visualizzati in questa unità.

Senso di regolazione

Aumentando il valore di uscita si dovrebbe ottenere un aumento del valore istantaneo. In questo caso si parla di un senso di regolazione normale.

PID_3Step non funziona con un guadagno proporzionale negativo. Per ridurre il valore istantaneo aumentando il valore di uscita, attivare la casella di opzione "Inversione del senso di regolazione".

Esempi

- Aprendo una valvola di scarico il livello di riempimento di un contenitore diminuisce.
- Aumentando la capacità di raffreddamento la temperatura diminuisce.

Comportamento all'avviamento dopo il reset

Per passare immediatamente, dopo il riavvio della CPU, all'ultimo modo di funzionamento attivo, attivare la casella di opzione "Avvia ultimo modo dopo il riavvio della CPU".

Se la casella è disattivata, PID_3Step rimane nel modo di funzionamento "Inattivo".

Setpoint V1

Procedimento

Per predefinire un setpoint fisso procedere nel modo seguente:

1. Selezionare "DB di istanza".
2. Inserire un setpoint, ad es. 80 °C.
3. Se necessario, cancellare una voce dell'istruzione.

Per predefinire un setpoint variabile procedere nel modo seguente:

1. Selezionare "Istruzione".
2. Inserire il nome della variabile REAL in cui è memorizzato il setpoint.

Alla variabile REAL è possibile assegnare diversi valori tramite comandi del programma, ad es. per modificare il setpoint con comandi a tempo.

Valore istantaneo V1

Se si utilizza direttamente il valore dell'ingresso analogico, PID_3Step lo riporta in scala nella grandezza fisica.

Se si desidera prima trattare il valore dell'ingresso analogico è necessario scrivere un proprio programma per il trattamento. Supponiamo che il valore istantaneo non sia direttamente proporzionale al valore nell'ingresso analogico. Il valore istantaneo trattato dovrà essere nel formato in virgola mobile.

Procedimento

Per utilizzare direttamente il valore dell'ingresso analogico procedere nel modo seguente:

1. Selezionare la voce "Input_PER" nella casella di riepilogo "Input".
2. Selezionare come sorgente "Istruzione".
3. Inserire l'indirizzo dell'ingresso analogico.

Per utilizzare il valore istantaneo trattato nel formato in virgola mobile procedere nel modo seguente:

1. Selezionare la voce "Input" nella casella di riepilogo "Input".
2. Selezionare come sorgente "Istruzione".
3. Inserire il nome della variabile in cui è memorizzato il valore istantaneo trattato.

Risposta di posizione V1

La configurazione della risposta di posizione dipende dall'organo attuatore utilizzato.

- Organo attuatore senza risposta di posizione
- Organo attuatore con segnali di riscontro digitali
- Organo attuatore con risposta di posizione analogica
- Organo attuatore con risposta di posizione analogica e segnali di riscontro

Organo attuatore senza risposta di posizione

Per configurare PID_3Step per un organo attuatore senza risposta di posizione procedere nel seguente modo:

1. Selezionare la voce "No Feedback" nella casella di riepilogo "Feedback".

Organo attuatore con segnali di riscontro digitali

Per configurare PID_3Step per un organo attuatore con segnali di riscontro procedere nel seguente modo:

1. Selezionare la voce "No Feedback" nella casella di riepilogo "Feedback".
2. Attivare la casella di opzione "Segnali di riscontro organo attuatore".
3. Selezionare la voce "Istruzione" come sorgente per Actuator_H e Actuator_L.
4. Inserire gli indirizzi degli ingressi digitali per Actuator_H e Actuator_L.

Organo attuatore con risposta di posizione analogica

Per configurare PID_3Step per un organo attuatore con risposta di posizione analogica procedere nel seguente modo:

1. Selezionare la voce "Feedback" o "Feedback_PER" nella casella di riepilogo "Feedback".
 - Per Feedback_PER utilizzare direttamente il valore dell'ingresso analogico. La scala di Feedback_PER viene configurata nelle impostazioni dell'organo attuatore.
 - Per Feedback trattare il valore dell'ingresso analogico con il programma utente.
2. Selezionare come sorgente "Istruzione".
3. Inserire l'indirizzo dell'ingresso analogico o la variabile del programma utente.

Organo attuatore con risposta di posizione analogica e segnali di riscontro

Per configurare PID_3Step per un organo attuatore con risposta di posizione analogica e segnali di riscontro procedere nel seguente modo:

1. Selezionare la voce "Feedback" o "Feedback_PER" nella casella di riepilogo "Feedback".
2. Selezionare come sorgente "Istruzione".
3. Inserire l'indirizzo dell'ingresso analogico o la variabile del programma utente.
4. Attivare la casella di opzione "Segnali di riscontro organo attuatore".
5. Selezionare la voce "Istruzione" come sorgente per Actuator_H e Actuator_L.
6. Inserire gli indirizzi degli ingressi digitali per Actuator_H e Actuator_L.

Valore di uscita V1

PID_3Step fornisce un valore di uscita analogico (Output_PER) e valori di uscita digitali (Output_UP, Output_DN). L'utilizzo di un determinato valore di uscita dipende dall'organo attuatore.

- Output_PER

L'attuatore, che dispone di un tempo di regolazione motore considerevole, è accessibile da un'uscita analogica e viene comandato da un segnale continuo, ad es. 0...10 V, 4...20 mA. Il valore di Output_PER corrisponde alla destinazione della valvola ad es. Output_PER = 13824, se l'apertura della valvola deve essere del 50 %.

PID_3Step considera che, per quanto concerne ad es. l'ottimizzazione automatica e il comportamento anti-windup, a causa del tempo di regolazione motore, il valore di uscita analogico agisce con ritardo sul processo. Se nel processo non è attivo un tempo di regolazione del motore rilevante (ad es. con le valvole magnetiche) in modo da consentire al valore di uscita di agire direttamente e pienamente nel processo, utilizzare in suo luogo PID_Compact.

- Output_UP, Output_DN

L'attuatore dispone di un tempo di regolazione motore considerevole e viene comandato da due uscite digitali.

Output_UP sposta la valvola verso lo stato di apertura.

Output_DN sposta la valvola verso lo stato di chiusura.

Il tempo di regolazione del motore viene considerato sia nel calcolo del valore di uscita analogico, sia nel calcolo dei valori di uscita digitali. Questo tempo è necessario soprattutto nell'ottimizzazione automatica e nel comportamento anti-windup per un corretto funzionamento. Configurare pertanto il tempo di regolazione, alla voce "Impostazioni dell'organo attuatore", con un valore necessario al motore per spostare l'attuatore dallo stato chiuso a quello aperto.

Procedimento

Per utilizzare il valore di uscita analogico procedere nel modo seguente:

1. Selezionare la voce "Output (analog)" nella casella di riepilogo "Output".
2. Selezionare "Istruzione".
3. Inserire l'indirizzo dell'uscita analogica.

Per utilizzare il valore di uscita digitale procedere nel modo seguente:

1. Selezionare la voce "Output (digital)" nella casella di riepilogo "Output".
2. Selezionare la voce "Istruzione" per Output_UP e Output_DN.
3. Inserire gli indirizzi delle uscite digitali.

Per trattare il valore di uscita con il programma utente procedere nel modo seguente:

1. Selezionare la voce adatta all'organo attuatore nella casella di riepilogo "Output".
2. Selezionare "Istruzione".
3. Indicare il nome della variabile utilizzata per il trattamento del valore di uscita.
4. Trasferire il valore di uscita trattato all'organo attuatore mediante un'uscita analogica o digitale della CPU.

6.3.1.2 Impostazioni del valore istantaneo V1

La finestra di configurazione "Impostazioni del valore istantaneo" consente di impostare i limiti assoluti del valore istantaneo per configurarne la normalizzazione.

Scala del valore istantaneo

Se nelle impostazioni di base è stato configurato l'utilizzo di Input_PER è necessario convertire il valore dell'ingresso analogico nella grandezza fisica del valore istantaneo. Nel campo di visualizzazione Input_PER viene visualizzata la configurazione attuale.

Se il valore istantaneo è direttamente proporzionale al valore dell'ingresso analogico, Input_PER viene riportato in scala in base ad una coppia di valori inferiore e una superiore.

1. Inserire nei campi di immissione "Valore istantaneo inferiore riportato in scala" e "Inferiore" la coppia di valori inferiore.
2. Inserire nei campi di immissione "Valore istantaneo superiore riportato in scala" e "Superiore" la coppia di valori superiore.

Nella configurazione HW sono memorizzate le preimpostazioni per le coppie di valori. Per utilizzare le coppie di valori nella configurazione HW procedere nel modo seguente:

1. Selezionare l'istruzione PID_3Step nell'editor di programma.
2. Nelle impostazioni di base collegare Input_PER con un ingresso analogico.
3. Fare clic nelle impostazioni del valore istantaneo sul pulsante "Impostazione automatica".
I valori esistenti vengono sovrascritti dai valori nella configurazione HW.

Controllo del valore istantaneo

Definire il limite superiore e inferiore assoluto del valore istantaneo. Come valori limite occorre indicare valori adeguati al sistema regolato. È importante scegliere i valori opportuni durante l'ottimizzazione per ottenere parametri PID ottimali. Il "Limite superiore valore istantaneo" preimpostato è il 120%. Nell'ingresso di periferia il valore istantaneo può superare al massimo del 18% il campo normale (campo di sovracomando). Con questa impostazione non vengono più segnalati errori di superamento del "Limite superiore valore istantaneo". Vengono riconosciuti solo la rottura conduttore e il cortocircuito e PID_3Step si comporta come in caso di errore.

ATTENZIONE

L'impianto può subire danneggiamenti.

Se si impostano valori molto elevati come limiti del valore istantaneo (ad es. $-3,4 \cdot 10^{38} \dots +3,4 \cdot 10^{38}$), il controllo del valore istantaneo viene disattivato. Così, a causa di un errore si possono verificare danni all'impianto. Configurare limiti del valore istantaneo adeguati al sistema regolato.

6.3.1.3 Impostazioni attuatore V1

Tempi specifici dell'organo attuatore

Per proteggere l'organo attuatore da eventuali danni occorre configurare il tempo di regolazione motore e i tempi minimi di attivazione e disattivazione. I dati sono disponibili nei dati tecnici dell'organo attuatore.

Il tempo di regolazione del motore è il tempo in secondi necessario al motore per spostare l'organo attuatore dallo stato chiuso a quello aperto. L'organo attuatore viene mosso in una direzione, al massimo per il 110% del tempo di regolazione del motore. Il tempo di regolazione del motore può essere misurato durante la messa in servizio.

Il tempo di regolazione del motore viene considerato sia nel calcolo del valore di uscita analogico, sia nel calcolo dei valori di uscita digitali. Questo tempo è necessario soprattutto nell'ottimizzazione automatica e nel comportamento anti-windup per un corretto funzionamento.

Se nel processo non è attivo un tempo di regolazione del motore rilevante (ad es. con le valvole magnetiche) in modo da consentire al valore di uscita di agire direttamente e pienamente nel processo, utilizzare in suo luogo PID_Compact.

Se si utilizzano Output_UP e Output_DN ridurre la frequenza di commutazione con il tempo di attivazione e disattivazione minimo.

Nel funzionamento automatico i tempi di attivazione o disattivazione calcolati vengono sommati e diventano effettivi solo se la somma è maggiore o uguale al tempo di attivazione o disattivazione minimo.

Nel funzionamento manuale un fronte di salita in Manual_UP o Manual_DN comanda l'organo attuatore almeno per il tempo di attivazione o disattivazione minimo.

Se nelle impostazioni di base è stato selezionato il valore di uscita analogico Output_PER, il tempo di inserzione e disinserzione minimo non viene analizzato e non è modificabile.

Comportamento in caso di errori

PID_3Step è preimpostato in modo tale per cui in caso di errore la regolazione resta attiva nella maggior parte dei casi. Se in funzionamento di regolazione si verificano errori frequenti, questa preimpostazione peggiora il comportamento del regolatore. In questo caso controllare il parametro Errorbits ed eliminare la causa dell'errore.

In caso di errori PID_3Step emette un valore di uscita configurabile:

- Valore attuale

PID_3Step è disattivato e non cambia più la posizione dell'organo attuatore.

- Valore attuale per la durata dell'errore

Le funzioni del regolatore PID_3Step sono disattivate e la posizione dell'organo attuatore non cambia più.

Se in funzionamento automatico si sono verificati i seguenti errori, PID_3Step ritorna a tale funzionamento non appena gli errori vengono eliminati.

- 0002h: Valore del parametro Input_PER non valido.
- 0200h: Valore del parametro Input non valido.
- 0800h: Errore del tempo di campionamento
- 1000h: Valore del parametro Setpoint non valido.
- 2000h: Valore del parametro Feedback_PER non valido.
- 4000h: Valore del parametro Feedback non valido.
- 8000h: Errore nella risposta di posizione digitale.

Se si verifica uno di questi errori in funzionamento manuale, PID_3Step rimane in questo modo di funzionamento.

Se si verifica un errore durante l'ottimizzazione o la misura del tempo di regolazione, PID_3Step viene disattivato.

- Valore di uscita sostitutivo

PID_3Step porta l'organo attuatore sul valore di uscita sostitutivo e si disattiva.

- Valore di uscita sostitutivo per la durata dell'errore

PID_3Step porta l'organo attuatore sul valore di uscita sostitutivo. Al raggiungimento del valore di uscita sostitutivo, PID_3Step si comporta come descritto al punto "Valore attuale per la durata dell'errore".

Il valore di uscita sostitutivo va inserito in "%".

In caso di utilizzo di Output_UP e Output_DN senza risposta di posizione analogica (Config.OutputPerOn = FALSE e Config.FeedbackOn = FALSE) possono essere configurati solo i valori di uscita sostitutivi 0 % e 100 %. Per permettere il raggiungimento del riscontro superiore o inferiore, l'organo attuatore si sposta in una direzione con il 110% del tempo di regolazione motore. I segnali di riscontro vengono considerati in base alla priorità.

Se si utilizza Output_PER o una risposta di posizione analogica (Config.OutputPerOn = TRUE o Config.FeedbackOn = TRUE) è possibile configurare qualsiasi valore di uscita sostitutivo compreso entro i limiti del valore di uscita.

Riporta in scala risposta di posizione

Se nelle impostazioni di base è stato configurato l'utilizzo di Feedback_PER è necessario convertire il valore dell'ingresso analogico in %. Nel campo di visualizzazione "Feedback" viene visualizzata la configurazione attuale.

Feedback_PER viene riportato in scala sulla base di una coppia di valori inferiore e una superiore.

1. Inserire nei campi di immissione "Riscontro inferiore" e "Inferiore" la coppia di valori inferiore.
2. Inserire nei campi di immissione "Riscontro superiore" e "Superiore" la coppia di valori superiore.

Il "riscontro inferiore" deve essere minore del "riscontro superiore"; il valore per "inferiore" deve essere minore del valore di "superiore".

I valori validi per "Riscontro superiore" e "Riscontro inferiore" dipendono da:

- Nessun Feedback, Feedback, Feedback_PER
- Output (analogico), Output (digitale)

Output	Feedback	Riscontro inferiore	Riscontro superiore
Output (digitale)	Nessun Feedback	non impostabile (0.0 %)	non impostabile (100.0 %)
Output (digitale)	Feedback	-100.0 % oppure 0.0 %	0.0 % oppure +100.0 %
Output (digitale)	Feedback_PER	-100.0 % oppure 0.0 %	0.0 % oppure +100.0 %
Output (analogico)	Nessun Feedback	non impostabile (0.0 %)	non impostabile (100.0 %)
Output (analogico)	Feedback	-100.0 % oppure 0.0 %	0.0 % oppure +100.0 %
Output (analogico)	Feedback_PER	-100.0 % oppure 0.0 %	0.0 % oppure +100.0 %

Limitazione del valore di uscita

Soltanto durante la misura del tempo di regolazione i limiti del valore di uscita possono essere superati o non venire raggiunti. Negli altri modi di funzionamento, il valore di uscita viene limitato su questi valori.

Nei campi di immissione "Valore di uscita limite superiore" e "Valore di uscita limite inferiore" inserire i limiti assoluti del valore di uscita. I limiti del valore di uscita devono essere compresi tra il "Riscontro inferiore" e il "Riscontro superiore".

Se manca un Feedback ed è impostato Output (digitale) non è possibile delimitare il valore di uscita. Le uscite digitali vengono resettate se Actuator_H = TRUE, se Actuator_L = TRUE oppure dopo un tempo di spostamento del 110% del tempo di regolazione del motore.

6.3.1.4 Impostazioni avanzate V1

Controllo valore istantaneo V1

La finestra "Controllo del valore istantaneo" consente di configurare il limite di avviso inferiore e superiore del valore istantaneo. Se durante il funzionamento viene superato uno dei limiti, nell'istruzione "PID_3Step" viene visualizzato un avviso:

- Nel parametro di uscita InputWarning_H, se è stato superato il limite di avviso superiore
- Nel parametro di uscita InputWarning_L, se non è stato raggiunto il limite di avviso inferiore

I limiti di avviso devono essere compresi entro il limite superiore e inferiore del valore istantaneo.

Se non si inserisce alcun valore, vengono utilizzati il limite superiore e inferiore del valore istantaneo.

Esempio

Limite superiore del valore istantaneo = 98 °C ; limite di avviso superiore = 90 °C

Limite di avviso inferiore = 10 °C ; limite inferiore del valore istantaneo = 0 °C

PID_3Step si comporta come descritto di seguito:

Valore istantaneo	InputWarning_H	InputWarning_L	Modo di funzionamento
> 98 °C	TRUE	FALSE	Inattivo
≤ 98 °C e > 90 °C	TRUE	FALSE	Funzionamento automatico
≤ 90 °C e ≥ 10 °C	FALSE	FALSE	Funzionamento automatico
< 10 °C e ≥ 0 °C	FALSE	TRUE	Funzionamento automatico
< 0 °C	FALSE	TRUE	Inattivo

Parametri PID V1

Nella finestra di configurazione "Parametri PID" vengono visualizzati i parametri PID. Durante l'ottimizzazione i parametri PID vengono adattati al proprio sistema regolato. Non è necessario, quindi, inserirli manualmente.

NOTA

I parametri PID attualmente effettivi si trovano nella struttura Retain.CtrlParams.

Per evitare un comportamento errato del regolatore PID, modificare online i parametri PID attualmente effettivi solo nel modo di funzionamento "Inattivo".

Se si vogliono modificare i parametri PID online nei modi di funzionamento "Automatico" o "Manuale", modificarli nella struttura CtrlParamsBackUp e applicare le modifiche nella struttura Retain.CtrlParams nel modo seguente:

- PID_3Step V1: applicare le modifiche con Config.LoadBackUp = TRUE
- PID_3Step V2: applicare le modifiche con LoadBackUp = TRUE

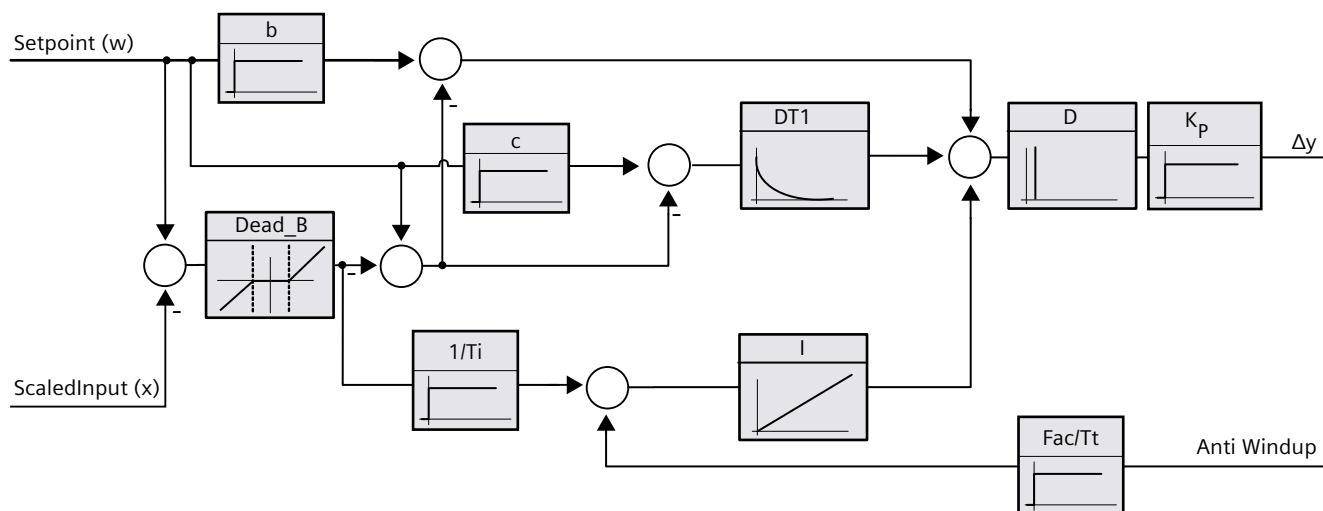
Le modifiche dei parametri PID eseguite online in modalità "Funzionamento automatico" possono causare gradini nel valore di uscita.

L'algoritmo PID utilizza la seguente formula:

$$\Delta y = K_p \cdot s \cdot \left[(b \cdot w - x) + \frac{1}{T_i \cdot s} (w - x) + \frac{T_D \cdot s}{a \cdot T_D \cdot s + 1} (c \cdot w - x) \right]$$

Δy	Valore di uscita dell'algoritmo PID
K_p	Guadagno proporzionale
s	Operatore di Laplace
b	Ponderazione del componente P
w	Setpoint
x	Valore istantaneo
T_i	Tempo di integrazione
a	Coefficiente per il ritardo derivativo (ritardo derivativo $T_1 = a \times T_D$)
T_D	Tempo derivativo
c	Ponderazione del componente D

Il seguente grafico mostra i parametri nell'algoritmo PID:



Tutti i parametri PID sono a ritenzione. Per l'inserimento manuale dei parametri PID, eseguire il caricamento completo di PID_3Step.

Caricamento degli oggetti tecnologici del dispositivo [\(Pagina 48\)](#)

Guadagno proporzionale

Questo valore indica il guadagno proporzionale del regolatore. PID_3Step non opera con un guadagno proporzionale negativo. Il senso di regolazione si inverte in Impostazioni di base > Modo di regolazione.

Tempo di integrazione

Il tempo di integrazione determina il comportamento temporale del componente I. La disattivazione del componente I avviene impostando il tempo di integrazione = 0,0. Se il tempo di integrazione viene modificato online da un altro valore a 0.0 in modalità "Funzionamento automatico", il componente I precedente viene cancellato e si verifica un gradino del valore di uscita.

Tempo derivativo

Il tempo derivativo determina il comportamento temporale del componente D. La disattivazione del componente D avviene impostando il tempo derivativo = 0,0.

Coefficiente ritardo derivativo

L'effetto del componente D viene ritardato dal coefficiente per il ritardo derivativo.

Ritardo derivativo = Tempo derivativo × Coefficiente ritardo derivativo

- 0.0: Il componente D ha effetto solo per un ciclo e quindi è quasi inattivo.
- 0.5: Questo valore ha dato buoni risultati nelle operazioni con i sistemi regolati con **una** costante temporale dominante.
- > 1.0: Maggiore è il coefficiente, maggiore sarà il ritardo dell'azione del componente D.

Ponderazione del componente P

In caso di variazioni del setpoint è possibile ridurre il componente P.

Sono adeguati valori da 0.0 a 1.0.

- 1.0: componente P pienamente attivo in caso di variazione del setpoint
- 0.0: componente P non attivo in caso di variazione del setpoint

In caso di variazioni del valore istantaneo il componente P è sempre pienamente attivo.

Ponderazione del componente D

In caso di variazioni del setpoint è possibile ridurre il componente D.

Sono adeguati valori da 0.0 a 1.0.

- 1.0: componente D pienamente attivo in caso di variazione del setpoint
- 0.0: componente D non attivo in caso di variazione del setpoint

In caso di variazioni del valore istantaneo il componente D è sempre pienamente attivo.

Tempo di campionamento algoritmo PID

Poiché il sistema regolato richiede un certo tempo per reagire a una modifica del valore di uscita, è utile non calcolare questo valore in ogni ciclo. Il tempo di campionamento dell'algoritmo PID è il tempo che trascorre tra due calcoli del valore di uscita. Viene determinato durante l'ottimizzazione e arrotondato a un multiplo del tempo di campionamento di PID_3Step. Tutte le altre funzioni di PID_3Step vengono eseguite ad ogni richiamo.

Ampiezza banda morta

La banda morta sopprime la componente di disturbo nello stato assestato del regolatore. L'ampiezza di banda morta indica la grandezza della banda morta. Con un'ampiezza di banda morta di 0,0, essa è disattivata.

Se sono stati configurati valori diversi da 1.0 per la ponderazione del componente P o per quella del componente D le modifiche dei setpoint influenzano il valore di uscita anche all'interno della banda morta.

Le modifiche del valore istantaneo all'interno della banda morta non influenzano il valore di uscita indipendentemente dalla ponderazione.

6.3.2 Messa in servizio di PID_3Step V1

6.3.2.1 Messa in servizio V1

Nel campo di lavoro "Ottimizzazione" si possono controllare il setpoint, il valore istantaneo e il valore di uscita in funzione del tempo. Il registratore supporta le seguenti funzioni per la messa in servizio:

- Ottimizzazione iniziale del regolatore
- ottimizzazione fine del regolatore
- Controllo della regolazione in corso nella finestra delle curve

Per tutte le funzione deve essere stabilito un collegamento online alla CPU.

Operazioni principali

- Selezionare il tempo di aggiornamento desiderato nella casella di riepilogo "Tempo di aggiornamento".
Tutti i valori nel campo di lavoro "Ottimizzazione" vengono aggiornati nel momento selezionato.
- Fare clic sul simbolo "Avvio" nel gruppo Misurazione se si vogliono utilizzare le funzioni per la messa in servizio.
Viene avviata la registrazione dei valori. Nella vista delle curve vengono immessi i valori attuali per il setpoint, il valore istantaneo e il valore di uscita. Viene abilitato l'utilizzo della finestra di messa in servizio.
- Fare clic sul simbolo "Stop" per disattivare le funzioni di messa in servizio.
I valori registrati nella vista delle curve possono essere ulteriormente analizzati.
- Chiudendo la finestra di messa in servizio si interrompe la registrazione nella vista delle curve e i valori registrati vengono cancellati.

6.3.2.2 Ottimizzazione iniziale V1

L'ottimizzazione iniziale rileva la risposta del processo a un impulso del valore di uscita e cerca il punto di flesso. Dall'incremento massimo e dal tempo morto del sistema regolato vengono calcolati i parametri PID ottimali.

Più stabile è il valore istantaneo, più facilmente e con maggiore precisione si possono determinare i parametri PID. Un disturbo del valore istantaneo è accettabile fintanto che l'incremento dello stesso è significativamente maggiore rispetto al disturbo. I parametri PID vengono salvati prima di essere nuovamente calcolati.

Il setpoint è congelato durante l'ottimizzazione iniziale.

Presupposto

- L'istruzione PID_3Step viene richiamata in un OB di schedulazione orologio.
- ManualEnable = FALSE
- PID_3Step si trova nel modo di funzionamento "Inattivo" o "Funzionamento manuale".
- Il setpoint e il valore istantaneo sono compresi entro i limiti configurati (vedere la configurazione di "Impostazione del valore istantaneo").

Procedimento

Per eseguire l'ottimizzazione iniziale procedere nel seguente modo:

1. Fare doppio clic nella navigazione del progetto sulla voce "PID_3Step > Messa in servizio".
2. Selezionare la voce "Ottimizzazione iniziale" nella casella di riepilogo "Tipo di ottimizzazione" del campo di lavoro "Ottimizzazione".
3. Fare clic sul simbolo "Start".
 - Viene creato un collegamento online.
 - Viene avviata la registrazione dei valori.
 - Viene avviata l'ottimizzazione iniziale.
 - Il campo "Stato" indica le operazioni in corso e gli eventuali errori. La barra di avanzamento indica lo stato di avanzamento dell'operazione in corso.

NOTA

Se la barra di avanzamento ha raggiunto il 100% e si deve presupporre un blocco dell'ottimizzazione, fare clic sul simbolo "Stop". Verificare la configurazione dell'oggetto tecnologico ed eventualmente riavviare l'ottimizzazione.

Risultato

Se l'ottimizzazione iniziale è stata eseguita senza messaggi di errore significa che i parametri PID sono stati ottimizzati. PID_3Step commuta sul funzionamento automatico e utilizza i parametri ottimizzati. Questi ultimi vengono mantenuti in caso di alimentazione OFF e nuovo avviamento della CPU.

Se non è possibile eseguire l'ottimizzazione iniziale, PID_3Step passa al modo di funzionamento "Inattivo".

6.3.2.3 Ottimizzazione fine V1

L'ottimizzazione fine genera un'oscillazione costante e limitata del valore istantaneo. I parametri PID per il punto di lavoro vengono ottimizzati partendo dall'ampiezza e dalla frequenza dell'oscillazione. A partire dai risultati vengono ricalcolati tutti i parametri PID. I parametri PID risultanti dall'ottimizzazione fine mostrano in genere un comportamento pilota e di disturbo migliore rispetto ai parametri PID dell'ottimizzazione iniziale.

PID_3Step cerca di generare automaticamente un'oscillazione che è maggiore rispetto al disturbo del valore istantaneo. L'ottimizzazione fine viene influenzata solo minimamente dalla stabilità del valore istantaneo. I parametri PID vengono salvati prima di essere nuovamente calcolati.

Il setpoint è congelato durante l'ottimizzazione fine.

Presupposti

- L'istruzione PID_3Step viene richiamata in un OB di schedulazione orologio.
- ManualEnable = FALSE
- Il tempo di regolazione del motore è stato configurato e misurato.
- Il setpoint e il valore istantaneo sono compresi entro i limiti configurati (vedere la configurazione di "Impostazione del valore istantaneo").
- Il circuito di regolazione si è assestato nel punto di lavoro. Il punto di lavoro è raggiunto quando il valore istantaneo corrisponde al setpoint.
- Non sono previste anomalie.
- PID_3Step si trova nel modo di funzionamento inattivo, automatico o manuale.

Procedimento a seconda della situazione di partenza

All'avvio, l'ottimizzazione fine si svolge nel modo seguente:

- Funzionamento automatico

Per migliorare i parametri PID esistenti mediante l'ottimizzazione, avviare l'ottimizzazione fine dal modo di funzionamento automatico.

PID_3Step esegue la regolazione con i parametri PID esistenti finché il circuito di regolazione è assestato e sono soddisfatti i requisiti per l'ottimizzazione fine. Solo in seguito viene avviata l'ottimizzazione fine.

- Inattivo o Funzionamento manuale

Viene sempre avviata per prima un'ottimizzazione iniziale. Viene quindi effettuata la regolazione con i parametri PID rilevati finché il circuito di regolazione è assestato e sono soddisfatti i requisiti per l'ottimizzazione fine. Solo in seguito viene avviata l'ottimizzazione fine.

Procedimento

Per eseguire l'ottimizzazione fine procedere nel seguente modo:

1. Selezionare la voce "ottimizzazione fine" nella casella di riepilogo "Tipo di ottimizzazione".
2. Fare clic sul simbolo "Start".
 - Viene creato un collegamento online.
 - Viene avviata la registrazione dei valori.
 - Viene avviata l'ottimizzazione fine.
 - Il campo "Stato" indica le operazioni in corso e gli eventuali errori. La barra di avanzamento indica lo stato di avanzamento dell'operazione in corso.

NOTA

Se la barra di avanzamento ha raggiunto il 100% e si deve presumere un blocco dell'ottimizzazione, fare clic sul simbolo "Stop" nel gruppo "Tipo di ottimizzazione". Verificare la configurazione dell'oggetto tecnologico ed eventualmente riavviare l'ottimizzazione.

Risultato

Se l'ottimizzazione fine è stata eseguita senza messaggi di errore significa che i parametri PID sono stati ottimizzati. PID_3Step commuta sul funzionamento automatico e utilizza i parametri ottimizzati. Questi ultimi vengono mantenuti in caso di alimentazione OFF e nuovo avviamento della CPU.

Se durante l'ottimizzazione fine si verificano degli errori, PID_3Step passa alla modalità di funzionamento "Inattivo".

6.3.2.4 Messa in servizio con parametri PID manuali V1

Procedimento

Per mettere in funzione PID_3Step con parametri PID manuali, procedere nel modo seguente:

1. Fare doppio clic nella navigazione del progetto sulla voce "PID_3Step > Configurazione".
2. Nella finestra di configurazione fare clic su "Impostazioni avanzate > Parametri PID".
3. Attivare la casella di opzione "Attiva immissione manuale".
4. Immettere i parametri PID.
5. Fare doppio clic nella navigazione del progetto sulla voce "PID_3Step > Messa in servizio".
6. Creare un collegamento online alla CPU.
7. Caricare i parametri PID nella CPU.
8. Fare clic sul simbolo "Attiva regolatore".

Risultato

PID_3Step commuta nel funzionamento automatico ed esegue la regolazione con i parametri PID attuali.

6.3.2.5 Misura del tempo di regolazione motore V1

Introduzione

Per ottenere un buon risultato di regolazione, PID_3Step richiede un tempo di regolazione del motore il più preciso possibile. I dati contenuti nella documentazione dell'organo attuatore sono valori medi per questo tipo di organi attuatori. Il valore può variare per l'organo attuatore effettivamente utilizzato.

Se si utilizzano organi attuatori con risposta di posizione o con segnali di riscontro, durante la messa in servizio è possibile misurare il tempo di regolazione del motore. Durante la misura del tempo di regolazione del motore, i limiti del valore di uscita non vengono considerati. L'organo attuatore può essere traslato fino al riscontro superiore o fino a quello inferiore.

Se non sono disponibili né una risposta di posizione né segnali di riscontro il tempo di regolazione del motore non può essere misurato.

Organi attuatori con risposta della posizione analogica

Per misurare il tempo di regolazione del motore con risposta di posizione procedere nel seguente modo:

Requisiti

- Nelle impostazioni di base deve essere selezionato Feedback o Feedback_PER e il segnale deve essere connesso.
- Deve esistere un collegamento online alla CPU.

1. Attivare la casella di scelta "Utilizza risposta di posizione".
2. Nel campo di immissione "Destinazione" inserire la destinazione del movimento dell'organo attuatore.

La risposta di posizione attuale (posizione di partenza) viene visualizzata. La differenza tra "Destinazione" e "Risposta di posizione" deve essere almeno pari al 50 % del campo del valore di uscita consentito.

3. Fare clic sull'icona  "Avvia misurazione del tempo di regolazione".

Risultato

L'organo attuatore viene portato dalla posizione di partenza alla destinazione. La misura del tempo si avvia immediatamente e termina quando l'organo attuatore ha raggiunto la destinazione. Il tempo di regolazione del motore si calcola sulla base della formula:

Tempo di regolazione motore = (Valore di uscita limite superiore – Valore di uscita limite inferiore) × Tempo di misura / IMPORTO (Destinazione – Posizione di partenza).

L'avanzamento e lo stato della misura del tempo di regolazione vengono visualizzati. Il tempo di regolazione misurato viene memorizzato nel blocco dati di istanza della CPU e visualizzato nel campo "Tempo di regolazione misurato". Se la misura del tempo di regolazione è terminata, PID_3Step passa al modo di funzionamento "Inattivo".

NOTA

Per applicare il tempo di regolazione del motore misurato nel progetto, fare clic sul simbolo  "Carica tempo di regolazione misurato".

Organi attuatori con segnali di riscontro

Per misurare il tempo di regolazione degli organi attuatori con segnali di riscontro, procedere nel seguente modo:

Requisiti

- Nelle impostazioni di base deve essere attivata la casella di opzione "Segnali di riscontro" e Actuator_H e Actuator_L devono essere connessi.
- Deve esistere un collegamento online alla CPU.

Per misurare il tempo di regolazione del motore con segnali di riscontro, procedere nel seguente modo:

1. Attivare la casella di scelta "Utilizza i segnali di riscontro dell'organo attuatore".
2. Selezionare la direzione in cui spostare l'organo attuatore.
 - Apri - Chiudi - Apri
L'organo attuatore viene dapprima portato verso il riscontro superiore, quindi verso quello inferiore e di nuovo verso quello superiore.
 - Chiudi - Apri - Chiudi
L'organo attuatore viene dapprima portato verso il riscontro inferiore, quindi verso quello superiore e di nuovo verso quello inferiore.
3. Fare clic sull'icona  "Avvia misurazione del tempo di regolazione".

Risultato

L'organo attuatore viene spostato nella direzione selezionata. La misura del tempo si avvia quando l'organo attuatore ha raggiunto il primo riscontro e termina quando lo raggiunge per la seconda volta. Dal tempo misurato diviso per due risulta il tempo di regolazione del motore.

L'avanzamento e lo stato della misura del tempo di regolazione vengono visualizzati. Il tempo di regolazione misurato viene memorizzato nel blocco dati di istanza della CPU e visualizzato nel campo "Tempo di regolazione misurato". Se la misura del tempo di regolazione è terminata, PID_3Step passa al modo di funzionamento "Inattivo".

Annullamento della misura del tempo di regolazione

Se la misura del tempo di regolazione viene annullata, PID_3Step passa immediatamente al modo di funzionamento "Inattivo". L'organo attuatore non viene più spostato. PID-3Step può essere riattivato nel registratore.

6.3.3 Simulazione di PID_3Step V1 con PLCSIM

NOTA

Simulazione con PLCSIM

Nella simulazione con PLCSIM il comportamento temporale del PLC simulato non è esattamente identico a quello di un PLC "reale". In un PLC simulato, il clock di tempo effettivo di un OB di schedulazione orologio può presentare oscillazioni maggiori rispetto ai PLC "reali".

Nella configurazione standard, PID_3Step rileva automaticamente gli intervalli tra i richiami e ne controlla le oscillazioni.

Nella simulazione di PID_3Step con PLCSIM può essere pertanto identificato un errore del tempo di campionamento (ErrorBits = DW#16#00000800).

Ciò comporta l'annullamento di ottimizzazioni in corso.

La reazione nel funzionamento automatico dipende dal valore della variabile ActivateRecoverMode.

Per evitare questo comportamento, si consiglia, nella simulazione con il PLCSIM, di configurare PID_3Step come indicato di seguito:

- CycleTime.EnEstimation = FALSE
 - CycleTime.EnMonitoring = FALSE
 - CycleTime.Value: Assegnare a questa variabile il clock di tempo dell'OB di schedulazione orologio richiamante espresso in secondi.
-

Impiego di PID_Temp

7.1 Oggetto tecnologico PID_Temp

L'oggetto tecnologico PID_Temp mette a disposizione un regolatore PID continuo con ottimizzazione integrata. PID_Temp è stato concepito espressamente per la regolazione della temperatura e si addice per applicazioni di solo riscaldamento oppure di riscaldamento/raffreddamento. Per questo scopo sono disponibili due uscite, una per il riscaldamento e una per il raffreddamento. PID_Temp può essere inoltre utilizzato anche per altri compiti di regolazione. PID_Temp può essere impiegato in cascata, nel funzionamento manuale o automatico.

PID_Temp rileva ininterrottamente il valore istantaneo misurato all'interno di un circuito di regolazione e lo confronta con il setpoint impostato. In base alle differenze di regolazione rilevate l'istruzione PID_Temp calcola il valore di uscita per il riscaldamento e/o il raffreddamento mediante il quale il valori istantaneo viene adeguato al setpoint. I valori di uscita sono formati dai tre seguenti componenti del regolatore PID:

- Componente P
Il componente P del valore di uscita aumenta in modo proporzionale alla differenza di regolazione.
- Componente I
Il componente I del valore di uscita aumenta fino a compensare la differenza di regolazione.
- Componente D
Il componente D aumenta con l'aumentare della velocità di variazione della differenza di regolazione. Il valore istantaneo viene adeguato il più rapidamente possibile al setpoint. Se la velocità di variazione della differenza di regolazione diminuisce nuovamente, il componente D si riduce.

L'istruzione PID_Temp calcola automaticamente i parametri P, I e D del sistema regolato in fase di "Ottimizzazione iniziale". I parametri possono essere ulteriormente ottimizzati mediante un'"ottimizzazione fine". Non è necessario, quindi, determinare manualmente i parametri.

Per applicazioni di riscaldamento/raffreddamento possono essere impiegati un fattore di raffreddamento fisso oppure due set di parametri PID.

Ulteriori informazioni

- Panoramica dei regolatori software ([Pagina 43](#))
- Inserire oggetti tecnologici ([Pagina 45](#))
- Configurare oggetti tecnologici ([Pagina 46](#))
- Configurazione di PID_Temp ([Pagina 159](#))

7.2 Configurazione di PID_Temp

7.2.1 Impostazioni di base

7.2.1.1 Introduzione

Nelle "Impostazioni di base", nella finestra di ispezione o di configurazione, si possono configurare le seguenti proprietà dell'oggetto tecnologico PID_Temp:

- Grandezza fisica
- Comportamento all'avviamento dopo il reset
- Sorgente e inserimento del setpoint (solo nella finestra di ispezione)
- Selezione del valore istantaneo
- Sorgente e inserimento del valore istantaneo (solo nella finestra di ispezione)
- Selezione del valore di uscita per il riscaldamento
- Sorgente e inserimento del setpoint (solo nella finestra di ispezione)
- Attivazione e selezione del valore di uscita per il raffreddamento
- Sorgente e inserimento del valore di uscita per il raffreddamento (solo nella finestra di ispezione)
- Attivazione di PID_Temp come master o slave in una cascata
- Numero di slave
- Selezione del master (solo nella finestra di ispezione)

Setpoint, valore istantaneo, valore di uscita per il riscaldamento e valore di uscita per il raffreddamento

La selezione della sorgente e l'indicazione di valori o variabili per setpoint, valore istantaneo, valore di uscita per il riscaldamento e valore di uscita per il raffreddamento, possono essere eseguite nella finestra di ispezione dell'editor di programma.

Per ogni valore selezionare la sorgente:

- DB di istanza:
Viene utilizzato il valore salvato nel DB di istanza. Il valore deve essere aggiornato dal programma utente nel DB di istanza. Nell'istruzione non può essere inserito alcun valore. È possibile effettuare le modifiche tramite HMI.
- Istruzione:
Viene utilizzato il valore interconnesso con l'istruzione. A ogni richiamo dell'istruzione il valore viene scritto nel DB di istanza. Le modifiche tramite HMI non sono possibili.

7.2.1.2 Modo di regolazione

Grandezza fisica

Nel gruppo "Modo di regolazione" selezionare la grandezza fisica e l'unità per il setpoint e il valore istantaneo. I valori verranno quindi visualizzati in questa unità.

Comportamento all'avviamento

1. Per passare al modo di funzionamento "Inattivo" dopo il nuovo avvio della CPU, disattivare la casella di scelta Attiva Mode dopo il riavvio della CPU".

Per passare al modo di funzionamento salvato in "Mode" dopo il nuovo avvio della CPU attivare la casella di scelta "Attiva Mode dopo il riavvio della CPU".

2. Nella casella di riepilogo "Imposta Mode su", selezionare il modo di funzionamento da attivare dopo il caricamento completo nel dispositivo.

Dopo l'esecuzione della funzione "Carica nel dispositivo", PID_Temp si avvia nel modo di funzionamento selezionato. Ad ogni riavvio PID_Temp si avvia nell'ultimo modo di funzionamento salvato alla voce "Mode".

In fase di selezione dell'ottimizzazione iniziale o dell'ottimizzazione fine, devono essere inoltre impostate o resettate le variabili Heat.EnableTuning e Cool.EnableTuning, al fine di consentire la scelta tra l'ottimizzazione per il riscaldamento e quella per il raffreddamento.

Esempio:

È stata attivata la casella di scelta "Attiva Mode dopo il riavvio della CPU" e nell'elenco "Imposta Mode su" è stata selezionata la voce "Ottimizzazione iniziale". Dopo l'esecuzione della funzione "Carica nel dispositivo" PID_Temp si avvia nel modo di funzionamento "Ottimizzazione iniziale". Se l'ottimizzazione iniziale è ancora attiva, dopo il riavvio della CPU, PID_Temp si avvia nuovamente in questo modo di funzionamento (riscaldamento/raffreddamento in funzione delle variabili Heat.EnableTuning e Cool.EnableCooling). Se l'ottimizzazione iniziale si è conclusa correttamente e il funzionamento automatico è attivo, dopo il riavvio della CPU, PID_Temp si avvia nel "Funzionamento automatico".

7.2.1.3 Setpoint

Procedura

Per predefinire un setpoint fisso procedere nel modo seguente:

1. Selezionare "DB di istanza".
2. Inserire un setpoint, ad es. 80°C.
3. Se necessario, cancellare una voce dell'istruzione.

Per predefinire un setpoint variabile, procedere nel modo seguente:

1. Selezionare "Istruzione".
2. Inserire il nome della variabile REAL in cui è memorizzato il setpoint.

Alla variabile REAL è possibile assegnare diversi valori tramite comandi del programma, ad es. per modificare il setpoint con comandi a tempo.

7.2.1.4 Valore istantaneo

Se si utilizza direttamente il valore dell'ingresso analogico, PID_Temp lo riporta in scala nella grandezza fisica.

Se si desidera prima trattare il valore dell'ingresso analogico è necessario scrivere un proprio programma per il trattamento. Supponiamo che il valore istantaneo non sia direttamente proporzionale al valore nell'ingresso analogico. Il valore istantaneo trattato dovrà essere nel formato in virgola mobile.

Procedimento

Per utilizzare direttamente il valore dell'ingresso analogico procedere nel modo seguente:

1. Selezionare la voce "Input_PER" nella casella di riepilogo "Input".
2. Selezionare come sorgente "Istruzione".
3. Inserire l'indirizzo dell'ingresso analogico.

Per utilizzare il valore istantaneo trattato nel formato in virgola mobile procedere nel modo seguente:

1. Selezionare la voce "Input" nella casella di riepilogo "Input".
2. Selezionare come sorgente "Istruzione".
3. Inserire il nome della variabile in cui è memorizzato il valore istantaneo trattato.

7.2.1.5 Valore di uscita riscaldamento e raffreddamento

L'istruzione PID_Temp mette a disposizione un regolatore PID con ottimizzazione integrata per processi di temperatura. PID_Temp si addice per applicazioni di riscaldamento oppure di riscaldamento/raffreddamento.

PID_Temp fornisce i seguenti valori di uscita. L'utilizzo di un determinato valore di uscita dipende dall'organo attuatore.

- **OutputHeat**

Valore di uscita per il riscaldamento (formato di virgola mobile): Il valore di uscita per il riscaldamento deve essere trattato con il programma utente, ad es. perché l'organo attuatore mostra un comportamento non lineare.

- **OutputHeat_PER**

Valore di uscita analogico riscaldamento: L'accesso all'organo attuatore per il riscaldamento avviene mediante un'uscita analogica ed il relativo comando mediante un segnale continuo, ad es. 0...10 V, 4...20 mA.

- **OutputHeat_PWM**

Valore di uscita con modulazione ad ampiezza d'impulso per il riscaldamento: L'organo attuatore per il riscaldamento viene comandato mediante un'uscita digitale. Tramite una modulazione di larghezza di impulso vengono calcolati i tempi di attivazione e disattivazione variabili.

- **OutputCool**

Valore di uscita per il raffreddamento (formato di virgola mobile): Il valore di uscita per il raffreddamento deve essere trattato con il programma utente, ad es. perché l'organo attuatore mostra un comportamento non lineare.

- **OutputCool_PER**

Valore di uscita analogico raffreddamento: L'accesso all'organo attuatore per il raffreddamento avviene mediante un'uscita analogica ed il relativo comando mediante un segnale continuo, ad es. 0...10 V, 4...20 mA.

- **OutputCool_PWM**

Valore di uscita con modulazione ad ampiezza d'impulso per il raffreddamento: L'organo attuatore per il raffreddamento viene comandato mediante un'uscita digitale. Tramite una modulazione di larghezza di impulso vengono calcolati i tempi di attivazione e disattivazione variabili.

Per essere disponibile, l'uscita di raffreddamento deve essere stata attivata nella casella di scelta "Attiva raffreddamento".

- Se questa casella di scelta è disattivata, il valore di uscita dell'algoritmo PID (PidOutputSum) viene riportato in scala ed emesso sulle uscite per il riscaldamento.
- Se questa casella di scelta è attivata, i valori di uscita positivi dell'algoritmo PID (PidOutputSum) vengono riportati in scala ed emessi sulle uscite per il riscaldamento. I valori negativi di uscita dell'algoritmo PID vengono riportati in scala ed emessi sulle uscite per il raffreddamento. Nelle impostazioni dell'uscita, è possibile scegliere tra due metodi per il calcolo del valore di uscita.

NOTA

Osservare quanto segue:

- Le uscite OutputHeat_PWM, OutputHeat_PER, OutputCool_PWM, OutputCool_PER vengono calcolate soltanto se sono state opportunamente selezionate nella casella di riepilogo.
 - L'uscita OutputHeat viene sempre calcolata.
 - L'uscita OutputCool viene calcolata se è stata attivata la casella di scelta per il raffreddamento.
 - La casella di scelta "Attiva raffreddamento" è disponibile soltanto se il regolatore non è stato configurato come master in una cascata.
-

Procedimento

Per utilizzare il valore di uscita analogico procedere nel modo seguente:

1. Nella casella di riepilogo "OutputHeat" o "OutputCool" selezionare la voce "OutputHeat_PER" o "OutputCool_PER".
2. Selezionare "Istruzione".
3. Inserire l'indirizzo dell'uscita analogica.

Per utilizzare il valore di uscita con modulazione di ampiezza impulsi procedere nel modo seguente:

1. Nella casella di riepilogo "OutputHeat" o "OutputCool" selezionare la voce "OutputHeat_PWM" o "OutputCool_PWM".
2. Selezionare "Istruzione".
3. Inserire l'indirizzo dell'uscita digitale.

Per trattare il valore di uscita con il programma utente procedere nel modo seguente:

1. Nella casella di riepilogo "OutputHeat" o "OutputCool" selezionare la voce "OutputHeat" o "OutputCool".
2. Selezionare "Istruzione".
3. Indicare il nome della variabile utilizzata per il trattamento del valore di uscita.
4. Trasferire il valore di uscita trattato all'organo attuatore mediante un'uscita analogica o digitale della CPU.

7.2.1.6 Cascata

Un'istanza PID_Temp che riceve il setpoint da un regolatore master sovraordinato e inoltra a sua volta il proprio valore di uscita ad un regolatore slave subordinato, è contemporaneamente regolatore master e regolatore slave. Per quest'istanza PID_Temp devono essere eseguite entrambe le configurazioni indicate nel seguito. E' il caso ad es. dell'istanza media PID_Temp nella regolazione in cascata con tre grandezze di misura concatenate e tre istanze PID_Temp.

Configurazione del regolatore come master in una cascata

Un regolatore master predefinisce con la relativa uscita il setpoint di un regolatore slave.

Per l'impiego di PID_Temp come master in una cascata, disattivare il raffreddamento nelle impostazioni di base. Per la configurazione dell'istanza PID_Temp come regolatore master in una cascata, attivare la casella di scelta "Il regolatore è il master". La selezione del valore di uscita per riscaldamento viene impostata automaticamente su OutputHeat.

OutputHeat_PWM e OutputHeat_PER non possono essere impiegati in un master in cascata.

Indicare infine il numero dei regolatori slave direttamente subordinati che ricevono il setpoint da questo regolatore master.

Se l'assegnazione del parametro OutputHeat del master al parametro Setpoint dello slave avviene senza l'impiego di una funzione di scala propria, può essere necessario un adeguamento dei limiti del valore di uscita e della graduazione di scala dell'uscita del master al campo del valore istantaneo e del setpoint dello slave. L'adeguamento può essere eseguito nelle impostazioni dell'uscita del master nel campo "OutputHeat / OutputCool".

Configurazione del regolatore come slave in una cascata

Un regolatore slave riceve il setpoint (parametri Setpoint) dall'uscita del suo regolatore master (parametri OutputHeat).

Per la configurazione dell'istanza PID_Temp come regolatore slave in una cascata, attivare la casella di scelta "Il regolatore è lo slave" nelle impostazioni di base.

Nella finestra di ispezione dell'editor di programma selezionare infine l'istanza PID_Temp da impiegare come regolatore master per questo slave. Con questa selezione i parametri Master e Setpoint del regolatore slave vengono interconnessi con quelli del regolatore master (le interconnessioni preesistenti a questi parametri sono state sovrascritte).

Quest'interconnessione consente lo scambio di informazioni e la predefinizione del setpoint tra master e slave. L'interconnessione nel parametro Setpoint del regolatore slave può essere all'occorrenza modificata a posteriori, ad es. per l'inserimento di un ulteriore filtro. La modifica a posteriore all'interconnessione nel parametro master non è consentita.

Nel regolatore master selezionato deve essere attivata la casella di scelta "Il regolatore è il master" ed il numero di slave deve essere correttamente configurato. Il regolatore master deve essere richiamato prima del regolatore slave nel medesimo OB di schedulazione orologio.

Ulteriori informazioni

Ulteriori informazioni sulla creazione del programma, sulla configurazione e la messa in servizio con l'impiego di PID_Temp nella regolazione in cascata, sono disponibili in Regolazione in cascata con PID_Temp [\(Pagina 193\)](#).

7.2.2 Impostazioni del valore iniziale

7.2.2.1 Limiti del valore istantaneo

Come valori limite è necessario definire un limite assoluto superiore e inferiore del valore istantaneo adeguato al sistema regolato. Non appena questi valori vengono superati in negativo o in positivo si verifica un errore (ErrorBits = 0001h). L'ottimizzazione viene interrotta se i limiti del valore istantaneo vengono superati. Il comportamento di PID_Temp in caso di errore nel funzionamento automatico può essere definito nelle impostazioni dell'uscita.

7.2.2.2 Riporta in scala valore istantaneo

Se è stato configurato l'utilizzo di Input_PER nelle impostazioni di base è necessario convertire il valore dell'ingresso analogico nella grandezza fisica del valore istantaneo. Nel campo di visualizzazione Input_PER viene visualizzata la configurazione attuale.

Se il valore istantaneo è direttamente proporzionale al valore dell'ingresso analogico, Input_PER viene riportato in scala in base a una coppia di valori inferiore e una superiore.

Procedimento

Per riportare in scala il valore istantaneo procedere nel modo seguente:

1. Immettere nei campi "Valore istantaneo inferiore riportato in scala" e "Inferiore" la coppia di valori inferiore.
2. Immettere nei campi "Valore istantaneo superiore riportato in scala" e "Superiore" la coppia di valori superiore.

Nella configurazione HW sono memorizzate coppie di valori predefinite. Per utilizzare le coppie di valori della configurazione HW procedere nel modo seguente:

1. Selezionare l'istruzione PID_Temp nell'editor di programma.
2. Nelle impostazioni di base collegare Input_PER con un ingresso analogico.
3. Nelle impostazioni del valore istantaneo fare clic sul pulsante "Impostazione automatica".

I valori esistenti vengono sovrascritti dai valori della configurazione HW.

7.2.3 Impostazioni dell'uscita

7.2.3.1 Impostazioni di base dell'uscita

Metodo per riscaldamento e raffreddamento

Se il raffreddamento è attivato nelle impostazioni di base, il calcolo del valore di uscita PID può essere eseguito con due metodi:

- Commutazione parametri PID Config.AdvancedCooling = TRUE
Il calcolo del valore di uscita per il raffreddamento avviene con un set di parametri PID proprio. Sulla base del valore di uscita calcolato e della differenza di regolazione, l'algoritmo PID stabilisce se i parametri PID debbano essere impiegati per il riscaldamento o per il raffreddamento. Questo metodo si addice quando l'attuatore di raffreddamento e quello di riscaldamento presentano un comportamento temporale e un guadagno diversi. L'ottimizzazione iniziale e l'ottimizzazione fine per il raffreddamento sono disponibili soltanto con questo metodo.
- Fattore di raffreddamento (Config.AdvancedCooling = FALSE):
Il calcolo del valore di uscita per il raffreddamento impiega i parametri PID di riscaldamento considerando il fattore di raffreddamento configurabile Config.CoolFactor. Questo metodo si addice quando l'organo attuatore del raffreddamento e quello del riscaldamento presentano lo stesso comportamento temporale ma guadagni diversi. Selezionando questo metodo, l'ottimizzazione iniziale e l'ottimizzazione fine, nonché il set di parametri PID per il raffreddamento non sono disponibili. Possono essere eseguite soltanto le ottimizzazioni per il riscaldamento.

Fattore di raffreddamento

Se come metodo per il riscaldamento/raffreddamento è stato selezionato il fattore di raffreddamento, quest'ultimo viene considerato come fattore nel calcolo del valore di uscita per il raffreddamento. In questo modo, per l'organo attuatore di riscaldamento e di raffreddamento, possono essere considerati guadagni diversi.

Il fattore di raffreddamento non viene impostato automaticamente o adeguato durante l'ottimizzazione. Esso deve essere configurato manualmente e in modo corretto con il rapporto "Guadagno dell'organo attuatore del riscaldamento / Guadagno dell'organo attuatore del raffreddamento".

Esempio: Fattore di raffreddamento = 2.0 sta ad indicare che il guadagno dell'organo attuatore del riscaldamento è il doppio di quello dell'organo attuatore del raffreddamento.

Il fattore di raffreddamento è efficace e modificabile soltanto se è stato selezionato come metodo di riscaldamento / raffreddamento.

Comportamento in caso di errori

ATTENZIONE

L'impianto può subire danneggiamenti.

Se in presenza di un errore vengono emessi "Valore attuale per la durata dell'errore" o "Valore di uscita sostitutivo per la durata dell'errore", PID_Temp rimane nel funzionamento automatico o manuale. In questo modo i limiti del valore istantaneo possono essere superati e l'impianto può essere danneggiato.

Configurare per il sistema regolato un comportamento da adottare in caso di errore che protegga l'impianto da danneggiamenti.

PID_Temp è preimpostato in modo da consentire che in caso di errore la regolazione rimanga in prevalenza attiva.

Se nel funzionamento di regolazione si verificano frequentemente errori, questa preimpostazione comporta un peggioramento del comportamento del regolatore. In questo caso controllare il parametro ErrorBits ed eliminare la causa dell'errore.

In caso di errori PID_Temp emette un valore di uscita configurabile:

- Zero (Inattivo)

In presenza di qualsiasi errore, PID_Temp commuta nel modo di funzionamento "Inattivo" ed emette i seguenti valori:

- 0.0 come valore di uscita PID (PidOutputSum)
- 0.0 come valore di uscita per il riscaldamento (OutputHeat) e valore di uscita per il raffreddamento (OutputCool)
- 0 come valore di uscita analogico per il riscaldamento (OutputHeat_PER) e valore di uscita analogico per il raffreddamento (OutputCool_PER)
- FALSE come valore di uscita PWM per il riscaldamento (OutputHeat_PWM) e valore di uscita PWM per il raffreddamento (OutputCool_PWM)

Questo comportamento si verifica indipendentemente dai limiti del valore di uscita configurati e dalla graduazione di scala. Il regolatore viene riattivato soltanto al verificarsi di un fronte di discesa sul reset o di un fronte di salita su ModeActivate.

- Valore attuale per la durata dell'errore

La reazione all'errore è in funzione dell'errore stesso e del modo di funzionamento.

Se nel funzionamento automatico si verificano uno o più dei seguenti errori PID_Temp rimane in questo modo di funzionamento:

- 0000001h: Il parametro Input non rientra nei limiti del valore istantaneo.
- 0000800h: Errore del tempo di campionamento
- 0040000h: Valore del parametro Disturbance non valido.
- 8000000h: Errore durante il calcolo dei parametri PID.

Se nel funzionamento automatico si verificano uno o più dei seguenti errori, PID_Temp commuta nel modo di funzionamento "Valore di uscita sostitutivo con controllo errori" ed emette l'ultimo valore di uscita PID valido (PidOutputSum):

- 0000002h: Valore del parametro Input_PER non valido.
- 0000200h: Valore del parametro Input non valido.
- 0000400h: Calcolo del valore di uscita non riuscito.
- 0001000h: Valore non valido nel parametro Setpoint oppure SubstituteSetpoint.

I valori sulle uscite per riscaldamento e raffreddamento, risultanti dal valore di uscita PID, derivano dalla graduazione di scala dell'uscita.

Non appena l'errore viene eliminato, PID_Temp torna al funzionamento automatico.

Se nel funzionamento manuale si verifica un errore, PID_Temp rimane in questo modo di funzionamento e continua ad impiegare il valore manuale come valore di uscita PID.

Se il valore manuale non è valido viene utilizzato il valore di uscita sostitutivo configurato.

Se né il valore manuale né il valore di uscita sostitutivo sono validi, viene utilizzato il limite inferiore PID del valore di uscita per il riscaldamento (Config.Output.Heat.PidLowerLimit).

Se durante un'ottimizzazione iniziale o un'ottimizzazione fine si verifica il seguente errore, PID_Temp rimane nel modo di funzionamento attivo:

- 0000020h: L'ottimizzazione iniziale non è consentita durante l'ottimizzazione fine.

Con tutti gli altri errori, PID_Temp interrompe l'ottimizzazione e passa al modo di funzionamento dal quale è stata avviata l'ottimizzazione.

- Valore di uscita sostitutivo per la durata dell'errore

PID_Temp si comporta come descritto nel "Valore attuale per la durata dell'errore", nel modo di funzionamento "Valore di uscita sostitutivo con controllo errori" emette tuttavia il valore di uscita sostitutivo configurato (SubstituteOutput) come valore di uscita PID (PidOutputSum).

I valori sulle uscite per riscaldamento e raffreddamento, risultanti dal valore di uscita PID, derivano dalla graduazione di scala dell'uscita.

Per i regolatori con uscita di raffreddamento attiva (Config.ActivateCooling = TRUE) indicare:

- un valore di uscita sostitutivo positivo, per emettere il valore sulle uscite per il riscaldamento
- un valore di uscita sostitutivo negativo, per emettere il valore sulle uscite per il raffreddamento

Al verificarsi del seguente errore, PID_Temp rimane nel modo di funzionamento "Valore di uscita sostitutivo con controllo errori" ed emette il limite inferiore del valore di uscita PID per il riscaldamento (Config.Output.Heat.PidLowerLimit):

- 0020000h: Valore della variabile SubstituteOutput non valido.

7.2.3.2 Limiti e graduazione di scala del valore di uscita

Il valore di uscita PID (PidOutputSum) viene calcolato automaticamente, a seconda del tipo di funzionamento, dall'algoritmo PID oppure predefinito dal valore manuale (ManualValue) o dal valore di uscita sostitutivo configurato (SubstituteOutput).

Il valore di uscita PID viene limitato in funzione della configurazione:

- Se nelle impostazioni di base il raffreddamento è stato disattivato (Config.ActivateCooling = FALSE), il valore viene circoscritto sul limite superiore del valore di uscita PID (riscaldamento) (Config.Output.Heat.PidUpperLimit) e sul limite inferiore del valore di uscita PID (riscaldamento) (Config.Output.Heat.PidLowerLimit).

Entrambi i valori limite possono essere configurati nella sezione "OutputHeat / OutputCool" sull'asse orizzontale della curva caratteristica della scala. Questi valori possono essere visualizzati nelle sezioni "OutputHeat_PWM / OutputCool_PWM" e "OutputHeat_PER / OutputCool_PER" ma non possono essere modificati.

- Se nelle impostazioni di base il raffreddamento è stato attivato (Config.ActivateCooling = TRUE), il valore viene circoscritto sul limite superiore del valore di uscita PID (riscaldamento) (Config.Output.Heat.PidUpperLimit) e sul limite inferiore del valore di uscita PID (raffreddamento) (Config.Output.Cool.PidLowerLimit).

Entrambi i valori limite possono essere configurati nella sezione "OutputHeat / OutputCool" sull'asse orizzontale della curva caratteristica della scala. Questi valori possono essere visualizzati nelle sezioni "OutputHeat_PWM / OutputCool_PWM" e "OutputHeat_PER / OutputCool_PER" ma non possono essere modificati.

Il limite inferiore del valore di uscita PID (riscaldamento) (Config.Output.Heat.PidLowerLimit) ed il limite superiore del valore di uscita PID (raffreddamento) (Config.Output.Cool.PidUpperLimit) non possono essere modificati e devono essere occupati dal valore 0.0.

Il valore di uscita PID viene riportato in scala ed emesso sulle uscite per il riscaldamento ed il raffreddamento. La graduazione di scala può essere predefinita separatamente per ciascuna uscita e stabilita rispettivamente da due coppie di valori costituite da un limite del valore di uscita PID e da un valore di scala:

Uscita	Coppia di valori	Parametro
OutputHeat	Coppia di valori 1	Limite superiore valore di uscita PID (riscaldamento) Config.Output.Heat.PidUpperLimit, Valore di uscita superiore riportato in scala (riscaldamento) Config.Output.Heat.UpperScaling
	Coppia di valori 2	Limite inferiore valore di uscita PID (riscaldamento) Config.Output.Heat.PidLowerLimit, Valore di uscita inferiore riportato in scala (riscaldamento) Config.Output.Heat.LowerScaling
OutputHeat_PWM	Coppia di valori 1	Limite superiore valore di uscita PID (riscaldamento) Config.Output.Heat.PidUpperLimit, Valore di uscita superiore PWM riportato in scala (riscaldamento) Config.Output.Heat.PwmUpperScaling

Il limite inferiore del valore di uscita PID (riscaldamento) (Config.Output.Heat.PidLowerLimit) deve avere il valore 0.0, se è attivato il raffreddamento (Config.ActivateCooling = TRUE).

Il limite superiore del valore di uscita PID (raffreddamento) (Config.Output.Cool.PidUpperLimit) deve sempre avere il valore 0.0.

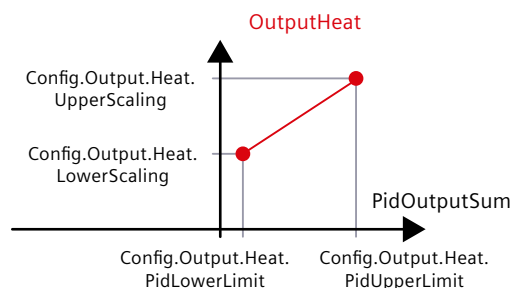
Uscita	Coppia di valori	Parametro
OutputHeat_PWM	Coppia di valori 2	Limite inferiore valore di uscita PID (riscaldamento) Config.Output.Heat.PidLowerLimit, Valore di uscita inferiore PWM riportato in scala (riscaldamento) Config.Output.Heat.PwmLowerScaling
OutputHeat_PER	Coppia di valori 1	Limite superiore valore di uscita PID (riscaldamento) Config.Output.Heat.PidUpperLimit, Valore di uscita analogico superiore riportato in scala (riscaldamento) Config.Output.Heat.PerUpperScaling
	Coppia di valori 2	Limite inferiore valore di uscita PID (riscaldamento) Config.Output.Heat.PidLowerLimit, Valore di uscita analogico inferiore riportato in scala (riscaldamento) Config.Output.Heat.PerLowerScaling
OutputCool	Coppia di valori 1	Limite inferiore valore di uscita PID (raffreddamento) Config.Output.Cool.PidLowerLimit, Valore di uscita superiore riportato in scala (raffreddamento) Config.Output.Cool.UpperScaling
	Coppia di valori 2	Limite superiore valore di uscita PID (raffreddamento) Config.Output.Cool.PidUpperLimit, Valore di uscita inferiore riportato in scala (raffreddamento) Config.Output.Cool.LowerScaling
OutputCool_PWM	Coppia di valori 1	Limite inferiore valore di uscita PID (raffreddamento) Config.Output.Cool.PidLowerLimit, Valore di uscita superiore PWM riportato in scala (raffreddamento) Config.Output.Cool.PwmUpperScaling
	Coppia di valori 2	Limite superiore valore di uscita PID (raffreddamento) Config.Output.Cool.PidUpperLimit, Valore di uscita inferiore PWM riportato in scala (raffreddamento) Config.Output.Cool.PwmLowerScaling
OutputCool_PER	Coppia di valori 1	Limite inferiore valore di uscita PID (raffreddamento) Config.Output.Cool.PidLowerLimit, Valore di uscita analogico superiore riportato in scala (raffreddamento) Config.Output.Cool.PerUpperScaling
	Coppia di valori 2	Limite superiore valore di uscita PID (raffreddamento) Config.Output.Cool.PidUpperLimit, Valore di uscita analogico inferiore riportato in scala (raffreddamento) Config.Output.Cool.PerLowerScaling

Il limite inferiore del valore di uscita PID (riscaldamento) (Config.Output.Heat.PidLowerLimit) deve avere il valore 0.0, se è attivato il raffreddamento (Config.ActivateCooling = TRUE).

Il limite superiore del valore di uscita PID (raffreddamento) (Config.Output.Cool.PidUpperLimit) deve sempre avere il valore 0.0.

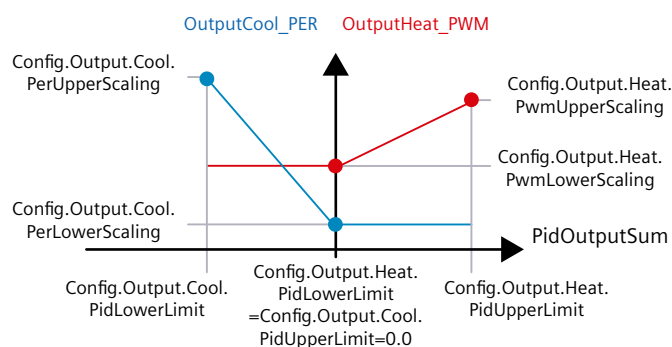
Esempio:

Graduazione di scala con impiego dell'uscita OutputHeat (raffreddamento disattivato; il limite inferiore del valore di uscita PID (riscaldamento) Config.Output.Heat.PidLowerLimit può essere diverso da 0.0):



Esempio:

Graduazione di scala con impiego delle uscite OutputHeat_PWM e OutputCool_PER (raffreddamento attivato; il limite inferiore del valore di uscita PID (riscaldamento) (Config.Output.Heat.PidLowerLimit) deve essere 0.0):



Ad eccezione del modo di funzionamento "Inattivo", il valore di un'uscita si colloca sempre tra il valore di uscita superiore e inferiore riportato in scala, ad es., nel caso di OutputHeat, sempre tra il valore di uscita superiore (riscaldamento) (Config.Output.Heat.UpperScaling) ed il valore di uscita inferiore riportati in scala (riscaldamento) (Config.Output.Heat.LowerScaling).

In caso di limitazione del valore sulla rispettiva uscita, devono essere adeguati anche questi valori della graduazione di scala.

I valori di scala di un'uscita possono essere configurati sugli assi verticali della curva caratteristica della scala. Ogni uscita dispone di due valori di scala propri. Per OutputHeat_PWM, OutputCool_PWM, OutputHeat_PER e OutputCool_PER, questi valori sono modificabili soltanto se nelle impostazioni di base è stata selezionata l'uscita corrispondente. In tutte le uscite per il raffreddamento, quest'ultimo deve essere inoltre attivato nelle impostazioni di base.

La vista della curva nella finestra di dialogo della messa in servizio visualizza solo i valori di OutputHeat e OutputCool a prescindere dall'uscita selezionata nelle impostazioni di base. Adeguare all'occorrenza anche i valori di scala per OutputHeat e OutputCool se si intende impiegare OutputHeat_PWM, OutputHeat_PER o OutputCool_PWM oppure OutputCool_PER e utilizzare la vista della curva nella finestra di dialogo della messa in servizio.

7.2.4 Impostazioni avanzate

7.2.4.1 Controllo del valore istantaneo

La finestra "Controllo del valore istantaneo" consente di configurare il limite di avviso inferiore e superiore del valore istantaneo. Se durante il funzionamento viene superato uno dei limiti, nell'istruzione "PID_Temp" viene visualizzato un avviso:

- Nel parametro di uscita InputWarning_H, se è stato superato il limite di avviso superiore
- Nel parametro di uscita InputWarning_L, se non è stato raggiunto il limite di avviso inferiore

I limiti di avviso devono essere compresi entro il limite superiore e inferiore del valore istantaneo.

Se non si inserisce alcun valore, vengono utilizzati il limite superiore e inferiore del valore istantaneo.

Esempio

Limite superiore del valore istantaneo = 98 °C ; limite di avviso superiore = 90 °C

Limite di avviso inferiore = 10 °C ; limite inferiore del valore istantaneo = 0 °C

PID_Temp si comporta come descritto di seguito:

Valore istantaneo	InputWarning_H	InputWarning_L	ErrorBits
> 98 °C	TRUE	FALSE	0001h
≤ 98 °C e > 90 °C	TRUE	FALSE	0000h
≤ 90 °C e ≥ 10 °C	FALSE	FALSE	0000h
< 10 °C e ≥ 0 °C	FALSE	TRUE	0000h
< 0 °C	FALSE	TRUE	0001h

La reazione di PID_Temp in caso di superamento positivo o negativo del limite superiore o inferiore del valore istantaneo viene configurata nelle impostazioni dell'uscita.

7.2.4.2 Limitazioni PWM

Il valore di uscita PID `PidOutputSum` viene riportato in scala e tramite la modulazione di larghezza di impulso trasformato in una sequenza di impulsi che viene emessa nei parametri di uscita `OutputHeat_PWM` e `OutputCool_PWM`.

Il tempo di campionamento "Algoritmo PID" è la distanza tra due calcoli del valore di uscita PID. Il tempo di campionamento viene utilizzato come durata periodo della modulazione di larghezza di impulso.

Durante il riscaldamento il valore di uscita PID viene sempre calcolato nel "Tempo di campionamento dell'algoritmo PID".

Il calcolo del valore di uscita PID durante il raffreddamento dipende dalla modalità selezionata per quest'ultimo nelle "Impostazioni di base Uscita":

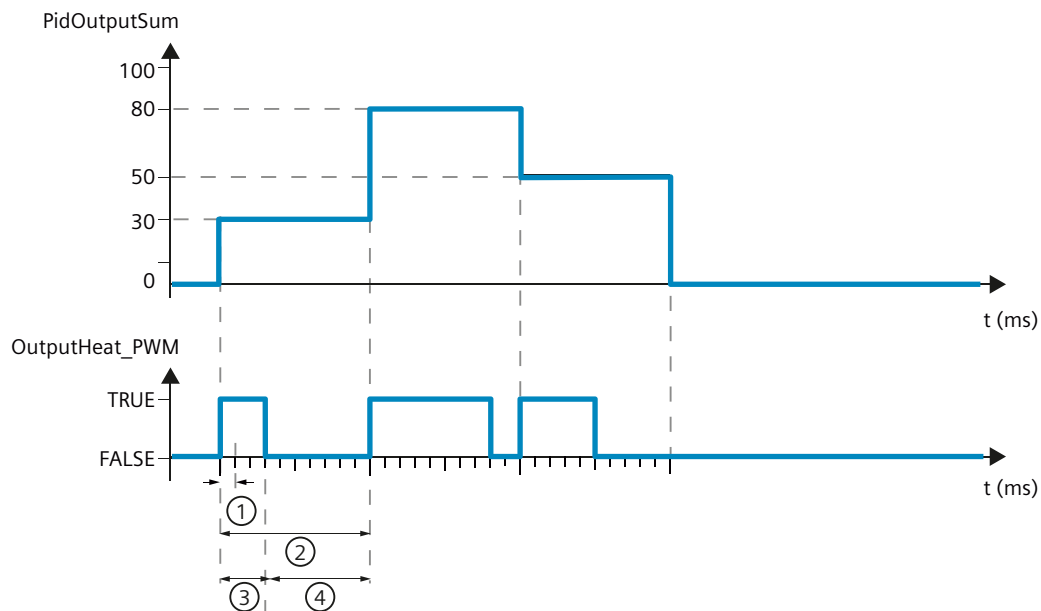
- Se viene impiegato il fattore di raffreddamento, vale il "Tempo di campionamento dell'algoritmo PID" per il riscaldamento.
- Se viene impiegata la commutazione dei parametri PID, vale il "Tempo di campionamento dell'algoritmo PID" per il raffreddamento.

Il tempo di campionamento dell'algoritmo PID per il riscaldamento e il raffreddamento viene determinato durante la prima ottimizzazione o durante l'ottimizzazione fine. Con l'impostazione manualmente dei parametri PID, è necessario configurare durante quest'operazione anche il tempo di campionamento dell'algoritmo PID per il riscaldamento e il raffreddamento.

`OutputHeat_PWM` e `OutputCool_PWM` vengono emessi durante il tempo di campionamento `PID_Temp`. Il tempo di campionamento `PID_Temp` corrisponde al tempo di ciclo dell'OB richiamante.

La durata dell'impulso è proporzionale al valore di uscita PID e ammonta sempre a un multiplo intero del tempo di campionamento `PID_Temp`.

Esempio di OutputHeat_PWM



- ① Tempo di campionamento PID_Temp
- ② Tempo di campionamento algoritmo PID per il riscaldamento
- ③ Durata dell'impulso
- ④ Durata pausa

Se arrotondati ad un multiplo intero del tempo di campionamento PID_Temp, il "Tempo ON minimo" e il "Tempo OFF minimo" possono essere impostati separatamente per il riscaldamento ed il raffreddamento.

Un impulso o una pausa non sono mai inferiori ai Tempi ON minimo o OFF minimo. Le imprecisioni che risultano da ciò vengono sommate e compensate nel ciclo successivo.

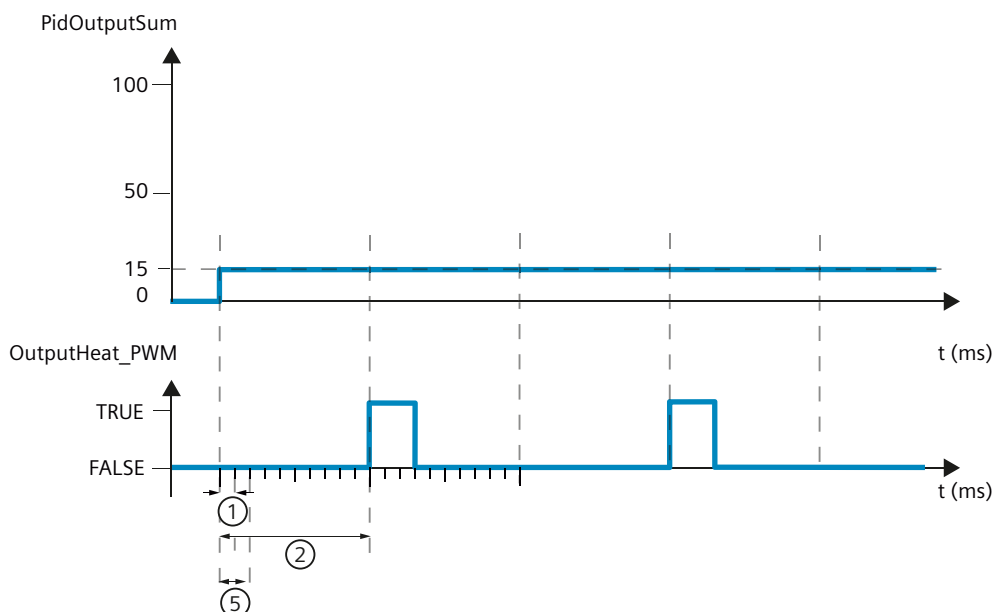
Esempio di OutputHeat_PWM

Il tempo di campionamento PID_Temp (corrispondente al tempo di ciclo dell'OB richiamante) è di 100 ms.

Il tempo di campionamento dell'algoritmo PID (corrispondente alla durata del periodo) è di 1000 ms.

Tempo ON minimo = 200 ms

Il valore di uscita PID `PidOutputSum` ammonta costantemente a 15 %. Il minimo impulso che PID_Temp può emettere corrisponde al 20 %. Nel primo ciclo non viene emesso alcun impulso. Nel secondo ciclo l'impulso non emesso del primo ciclo viene aggiunto all'impulso del secondo ciclo.



- ① Tempo di campionamento PID_Temp
- ② Tempo di campionamento algoritmo PID per il riscaldamento
- ⑤ Tempo minimo di attivazione

Per ridurre la frequenza di commutazione e non sovraccaricare l'organo attuatore, prolungare i tempi di ON e OFF minimi.

Se come uscita nelle impostazioni di base sono stati selezionati `OutputHeat`, `OutputCool` o `OutputHeat_PER` oppure `OutputCool_PER`, il tempo on ed il tempo off minimi non vengono analizzati e non sono modificabili.

Se con l'impiego di `OutputHeat_PWM` o `OutputCool_PWM`, il "tempo di campionamento dell'algoritmo PID" (`Retain.CtrlParams.Heat.Cycle` o `Retain.CtrlParams.Cool.Cycle`) e pertanto la durata periodo della modulazione di larghezza di impulso, fossero particolarmente lunghi è possibile, per migliorare il livellamento del valore istantaneo, indicare, divergendo dai valori normalmente previsti, una durata periodo più breve nei parametri `Config.Output.Heat.PwmPeriode` o `Config.Output.Cool.PwmPeriode` (vedere anche `Periodo Pwm` variabile [\(Pagina 433\)](#)).

NOTA

I tempi di attivazione e di disattivazione minimi hanno effetto solo sui parametri di uscita `OutputHeat_PWM` o `OutputCool_PWM` e non vengono utilizzati per generatori di impulsi eventualmente integrati nella CPU.

7.2.4.3 Parametri PID

Nella finestra di configurazione "Parametri PID" vengono visualizzati i parametri PID.

Se nelle impostazioni di base è stato attivato il raffreddamento e nelle impostazioni dell'uscita è stata selezionata la commutazione dei parametri PID come metodo per il riscaldamento/raffreddamento, sono disponibili due set di parametri: uno per il riscaldamento e uno per il raffreddamento.

Sulla base del valore di uscita calcolato e della differenza di regolazione, l'algoritmo PID stabilisce in questo caso se i parametri PID debbano essere impiegati per il riscaldamento o per il raffreddamento.

Se il raffreddamento è disattivato oppure se come metodo per il riscaldamento/raffreddamento è stato selezionato il fattore di raffreddamento, per il riscaldamento viene sempre impiegato il set di parametri.

Durante l'ottimizzazione i parametri PID vengono adeguati al sistema regolato ad eccezione dell'ampiezza della banda morta che deve essere configurata manualmente.

NOTA

I parametri PID attualmente effettivi si trovano nella struttura Retain.CtrlParams.

Per evitare un comportamento errato del regolatore PID, modificare online i parametri PID attualmente effettivi solo nel modo di funzionamento "Inattivo".

Se si vogliono modificare i parametri PID online nei modi di funzionamento "Automatico" o "Manuale", modificarli nella struttura CtrlParamsBackUp e applicare le modifiche con LoadBackUp = TRUE nella struttura Retain.CtrlParams.

Le modifiche dei parametri PID eseguite online in modalità "Funzionamento automatico" possono causare gradini nel valore di uscita.

PID_Temp è un regolatore PIDT1 con anti-windup e ponderazione dei componenti P e D.

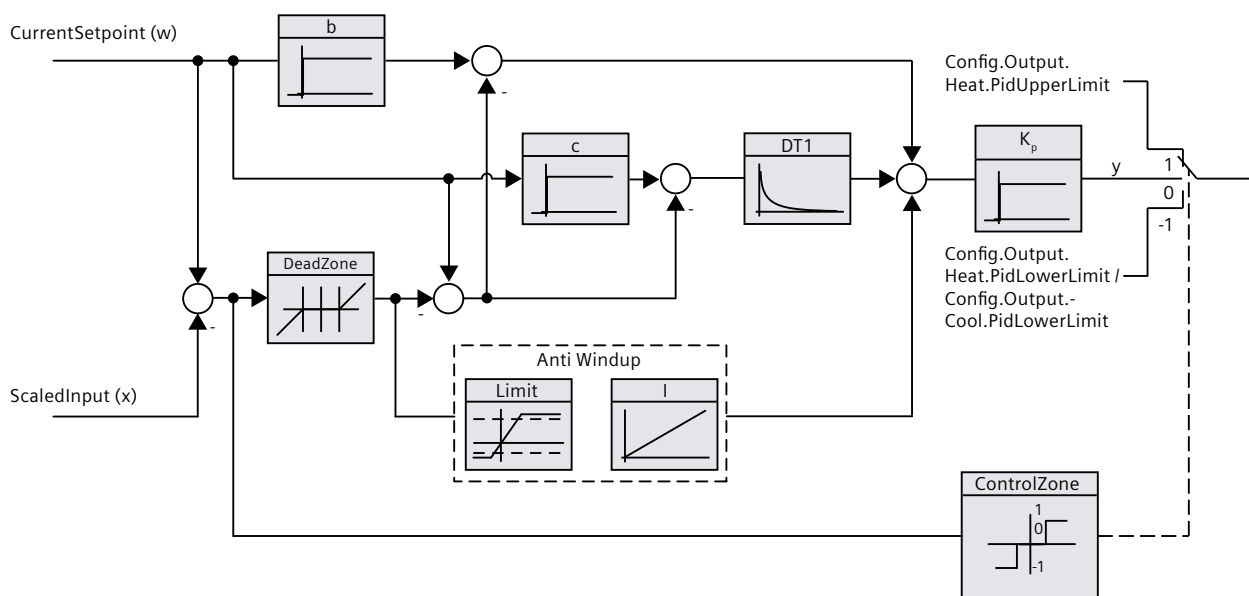
L'algoritmo PID utilizza la seguente formula (la zona di regolazione e la banda morta sono disattivate):

$$y = K_p \left[(b \cdot w - x) + \frac{1}{T_i \cdot s} (w - x) + \frac{T_d \cdot s}{a \cdot T_d \cdot s + 1} (c \cdot w - x) \right]$$

Simbolo	Descrizione	Parametri dell'istruzione PID_Temp
y	Valore di uscita dell'algoritmo PID	-
K _p	Guadagno proporzionale	Retain.CtrlParams.Heat.Gain Retain.CtrlParams.Cool.Gain CoolFactor
s	Operatore di Laplace	-
b	Ponderazione del componente P	Retain.CtrlParams.Heat.PWeighting Retain.CtrlParams.Cool.PWeighting
w	Setpoint	CurrentSetpoint
x	Valore istantaneo	ScaledInput
T _i	Tempo di integrazione	Retain.CtrlParams.Heat.Ti Retain.CtrlParams.Cool.Ti

Simbolo	Descrizione	Parametri dell'istruzione PID_Temp
T_D	Tempo derivativo	Retain.CtrlParams.Heat.Td Retain.CtrlParams.Cool.Td
a	Coefficiente per il ritardo di differenziazione (Ritardo derivativo $T1 = a \times T_D$)	Retain.CtrlParams.Heat.TdFiltRatio Retain.CtrlParams.Cool.TdFiltRatio
c	Ponderazione del componente D	Retain.CtrlParams.Heat.DWeighting Retain.CtrlParams.Cool.DWeighting
DeadZone	Ampiezza banda morta	Retain.CtrlParams.Heat.DeadZone Retain.CtrlParams.Cool.DeadZone
ControlZone	Ampiezza della zona di regolazione	Retain.CtrlParams.Heat.ControlZone Retain.CtrlParams.Cool.ControlZone

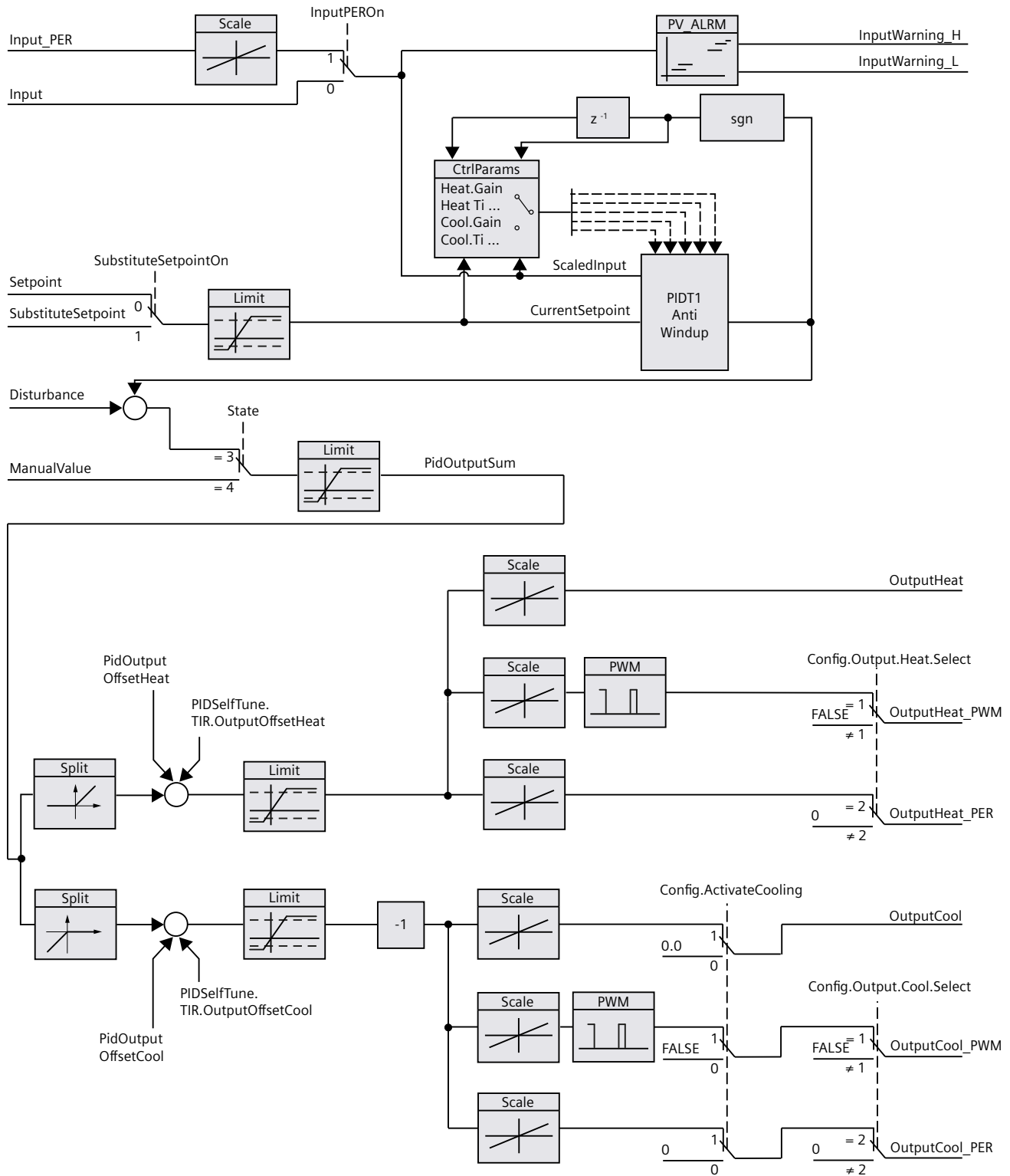
Il seguente grafico mostra i parametri nell'algoritmo PID:



Tutti i parametri PID sono a ritenzione. Per l'inserimento manuale dei parametri PID, eseguire il caricamento completo di PID_Temp(Caricamento degli oggetti tecnologici del dispositivo [\(Pagina 48\)](#)).

Schema a blocchi di PID_Temp

Il seguente schema a blocchi illustra l'integrazione dell'algoritmo PID in PID_Temp.



Guadagno proporzionale

Il valore indica il guadagno proporzionale del regolatore. PID_Temp non opera con il guadagno proporzionale negativo e supporta soltanto il senso di regolazione normale, in altri termini, con un incremento del valore di uscita PID (PidOutputSum) deve essere realizzato un incremento del valore istantaneo.

Tempo di integrazione

Il tempo di integrazione determina il comportamento temporale del componente I. La disattivazione del componente I avviene impostando il tempo di integrazione = 0.0. Se il tempo di integrazione viene modificato online da un altro valore a 0.0 in modalità "Funzionamento automatico", il componente I precedente viene cancellato e si verifica un gradino del valore di uscita.

Tempo derivativo

Il tempo derivativo determina il comportamento temporale del componente D. La disattivazione del componente D avviene impostando il tempo derivativo = 0.0.

Coefficiente ritardo derivativo

L'effetto del componente D viene ritardato dal coefficiente per il ritardo derivativo.

Ritardo derivativo = Tempo derivativo × Coefficiente ritardo derivativo

- 0.0: Il componente D ha effetto solo per un ciclo e quindi è quasi inattivo.
- 0.5: Questo valore ha dato buoni risultati nelle operazioni con i sistemi regolati con una costante temporale dominante.
- > 1.0: Maggiore è il coefficiente, maggiore sarà il ritardo dell'azione del componente D.

Ponderazione del componente P

In caso di variazioni del setpoint è possibile ridurre il componente P.

Sono adeguati valori da 0.0 a 1.0.

- 1.0: componente P pienamente attivo in caso di variazione del setpoint
- 0.0: componente P non attivo in caso di variazione del setpoint

In caso di variazioni del valore istantaneo il componente P è sempre pienamente attivo.

Ponderazione del componente D

In caso di variazioni del setpoint è possibile ridurre il componente D.

Sono adeguati valori da 0.0 a 1.0.

- 1.0: componente D pienamente attivo in caso di variazione del setpoint
- 0.0: componente D non attivo in caso di variazione del setpoint

In caso di variazioni del valore istantaneo il componente D è sempre pienamente attivo.

Tempo di campionamento algoritmo PID

Poiché il sistema regolato richiede un certo tempo per reagire a una modifica del valore di uscita, è utile non calcolare questo valore in ogni ciclo. Il tempo di campionamento "Algoritmo PID" è la distanza tra due calcoli del valore di uscita PID. Questo tempo viene rilevato durante l'ottimizzazione e arrotondato ad un multiplo del tempo di campionamento PID_Temp (tempo di ciclo dell'OB di schedulazione orologio). Tutte le altre funzioni di PID_Temp vengono eseguite ad ogni richiamo.

In caso di impiego di OutputHeat_PWM e OutputCool_PWM, come durata periodo della modulazione ampiezza impulso viene utilizzato il tempo di campionamento dell'algoritmo PID. La precisione del segnale di uscita viene determinata dal rapporto tra tempo di campionamento dell'algoritmo PID e tempo di ciclo dell'OB. Il tempo di ciclo non dovrebbe superare un decimo del tempo di campionamento dell'algoritmo PID.

Il tempo di campionamento dell'algoritmo PID utilizzato come durata periodo della modulazione ampiezza impulso in OutputCool_PWM, dipende dal metodo selezionato nelle "Impostazioni di base Uscita" per il riscaldamento/raffreddamento.

- Se viene impiegato il fattore di raffreddamento, il "Tempo di campionamento dell'algoritmo PID per il riscaldamento" vale anche per OutputCool_PWM.
- Se viene impiegata la commutazione dei parametri PID, il "Tempo di campionamento dell'algoritmo PID per il raffreddamento" vale come durata periodo per OutputCool_PWM.

Se con l'impiego di OutputHeat_PWM o OutputCool_PWM, il tempo di campionamento dell'algoritmo PID e pertanto la durata periodo della modulazione ampiezza impulso, fossero particolarmente lunghi è possibile, per migliorare il livellamento del valore istantaneo, indicare, divergendo dai valori normalmente previsti, una durata periodo più breve nei parametri Config.Output.Heat.PwmPeriode e Config.Output.Cool.PwmPeriode

Ampiezza banda morta

Se il valore istantaneo è disturbato, la componente di disturbo si ripercuote anche sul valore di uscita. In caso di elevato guadagno del regolatore e di componente D attivo il valore di uscita può oscillare molto. Se il valore istantaneo all'interno della banda morta si colloca intorno al setpoint, la differenza di regolazione viene soppressa evitando così la reazione dell'algoritmo PID e consentendo di ridurre inutili oscillazioni del valore di uscita.

L'ampiezza della banda morta per il riscaldamento o raffreddamento non viene impostata automaticamente durante l'ottimizzazione L'ampiezza della banda morta deve essere configurata manualmente in modo corretto. La disattivazione della banda morta avviene con l'ampiezza = 0.0.

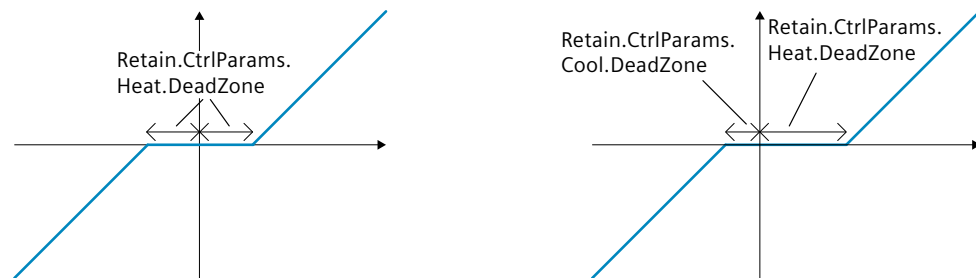
Con la banda morta attivata, può instaurarsi una differenza di regolazione permanente (divergenza tra setpoint e valore istantaneo). Questo può avere ripercussioni negative sull'esecuzione dell'ottimizzazione fine.

Se nelle impostazioni di base è stato attivato il raffreddamento, e nelle impostazioni dell'uscita è stata selezionata la commutazione dei parametri PID come metodo per il riscaldamento/raffreddamento, la banda morta si colloca tra "Setpoint - Ampiezza banda morta (riscaldamento)" e "Setpoint - Ampiezza banda morta (raffreddamento)".

Se nelle impostazioni di base è stato disattivato il raffreddamento o se viene impiegato il fattore di raffreddamento, la banda morta si colloca simmetricamente tra "Setpoint - Ampiezza banda morta (riscaldamento)" e "Setpoint - Ampiezza banda morta (riscaldamento)".

Se sono stati configurati valori diversi da 1.0 per la ponderazione del componente P o per quella del componente D le modifiche dei setpoint influenzano il valore di uscita anche all'interno della banda morta.

Le modifiche del valore istantaneo all'interno della banda morta non influenzano il valore di uscita indipendentemente dalla ponderazione.



Banda morta con raffreddamento disattivato o con fattore di raffreddamento (a sinistra) oppure con raffreddamento attivato e commutazione dei parametri PID (a destra). L'asse orizzontale x / indica la differenza di regolazione = setpoint - valore istantaneo. L'asse verticale y / indica il segnale di uscita della banda morta inoltrato all'algoritmo PID.

Ampiezza della zona di regolazione

Se il valore istantaneo abbandona la zona di regolazione intorno al setpoint, viene emesso il valore di uscita minimo o massimo. In questo modo il valore istantaneo raggiunge più velocemente il setpoint.

Se il valore istantaneo all'interno della zona di regolazione si trova intorno al setpoint, viene calcolato il valore di uscita dell'algoritmo PID.

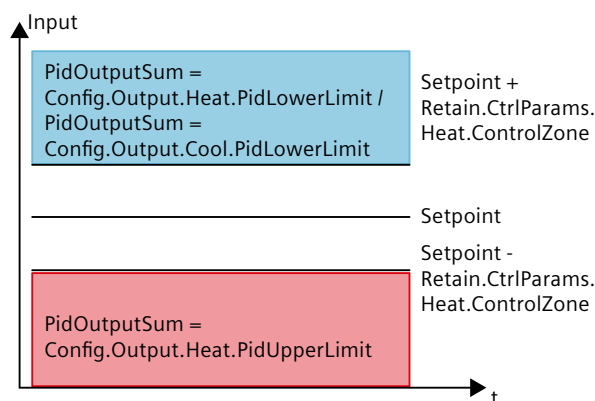
L'ampiezza della zona di regolazione per riscaldamento e raffreddamento viene impostata automaticamente soltanto durante l'ottimizzazione iniziale se come struttura del regolatore per riscaldamento o raffreddamento è stato selezionata l'opzione "PID (temperatura)".

La disattivazione della zona di regolazione avviene con l'ampiezza = 3.402822e+38.

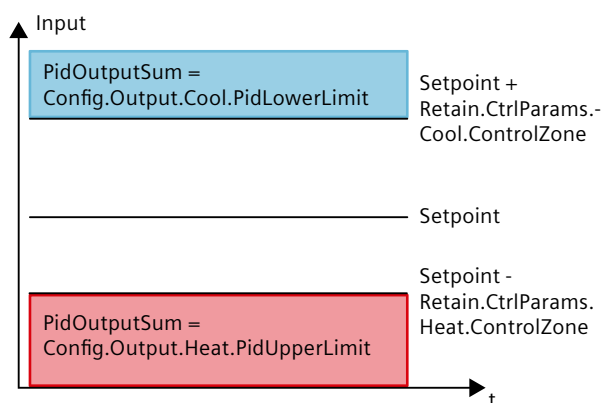
Se nelle impostazioni di base è stato disattivato il raffreddamento o se viene impiegato il fattore di raffreddamento, la zona di regolazione si colloca simmetricamente tra "Setpoint - Ampiezza zona di regolazione (riscaldamento)" e "Setpoint - Ampiezza zona di regolazione (riscaldamento)".

Se nelle impostazioni di base è stato attivato il raffreddamento, e nelle impostazioni dell'uscita è stata selezionata la commutazione dei parametri PID come metodo per il

riscaldamento/raffreddamento, la zona di regolazione si colloca tra "Setpoint - Ampiezza zona di regolazione (riscaldamento)" e "Setpoint - Ampiezza zona di regolazione (raffreddamento)".



Zona di regolazione con raffreddamento disattivato o con fattore di raffreddamento.



Zona di regolazione con raffreddamento attivato e commutazione dei parametri PID.

Regole per l'ottimizzazione

Nella casella di riepilogo "Struttura del regolatore" selezionare se debbano essere calcolati i parametri PI o PID. Le regole per l'ottimizzazione riscaldamento e raffreddamento possono essere predefinite separatamente.

- PID (temperatura)

Calcola i parametri PID durante l'ottimizzazione iniziale e fine.

L'ottimizzazione iniziale è concepita per processi di temperatura e determina un comportamento di regolazione più lento e piuttosto asintotico con oscillazioni più basse rispetto all'ottimizzazione con l'opzione "PID". L'ottimizzazione fine è identica all'ottimizzazione con l'opzione "PID".

Soltanto selezionando quest'opzione, l'ampiezza della zona di regolazione viene rilevata automaticamente durante l'ottimizzazione iniziale.

- PID

Calcola i parametri PID durante l'ottimizzazione iniziale e fine.

- PI
Calcola i parametri PI durante l'ottimizzazione iniziale e l'ottimizzazione fine.
- Personalizzato
Se, da un programma utente o dalla vista parametri, sono state impostate diverse strutture del regolatore per l'ottimizzazione iniziale e l'ottimizzazione fine, nella casella di riepilogo verrà visualizzata la voce "Personalizzato".

7.3 Messa in servizio di PID_Temp

7.3.1 Messa in servizio

La finestra di messa in servizio facilita la messa in servizio del regolatore PID. La vista delle curve consente di controllare il valore del setpoint, il valore istantaneo e il valore di uscita per riscaldamento e raffreddamento lungo l'asse temporale. La finestra di messa in servizio supporta le seguenti funzioni:

- Ottimizzazione iniziale del regolatore
- Ottimizzazione fine del regolatore
Questa opzione consente una regolazione precisa dei parametri PID.
- Controllo della regolazione in corso nella finestra delle curve
- Test del sistema regolato mediante impostazione manuale del valore di uscita PID e del valore sostitutivo
- Backup dei valori attuali dei parametri PID nel progetto offline.

Per tutte le funzione deve essere stabilito un collegamento online alla CPU.

Dai pulsanti "Controlla tutto"  o "Avvio" nella vista delle curve viene creato, se non preesistente, il collegamento online alla CPU e abilitato l'uso della finestra di messa in servizio.

Operazioni nella vista delle curve

- Nella casella di riepilogo "Tempo di campionamento", selezionare il tempo di campionamento desiderato.
I valori nella vista delle curve vengono aggiornati con il tempo di campionamento selezionato.
- Per accedere alla vista delle curve, fare clic sul simbolo "Avvio" nel gruppo Misurazione.
Viene avviata la registrazione dei valori. Nella vista delle curve vengono immessi i valori attuali per il setpoint, il valore istantaneo e i valori di uscita per riscaldamento e raffreddamento.
- Fare clic sul simbolo "Stop" per uscire dalla vista delle curve.
I valori registrati nella vista delle curve possono essere ulteriormente analizzati.

Chiudendo la finestra di messa in servizio si interrompe la registrazione nella vista delle curve e i valori registrati vengono cancellati.

7.3.2 Ottimizzazione iniziale

L'ottimizzazione iniziale rileva la risposta del processo a un gradino del valore di uscita e cerca il punto di flesso. Dall'incremento massimo e dal tempo morto del sistema regolato vengono calcolati i parametri PID ottimali. I parametri PID migliori si ottengono eseguendo l'ottimizzazione iniziale e l'ottimizzazione fine.

Più stabile è il valore istantaneo, più facilmente e con maggiore precisione si possono determinare i parametri PID. Un disturbo del valore istantaneo è accettabile fintanto che l'incremento dello stesso è significativamente maggiore rispetto al disturbo. Ciò si verifica con la maggiore probabilità nei modi "Inattivo" o "Funzionamento manuale". I parametri PID vengono salvati prima di essere nuovamente calcolati.

PID_Temp consente, in funzione della configurazione, diversi tipi di ottimizzazione iniziale:

- **Ottimizzazione iniziale riscaldamento**
Sul valore di uscita del riscaldamento viene emesso un gradino, vengono calcolati i parametri PID per il riscaldamento e nel funzionamento automatico viene poi eseguita la regolazione sul setpoint.
- **Ottimizzazione iniziale riscaldamento e raffreddamento**
Viene emesso un gradino sul valore di uscita del riscaldamento.
Non appena il valore istantaneo si avvicina al setpoint, viene emesso un gradino sul valore di uscita del raffreddamento.
Vengono calcolati i parametri PID per il riscaldamento (struttura Retain.CtrlParams.Heat) ed il raffreddamento (struttura Retain.CtrlParams.Cool) e nel funzionamento automatico viene infine eseguita la regolazione sul setpoint.
- **Ottimizzazione iniziale raffreddamento**
Viene emesso un gradino sul valore di uscita del raffreddamento.
Vengono calcolati i parametri PID per il raffreddamento e nel funzionamento automatico viene poi eseguita la regolazione sul setpoint.

L'ottimizzazione dei parametri PID per riscaldamento e raffreddamento offre un miglior comportamento di regolazione eseguendo la funzione "Ottimizzazione iniziale riscaldamento" seguita da "Ottimizzazione iniziale raffreddamento" in luogo della sola funzione "Ottimizzazione iniziale riscaldamento e raffreddamento". L'esecuzione dell'ottimizzazione in due fasi richiede tuttavia più tempo.

Requisiti generali

- L'istruzione PID_Temp viene richiamata in un OB di schedulazione orologio.
- ManualEnable = FALSE
- Reset = FALSE
- PID_Temp si trova nella modalità "Inattivo", "Funzionamento manuale" o "Funzionamento automatico".
- Il setpoint e il valore istantaneo sono compresi entro i limiti configurati (vedere la configurazione Controllo del valore istantaneo [\(Pagina 180\)](#)).

Presupposti per l'ottimizzazione iniziale riscaldamento

- La differenza tra setpoint e valore istantaneo supera il 30 % della differenza tra limite superiore e limite inferiore del valore istantaneo.
- La distanza tra setpoint e valore istantaneo supera il 50 % del setpoint.
- Il setpoint è maggiore del valore istantaneo.

Presupposti per l'ottimizzazione iniziale riscaldamento e raffreddamento

- L'uscita di raffreddamento nelle "Impostazioni di base" deve essere attivata (Config.ActivateCooling = TRUE).
- La commutazione dei parametri PID nelle "Impostazioni di base valore di uscita" deve essere attivata (Config.AdvancedCooling = TRUE).
- La differenza tra setpoint e valore istantaneo supera il 30 % della differenza tra limite superiore e limite inferiore del valore istantaneo.
- La distanza tra setpoint e valore istantaneo supera il 50 % del setpoint.
- Il setpoint è maggiore del valore istantaneo.

Presupposti per l'ottimizzazione iniziale raffreddamento

- L'uscita di raffreddamento nelle "Impostazioni di base" deve essere attivata (Config.ActivateCooling = TRUE).
- La commutazione dei parametri PID nelle "Impostazioni di base valore di uscita" deve essere attivata (Config.AdvancedCooling = TRUE).
- Una delle funzioni "Ottimizzazione iniziale riscaldamento" oppure "Ottimizzazione iniziale raffreddamento" deve essere stata eseguita correttamente (PIDSelfTune.SUT.ProcParHeatOk = TRUE). Per tutte le ottimizzazioni si sconsiglia l'impiego del medesimo setpoint.
- La differenza tra setpoint e valore istantaneo è inferiore al 5 % della differenza tra limite superiore e limite inferiore del valore istantaneo.

Procedimento

Per eseguire l'ottimizzazione iniziale procedere nel modo seguente:

1. Fare doppio clic nella navigazione del progetto sulla voce "PID_Temp > Messa in servizio".
2. Attivare il pulsante "Controlla tutto"  oppure avviare la vista delle curve.
Viene creato un collegamento online.
3. Nella casella di riepilogo "Tipo di ottimizzazione", selezionare l'opzione desiderata per l'ottimizzazione iniziale.
4. Fare clic sul simbolo "Start".
 - Viene avviata l'ottimizzazione iniziale.
 - Il campo "Stato" indica le operazioni in corso e gli eventuali errori. La barra di avanzamento indica lo stato di avanzamento dell'operazione in corso.

NOTA

Se la barra di avanzamento (variabile "Progress") non visualizza per un certo lasso di tempo alcun progresso lasciando presupporre un blocco dell'operazione di ottimizzazione, fare clic sul simbolo "Stop". Verificare la configurazione dell'oggetto tecnologico ed eventualmente riavviare l'ottimizzazione.

Risultato

Se l'ottimizzazione iniziale è stata eseguita senza messaggi di errore significa che i parametri PID sono stati ottimizzati. PID_Temp commuta sul funzionamento automatico e utilizza i parametri ottimizzati. Questi ultimi vengono mantenuti in caso di alimentazione OFF e nuovo avviamento della CPU.

Se un'ottimizzazione iniziale non è possibile, PID_Temp reagisce come configurato nel comportamento in caso di errore.

7.3.3 Ottimizzazione fine

L'ottimizzazione fine genera un'oscillazione costante e limitata del valore istantaneo. I parametri PID per il punto di lavoro vengono ottimizzati partendo dall'ampiezza e dalla frequenza dell'oscillazione. Sulla base dei risultati tutti i parametri PID vengono ricalcolati. I parametri PID risultanti dall'ottimizzazione fine mostrano in genere un comportamento pilota e di disturbo migliore rispetto ai parametri PID dell'ottimizzazione iniziale. I parametri PID migliori si ottengono eseguendo l'ottimizzazione iniziale e l'ottimizzazione fine.

PID_Temp tenta di generare automaticamente un'oscillazione che sia maggiore rispetto al fruscio del valore istantaneo. L'ottimizzazione fine viene influenzata solo minimamente dalla stabilità del valore istantaneo. I parametri PID vengono salvati prima di essere nuovamente calcolati.

PID_Temp consente, in funzione della configurazione, diversi tipi di ottimizzazione fine:

- Ottimizzazione fine riscaldamento:
PID_Temp crea, con modifiche periodiche, un'oscillazione sul valore di uscita di riscaldamento del valore istantaneo e calcola i parametri PID per il riscaldamento ().
- Ottimizzazione fine raffreddamento:
PID_Temp crea, con modifiche periodiche, un'oscillazione sul valore di uscita di raffreddamento del valore istantaneo e calcola i parametri PID per il raffreddamento ().

Offset di ottimizzazione provvisorio per regolatori di riscaldamento/raffreddamento

Se PID_Temp viene impiegato come regolatore di riscaldamento/raffreddamento (Config.ActivateCooling = TRUE), affinché possa essere creata un'oscillazione del valore istantaneo, e l'ottimizzazione fine possa essere eseguita correttamente, il valore di uscita PID (PidOutputSum) sul setpoint deve soddisfare i seguenti requisiti:

- Valore di uscita PID positivo per l'ottimizzazione fine del riscaldamento
- Valore di uscita PID negativo per l'ottimizzazione fine del raffreddamento

Se questa condizione non viene soddisfatta è possibile predefinire un offset provvisorio per l'ottimizzazione fine che verrà emesso sull'uscita attiva opposta:

- Offset per l'uscita di raffreddamento (PIDSelfTune.TIR.OutputOffsetCool) nell'ottimizzazione fine del riscaldamento.
Prima dell'avvio dell'ottimizzazione, predefinire per quest'ultima un offset negativo per il raffreddamento che sia inferiore al valore di uscita PID (PidOutputSum) sul setpoint nello stato stazionario.
- Offset per l'uscita di riscaldamento (PIDSelfTune.TIR.OutputOffsetHeat) nell'ottimizzazione fine del raffreddamento

Prima dell'avvio dell'ottimizzazione, predefinire per quest'ultima un offset positivo per il riscaldamento che sia superiore al valore di uscita PID (PidOutputSum) sul setpoint nello stato stazionario.

L'offset predefinito viene poi compensato dall'algoritmo PID consentendo così al valore istantaneo di mantenersi sul setpoint. Sulla base della grandezza dell'offset è possibile adeguare il valore di uscita PID affinché quest'ultimo soddisfi la condizione in precedenza indicata.

Per evitare maggiori sovraoscillazioni del valore istantaneo durante la predefinizione dell'offset, quest'ultimo può essere incrementato anche in più fasi.

Se PID_Temp abbandona il modo di funzionamento Ottimizzazione fine, l'offset di ottimizzazione viene resettato.

Esempio: Predefinizione dell'offset per l'ottimizzazione fine raffreddamento

- Senza offset
 - Setpoint (Setpoint) = valore istantaneo (ScaledInput) = 80 °C
 - Valore di uscita PID (PidOutputSum) = 30.0
 - Valore di uscita per il riscaldamento (OutputHeat) = 30.0
 - Valore di uscita per il raffreddamento (OutputCool) = 0.0

Un'oscillazione del valore istantaneo intorno al setpoint non può essere creata soltanto con l'uscita di raffreddamento. L'ottimizzazione fine in questo caso non riuscirebbe.
- Con offset per l'uscita di riscaldamento (PIDSelfTune.TIR.OutputOffsetHeat) = 80.0
 - Setpoint (Setpoint) = valore istantaneo (ScaledInput) = 80 °C
 - Valore di uscita PID (PidOutputSum) = -50.0
 - Valore di uscita per il riscaldamento (OutputHeat) = 80.0
 - Valore di uscita per il raffreddamento (OutputCool) = -50.0

Predefinendo un offset per l'uscita di riscaldamento, l'uscita di raffreddamento può creare soltanto un'oscillazione del valore istantaneo intorno al setpoint. L'ottimizzazione fine può essere così eseguita correttamente.

Requisiti generali

- L'istruzione PID_Temp viene richiamata in un OB di schedulazione orologio.
- ManualEnable = FALSE
- Reset = FALSE
- Il setpoint e il valore istantaneo sono compresi entro i limiti configurati (vedere la configurazione di "Impostazione del valore istantaneo").
- Il circuito di regolazione si è assestato nel punto di lavoro. Il punto di lavoro è raggiunto quando il valore istantaneo corrisponde al setpoint.
Con la banda morta attivata, può instaurarsi una differenza di regolazione permanente (divergenza tra setpoint e valore istantaneo). Questo può avere ripercussioni negative sull'esecuzione dell'ottimizzazione fine.
- Non sono previste anomalie.
- PID_Temp si trova nel modo di funzionamento inattivo, automatico o manuale.

Requisiti per l'ottimizzazione fine del riscaldamento

- Heat.EnableTuning = TRUE
- Cool.EnableTuning = FALSE
- Se PID_Temp è stato configurato come regolatore di riscaldamento/raffreddamento (Config.ActivateCooling = TRUE), sul punto di lavoro sul quale eseguire l'ottimizzazione, deve essere attiva l'uscita di riscaldamento.
PidOutputSum > 0.0 (vedere offset di ottimizzazione)

Requisiti per l'ottimizzazione fine raffreddamento

- Heat.EnableTuning = FALSE
- Cool.EnableTuning = TRUE
- L'uscita di raffreddamento deve essere attivata (Config.ActivateCooling = TRUE).
- La commutazione dei parametri PID deve essere attivata (Config.AdvancedCooling = TRUE).
- L'uscita di raffreddamento deve essere attiva sul punto di lavoro sul quale eseguire l'ottimizzazione.
PidOutputSum < 0.0 (vedere offset di ottimizzazione)

Procedura a seconda della situazione di partenza

L'ottimizzazione fine può essere avviata dai modi di funzionamento "Inattivo", "Funzionamento automatico" o "Funzionamento manuale",

All'avvio, l'ottimizzazione fine si svolge nel modo seguente:

- Funzionamento automatico con PIDSelfTune.TIR.RunIn = FALSE (preassegnazione)
Per migliorare i parametri PID esistenti mediante un'ottimizzazione, avviare l'ottimizzazione fine dal modo di funzionamento automatico.
PID_Temp esegue la regolazione con i parametri PID esistenti finché il circuito di regolazione non si assesta e finché non sono soddisfatti i requisiti per un'ottimizzazione fine. Solo in seguito viene avviata l'ottimizzazione fine.
- Inattivo, funzionamento manuale o automatico con PIDSelfTune.TIR.RunIn = TRUE
Viene effettuato il tentativo di raggiungere il setpoint con il valore di uscita minimo o massimo (regolazione a due punti):
 - Riscaldamento con valore di uscita minimo o massimo nell'ottimizzazione fine riscaldamento.
 - Raffreddamento con valore di uscita minimo o massimo nell'ottimizzazione fine raffreddamento.

Ciò può causare una maggiore sovraoscillazione. Una volta raggiunto il setpoint, ha inizio l'ottimizzazione fine.

Qualora il setpoint non possa essere raggiunto, PID_Temp non interrompe automaticamente l'ottimizzazione.

Procedura

Per eseguire l'ottimizzazione fine procedere nel seguente modo:

1. Fare doppio clic nella navigazione del progetto sulla voce "PID_Temp > Messa in servizio".
2. Attivare il pulsante "Controlla tutto"  oppure avviare la vista delle curve.
Viene creato un collegamento online.
3. Nella casella di riepilogo "Tipo di ottimizzazione", selezionare l'opzione desiderata per l'ottimizzazione fine.
4. Indicare all'occorrenza (vedere offset di ottimizzazione) un offset di ottimizzazione e attendere il raggiungimento dello stato stazionario.
5. Fare clic sul simbolo "Start".
 - Viene avviata l'ottimizzazione fine.
 - Il campo "Stato" indica le operazioni in corso e gli eventuali errori.
La barra di avanzamento indica lo stato di avanzamento dell'operazione in corso.

NOTA

Se la barra di avanzamento (variabile "Progress") non visualizza per un certo lasso di tempo alcun progresso lasciando presupporre un blocco dell'ottimizzazione, fare clic sul simbolo "Stop" nel gruppo "Tipo di ottimizzazione" Verificare la configurazione dell'oggetto tecnologico ed eventualmente riavviare l'ottimizzazione.

Soprattutto nelle fasi successive, in caso di mancato raggiungimento del setpoint, non si avrà l'interruzione automatica dell'ottimizzazione.

- "Tentativo di raggiungere il setpoint con la regolazione a due punti per il riscaldamento".
 - "Tentativo di raggiungere il setpoint con la regolazione a due punti per il raffreddamento".
-

Risultato

Se durante l'ottimizzazione fine non si verificano errori, i parametri PID sono stati ottimizzati. PID_Temp commuta sul funzionamento automatico e utilizza i parametri ottimizzati. Questi ultimi vengono mantenuti in caso di alimentazione OFF e nuovo avviamento della CPU.

Se durante l'ottimizzazione fine si sono verificati errori, PID_Temp reagisce come configurato nel comportamento in caso di errore.

7.3.4 Modo "Funzionamento manuale"

Nel seguito sono descritte le operazioni da eseguire nel modo "Funzionamento manuale" nella finestra di messa in servizio dell'oggetto tecnologico "PID_Temp".

Il funzionamento manuale è possibile anche in presenza di un errore.

Presupposti

- L'istruzione "PID_Temp" deve essere richiamata in un OB di schedulazione orologio.
- Deve esistere un collegamento online alla CPU.
- La CPU deve trovarsi nello stato di funzionamento "STOP".

Procedimento

Per testare il sistema regolato impostando un valore manuale, utilizzare l'opzione "Funzionamento manuale" della finestra di messa in servizio.

Per predefinire un valore manuale, procedere nel modo seguente:

1. Fare doppio clic nella navigazione del progetto sulla voce "PID_Temp > Messa in servizio".
2. Attivare il pulsante "Controlla tutto"  oppure avviare la vista delle curve.
Viene creato un collegamento online.
3. Attivare la casella di controllo "Funzionamento manuale" nell'area "Stato online del regolatore".
PID_Temp opera nel funzionamento manuale. L'ultimo valore di uscita attuale rimane attivo.
4. Nel campo editabile inserire il valore manuale espresso nell'unità %.
Se nelle impostazioni di base è stato attivato il raffreddamento, il valore manuale deve essere indicato come segue:
 - un valore di uscita manuale positivo, per emettere il valore sulle uscite del riscaldamento
 - un valore di uscita manuale negativo, per emettere il valore sulle uscite del raffreddamento.
5. Fare clic sul simbolo .

Risultato

Il valore manuale viene scritto nella CPU e applicato immediatamente.

Per fare in modo che il valore di uscita venga nuovamente impostato dal regolatore PID disattivare la casella di controllo "Valore manuale".

La commutazione nel funzionamento automatico è bumpless.

7.3.5 Setpoint sostitutivo

Nel seguito sono descritte le modalità di impiego del setpoint sostitutivo nella finestra di messa in servizio dell'oggetto tecnologico "PID_Temp".

Presupposti

- L'istruzione "PID_Temp" deve essere richiamata in un OB di schedulazione orologio.
- Deve esistere un collegamento online alla CPU.
- La CPU deve trovarsi nello stato di funzionamento "STOP".

Procedimento

Se si intende impiegare provvisoriamente per il setpoint un valore diverso da quello indicato nel parametro "Setpoint", (ad es. per l'ottimizzazione di uno slave in una cascata), utilizzare il setpoint sostitutivo nella finestra di messa in servizio.

Per predefinire il setpoint sostitutivo, procedere nel modo seguente:

1. Fare doppio clic nella navigazione del progetto sulla voce "PID_Temp > Messa in servizio".
2. Attivare il pulsante "Controlla tutto"  oppure avviare la vista delle curve.
Viene creato un collegamento online.
3. Attivare la casella di controllo "Subst.Setpoint" nell'area "Stato online del regolatore".
Il setpoint sostitutivo (variabile SubstituteSetpoint) viene inizializzato e utilizzato con il setpoint aggiornato per ultimo.
4. Nella casella editabile inserire il setpoint sostitutivo.
5. Fare clic sul simbolo .

Risultato

Il setpoint sostitutivo viene scritto nella CPU e applicato immediatamente.

Disattivare la casella di controllo "Subst.Setpoint", per tornare ad utilizzare come setpoint il valore nel parametro "Setpoint".

La commutazione non avviene bumpless.

7.3.6 Messa in servizio in cascata

Informazioni sulla messa in servizio in cascata con PID_Temp si trovano in Messa in servizio (Pagina 197).

7.4 Regolazione in cascata con PID_Temp

7.4.1 Introduzione

Nella regolazione in cascata vengono interconnessi più circuiti di regolazione. Gli slave ricevono il setpoint (Setpoint) dal valore di uscita (OutputHeat) del rispettivo master subordinato.

La configurazione della regolazione in cascata presuppone che il sistema regolato sia suddivisibile in più parti di circuito ciascuna caratterizzata da una propria grandezza di misura.

La predefinitone del setpoint per la grandezza di regolazione viene eseguita sul master esterno.

Il valore di uscita dello slave più interno viene inoltrato all'organo attuatore ed è pertanto attivo nel sistema regolato.

Rispetto alla regolazione ad un loop, l'impiego della regolazione in cascata offre i seguenti sostanziali vantaggi:

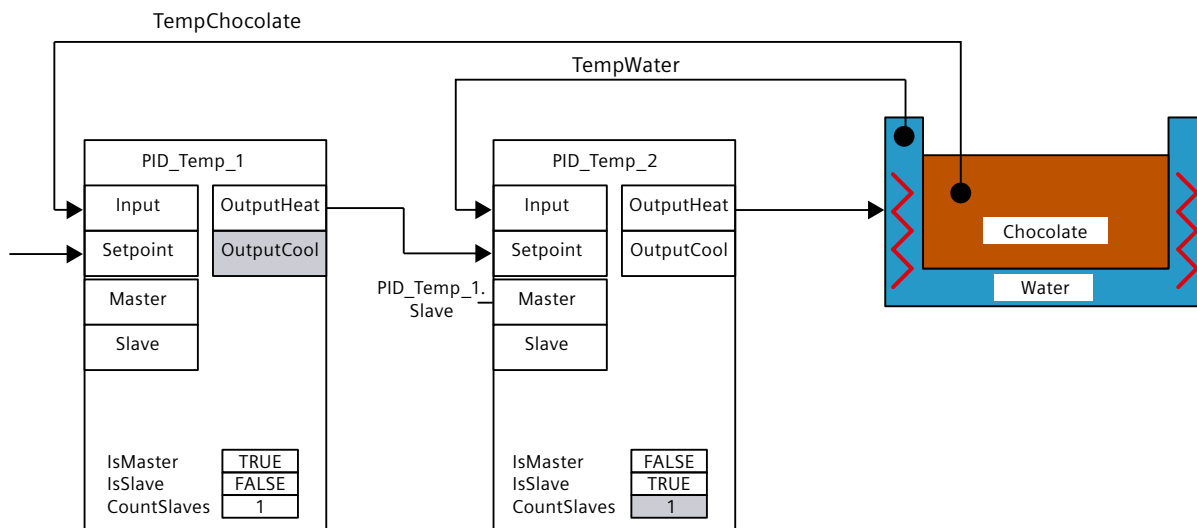
- Per effetto di circuiti di regolazione supplementari subordinati, eventuali disturbi vengono velocemente compensati con conseguente sensibile riduzione della relativa influenza sulla grandezza di regolazione. Il comportamento di disturbo viene così ridotto.
- I circuiti di regolazione subordinati svolgono un'azione di linearizzazione. Le ripercussioni negative sulla grandezza di regolazione di questa non linearità vengono in questo modo mitigate.

Soprattutto se impiegato nella regolazione in cascata, PID_Temp offre le seguenti funzionalità:

- Predefinitone di un setpoint sostitutivo
- Scambio di informazioni di stato tra master e slave (ad es. modo di funzionamento attuale)
- Diversi modi Anti-Wind-Up (reazione del master alla limitazione del relativo slave)

Esempio

Il seguente schema a blocchi illustra una regolazione in cascata con PID_Temp sulla base dell'esempio semplificato di un fusore per cioccolato:



Il master PID_Temp_1 confronta il valore istantaneo della temperatura della cioccolata (TempChocolate) con la predefinitiva personalizzata del setpoint sul rispettivo parametro Setpoint. Il valore di uscita OutputHeat forma il setpoint dello slave PID_Temp_2.

PID_Temp_2 cerca di regolare su questo setpoint il valore istantaneo della temperatura dell'acqua del bagnomaria (TempWater). Il valore di uscita di PID_Temp_2 viene inoltrato direttamente all'organo attuatore del sistema di regolazione (riscaldamento del bagnomaria) influenzando così la temperatura dell'acqua. A sua volta, la temperatura dell'acqua del bagnomaria influenza la temperatura della cioccolata.

FAQ

Ulteriori informazioni sono disponibili nelle seguenti FAQ alla sezione Siemens Industry Online Support.

- ID articolo: 103526819 (<https://support.industry.siemens.com/cs/ww/en/view/103526819>)

Vedere anche

[Creazione del programma \(Pagina 195\)](#)

7.4.2 Creazione del programma

Per la creazione del programma osservare le seguenti regole:

- Numero di istanze PID_Temp
In un OB di schedulazione orologio devono essere richiamate tante istanze PID_Temp diverse quante sono le grandezze di misura concatenate nel processo.
Nell'esempio sono disponibili due grandezze di misura concatenate: TempChocolate e TempWater. Pertanto sono necessarie due istanze PID_Temp.
- Ordine di richiamo
Il master deve essere richiamato prima dei relativi slave nello stesso OB di schedulazione orologio.
Il master più esterno nel quale viene predefinito il setpoint dell'utente, deve essere richiamato per primo.
Lo slave che riceve il setpoint dal questo master viene richiamato subito dopo, etc.
Lo slave più interno, che agisce con il suo valore di uscita sull'organo attuatore del processo, viene richiamato per ultimo.
Nell'esempio PID_Temp_1 viene richiamato prima di PID_Temp_2.
- Interconnessione delle grandezze di misura
Il master più esterno viene interconnesso con la grandezza di misura più esterna che deve essere regolata al setpoint utente.
Lo slave più interno viene interconnesso con la grandezza di misura più interna che viene influenzata direttamente dall'organo attuatore.
L'interconnessione delle grandezze di misura con PID_Temp avviene con i parametri Input o Input_PER.
Nell'esempio la grandezza di misura TempChocolate più esterna viene interconnessa con PID_Temp_1 e la grandezza di misura TempWater viene interconnessa con PID_Temp_2.
- Interconnessione del valore di uscita del master con il setpoint dello slave
Il valore di uscita (OutputHeat) di un master deve essere assegnato al setpoint (Setpoint) del rispettivo slave.
L'interconnessione può essere eseguita manualmente nell'editor di programma oppure automaticamente selezionando il master nelle impostazioni di base della finestra di ispezione dello slave.
All'occorrenza possono essere inserite funzioni del filtro o di scala proprie, finalizzate ad es. ad adeguare il campo del valore di uscita del master al campo del valore istantaneo o del setpoint dello slave.
Nell'esempio OutputHeat di PID_Temp_1 viene assegnato a Setpoint di PID_Temp_2.
- Interconnessione dell'interfaccia per lo scambio di informazioni tra master e slave
Il parametro "Slave" di un master deve essere assegnato ai parametri "Master" di tutti gli slave direttamente subordinati (che ricevono il setpoint da questo master). Per consentire l'interconnessione di un master con più slave e la visualizzazione della stessa nella finestra di ispezione dello slave nelle impostazioni di base, l'assegnazione deve essere effettuata dall'interfaccia dello slave.
L'interconnessione può essere eseguita manualmente nell'editor di programma oppure automaticamente selezionando il master nelle impostazioni di base della finestra di ispezione dello slave.

Le funzionalità Anti-Wind-Up e l'analisi dei modi operativi degli slave nel master possono essere eseguite correttamente soltanto se è avvenuta quest'interconnessione.

Nell'esempio il parametro "Slave" di PID_Temp_1 viene assegnato al parametro Master di PID_Temp_2.

Codice di programma dell'esempio nel linguaggio SCL (senza assegnazione del valore di uscita dello slave all'organo attuatore):

```
"PID_Temp_1" (Input:="TempChocolate");
"PID_Temp_2" (Input:="TempWater", Master := "PID_Temp_1".Slave,
Setpoint := "PID_Temp_1".OutputHeat);
```

Vedere anche

[Variabile ActivateRecoverMode PID_Temp \(Pagina 430\)](#)

7.4.3 Configurazione

La configurazione può essere eseguita nel programma utente, nell'editor di configurazione o nella finestra di ispezione del richiamo di PID_Temp.

In caso di impiego di PID_Temp nella regolazione in cascata, accertarsi che la configurazione delle impostazioni trattate nel seguito avvenga correttamente.

Un'istanza PID_Temp che riceve il setpoint da un master sovraordinato e inoltra a sua volta il proprio valore di uscita ad uno slave subordinato, è contemporaneamente master e slave. Per quest'istanza PID_Temp devono essere eseguite entrambe le configurazioni indicate nel seguito. E' il caso ad es. dell'istanza media PID_Temp nella regolazione in cascata con tre grandezze di misura concatenate e tre istanze PID_Temp.

Configurazione di un master

Impostazione nell'editor di configurazione o nella finestra di ispezione	Parametri DB	Descrizione
Impostazioni di base → cascata: Attivare la casella di scelta "Il regolatore è un master"	Config.Cascade.IsMaster = TRUE	Attiva questo regolatore come master in una cascata
Impostazioni di base → cascata: Numero di slave	Config.Cascade.CountSlaves	Numero di slave direttamente subordinati che ricevono il setpoint da questo master.
Impostazioni di base → parametri di ingresso/uscita: Selezione del valore di uscita (riscaldamento) = OutputHeat	Config.Output.Heat.Select = 0	Il master impiega soltanto i parametri di uscita OutputHeat. OutputHeat_PWM e OutputHeat_PER sono disattivati.
Impostazioni di base → parametri di ingresso/uscita: Disattivare la casella di scelta "Attiva raffreddamento"	Config.ActivateCooling = FALSE	In un master il raffreddamento deve essere disattivato.
Impostazioni dell'uscita → Limiti e graduazione di scala del valore di uscita → OutputHeat / OutputCool: Limite inferiore del valore di uscita (riscaldamento), Limite superiore del valore di uscita (riscaldamento), Valore inferiore di uscita riportato in sca-	Config.Output.Heat.PidLowerLimit, Config.Output.Heat.PidUpperLimit, Config.Output.Heat.LowerScaling, Config.Output.Heat.UpperScaling	Se l'assegnazione del parametro OutputHeat del master al parametro Setpoint dello slave avviene senza l'impiego di una funzione di scala propria, può essere necessario un adeguamento dei limiti del valore di uscita e della graduazione di scala dell'uscita del master al campo del valore istantaneo e del setpoint dello slave.

Impostazione nell'editor di configurazione o nella finestra di ispezione	Parametri DB	Descrizione
la (riscaldamento), Valore superiore di uscita riportato in scala (riscaldamento)		
Questa variabile non è disponibile nella finestra di ispezione o nella vista delle funzioni dell'editor di configurazione. La variabile può essere modificata nella vista parametri dell'editor di configurazione.	Config.Cascade.AntiWindUpMode	Il modo Anti-Wind-Up determina le modalità di trattamento del componente I di questo master qualora gli slave direttamente subordinati raggiungano i limiti del valore di uscita. È possibile quanto segue: • AntiWindUpMode = 0: La funzionalità AntiWindUp è disattivata. Il master non reagisce alla limitazione del rispettivo slave. • AntiWindUpMode = 1 (default): Il componente I del master viene ridotto nella proporzione degli "Slave nella limitazione" rispetto al "Numero di slave". Ciò consente di ridurre le ripercussioni della limitazione sul comportamento di regolazione. • AntiWindUpMode = 2: Il componente I del master viene mantenuto fintantoché lo slave si trova nella limitazione.

Configurazione di uno slave

Impostazione nell'editor di configurazione o nella finestra di ispezione	Parametri DB	Descrizione
Impostazioni di base → cascata: Attivare la casella di scelta "Il regolatore è uno slave"	Config.Cascade.IsSlave = TRUE	Attiva questo regolatore come slave in una cascata

7.4.4 Messa in servizio

Dopo la compilazione ed il caricamento del programma è possibile avviare la messa in servizio della regolazione in cascata.

Iniziare la messa in servizio (esecuzione di un'ottimizzazione o di una commutazione nel funzionamento automatico con i parametri PID) con lo slave più interno e proseguire verso l'esterno fino al raggiungimento dell'ultimo master.

Nell'esempio sovrastante la messa in servizio si avvia con PID_Temp_2 e prosegue poi con PID_Temp_1.

Ottimizzazione dello slave

L'ottimizzazione di PID_Temp richiede un setpoint costante. Per l'ottimizzazione di uno slave attivare pertanto il rispettivo setpoint sostitutivo (variabili SubstituteSetpoint e SubstituteSetpointOn) oppure commutare il relativo master nel funzionamento manuale con il valore manuale corrispondente. In questo modo si garantisce che il setpoint dello slave rimanga costante durante l'ottimizzazione.

Ottimizzazione del master

Affinché un master sia in grado di influenzare il processo o di eseguire l'ottimizzazione, tutti gli slave che seguono devono trovarsi nel funzionamento automatico ed avere disattivato il setpoint sostitutivo. Tramite l'interfaccia per lo scambio di informazioni tra master e slave (parametro Master e parametro Slave), un master effettua l'analisi di queste condizioni e segnala lo stato attuale alle variabili AllSlaveAutomaticState e NoSlaveSubstituteSetpoint. Nell'editor per la messa in servizio vengono emessi i messaggi di stato corrispondenti.

Messaggio di stato nell'editor per la messa in servizio del master	Parametri DB del master	Rimedio
Uno o più slave non si trovano nel funzionamento automatico.	AllSlaveAutomaticState = FALSE, NoSlaveSubstituteSetpoint = TRUE	Eseguire innanzitutto la messa in servizio di tutti gli slave che seguono. Prima di procedere all'ottimizzazione o di attivare il funzionamento manuale o automatico del master, accertarsi che siano soddisfatte le seguenti condizioni:
In uno o più slave è attivato il setpoint sostitutivo.	AllSlaveAutomaticState = TRUE, NoSlaveSubstituteSetpoint = FALSE	- Tutti gli slave che seguono si devono trovare nel funzionamento automatico (State = 3). - In tutti gli slave che seguono il setpoint sostitutivo (SubstituteSetpointOn = FALSE) deve essere disattivato.
Uno o più slave non si trovano nel funzionamento automatico e hanno attivato il setpoint sostitutivo.	AllSlaveAutomaticState = FALSE, NoSlaveSubstituteSetpoint = FALSE	

All'avvio dell'ottimizzazione iniziale o dell'ottimizzazione fine in un master, PID_Temp interrompe l'operazione ed emette un messaggio di errore con ErrorBits = DW#16#0200000, nei seguenti casi:

- Uno o più slave non si trovano nel funzionamento automatico (AllSlaveAutomaticState = FALSE)
- In uno o più slave è attivato il setpoint sostitutivo (NoSlaveSubstituteSetpoint = FALSE).

La successiva commutazione del modo di funzionamento è in funzione di ActivateRecoverMode.

7.4.5 Setpoint sostitutivo

Per la predefinitone del setpoint, PID_Temp fornisce, accanto al parametro Setpoint, un setpoint sostitutivo nella variabile SubstituteSetpoint. Questo setpoint può essere attivato con SubstituteSetpointOn = TRUE oppure selezionando la casella di scelta corrispondente nell'editor di messa in servizio.

Il setpoint sostitutivo consente l'assegnazione provvisoria del setpoint direttamente allo slave, ad es. per la messa in servizio dell'ottimizzazione.

In questo caso l'interconnessione nel programma del valore di uscita del master con il setpoint dello slave, essenziale per il normale funzionamento del regolamento in cascata, non deve essere modificata.

Affinché un master sia in grado di influenzare il processo o di eseguire l'ottimizzazione, in tutti gli slave che seguono deve essere stato disattivato il setpoint sostitutivo.

Il setpoint attualmente effettivo, così come viene utilizzato dall'algoritmo PID per il calcolo, può essere monitorato nella variabile CurrentSetpoint.

7.4.6 Modi di funzionamento e reazione agli errori

Il master o lo slave di un'istanza PID_Temp non modificano il modo di funzionamento di quest'istanza.

Se in uno dei suoi slave si verifica un errore, il master rimane nel suo attuale modo di funzionamento.

Se in uno dei suoi master si verifica un errore, lo slave rimane nel suo attuale modo di funzionamento. Tuttavia la continuazione del funzionamento dello slave dipende dal tipo di errore e dalla reazione allo stesso configurata nel master, in quanto il valore di uscita di quest'ultimo viene impiegato come setpoint dello slave:

- Se nel master è stato configurato `ActivateRecoverMode = TRUE` e l'errore impedisce il calcolo di `OutputHeat`, quest'errore non si ripercuote sullo slave.
- Se nel master è stato configurato `ActivateRecoverMode = TRUE` e l'errore impedisce il calcolo di `OutputHeat`, in funzione di `SetSubstituteOutput`, il master emette l'ultimo valore di uscita valido oppure il valore di uscita sostitutivo configurato `SubstituteOutput`. Quest'ultimo viene poi impiegato come setpoint dallo slave.

PID_Temp è preimpostato in modo da consentire che in questo caso venga emesso il valore di uscita sostitutivo 0.0 (`ActivateRecoverMode = TRUE`, `SetSubstituteOutput = TRUE`, `SubstituteOutput = 0.0`). Configurare un valore di uscita sostitutivo idoneo alla propria applicazione oppure attivare l'impiego dell'ultimo valore di uscita PID valido (`SetSubstituteOutput = FALSE`).

- Se nel master è stato configurato `ActivateRecoverMode = FALSE`, in caso di errore esso commuta nel modo di funzionamento "Inattivo" ed emette `OutputHeat = 0.0`. Lo slave impiega quindi 0.0 come setpoint.

La reazione all'errore è indicata nelle impostazioni dell'uscita nell'editor di configurazione.

7.5 Regolazione multizona con PID_Temp

Introduzione

Nella regolazione multizona viene eseguita la regolazione simultanea, a temperature diverse, di aree parziali, le cosiddette zone, di un impianto. Una peculiarità della regolazione multizona consiste nell'influenza reciproca delle zone di temperatura dovuta all'accoppiamento termico, in altri termini, il valore istantaneo di una zona può influire sul valore istantaneo di un'altra zona per effetto dell'accoppiamento termico. L'intensità di quest'influenza dipende dalla configurazione dell'impianto e dai punti di lavoro selezionati delle zone.

Esempio: impianto di estrusione con riferimento, tra l'altro, al relativo impiego nella lavorazione delle materie plastiche.

Per una lavorazione ottimale, la miscela dei materiali che attraversa l'estrusore deve essere regolata a temperature diverse. All'imboccatura dell'estrusore possono essere richieste temperature diverse che sugli ugelli di espulsione. Le singole zone di temperatura si influenzano reciprocamente per effetto dell'accoppiamento termico.

Quanto PID_Temp viene impiegato nella regolazione multizona, ciascuna zona di temperatura viene regolata da una propria istanza PID_Temp.

In caso di impiego di PID_Temp nella regolazione multizona, attenersi alle seguenti indicazioni.

Ottimizzazione iniziale separata per riscaldamento e raffreddamento

La prima messa in servizio di un impianto comincia generalmente con l'ottimizzazione iniziale al fine di eseguire una prima impostazione dei parametri PID e di regolare il punto di lavoro. Nella regolazione multizona, l'ottimizzazione iniziale viene spesso eseguita contemporaneamente per tutte le zone.

PID_Temp offre per i regolatori con raffreddamento attivato e commutazione dei parametri PID come metodo per il riscaldamento/raffreddamento (Config.ActivateCooling = TRUE, Config.AdvancedCooling = TRUE), la possibilità di eseguire l'ottimizzazione iniziale in un'unica fase per il riscaldamento ed il raffreddamento (Mode = 1, Heat.EnableTuning = TRUE, Cool.EnableTuning = TRUE).

Si sconsiglia tuttavia di ricorrere a questo tipo di ottimizzazione nella regolazione multizona con ottimizzazione iniziale simultanea di più istanze PID_Temp. Eseguire invece l'ottimizzazione iniziale del riscaldamento (Mode = 1, Heat.EnableTuning = TRUE, Cool.EnableTuning = FALSE) e l'ottimizzazione iniziale del raffreddamento (Mode = 1, Heat.EnableTuning = FALSE, Cool.EnableTuning = TRUE) in fasi separate.

L'ottimizzazione iniziale del raffreddamento deve essere avviata soltanto quando tutte le zone hanno concluso l'ottimizzazione iniziale del riscaldamento e raggiunto il punto di lavoro.

Questa procedura consente di ridurre durante l'ottimizzazione l'influenza reciproca delle zone dovuta ad accoppiamenti termici.

Adeguamento del tempo di ritardo

In caso di impiego di PID_Temp in una regolazione multizona caratterizzata da forti accoppiamenti termici tra le zone, accertarsi che l'adeguamento del tempo di ritardo per l'ottimizzazione iniziale con PIDSelfTune.SUT.AdaptDelayTime = 0 sia disattivato. Diversamente la rilevazione del tempo di ritardo può essere falsata se durante l'adeguamento del tempo di ritardo (in questa fase il riscaldamento è disattivato), l'influsso termico di altre zone impedisce il raffreddamento di questa zona.

Spegnimento provvisorio del raffreddamento

Nei regolatori con raffreddamento attivato (Config.ActivateCooling = TRUE), PID_Temp consente, nel funzionamento automatico, la disattivazione temporanea dello stesso impostando DisableCooling = TRUE.

Quest'accorgimento è volto ad evitare, durante la messa in servizio, che nel funzionamento automatico questo regolatore esegua il raffreddamento prima che i regolatori di altre zone abbiano concluso l'ottimizzazione del riscaldamento. L'accoppiamento termico tra le zone potrebbe altrimenti influenzare negativamente l'ottimizzazione.

Procedimento

Durante la messa in servizio di regolazioni multizona con accoppiamenti termici di rilievo, procedere come indicato nel seguito:

1. Per tutti i regolatori con raffreddamento attivato impostare DisableCooling = TRUE.
2. Per tutti i regolatori impostare PIDSelfTune.SUT.AdaptDelayTime = 0.
3. Predefinire i setpoint desiderati (parametro Setpoint) e avviare contemporaneamente per tutti i regolatori l'ottimizzazione iniziale del riscaldamento (Mode = 1, Heat.EnableTuning = TRUE, Cool.EnableTuning = FALSE).
4. Attendere che tutti i regolatori abbiano concluso l'ottimizzazione iniziale del riscaldamento.
5. Per tutti i regolatori con raffreddamento attivato impostare DisableCooling = FALSE.
6. Attendere che i valori istantanei di tutte le zone si siano stabilizzati e avvicinati al rispettivo setpoint.
Il mancato raggiungimento del setpoint in una determinata zona, in un lasso di tempo continuo, è riconducibile ad una potenza troppo bassa dell'organo attuatore.
7. Avviare l'ottimizzazione iniziale del raffreddamento (Mode = 1, Heat.EnableTuning = FALSE, Cool.EnableTuning = TRUE) in tutti i regolatori con raffreddamento attivato.

NOTA

Limite del limite del valore istantaneo

Se nel funzionamento automatico il raffreddamento viene disattivato con DisableCooling = TRUE, può accadere che, fintantoché DisableCooling = TRUE, il valore istantaneo superi i propri valori limite ed il setpoint. In caso di impiego di DisableCooling, monitorare i valori istantanei e se necessario intervenire.

NOTA

Regolazioni multizona

Gli accoppiamenti termici tra le zone nelle regolazioni multizona possono causare forti sovraoscillazioni, il superamento temporaneo o permanente dei valori limite, divergenze temporanee o permanenti nella regolazione durante la messa in servizio e nel funzionamento. Monitorare i valori istantanei e all'occorrenza intervenire. A seconda dell'impianto può presentarsi la necessità di seguire procedure divergenti da quelle descritte in questa sede.

Sincronizzazione di più ottimizzazioni fini

Se l'ottimizzazione fine viene avviata con PIDSelfTune.TIR.RunIn = FALSE nel funzionamento automatico, PID_Temp cerca di raggiungere il setpoint con la regolazione PID ed i parametri PID attuali. L'ottimizzazione vera e propria si avvia soltanto al raggiungimento del setpoint. Il tempo necessario al raggiungimento del setpoint può essere diverso per le singole zone di una regolazione multizona.

Se l'ottimizzazione fine deve essere eseguita simultaneamente per più zone, PID_Temp consente la sincronizzazione della stessa, realizzabile rimandando, dopo il raggiungimento del setpoint, le fasi di ottimizzazione successive.

Procedimento

Quest'accorgimento è volto a garantire che all'avvio delle fasi di ottimizzazione vere e proprie tutti i regolatori abbiano raggiunto il setpoint. Ciò consente di ridurre durante l'ottimizzazione l'influenza reciproca delle zone dovuta ad accoppiamenti termici.

Nei regolatori di cui si intende eseguire simultaneamente l'ottimizzazione fine delle zone, procedere come indicato nel seguito:

1. Per tutti i regolatori impostare `PIDSelfTune.TIR.WaitForControlIn = TRUE`.
Questi regolatori devono trovarsi nel funzionamento automatico con `PIDSelfTune.TIR.RunIn = FALSE`.
2. Predefinire i setpoint desiderati (parametro Setpoint) e avviare l'ottimizzazione fine per tutti i regolatori.
3. Attendere fino a quando `PIDSelfTune.TIR.ControlInReady = TRUE` è impostato in tutti i regolatori.
4. Per tutti i regolatori impostare `PIDSelfTune.TIR.FinishControlIn = TRUE`.

In questo modo l'ottimizzazione vera e propria si avvia contemporaneamente in tutti i regolatori.

7.6 Regolazione override con PID_Temp

Regolazione override

Nella regolazione override due o più regolatori si dividono un attuatore comune. In qualsiasi momento soltanto un regolatore accede all'attuatore e è attivo nel processo.

La scelta del regolatore preposto ad accedere all'attuatore avviene sulla base di un criterio logico. La decisione avviene spesso sulla base di un confronto dei valori di uscita di tutti i regolatori, partendo ad es. dalla selezione del massimo, sarà il regolatore con il valore di uscita più elevato ad avere accesso all'attuatore.

La scelta basata sul valore di uscita richiede che tutti i regolatori si operino nel funzionamento automatico. I regolatori che non esercitano alcuna azione sull'attuatore vengono adeguati tramite inseguimento. Questa procedura è necessaria al fine di evitare effetti windup e le conseguenti ripercussioni negative sul comportamento di regolazione, nonché per evitare la commutazione tra i regolatori.

Dalla versione V. 1.1 PID_Temp supporta la regolazione override. In questa versione il regolatore si avvale di una semplice procedura di inseguimento dei regolatori inattivi: Tramite le variabili `OverwriteInitialOutputValue` e `PIDCtrl.PIDInit` è possibile preassegnare al componente I del regolatore nel funzionamento automatico un valore tale da lasciar supporre che nell'ultimo ciclo l'algoritmo PID abbia calcolato `PidOutputSum = OverwriteInitialOutputValue` per il valore di uscita PID. A questo scopo `OverwriteInitialOutputValue` viene interconnesso al valore di uscita PID del regolatore che ha attualmente accesso all'attuatore. Impostando il bit `PIDCtrl.PIDInit` viene avviata la preassegnazione dei valori al componente I, e viene attivato il riavvio del clock del regolatore e del periodo PWM. Il calcolo successivo del valore di uscita PID nel ciclo attuale avviene sulla base dei valori preassegnati (e adeguati in tutti i regolatori), al componente I, nonché sulla base dei componenti I e P della differenza di regolazione attuale. Durante il richiamo di `PIDCtrl.PIDInit = TRUE` il componente D non è attivo e non viene quindi incluso nel calcolo del valore di uscita.

Questo comportamento è volto a garantire che il calcolo del valore di uscita PID attuale, e pertanto la scelta del regolatore preposto ad accedere all'attuatore, avvenga esclusivamente sulla base dello stato attuale del processo e dei parametri PI. Gli effetti windup nei regolatori inattivi, con conseguenti scelte errate nella logica di commutazione, vengono così evitati.

Presupposti

- PIDCtrl.PIDInit è valido soltanto se il componente I è attivo (variabili Retain.CtrlParams.Heat.Ti e Retain.CtrlParams.Cool.Ti > 0.0).
- I valori a PIDCtrl.PIDInit e OverwriteInitialOutputValue nel programma utente devono essere assegnati dall'operatore (vedere l'esempio in basso). PID_Temp non apporta automaticamente alcuna modifica a queste variabili.
- PIDCtrl.PIDInit è valido soltanto se PID_Temp si trova nel funzionamento automatico (parametro State = 3).
- Se possibile, selezionare un tempo di campionamento dell'algoritmo PID (variabili Retain.CtrlParams.Heat.Cycle e Retain.CtrlParams.Cool.Cycle) identico per tutti i regolatori e richiamare questi ultimi nello stesso OB di schedulazione orologio. Ciò garantisce che la commutazione non avvenga durante il clock del regolatore oppure durante il periodo PWM.

NOTA

Adattamento continuo dei limiti del valore di uscita

In luogo dell'inseguimento attivo qui descritto per i regolatori senza accesso all'attuatore, in altri sistemi di regolazione si realizza, in via alternativa, un adeguamento continuo dei limiti del valore di uscita.

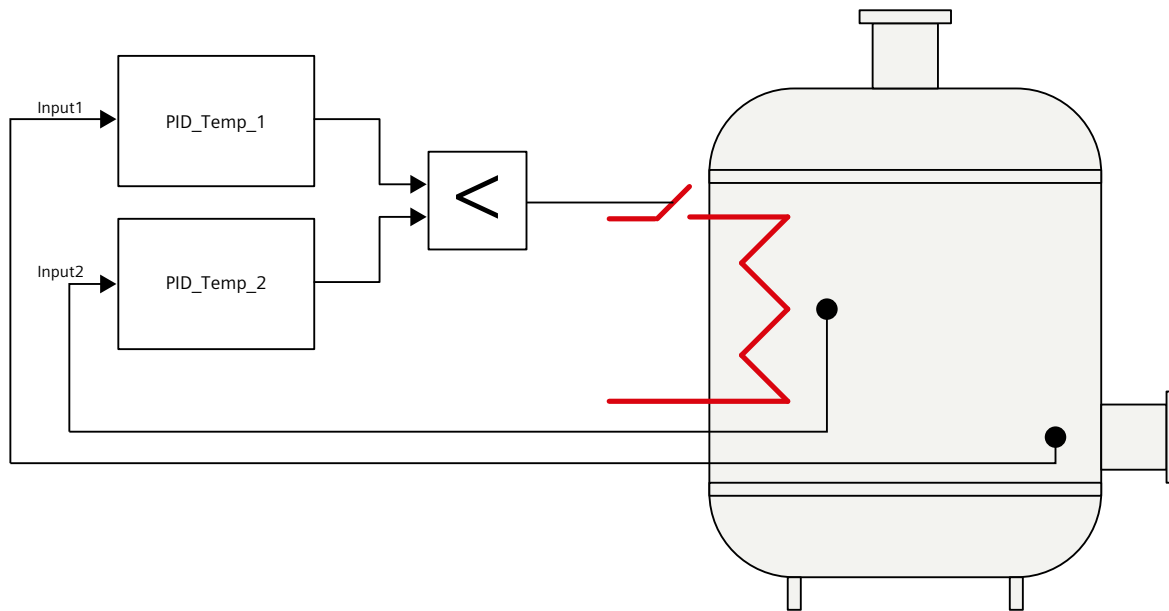
In PID_Temp questa procedura non è possibile in quanto il funzionamento automatico non supporta la modifica dei limiti del valore di uscita.

Esempio: Viene impiegato nella regolazione di una caldaia di dimensioni maggiori.

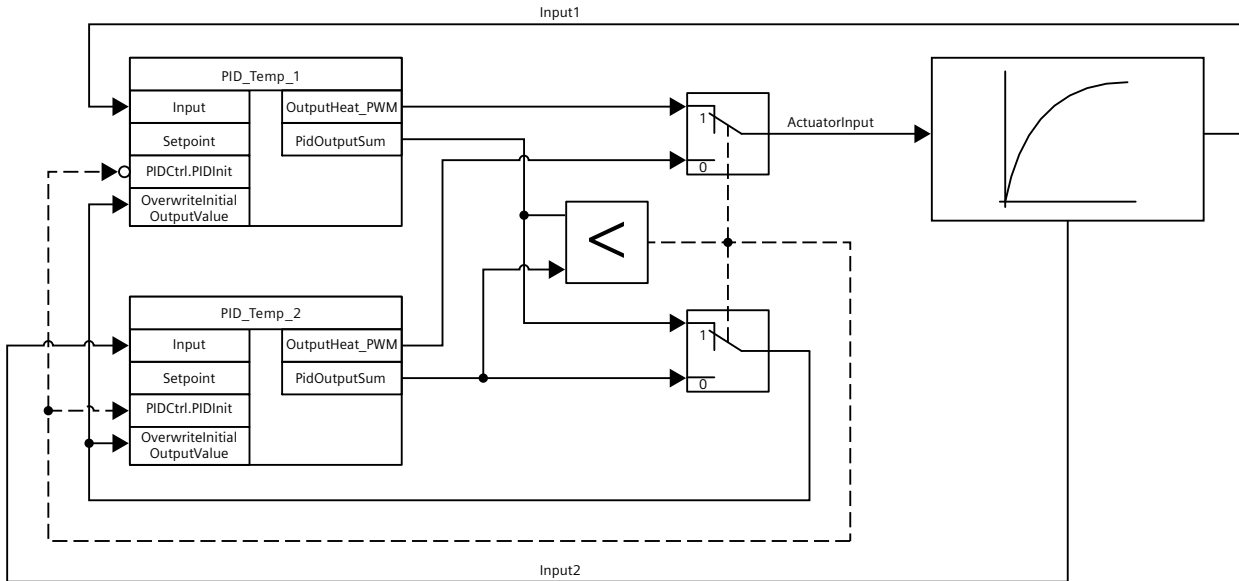
PID_Temp viene impiegato nella regolazione di una caldaia di dimensioni maggiori.

Il compito principale consiste nella regolazione della temperatura Input1. Per questo scopo trova impiego il regolatore PID_Temp_1. Inoltre, con l'ausilio del regolatore di limite PID_Temp_2, la temperatura Input2 deve essere mantenuta al di sotto del limite superiore in un punto di misura supplementare.

Questi due valori della temperatura vengono influenzati da un unico riscaldamento. Il valore di uscita del regolatore corrisponde alla capacità di riscaldamento:



Il riscaldamento viene comandato tramite il valore di uscita di PID_Temp con modulazione ampiezza impulso (parametro OutputHeat_PWM) descrivendo la variabile di programma ActuatorInput. La predefinizione del setpoint per la temperatura Input1 portata avviene nel parametro PID_Temp_1.Setpoint. Il limite superiore della temperatura per il punto di misura supplementare viene predefinito come setpoint nel parametro PID_Temp_2.Setpoint.



Entrambi i regolatori dispongono di un unico riscaldamento che funge da attuatore comune. La logica di scelta del regolatore preposto ad accedere all'attuatore avviene in questo caso tramite selezione del minimo del valore di uscita PID (nel formato reale, parametro PidOutputSum). Poiché il valore di uscita PID corrisponde alla capacità di riscaldamento, il controllo viene assegnato al regolatore che richiede la capacità di riscaldamento più bassa.

Nel funzionamento normale dell'impianto, il valore istantaneo della grandezza di regolazione principale corrisponde al setpoint. Il regolatore principale PID_Temp_1 si è assestato su un valore di uscita PID stazionario PID_Temp_1.PidOutputSum. Nel funzionamento normale, il valore istantaneo del regolatore di limite Input2 si colloca nettamente al di sotto del limite superiore, predefinito come setpoint per PID_Temp_2. Per aumentare la capacità di riscaldamento, al fine di incrementare il valore istantaneo, il regolatore di limite calcolerà un valore di uscita PID PID_Temp_2.PidOutputSum superiore a quello del regolatore principale PID_Temp_1.PidOutputSum. La selezione del minimo prevista dalla logica di commutazione, consente pertanto al regolatore principale PID_Temp_1 di continuare ad accedere all'attuatore. Inoltre con le assegnazioni PID_Temp_2.OverwriteInitialOutputValue = PID_Temp_1.PidOutputSum e PID_Temp_2.PIDCtrl.PIDInit = TRUE si garantisce l'adeguamento tramite inseguimento di PID_Temp_2.

Se Input2 si avvicina al limite superiore o lo supera ad es. a causa di un guasto, il regolatore di limite PID_Temp_2, per ridurre la capacità di riscaldamento e quindi Input2, calcola un valore di uscita PID inferiore. Se PID_Temp_2.PidOutputSum è inferiore a PID_Temp_1.PidOutputSum, tramite selezione del minimo, il regolatore di limite PID_Temp_2 accede all'attuatore e riduce la capacità di riscaldamento. Con le assegnazioni PID_Temp_1.OverwriteInitialOutputValue = PID_Temp_2.PidOutputSum e PID_Temp_1.PIDCtrl.PIDInit = TRUE si garantisce l'adeguamento tramite inseguimento di PID_Temp_1.

Nel punto di misura supplementare Input2 la temperatura scende. La temperatura della grandezza di regolazione principale Input1 si abbassa a sua volta e non può più essere mantenuta sul valore del setpoint.

Una volta eliminato il guasto, Input2 scende ulteriormente e la capacità di riscaldamento dal regolatore di limite viene ulteriormente incrementata. Non appena il regolatore principale calcola come valore di uscita una capacità di riscaldamento inferiore, l'impianto torna al funzionamento normale così da consentire al regolatore PID_Temp_1 di accedere nuovamente all'attuatore. Quest'esempio può essere realizzato con il seguente codice di programma SCL:

```
"PID_Temp_1"(Input := "Input1");
"PID_Temp_2"(Input := "Input2");
IF "PID_Temp_1".PidOutputSum <= "PID_Temp_2".PidOutputSum THEN
  "ActuatorInput" := "PID_Temp_1".OutputHeat_PWM;
  "PID_Temp_1".PIDCtrl.PIDInit := FALSE;
  "PID_Temp_2".PIDCtrl.PIDInit := TRUE;
  "PID_Temp_2".OverwriteInitialOutputValue := "PID_Temp_1".PidOutputSum;
ELSE
  "ActuatorInput" := "PID_Temp_2".OutputHeat_PWM;
  "PID_Temp_1".PIDCtrl.PIDInit := TRUE;
  "PID_Temp_2".PIDCtrl.PIDInit := FALSE;
  "PID_Temp_1".OverwriteInitialOutputValue := "PID_Temp_2".PidOutputSum;
END_IF;
```

7.7 Simulazione di PID_Temp con PLCSIM

NOTA

Simulazione con PLCSIM

La simulazione di PID_Temp con PLCSIM per la CPU S7-1200 non è supportata.

PID_Temp può essere simulata con PLCSIM solo per la CPU S7-1500 e la CPU S7-1200 G2.

Nella simulazione con PLCSIM il comportamento temporale del PLC simulato non è esattamente identico a quello di un PLC "reale". In un PLC simulato, il clock di tempo effettivo di un OB di schedulazione orologio può presentare oscillazioni maggiori rispetto ai PLC "reali".

Nella configurazione standard, PID_Temp rileva automaticamente gli intervalli tra i richiami e ne controlla le oscillazioni.

Nella simulazione di PID_Temp con PLCSIM può essere pertanto identificato un errore del tempo di campionamento (ErrorBits = DW#16#00000800).

Ciò comporta l'annullamento di ottimizzazioni in corso.

La reazione nel funzionamento automatico dipende dal valore della variabile ActivateRecoverMode.

Per evitare questo comportamento, si consiglia, nella simulazione con il PLCSIM, di configurare PID_Temp come indicato nel seguito:

- CycleTime.EnEstimation = FALSE
 - CycleTime.EnMonitoring = FALSE
 - CycleTime.Value: Assegnare a questa variabile il clock di tempo dell'OB di schedulazione orologio richiamaente espresso in secondi.
-

Applicare le funzioni di base PID

8.1 CONT_C

8.1.1 Oggetto tecnologico CONT_C

L'oggetto tecnologico CONT_C è un regolatore PID continuo per il funzionamento automatico e manuale. Esso corrisponde al blocco dati di istanza dell'istruzione CONT_C. Con l'istruzione PULSEGEN si può configurare un regolatore a impulsi.

La componente proporzionale, integrale (INT) e differenziale (DIF) si accendono in modo parallelo e possono essere singolarmente accese o spente. In questo modo si possono parametrizzare i regolatori P, I, PI, PD e PID.

S7-1500

Tutti i parametri e le variabili dell'oggetto tecnologico sono a ritenzione e possono essere modificati durante il caricamento nel dispositivo soltanto se CONT_C è stato caricato completamente.

Vedere anche

[Panoramica dei regolatori software \(Pagina 43\)](#)

[Inserire oggetti tecnologici \(Pagina 45\)](#)

[Configurare oggetti tecnologici \(Pagina 46\)](#)

[Caricamento degli oggetti tecnologici del dispositivo \(Pagina 48\)](#)

[CONT_C \(Pagina 436\)](#)

8.1.2 Configurare la differenza di regolazione CONT_C

Utilizzo del valore istantaneo di periferia

Per utilizzare il valore istantaneo nel formato periferia sul parametro di ingresso PV_PER procedere nel modo seguente:

1. Attivare la casella di controllo "Valore istantaneo IO".
2. Se il valore istantaneo è presente come grandezza fisica, rilevare il fattore e l'offset di normalizzazione in termini percentuali.

Il valore istantaneo viene quindi determinato secondo la formula seguente:

$$PV = PV\_PER \times PV\_FAC + PV\_OFF$$

Utilizzo del valore istantaneo interno

Per utilizzare il valore istantaneo in formato di virgola mobile nel parametro di ingresso PV_IN procedere nel modo seguente:

1. Disattivare la casella di controllo "Valore istantaneo IO".

Differenza di regolazione

Alle seguenti condizioni si imposta un'ampiezza di banda morta:

- Il segnale del valore istantaneo è disturbato.
- Il guadagno del regolatore è elevato.
- Il componente D è attivo.

In questo caso la rumorosità del valore istantaneo causa forti oscillazioni del valore regolante. La banda morta sopprime la rumorosità nello stato stazionario del regolatore. L'ampiezza della banda morta indica le dimensioni della banda morta. Con un'ampiezza di 0.0 la banda morta è disattivata.

Vedere anche

[Funzionamento di CONT_C \(Pagina 437\)](#)

8.1.3 Configurazione dell'algoritmo di regolazione CONT_C

In generale

Per stabilire quali componenti dell'algoritmo di regolazione sono attivate, procedere nel modo seguente:

1. Selezionare una voce dall'elenco "Struttura del regolatore".
Si possono inserire solo i parametri richiesti per la struttura del regolatore selezionata.

Componente P

1. Se la struttura del regolatore contiene un componente P, immettere il "Guadagno proporzionale".

Componente I

1. Se la struttura del regolatore contiene un componente I, indicare il tempo di integrazione.
2. Per assegnare al componente I un valore di inizializzazione, attivare la casella "Inizializza componente I" e inserire il valore di inizializzazione.
3. Per impostare il componente I in modo permanente su questo valore di inizializzazione, attivare la casella "Arresta componente I".

Componente D

1. Se la struttura del regolatore contiene un componente D, immettere il tempo derivativo, la ponderazione del componente D e il tempo di ritardo.

8.1 CONT_C

Vedere anche

[Funzionamento di CONT_C \(Pagina 437\)](#)

8.1.4 Configurazione del valore regolante CONT_C

In generale

E' possibile impostare CONT_C nella modalità manuale o automatica.

1. Per assegnare un valore regolante manuale, attivare la casella "Attiva modalità manuale".
Si può impostare sul parametro di ingresso MAN un valore regolante manuale.

Limiti del valore regolante

Al valore regolante vengono fissati un limite superiore e uno inferiore, in modo tale che egli possa accettare solo valori validi. Non è possibile disattivare tale limitazione. Il superamento dei limiti viene visualizzato mediante i parametri di uscita QLMN_HLM e QLMN_LLM.

1. Immettere un valore per il limite superiore e inferiore.
Se il valore regolante è una grandezza fisica, le unità dei limiti superiore e inferiore del valore regolante devono coincidere.

Normalizzazione

Il valore regolante può essere normalizzato per l'emissione come valore a virgola mobile e di periferia mediante un fattore e un offset secondo la formula seguente:

Valore normalizzato = valore regolante x fattore + offset

Vengono preassegnati un fattore di 1.0 e un offset di 0.0.

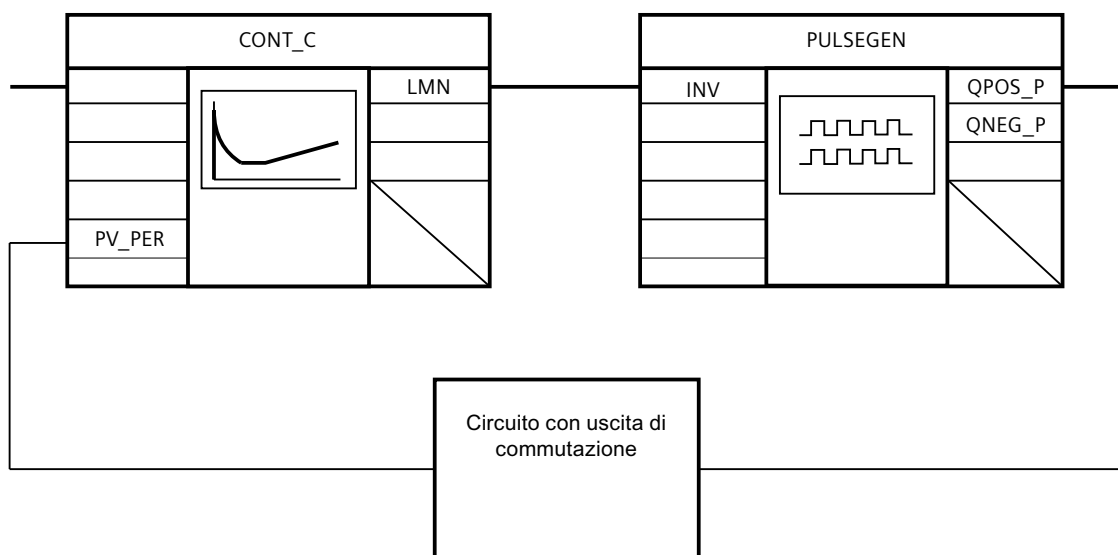
1. Immettere un valore per il fattore e l'offset.

Vedere anche

[Funzionamento di CONT_C \(Pagina 437\)](#)

8.1.5 Programmazione del regolatore a impulsi

Con il regolatore continuo CONT_C e il generatore di impulsi PULSEGEN è possibile realizzare un regolatore a valore fisso con uscita di commutazione per organi attuatori proporzionali. La seguente figura mostra l'andamento generale del segnale del circuito di regolazione.



Il regolatore continuo CONT_C genera il valore regolante LMN che viene convertito dal generatore di impulsi PULSEGEN in segnali impulso-pausa QPOS_P o QNEG_P.

Vedere anche

[PULSEGEN \(Pagina 447\)](#)

8.1.6 Messa in funzione di CONT_C

Presupposti

- L'istruzione e l'oggetto tecnologico sono stati caricati sulla CPU.

Procedimento

Per rilevare manualmente i parametri PID ottimali, procedere nel modo seguente:

1. Fare clic sul simbolo "Start".
Viene creato un collegamento online (se non è ancora attivo). Vengono registrati i valori attuali per setpoint, valore istantaneo e valore regolante.
2. Immettere nuovi parametri PID nei campi "P", "I", "D" e "Tempo di ritardo".
3. Fare clic nel gruppo "Ottimizzazione" sull'icona  "Invia parametri alla CPU".
4. Nel gruppo "Valori attuali" attivare l'opzione "Assegna setpoint".
5. Immettere un nuovo setpoint e nel gruppo "Valori attuali" fare clic sull'icona .

8.2 CONT_S

6. Se necessario, disattivare l'opzione "Funzionamento manuale".

Il regolatore funziona con i nuovi parametri PID ed esegue la regolazione in base al nuovo setpoint.

7. Controllare la qualità dei parametri PID in base all'andamento delle curve.
8. Ripetere i passi da 2 a 6 finché non si ottiene un risultato di regolazione soddisfacente.

8.2 CONT_S

8.2.1 Oggetto tecnologico CONT_S

L'oggetto tecnologico CONT_S mette a disposizione un regolatore a passi per organi attuatori con comportamento di integrazione e consente la regolazione di processi di controllo della temperatura con segnali di uscita del valore regolante binari. Questo oggetto tecnologico corrisponde al blocco dati di istanza dell'istruzione CONT_S. Il funzionamento si basa sull'algoritmo di regolazione PI del regolatore digitale. Il regolatore a passi funziona senza risposta di posizione. Sono possibili sia il funzionamento manuale che quello automatico.

S7-1500

Tutti i parametri e le variabili dell'oggetto tecnologico sono a ritenzione e possono essere modificati durante il caricamento nel dispositivo soltanto se CONT_S è stato caricato completamente.

Vedere anche

[Panoramica dei regolatori software \(Pagina 43\)](#)

[Inserire oggetti tecnologici \(Pagina 45\)](#)

[Configurare oggetti tecnologici \(Pagina 46\)](#)

[Caricamento degli oggetti tecnologici del dispositivo \(Pagina 48\)](#)

[CONT_S \(Pagina 442\)](#)

8.2.2 Configurare la differenza di regolazione CONT_S

Utilizzo del valore istantaneo di periferia

Per utilizzare il valore istantaneo nel formato periferia sul parametro di ingresso PV_PER procedere nel modo seguente:

1. Attivare la casella di controllo "Valore istantaneo IO".
2. Se il valore istantaneo è presente come grandezza fisica, rilevare il fattore e l'offset di normalizzazione in termini percentuali.

Il valore istantaneo viene quindi determinato secondo la formula seguente:

$$PV = PV\_PER \times PV\_FAC + PV\_OFF$$

Utilizzo del valore istantaneo interno

Per utilizzare il valore istantaneo in formato di virgola mobile nel parametro di ingresso PV_IN procedere nel modo seguente:

1. Disattivare la casella di controllo "Valore istantaneo IO".

Differenza di regolazione

Alle seguenti condizioni si imposta un'ampiezza di banda morta:

- Il segnale del valore istantaneo è disturbato.
- Il guadagno del regolatore è elevato.
- Il componente D è attivo.

In questo caso la rumorosità del valore istantaneo causa forti oscillazioni del valore regolante. La banda morta sopprime la rumorosità nello stato stazionario del regolatore. L'ampiezza è il parametro che indica le dimensioni della banda morta. Con un'ampiezza di 0.0 la banda morta è disattivata.

Vedere anche

[Funzionamento di CONT_S \(Pagina 443\)](#)

8.2.3 Configurazione dell'algoritmo di regolazione CONT_S

Algoritmo PI

1. Immettere la "amplificazione proporzionale" per la componente P.
2. Immettere il tempo di integrazione per il comportamento temporale della componente I.
Con un tempo di integrazione di 0.0 la componente I è spenta.

Vedere anche

[Funzionamento di CONT_S \(Pagina 443\)](#)

8.2.4 Configurazione del valore regolante CONT_S

In generale

E' possibile impostare CONT_S nella modalità manuale o automatica.

1. Per impostare un valore regolante manuale, attivare la casella "Attiva funzionamento manuale".
Sui parametri di ingresso LMNUP e LMNDN immettere un valore regolante manuale.

Generatore di impulsi

1. Impostare la durata minima dell'impulso e la durata minima di pausa.
I valori devono essere maggiori o uguali al ciclo temporale sul parametro di ingresso CYCLE. In questo modo si riduce la frequenza di commutazione.
2. Immettere il tempo di regolazione del motore.
Il valore deve essere maggiore o uguale al ciclo temporale sul parametro di ingresso CYCLE.

Vedere anche

[Funzionamento di CONT_S \(Pagina 443\)](#)

8.2.5 Messa in funzione di CONT_S

Presupposti

- L'istruzione e l'oggetto tecnologico sono stati caricati sulla CPU.

Procedimento

Per rilevare manualmente i parametri PID ottimali, procedere nel modo seguente:

1. Fare clic sul simbolo "Start".

Viene creato un collegamento online (se non è ancora attivo). Vengono registrati i valori attuali per setpoint, valore istantaneo e valore regolante.

2. Nei campi "P" e "I" inserire un nuovo coefficiente proporzionale e un nuovo tempo di integrazione.

3. Fare clic nel gruppo "Ottimizzazione" sull'icona  "Invia parametri alla CPU".

4. Nel gruppo "Valori attuali" attivare l'opzione "Assegna setpoint".

5. Immettere un nuovo setpoint e nel gruppo "Valori attuali" fare clic sull'icona .

6. Se necessario, disattivare l'opzione "Funzionamento manuale".

Il regolatore funziona con i nuovi parametri ed esegue la regolazione in base al nuovo setpoint.

7. Controllare la qualità dei parametri PID in base all'andamento delle curve.

8. Ripetere i passi da 2 a 6 finché non si ottiene un risultato di regolazione soddisfacente.

8.3 TCONT_CP

8.3.1 Oggetto tecnologico TCONT_CP

L'oggetto tecnologico TCONT_CP mette a disposizione un regolatore di temperatura continuo con generatore di impulsi. Esso corrisponde al blocco dati di istanza dell'istruzione TCONT_CP. Il funzionamento si basa sull'algoritmo di regolazione PID del regolatore digitale. Sono possibili sia il funzionamento manuale che quello automatico.

L'istruzione TCONT_CP calcola automaticamente i parametri P, I e D dei sistemi regolati in fase di "Ottimizzazione iniziale". I parametri possono essere ulteriormente ottimizzati mediante una "Ottimizzazione fine". In alternativa è possibile determinare i parametri PID manualmente.

S7-1500

Tutti i parametri e le variabili dell'oggetto tecnologico sono a ritenzione e possono essere modificati durante il caricamento nel dispositivo soltanto se TCONT_CP è stato caricato completamente.

Vedere anche

[Panoramica dei regolatori software \(Pagina 43\)](#)

[Inserire oggetti tecnologici \(Pagina 45\)](#)

[Configurare oggetti tecnologici \(Pagina 46\)](#)

[Caricamento degli oggetti tecnologici del dispositivo \(Pagina 48\)](#)

[TCONT_CP \(Pagina 456\)](#)

8.3.2 Configurare TCONT_CP

8.3.2.1 Differenza di regolazione

Utilizzo del valore istantaneo di periferia

Per utilizzare il parametro di ingresso PV_PER procedere nel modo seguente:

1. Selezionare la voce "Periferia" dall'elenco "Sorgente".
2. Scegliere il "tipo di sensore".
A seconda del tipo di sensore, il valore istantaneo viene normalizzato secondo formule diverse.
 - Standard
termocoppie; PT100/Ni100
$$PV = 0.1 \times PV\_PER \times PV\_FAC + PV\_OFFS$$
 - Climat;
PT100/Ni100
$$PV = 0.01 \times PV\_PER \times PV\_FAC + PV\_OFFS$$
 - Corrente/tensione
$$PV = 100/27648 \times PV\_PER \times PV\_FAC + PV\_OFFS$$
3. Rilevare il fattore e l'offset per la normalizzazione del valore istantaneo di periferia.

Utilizzo del valore istantaneo interno

Per utilizzare il parametro di ingresso PV_IN procedere nel modo seguente:

1. Selezionare la voce "Interno" dall'elenco "Sorgente".

Differenza di regolazione

Alle seguenti condizioni si imposta un'ampiezza di banda morta:

- Il segnale del valore istantaneo è disturbato.
- Il guadagno del regolatore è elevato.
- Il componente D è attivo.

In questo caso la rumorosità del valore istantaneo causa forti oscillazioni del valore regolante. La banda morta sopprime la rumorosità nello stato stazionario del regolatore. L'ampiezza è il parametro che indica le dimensioni della banda morta. Con un'ampiezza di 0.0 la banda morta è disattivata.

Vedere anche

[Funzionamento di TCONT_CP \(Pagina 457\)](#)

8.3.2.2 Algoritmo di regolazione

In generale

1. Immettere il "Tempo di campionamento dell'algoritmo PID".
Il tempo di campionamento del regolatore non dovrebbe superare il 10% del tempo di integrazione rilevato (TI) del regolatore.
2. Se la struttura del regolatore contiene un componente P, immettere il "Guadagno proporzionale".
Un guadagno proporzionale negativo inverte il senso di regolazione.

Componente P

Le variazioni del setpoint possono causare sovraelongazioni determinate dal componente P. Con la ponderazione del componente P è possibile stabilire fino a che punto debba agire il componente P in caso di variazione del setpoint. L'attenuazione del componente P si ottiene con una compensazione del componente I.

1. Per attenuare il componente I in caso di modifiche del setpoint, immettere la "Ponderazione del componente P".
 - 1.0: Effetto pieno del componente P in caso di modifica del setpoint
 - 0.0: Mancato effetto del componente P in caso di modifica del setpoint

Componente I

Durante la definizione dei limiti del valore regolante il componente I viene bloccato. Con una differenza di regolazione che muove il componente I in direzione del campo di regolazione interno, il componente I viene di nuovo sbloccato.

1. Se la struttura del regolatore contiene un componente I, immettere il "tempo di integrazione".
Con un tempo di integrazione di 0.0 il componente I è spento.
2. Per assegnare al componente I un valore di inizializzazione, attivare la casella "Inizializza componente I" e immettere il "valore di inizializzazione".
In caso di riavvio o COM_RST = TRUE il componente I ritorna su questo valore.

Componente D

1. Se la struttura del regolatore contiene un componente D, immettere il tempo derivativo (TD) e il coefficiente DT1 (D_F).
Se il componente D è inserito attenersi alla seguente equazione:
 $TD = 0.5 \times CYCLE \times D\_F$
Il tempo di ritardo viene calcolato secondo la formula:
 $Tempo\ di\ ritardo = TD/D\_F$

Parametrizzare il regolatore PD con il punto di lavoro

1. Immettere il tempo di integrazione 0.0.
2. Attivare la casella "Inizializza componente I".
3. Immettere il punto di lavoro come valore di inizializzazione.

Parametrizzare il regolatore P con il punto di lavoro

1. Parametrizzare un regolatore PD con il punto di lavoro.
2. Immettere il tempo di derivazione 0.0.
Il componente D viene disattivato.

Zona di regolazione

La zona di regolazione delimita il campo dei valori della differenza di regolazione. Se la differenza di regolazione si trova al di fuori di questo campo dei valori, si utilizzano i limiti del valore regolante.

All'ingresso nella zona di regolazione, il componente D acceso comporta una riduzione molto rapida della grandezza regolante. Pertanto, la zona di regolazione è opportuna solo con componente D acceso. Senza zona di regolazione, solo il componente P che si riduce ridurrebbe essenzialmente la grandezza regolante. La zona di regolazione comporta un rapido assestamento senza oscillazioni in eccesso in un senso o nell'altro, quando la grandezza regolante emessa minima o massima è distante dalla grandezza regolante necessaria per il nuovo punto di lavoro.

1. Nel gruppo "zona di regolazione" attivare la casella "Attiva".
2. Immettere nel campo "Larghezza" un valore dal quale il valore istantaneo può differire in un senso o nell'altro.

Vedere anche

[Funzionamento di TCONT_CP \(Pagina 457\)](#)

8.3.2.3 Valore regolante regolatore continuo

Limiti del valore regolante

Al valore regolante vengono fissati un limite superiore e uno inferiore, in modo tale che egli possa accettare solo valori validi. Non è possibile disattivare tale limitazione. Il superamento dei limiti viene visualizzato mediante i parametri di uscita QLMN_HLM e QLMN_LLM.

1. Immettere un valore per il limite superiore e inferiore.

Normalizzazione

Il valore regolante può essere normalizzato per l'emissione come valore a virgola mobile e di periferia mediante un fattore e un offset secondo la formula seguente:

Valore normalizzato = valore regolante x fattore + offset

Vengono preassegnati un fattore di 1.0 e un offset di 0.0.

1. Immettere un valore per il fattore e l'offset.

Generatore di impulsi

Per un regolatore continuo il generatore di impulsi deve essere spento.

1. Nel gruppo "generatore di impulsi" disattivare la casella "Attiva".

Vedere anche

[Funzionamento di TCONT_CP \(Pagina 457\)](#)

8.3.2.4 Valore regolante regolatore ad impulsi

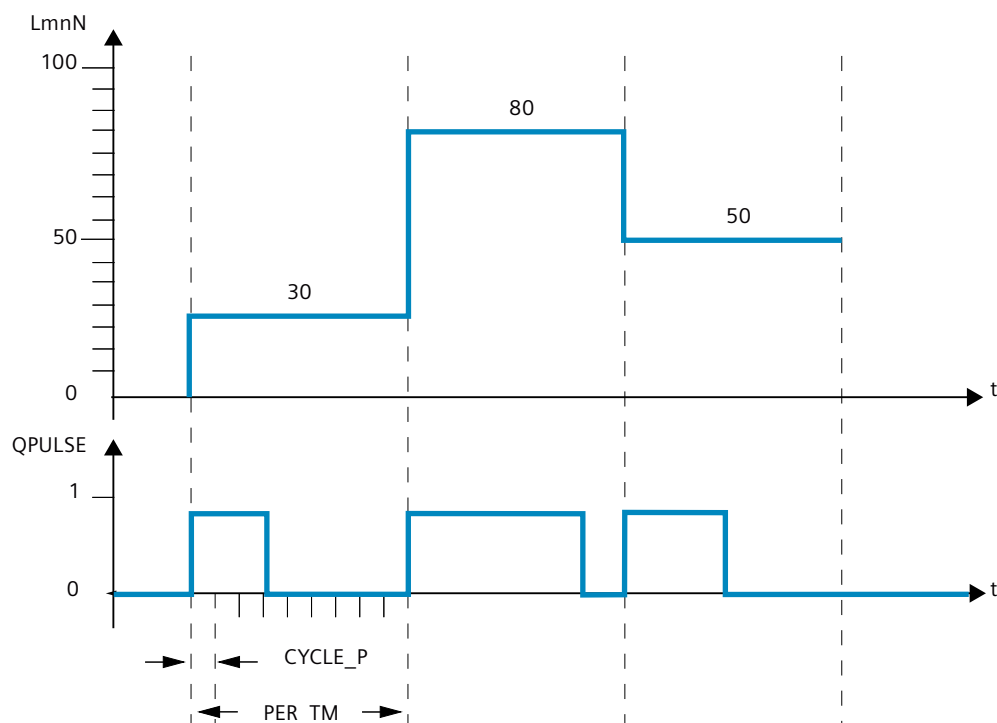
Generatore di impulsi

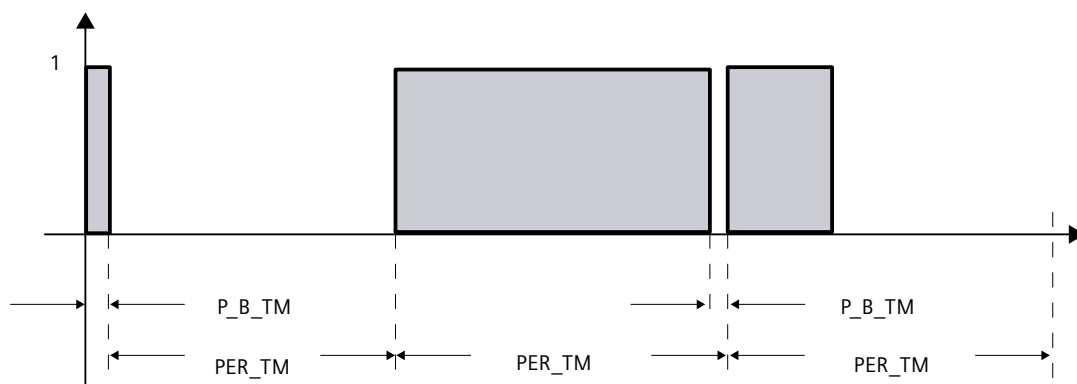
Il valore regolante analogico (LmnN) può essere emesso mediante una modulazione di larghezza di impulso sul parametro di uscita QPULSE come una sequenza di impulsi.

Per utilizzare il generatore di impulsi, procedere nel modo seguente:

1. Nel gruppo "Generatore di impulsi" attivare la casella "Attiva".
2. Immettere il "Tempo di campionamento del generatore di impulsi", la "Durata minima di impulso/durata di pausa" e la "Durata per periodo".

I grafici seguenti illustrano la correlazione tra "Tempo di campionamento del generatore di impulsi" (CYCLE_P), la "Durata minima di impulso/durata della pausa" (P_B_TM) e la "Durata per periodo" (PER_TM):





Tempo di campionamento del generatore di impulsi

Il tempo di campionamento del generatore di impulsi deve coincidere con lo scatto dell'OB di schedulazione orologio richiamato. La durata dell'impulso generato è sempre pari a un multiplo a cifra tonda di questo valore. Per una risoluzione sufficientemente precisa del valore regolante dovrebbe valere il rapporto seguente:

$$\text{CYCLE\_P} \leq \text{PER\_TM}/50$$

Durata minima di impulso / durata della pausa

Mediante la durata minima di impulso / durata della pausa si evitano tempi brevi di accensione e spegnimento sull'organo attuatore. Un impulso inferiore a P_B_TM viene represso.

Si raccomandano i valori $P\_B\_TM \leq 0.1 \times \text{PER\_TM}$.

Durata del periodo

La durata del periodo non dovrebbe superare il 20% del tempo di integrazione rilevato del regolatore (TI):

$$\text{PER\_TM} \leq \text{TI}/5$$

Esempio di effetto dei parametri CYCLE_P, CYCLE e PER_TM:

Durata del periodo PER_TM = 10 s

Tempo di campionamento algoritmo PID CYCLE = 1 s

Tempo di campionamento generatore di impulsi CYCLE_P = 100 ms.

Ogni secondo viene calcolato un nuovo valore regolante, ogni 100 ms avviene il confronto tra il valore regolante e la lunghezza di impulso o durata finora emessa.

- Quando un impulso viene emesso, vi sono 2 possibilità:
 - il valore regolante calcolato è superiore alla lunghezza di impulso/PER_TM finora registrata. L'impulso viene prolungato.
 - il valore regolante calcolato è inferiore o pari alla lunghezza di impulso/PER_TM finora registrata. Nessun segnale di impulso viene più emesso.
- Quando nessun impulso viene emesso, vi sono 2 possibilità:
 - Il valore (100% - valore regolante calcolato) è superiore alla lunghezza di pausa/PER_TM finora registrata. La pausa viene prolungata.
 - Il valore (100% - valore regolante calcolato) è inferiore o pari alla lunghezza di pausa/PER_TM finora registrata. Un segnale di impulso viene emesso.

Vedere anche

[Funzionamento di TCONT_CP \(Pagina 457\)](#)

[Funzionamento del generatore di impulsi \(Pagina 466\)](#)

8.3.3 Messa in funzione di TCONT_CP

8.3.3.1 Ottimizzazione TCONT_CP

Possibilità di impiego

L'ottimizzazione di regolazione può essere utilizzata per un processo di riscaldamento o raffreddamento puri del tipo di circuito I. Il blocco, tuttavia, può essere utilizzato anche per circuiti di ordine superiore, cioè del tipo II o III.

I parametri PU/PID vengono automaticamente rilevati e impostati. La progettazione del regolatore prevede una reazione ottimale al comportamento di disturbo. I parametri che ne risultano potrebbero causare sovraoscillazioni dal 10 % al 40 % dell'altezza del gradino nel caso di valori di setpoint.

Fasi dell'ottimizzazione di regolazione

Nell'ottimizzazione di regolazione si susseguono singole fasi che si possono leggere sul parametro PHASE .

PHASE = 0

Non avviene alcuna ottimizzazione. TCONT_CP lavora in modalità automatica o manuale.

Durante PHASE = 0 adoperarsi affinché il sistema regolato soddisfi i requisiti di un'ottimizzazione.

Al termine dell'ottimizzazione TCONT_CP commuta di nuovo in PHASE = 0.

PHASE = 1

TCONT_CP è pronto per l'ottimizzazione. PHASE = 1 può essere avviato solo quando i requisiti per un'ottimizzazione sono soddisfatti

Durante PHASE = 1 vengono rilevati i seguenti valori:

- Disturbo del valore istantaneo NOISE_PV
- Incremento iniziale PVDTO
- Valore medio del valore regolante
- Tempo di campionamento algoritmo PID CYCLE
- Tempo di campionamento del generatore di impulsi CYCLE_P

PHASE = 2

Nella fase 2 il punto di flesso del valore istantaneo viene cercato con un valore regolante costante. Questo evita che il punto di flesso venga rilevato troppo presto a causa della rumorosità del PV.

Nel regolatore a impulsi il PV viene mediato nel corso di N cicli di impulsi e in seguito fornito alla parte di regolazione. In quest'ultima viene eseguita un'ulteriore mediazione del PV: all'inizio la mediazione è disattivata, ovvero viene sempre eseguita solo in 1 ciclo. Finché la rumorosità supera un dato valore, il numero di cicli viene raddoppiato.

Vengono rilevati il periodo e l'ampiezza della rumorosità. Solo se, durante il periodo previsto, il gradiente rimane sempre inferiore all'incremento iniziale, la ricerca del punto di flesso viene interrotta e la fase 2 si conclude. TU e T_P_INF vengono tuttavia calcolati nel punto di flesso effettivo.

L'ottimizzazione viene conclusa solo se sussistono anche le seguenti condizioni:

1. Il valore istantaneo è distante dal punto di flesso di un valore superiore a $2 \cdot \text{NOISE\_PV}$.
2. Il valore istantaneo ha superato del 20 % il punto di flesso.

NOTA

In caso di eccitazione tramite gradino del setpoint, l'ottimizzazione termina al più tardi quando il valore istantaneo ha eseguito almeno il 75 % del gradino del setpoint (SP_INT-PV0) (vedere più sotto).

8.3 TCONT_CP

PHASE = 3, 4, 5

Le fasi 3, 4 e 5 durano rispettivamente un ciclo.

Nella fase 3 i parametri PI/PID validi prima dell'ottimizzazione vengono salvati e i parametri di processo calcolati.

Nella fase 4 i nuovi parametri PI/PID vengono calcolati.

Nella fase 5 viene calcolato il nuovo valore regolante e assegnato al sistema regolato.

PHASE = 7

Il tipo di circuito viene verificato nella fase 7, in quanto dopo l'ottimizzazione TCONT_CP commuta sempre nella modalità automatica. Il funzionamento automatico inizia con $LMN = LMN0 + 0.75 \cdot TUN\_DLMN$ come valore regolante. La verifica del tipo di circuito avviene in **modalità automatica** con i parametri di regolazione calcolati di nuovo e termina al più tardi $0,35 \cdot TA$ (tempo di compensazione) dopo il punto di flesso. Se l'ordine di processo differisce fortemente dal valore stimato, i parametri di regolazione vengono calcolati di nuovo e lo STATUS_D sovrastimato di 1, altrimenti i parametri di regolazione rimangono immutati.

La modalità di ottimizzazione è quindi terminata e il TCONT_CP si trova di nuovo in PHASE = 0. Dal parametro STATUS_H è possibile riconoscere se l'ottimizzazione si è conclusa con successo.

Interruzione anticipata dell'ottimizzazione

Nelle fasi 1, 2 oppure 3 è possibile, resettando $TUN\_ON = FALSE$, interrompere l'ottimizzazione senza che vengano calcolati nuovi parametri. Il regolatore si avvia in funzionamento automatico con $LMN = LMN0 + TUN\_DLMN$. Se prima dell'ottimizzazione il regolatore era impostato sul funzionamento manuale, viene emesso il valore regolante manuale precedente.

Se l'ottimizzazione viene interrotta nelle fasi 4, 5 oppure 7 mediante $TUN\_ON = FALSE$ vengono mantenuti i parametri fino a quel momento calcolati.

8.3.3.2 Requisiti per l'ottimizzazione

Comportamento transitorio di assestamento

Il processo deve presentare un comportamento transitorio di assestamento stabile, ritardato ed asintotico.

Dopo un gradino della grandezza regolante, il valore istantaneo deve passare a uno stato stazionario. Sono quindi esclusi i processi che presentano un comportamento oscillatorio anche senza regolazione e i sistemi regolati senza compensazione (integratore nel sistema regolato).

AVVERTENZA

Può causare la morte, ferite gravi o ingenti danni materiali.

In fase di ottimizzazione, il parametro MAN_ON non è attivo. Ciò può causare valori indesiderati - o eccessivi - del valore regolante o di quello istantaneo.

Il valore regolante viene predefinito tramite l'ottimizzazione. Per interrompere l'ottimizzazione impostare innanzitutto TUN_ON = FALSE. In questo modo anche MAN_ON torna ad essere effettivo.

Garanzia di uno stato iniziale quasi stazionario (fase 0)

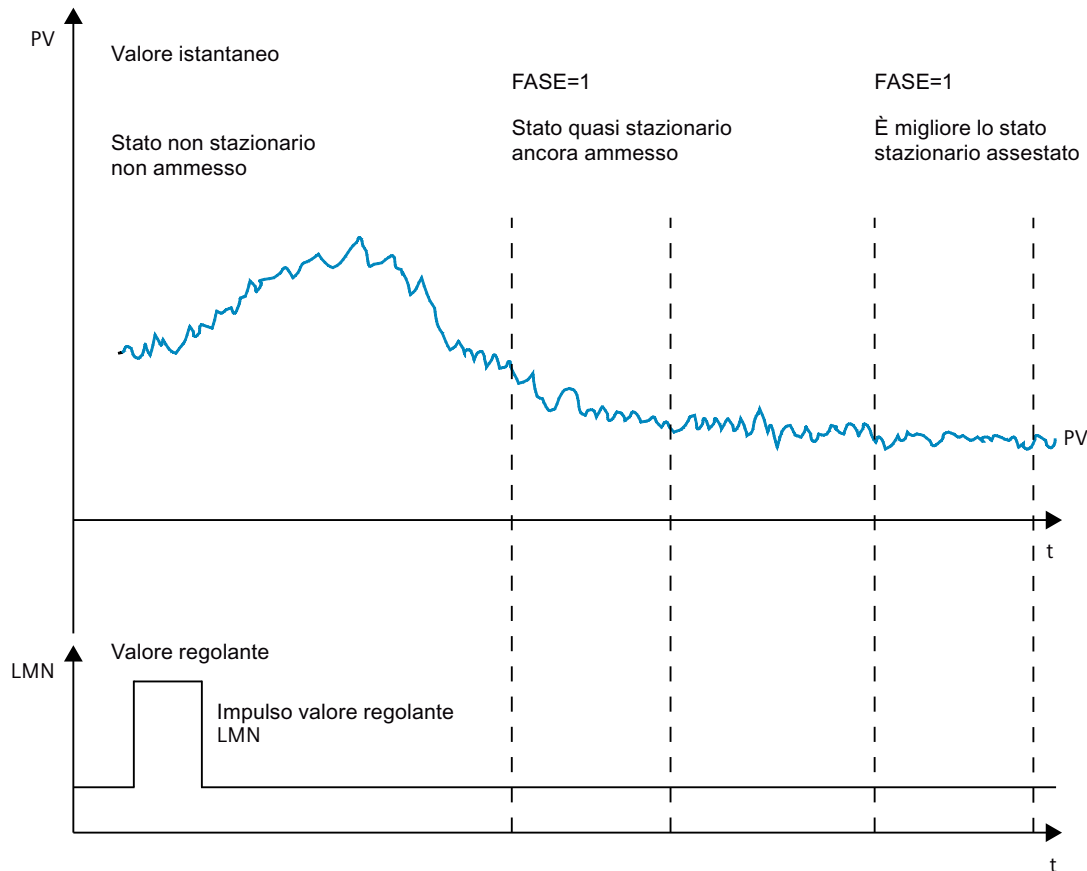
Se la grandezza regolata presentasse oscillazioni a bassa frequenza dovute p. es. al fatto che i parametri del regolatore sono stati impostati in modo errato, sarà necessario controllare il regolatore in modo manuale prima di iniziare l'ottimizzazione e attendere che le oscillazioni cessino. In alternativa è possibile ricorrere ad un regolatore PI impostato con valori gradualmente (piccola amplificazione d'anello, tempo d'integrazione lungo).

Attendere ora che venga raggiunto lo stato stazionario, ovvero che il valore istantaneo e quello regolante abbiano raggiunto lo stato di assestamento. È concesso anche uno stato transitorio di assestamento asintotico oppure una lenta deriva del valore istantaneo (stato quasi-stazionario, vedere la figura seguente). La grandezza regolante deve essere costante ovvero oscillare intorno a un valore medio costante.

NOTA

Evitare di modificare la grandezza regolante immediatamente prima dell'avvio dell'ottimizzazione. La grandezza regolante potrebbe essere modificata anche inavvertitamente, creando condizioni di prova (p. es. chiudendo la porta di un forno). Se ciò si fosse verificato, attendere almeno che il valore istantaneo si sia assestato in modo asintotico in uno stato stazionario. Per migliorare i parametri tuttavia è preferibile attendere che si concluda completamente lo stato transitorio di assestamento.

Nell'immagine seguente viene rappresentato l'assestamento nello stato stazionario:



Linearità e zona di lavoro

Il processo deve presentare un comportamento lineare nella zona di lavoro. Un comportamento non lineare si verifica ad es. con il variare dello stato di un materiale. L'ottimizzazione deve essere eseguita in una parte lineare della zona di lavoro.

Ciò sta ad indicare che, sia per l'ottimizzazione sia per la normale regolazione, gli effetti non lineari all'interno di questa area di lavoro devono essere tali da poter essere considerati trascurabili. Cambiando l'area di lavoro, è tuttavia possibile ottimizzare nuovamente il processo se l'ottimizzazione viene eseguita in un punto circoscritto della nuova zona di lavoro e se, durante l'ottimizzazione, non viene attraversata la zona non lineare.

Se sono note determinate mancanze di linearità statiche (ad es. curve caratteristiche di valvole), è comunque opportuno compensarle preventivamente con una poligonale, per rendere lineare il comportamento del processo.

Fattori di disturbo nei processi di temperatura

I fattori di disturbo, quali la trasmissione di calore tra aree vicine, non devono avere eccessiva influenza sulla temperatura dell'intero processo. In caso di ottimizzazione delle aree di un estrusore, ad es., tutte le aree devono essere riscaldate contemporaneamente.

8.3.3.3 Possibilità dell'ottimizzazione

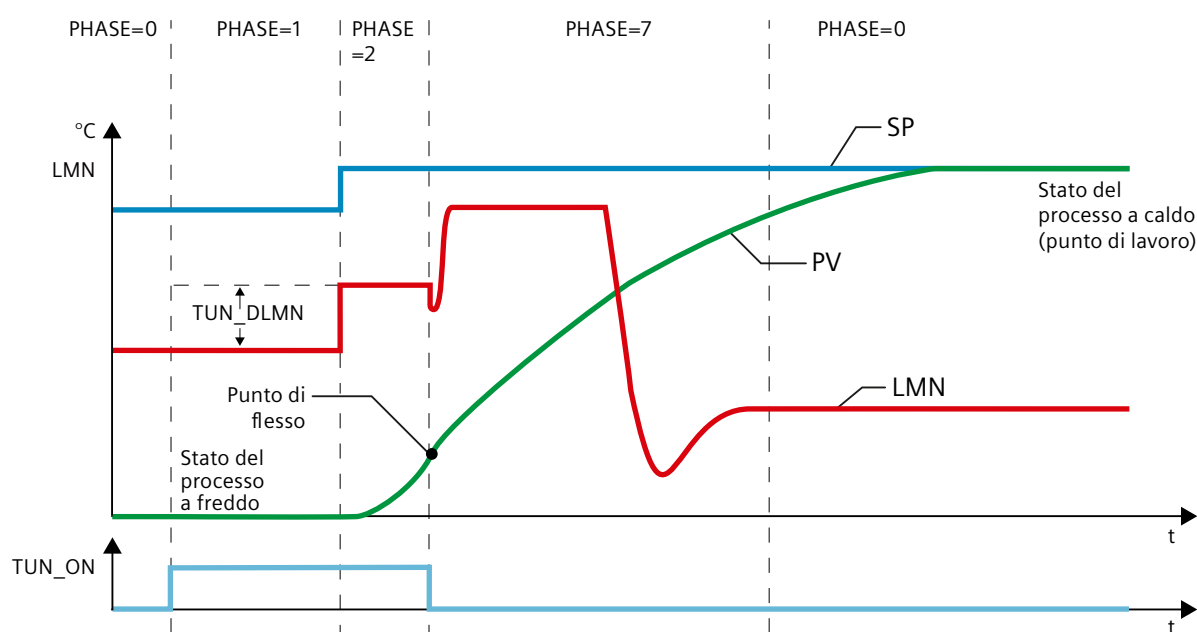
Per l'ottimizzazione vi sono le seguenti possibilità:

- Ottimizzazione iniziale
- Ottimizzazione fine
- Riottimizzazione manuale nel funzionamento di regolazione

Ottimizzazione iniziale

Durante quest'ottimizzazione, il punto di lavoro viene guidato da uno stato freddo mediante un gradino di setpoint.

Con TUN_ON = TRUE si ristabilisce la disponibilità all'ottimizzazione. Il regolatore commuta da PHASE = 0 a PHASE = 1.



La grandezza regolante di ottimizzazione ($LMN0 + TUN\_DLMN$) viene inserita in seguito alla variazione del setpoint (passaggio fase 1 -> 2). Il setpoint diventa tuttavia attivo solo una volta raggiunto il punto di flesso (solo allora si passa al funzionamento automatico).

Il delta del valore regolante (TUN_DLMN) deve essere definito dall'utente in base alla variazione del valore istantaneo consentita. Il segno davanti a TUN_DLMN deve corrispondere alla variazione prevista (considerare il senso della regolazione).

Il gradino del setpoint e TUN_DLMN devono essere adattati tra loro. Se TUN_DLMN è troppo elevato, è possibile che il punto di flesso non venga trovato entro il 75% del gradino del setpoint.

Le dimensioni di TUN_DLMN devono essere tali da consentire al valore istantaneo di raggiungere almeno il 22 % del gradino del setpoint. In caso contrario il procedimento rimane nel funzionamento di ottimizzazione (fase 2).

Rimedio: ridurre il setpoint durante la ricerca del punto di flesso.

NOTA

Nei processi con ritardi molto elevati è consigliabile impostare per l'ottimizzazione un setpoint di destinazione leggermente inferiore rispetto al punto di lavoro desiderato e controllare esattamente sia i bit di stato che PV (rischio di sovraelongazione).

Ottimizzazione solo nel campo lineare:

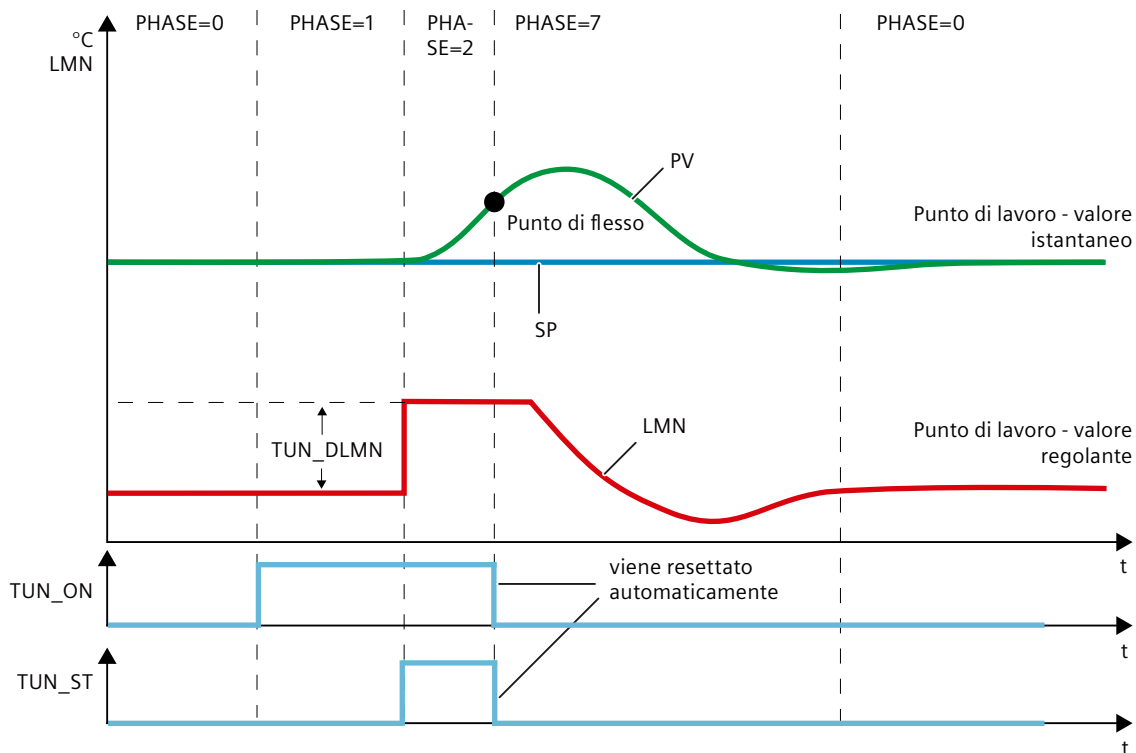
Alcuni sistemi regolati (ad es. i crogioli di fusione in zinco o magnesio) passano, poco prima del punto di lavoro, attraverso un campo non lineare (modifica dello stato del materiale).

Selezionando in modo opportuno il gradino del setpoint è possibile limitare l'ottimizzazione al campo lineare. Quando il valore istantaneo ha attraversato il 75 % del gradino del setpoint ($SP\_INT - PV0$) l'ottimizzazione viene conclusa.

Allo stesso tempo, TUN_DLMN dovrebbe essere ridotto quanto basta per far sì che il punto di flesso venga individuato prima del raggiungimento del 75 % del gradino del setpoint.

Ottimizzazione fine

Durante quest'ottimizzazione il processo con setpoint costante viene stimolato da uno gradino del valore regolante.



La grandezza regolante dell'ottimizzazione ($LMN0 + TUN\_DLMN$) viene attivata impostando il bit di start TUN_ST (passaggio fase 1 -> 2). Se si modifica il setpoint, il nuovo valore diventa attivo quando viene raggiunto il punto di flesso (solo allora viene attivato il funzionamento automatico).

Il delta del valore regolante (TUN_DLMN) deve essere definito dall'utente in base alla variazione del valore istantaneo consentita. Il segno davanti a TUN_DLMN deve corrispondere alla variazione prevista (tenere in considerazione il senso della regolazione).

ATTENZIONE

In caso di eccitazione mediante TUN_ST non si ha disattivazione di sicurezza al 75 %. L'ottimizzazione termina quando viene raggiunto il punto di flesso. Nei circuiti con forte rumorosità è tuttavia possibile che il punto di flesso venga nettamente superato.

Riottimizzazione manuale nel funzionamento di regolazione

Per fare in modo che non si verifichino sovraelongazioni nel setpoint attenersi alle misure indicate nel seguito:

- Adeguamento della zona di regolazione
- Ottimizzazione del comportamento pilota
- Attenuazione dei parametri di regolazione
- Modifica dei parametri di regolazione

8.3.3.4 Risultato dell'ottimizzazione

La cifra a sinistra di STATUS_H indica lo stato di ottimizzazione

STATUS_H	Risultato
0	Default ovvero non sono (ancora) stati trovati nuovi parametri del regolatore
10000	Sono stati trovati parametri adeguati per il regolatore
2xxxx	Sono stati trovati parametri di regolazione tramite valori stimati; verificare il comportamento di regolazione o cercare il messaggio di diagnostica STATUS_H e ripetere l'ottimizzazione del regolatore.
3xxxx	Si sono verificati errori di servizio; cercare il messaggio di diagnostica STATUS_H e ripetere l'ottimizzazione del regolatore.

Nella fase 1 sono già stati verificati i tempi di campionamento CYCLE e CYCLE_P.

Nel TCONT_CP vengono aggiornati i parametri di regolazione seguenti:

- P (coefficiente proporzionale GAIN)
- I (tempo di integrazione TI)
- D (tempo derivativo TD)
- Ponderazione del componente P PFAC_SP
- Coefficiente DT1 (D_F)
- Zona di regolazione on/off CONZ_ON
- Larghezza della zona di regolazione CON_ZONE

La zona di regolazione viene attivata solo se il tipo di circuito (I e II) e il regolatore PID sono adatti (CONZ_ON = TRUE).

In funzione di PID_ON viene utilizzato il regolatore PI o PID. I vecchi parametri del regolatore vengono memorizzati e possono essere riattivati tramite UNDO_PAR. Inoltre vengono salvati un set di parametri PI e un set di parametri PID nelle strutture PI_CON e PID_CON. Tramite LOAD_PID e l'impostazione di PID_ON è possibile scegliere anche in seguito i parametri PI o PID ottimizzati.

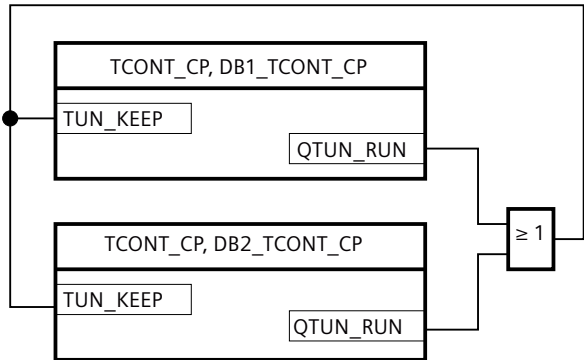
8.3.3.5 Ottimizzazione parallela dei canali del regolatore

Zone adiacenti (elevato accoppiamento termico)

Se la temperatura è regolata da due o più regolatori, ad es. su un pannello (ovvero due riscaldamenti e due valori istantanei misurati con forte accoppiamento termico) è possibile ottimizzare parallelamente entrambi i regolatori:

- 1. Combinare tramite OR le due uscite QTUN_RUN.
- 2. Collegare i due ingressi TUN_KEEP ciascuno a un'uscita dell'elemento OR.
- 3. Avviare i due regolatori impostando contemporaneamente un gradino del valore di riferimento o TUN_ST.

La seguente figura illustra l'ottimizzazione parallela dei canali del regolatore:



Vantaggio:

Entrambi i regolatori emettono LMNO + TUN_DLMN finché non sono usciti dalla fase 2. In questo modo si evita che il regolatore che conclude prima l'ottimizzazione, modificando la propria grandezza regolante, possa alterare il risultato dell'ottimizzazione dell'altro regolatore.

ATTENZIONE
Una volta raggiunto il 75 % del gradino del valore di riferimento termina la fase 2 e viene resettata l'uscita QTUN_RUN. Il funzionamento automatico viene tuttavia attivato solo quando anche TUN_KEEP assume valore 0.

Zone adiacenti (ridotto accoppiamento termico)

In generale vale la regola che l'ottimizzazione e la successiva regolazione devono procedere allo stesso modo. Se nel funzionamento di produzione le zone vengono elaborate contemporaneamente in modo parallelo, così da presentare sempre le stesse differenze di temperatura, anche durante l'ottimizzazione si dovrebbe aumentare il livello della temperatura delle zone adiacenti in modo analogo.

Le differenze di temperatura all'inizio del tentativo non sono rilevanti perché vengono compensate da un riscaldamento iniziale ($\rightarrow$ aumento iniziale = 0).

8.3.3.6 Immagini degli errori e rimedi

Blocco degli errori di servizio

Errori operativi	Stato e rimedi	Commento
Impostazione simultanea di TUN_ON e del gradino del valore di riferimento o TUN_ST	Passaggio alla fase 1, ma senza avvio dell'ottimizzazione. - SP_INT = SP_{precedente} o - TUN_ST = FALSE	La variazione del valore di riferimento viene ripresa. Si evita così che il regolatore utilizzi il nuovo valore di riferimento e lasci inutilmente il punto di lavoro stazionario.
TUN_DLMN effettivo < 5 % (fine della fase 1)	STATUS_H = 30002 - Passaggio alla fase 0 - TUN_ON = FALSE - SP = SP_{precedente}	Interruzione dell'ottimizzazione. La variazione del valore di riferimento viene ripresa. Si evita così che il regolatore utilizzi il nuovo valore di riferimento e lasci inutilmente il punto di lavoro stazionario.

Punto di flesso non raggiunto (solo in caso di eccitazione tramite gradino del valore di riferimento)

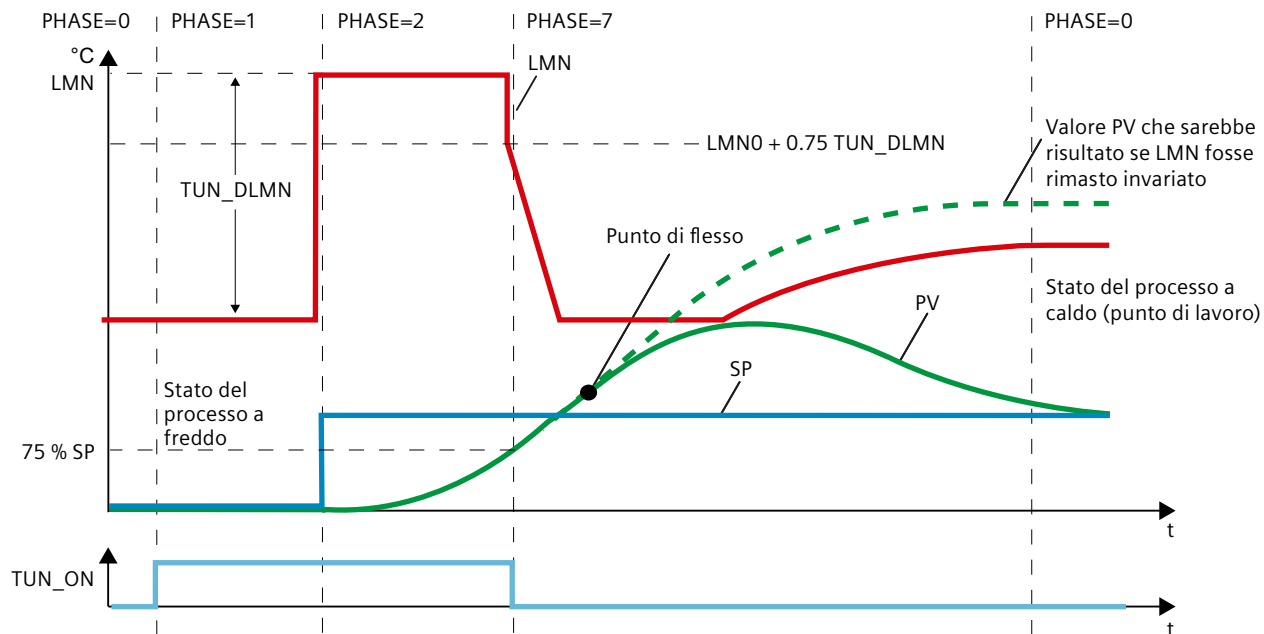
L'ottimizzazione termina al più tardi quando il valore istantaneo ha eseguito il 75 % del gradino del valore di riferimento (SP-INT-PV0). Questa situazione viene segnalata da "Punto di flesso non raggiunto" in STATUS_H (2xx2x).

È sempre valido il valore di riferimento impostato nel momento considerato. Riducendo il valore di riferimento è anche possibile fare in modo che l'ottimizzazione termini anticipatamente.

Nei circuiti di temperatura tipici, per evitare sovraelongazioni è generalmente sufficiente che l'ottimizzazione venga interrotta una volta raggiunto il 75 % del gradino del valore di riferimento. Soprattutto nei circuiti con un ritardo molto elevato ($TU/TA > 0.1$, tipo di circuito III) è tuttavia necessario procedere con molta attenzione. In caso di eccitazione eccessiva del valore regolante, il valore istantaneo può subire forti sovraelongazioni rispetto al gradino del valore di riferimento (fino al fattore 3).

Se, nei circuiti di ordine superiore, il punto di flesso è ancora molto distante dopo che è stato raggiunto il 75 % del gradino del valore di riferimento, si verificano forti sovraelongazioni. Inoltre i parametri del regolatore sono troppo poco gradualmente. Sarà quindi necessario attenuarli oppure ripetere l'operazione.

La seguente figura rappresenta la sovralongazione del valore istantaneo in caso di eccitazione eccessiva (tipo di circuito III):



Nei circuiti di temperatura l'interruzione prima del raggiungimento del punto di flesso non è critica rispetto ai parametri del regolatore.

Nell'eventuale nuovo tentativo si dovrà ridurre TUN_DLMN o aumentare il gradino del valore di riferimento.

Principio: il valore regolante per l'ottimizzazione deve essere adeguato al gradino del valore di riferimento.

Errori di stima del tempo di ritardo o dell'ordinamento

Il tempo di ritardo (STATUS_H = 2x1xx o 2x3xx) o l'ordinamento (STATUS_H = 21xxx o 22xxx) non sono stati rilevati correttamente. L'elaborazione prosegue con un valore stimato inadatto ad ottimizzare i parametri del regolatore.

Ripetere l'ottimizzazione facendo attenzione che non si verifichino disturbi nel valore istantaneo.

NOTA

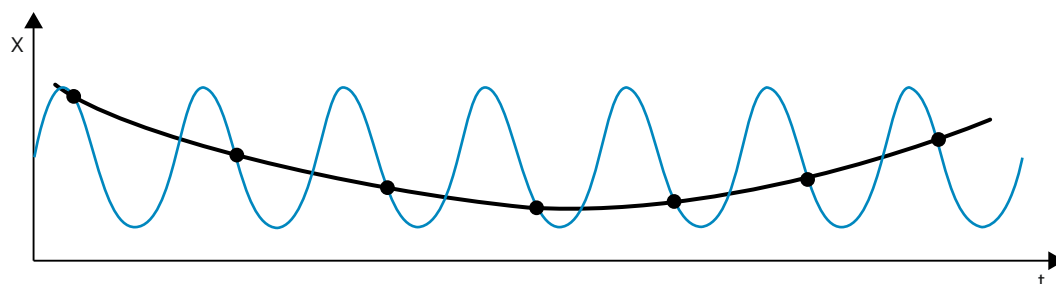
L'eccezione di un puro circuito PT1 viene segnalata anche mediante STATUS_H = 2x1xx (TU <= 3 * CYCLE). In questo caso non è necessario fare un nuovo tentativo. Se la regolazione oscilla si dovranno attenuare i parametri del regolatore.

Qualità dei segnali di misura (rumorosità di misura, disturbi a bassa frequenza)

Il risultato dell'ottimizzazione può essere stato compromesso dalla rumorosità di misura o da disturbi a bassa frequenza. Tener conto di quanto segue:

- Per la rumorosità di misura, scegliere una frequenza di campionamento piuttosto alta che bassa. Il campionamento dovrebbe essere eseguito almeno due volte entro un periodo di rumorosità del valore istantaneo. Nel funzionamento a impulsi è utile utilizzare il filtraggio del valore medio integrato. Per poterlo fare è tuttavia necessario che il valore istantaneo PV venga trasmesso all'istruzione nel ciclo di impulsi rapido. La misura della rumorosità non deve superare di oltre il 5 % la variazione del segnale utile.
- I disturbi ad alta frequenza non possono più essere filtrati ed eliminati con TCONT_CP, ma dovrebbero già essere stati eliminati nel rilevatore del valore di misura in modo da evitare il cosiddetto effetto aliasing.

La seguente figura rappresenta l'effetto aliasing in caso di tempo di campionamento troppo elevato:



- Nei disturbi a bassa frequenza è relativamente facile scegliere una frequenza di campionamento sufficientemente alta. In seguito TCONT_CP dovrà tuttavia generare un segnale di misura omogeneo utilizzando un intervallo elevato di filtraggio del valore medio. Il filtraggio del valore medio deve protrarsi per almeno due periodi di rumorosità. In questo modo, internamente al blocco si generano rapidamente tempi di campionamento elevati che compromettono la precisione dell'ottimizzazione. Una precisione sufficientemente elevata è garantita da almeno 40 periodi di rumorosità entro il punto di flesso.

Possibile rimedio in caso di nuovo tentativo:

Aumento di TUN_DLMN.

Sovraelongazione

Nelle seguenti situazioni può verificarsi una sovraelongazione:

Situazione	Causa	Rimedio
Fine dell'ottimizzazione	• Eccitazione a causa di una forte variazione del valore di riferimento rispetto al suo gradino (vedere più sopra). • Regolatore PI attivato da PID_ON = FALSE.	• Aumentare il gradino del valore di riferimento o ridurre quello del valore regolante • Se il processo consente di utilizzare un regolatore PID avviare l'ottimizzazione con PID_ON = TRUE.

8.3 TCONT_CP

Situazione	Causa	Rimedio
Ottimizzazione nella fase 7	Sono stati dapprima rilevati parametri del regolatore più gradualì (tipo di circuito III) che possono determinare una sovraelonzazione nella fase 7.	-
Regolazione	Regolatore PI e con PFAC_SP = 1.0 per il tipo di circuito I.	Se il processo consente di utilizzare un regolatore PID avviare l'ottimizzazione con PID_ON = TRUE.

8.3.3.7 Esecuzione dell'ottimizzazione iniziale

Presupposti

- L'istruzione e l'oggetto tecnologico sono caricati sulla CPU.

Procedimento

Per determinare manualmente i parametri PID ottimali alla prima messa in servizio procedere nel modo seguente:

1. Fare clic sul simbolo "Start".
Viene creato un collegamento online (se non è ancora attivo). Vengono registrati i valori attuali per setpoint, valore istantaneo e valore regolante.
2. Nella casella "Modo" selezionare la voce "Ottimizzazione iniziale".
TCONT_CP è pronto per l'ottimizzazione.
3. Nel campo "Gradino del valore regolante" immettere il valore pari al quale incrementare il valore regolante.
4. Indicare un setpoint nel campo "Setpoint". Il gradino del valore regolante diventa effettivo solo con un altro setpoint.
5. Fare clic sul simbolo  "Avvia ottimizzazione".
Viene avviata l'ottimizzazione iniziale. Viene visualizzato lo stato dell'ottimizzazione.

8.3.3.8 Esecuzione dell'ottimizzazione fine

Presupposti

- L'istruzione e l'oggetto tecnologico sono caricati sulla CPU.

Procedimento

Per determinare i parametri PID ottimali nel punto di lavoro procedere nel modo seguente:

1. Fare clic sul simbolo "Start".
Viene creato un collegamento online (se non è ancora attivo). Vengono registrati i valori attuali per setpoint, valore istantaneo e valore regolante.
2. Nella casella "Modo" selezionare la voce "Ottimizzazione fine".
TCONT_CP è pronto per l'ottimizzazione.
3. Nel campo "Gradino del valore regolante" immettere il valore pari al quale incrementare il valore regolante.
4. Fare clic sul simbolo  "Avvia ottimizzazione".
Viene avviata l'ottimizzazione fine. Viene visualizzato lo stato dell'ottimizzazione.

8.3.3.9 Interruzione dell'ottimizzazione iniziale o fine

Per interrompere un'ottimizzazione iniziale o fine fare clic sul simbolo  "Arresta l'ottimizzazione".

Se i parametri PID non sono ancora stati calcolati e salvati TCONT_CP si avvia in funzionamento automatico con $LMN = LMNO + TUN\_DLMN$. Se prima dell'ottimizzazione il regolatore era impostato sul funzionamento manuale, viene emesso il valore regolante manuale precedente.

Se i parametri PID calcolati sono già stati salvati, TCONT_CP si avvia in funzionamento automatico e funziona con i parametri PID rilevati fino a quel momento.

8.3.3.10 Riottimizzazione manuale nel funzionamento di regolazione

Per fare in modo che non si verifichino sovraelongazioni nel setpoint attenersi alle misure indicate nel seguito:

Adeguamento della zona di regolazione

Durante l'ottimizzazione TCONT_CP rileva la zona di regolazione CON_ZONE e la attiva se il tipo di circuito (tipo I e II) e il regolatore PID sono adeguati (CONZ_ON = TRUE). La zona di regolazione può essere modificata o disattivata completamente nel funzionamento di regolazione (con CONZ_ON = FALSE).

NOTA

L'attivazione della zona di regolazione nei circuiti di ordine superiore (tipo di circuito III) normalmente non è di alcun vantaggio in quanto questa zona diventerebbe più grande del campo di regolazione ottenibile con una grandezza regolante del 100%. Nemmeno l'attivazione della zona di regolazione per il regolatore PI presenta vantaggi particolari.

Prima di attivare manualmente la zona di regolazione accertarsi che l'ampiezza della zona impostata non sia troppo piccola. Se l'ampiezza della zona di regolazione impostata è insufficiente si verificano oscillazioni nell'andamento della grandezza regolante e del valore istantaneo.

Attenuazione continua del comportamento pilota con PFAC_SP

Il comportamento pilota può essere attenuato con il parametro PFAC_SP. Questo parametro stabilisce il grado di efficacia della componente P in caso di gradini del setpoint.

A prescindere dal tipo di circuito, l'ottimizzazione imposta PFAC_SP a 0.8; il valore può essere modificato successivamente. Per limitare al 2 % ca. la sovraelongazione in caso di gradini del punto di riferimento (con parametri del regolatore altrimenti corretti) sono sufficienti i seguenti valori per PFAC_SP:

	Tipo di circuito I	Tipo di circuito II	Tipo di circuito III
	Tipico circuito di temperatura	Area di passaggio	Circuito di temperatura di ordine superiore
PI	0.8	0.82	0.8
PID	0.6	0.75	0.96

Adeguare il coefficiente preimpostato (0.8) soprattutto nei seguenti casi:

- Tipo di circuito di regolazione I con PID (0.8 -> 0.6): se PFAC_SP = 0.8 i gradini del setpoint entro la zona di regolazione determinano un'ulteriore sovraelongazione di ca. 18 %.
- Tipo di circuito di regolazione III con PID (0.8 -> 0.96): i gradini del setpoint con PFAC_SP = 0.8 vengono attenuati eccessivamente. Ciò causa un dispendio di tempo nel raggiungimento del setpoint.

Attenuazione dei parametri di regolazione

Se nel circuito di regolazione chiuso si verificano oscillazioni o sono presenti sovraelongazioni dopo i gradini del setpoint, è possibile ridurre il guadagno del regolatore GAIN (ad es. all'80 % del valore originale) e aumentare il tempo di integrazione TI (ad es. al 150 % del valore originale). Se il valore regolante analogico del regolatore continuo viene convertito in segnali regolanti binari tramite un generatore di impulsi, è possibile che si verifichino lievi oscillazioni continue dovute agli effetti di quantizzazione. Queste possono essere eliminate aumentando la banda morta DEADB_W.

Modifica dei parametri di regolazione

Se si desidera modificare i parametri del regolatore, procedere come indicato nel seguito:

1. Salvare i parametri attuali con SAVE_PAR.
2. Modificare i parametri.
3. Testare il comportamento del regolatore.

Se i nuovi parametri non costituiscono un miglioramento, ripristinarli con quelli precedenti tramite UNDO_PAR.

8.3.3.11 Esecuzione manuale dell'ottimizzazione fine

Presupposti

- L'istruzione e l'oggetto tecnologico sono stati caricati sulla CPU.

Procedimento

Per rilevare manualmente i parametri PID ottimali, procedere nel modo seguente:

1. Fare clic sul simbolo "Start".
Viene creato un collegamento online (se non è ancora attivo). Vengono registrati i valori attuali per setpoint, valore istantaneo e valore regolante.
2. Nella casella "Modo" selezionare la voce "Manuale".
3. Immettere i nuovi parametri PID.
4. Fare clic nel gruppo "Ottimizzazione" sull'icona  "Invia parametri alla CPU".
5. Nel gruppo "Valori attuali" attivare l'opzione "Assegna setpoint".
6. Immettere un nuovo setpoint e nel gruppo "Valori attuali" fare clic sull'icona .
7. Se necessario, disattivare l'opzione "Funzionamento manuale".
Il regolatore funziona con i nuovi parametri PID ed esegue la regolazione in base al nuovo setpoint.
8. Controllare la qualità dei parametri PID in base all'andamento delle curve.
9. Ripetere i passi da 3 a 8 finché non si ottiene un risultato di regolazione soddisfacente.

8.4 TCONT_S

8.4.1 Oggetto tecnologico TCONT_S

L'oggetto tecnologico TCONT_S mette a disposizione un regolatore a passi per organi attuatori con comportamento di integrazione e consente la regolazione di processi di controllo della temperatura con segnali di uscita del valore regolante binari. Questo oggetto tecnologico corrisponde al blocco dati di istanza dell'istruzione TCONT_S. Il funzionamento si basa sull'algoritmo di regolazione PI del regolatore digitale. Il regolatore a passi funziona senza risposta di posizione. Sono possibili sia il funzionamento manuale che quello automatico.

S7-1500

Tutti i parametri e le variabili dell'oggetto tecnologico sono a ritenzione e possono essere modificati durante il caricamento nel dispositivo soltanto se TCONT_S è stato caricato completamente.

Vedere anche

[Panoramica dei regolatori software \(Pagina 43\)](#)

[Inserire oggetti tecnologici \(Pagina 45\)](#)

[Configurare oggetti tecnologici \(Pagina 46\)](#)

[Caricamento degli oggetti tecnologici del dispositivo \(Pagina 48\)](#)

[TCONT_S \(Pagina 478\)](#)

8.4.2 Configurare la differenza di regolazione TCONT_S

Utilizzo del valore istantaneo di periferia

Per utilizzare il parametro di ingresso PV_PER procedere nel modo seguente:

1. Selezionare la voce "Periferia" dall'elenco "Sorgente".
2. Scegliere il "Tipo di sensore".
A seconda del tipo di sensore, il valore istantaneo viene normalizzato secondo formule diverse.
 - Standard
termocoppie; PT100/Ni100
 $PV = 0.1 \times PV\_PER \times PV\_FAC + PV\_OFFS$
 - Climat;
PT100/Ni100
 $PV = 0.01 \times PV\_PER \times PV\_FAC + PV\_OFFS$
 - Corrente/tensione
 $PV = 100/27648 \times PV\_PER \times PV\_FAC + PV\_OFFS$
3. Rilevare il fattore e l'offset per la normalizzazione del valore istantaneo di periferia.

Utilizzo del valore istantaneo interno

Per utilizzare il parametro di ingresso PV_IN procedere nel modo seguente:

1. Selezionare la voce "Interno" dall'elenco "Sorgente".

Differenza di regolazione

Alle seguenti condizioni si imposta un'ampiezza di banda morta:

- Il segnale del valore istantaneo è disturbato.
- Il guadagno del regolatore è elevato.
- Il componente D è attivo.

In questo caso la rumorosità del valore istantaneo causa forti oscillazioni del valore regolante. La banda morta sopprime la rumorosità nello stato stazionario del regolatore. L'ampiezza è il parametro che indica le dimensioni della banda morta. Con un'ampiezza di 0.0 la banda morta è disattivata.

Vedere anche

[Funzionamento di TCONT_S \(Pagina 480\)](#)

8.4.3 Configurare l'algoritmo di regolazione TCONT_S

In generale

1. Immettere il "Tempo di campionamento dell'algoritmo PID".
Il tempo di campionamento del regolatore non dovrebbe superare il 10% del tempo di integrazione rilevato (TI) del regolatore.
2. Se la struttura del regolatore contiene un componente P, immettere il "Guadagno proporzionale".
Un guadagno proporzionale negativo inverte il senso di regolazione.

Componente P

Le variazioni del setpoint possono causare sovraelongazioni determinate dal componente P. Con la ponderazione del componente P è possibile stabilire fino a che punto debba agire il componente P in caso di variazione del setpoint. L'attenuazione del componente P si ottiene con una compensazione del componente I.

1. Per attenuare il componente I in caso di modifiche del setpoint, immettere la "Ponderazione del componente P".
 - 1.0: Effetto pieno del componente P in caso di modifica del setpoint
 - 0.0: Mancato effetto del componente P in caso di modifica del setpoint

8.4 TCONT_S

Componente I

1. Se la struttura del regolatore contiene un componente I, immettere il "tempo di integrazione".
Con un tempo di integrazione di 0.0 il componente I è spento.

Vedere anche

[Funzionamento di TCONT_S \(Pagina 480\)](#)

8.4.4 Configurare il valore regolante TCONT_S

Generatore di impulsi

1. Impostare la durata minima dell'impulso e la durata minima di pausa.
I valori devono essere maggiori o uguali al ciclo temporale sul parametro di ingresso CYCLE. In questo modo si riduce la frequenza di commutazione.
2. Immettere il tempo di regolazione del motore.
Il valore deve essere maggiore o uguale al ciclo temporale sul parametro di ingresso CYCLE.

Vedere anche

[Funzionamento di TCONT_S \(Pagina 480\)](#)

8.4.5 Messa in funzione di TCONT_S

Presupposti

- L'istruzione e l'oggetto tecnologico sono stati caricati sulla CPU.

Procedimento

Per rilevare manualmente i parametri PID ottimali, procedere nel modo seguente:

1. Fare clic sul simbolo "Start".
Viene creato un collegamento online (se non è ancora attivo). Vengono registrati i valori attuali per setpoint, valore istantaneo e valore regolante.
2. Immettere i nuovi parametri PI nei campi "P", "I" e "Ponderazione del componente P".
3. Fare clic nel gruppo "Ottimizzazione" sull'icona  "Invia parametri alla CPU".
4. Nel gruppo "Valori attuali" attivare l'opzione "Assegna setpoint".
5. Immettere un nuovo setpoint e nel gruppo "Valori attuali" fare clic sull'icona .
6. Se necessario, disattivare l'opzione "Funzionamento manuale".
Il regolatore funziona con i nuovi parametri ed esegue la regolazione in base al nuovo setpoint.
7. Controllare la qualità dei parametri PID in base all'andamento delle curve.
8. Ripetere i passi da 2 a 6 finché non si ottiene un risultato di regolazione soddisfacente.

Funzioni ausiliarie

9.1 Polyline

Polyline

L'istruzione Polyline fornisce una funzione con la caratteristica del tratto poligonale i cui punti di interpolazione consentono ad es. di linearizzare il comportamento di sensori non lineari.

Ulteriori informazioni

Compatibilità con CPU e FW ([Pagina 489](#))

Descrizione Polyline ([Pagina 490](#))

Funzionamento Polyline ([Pagina 493](#))

Parametro di ingresso Polyline ([Pagina 496](#))

Parametro di uscita Polyline ([Pagina 497](#))

Variabili statiche Polyline ([Pagina 497](#))

Parametro ErrorBits ([Pagina 498](#))

9.2 SplitRange

SplitRange

L'istruzione SplitRange suddivide il campo dei valori di uscita del regolatore PID in diversi sottocampi. Questi sottocampi consentono la regolazione di un processo influenzato da diversi attuatori.

Ulteriori informazioni

Compatibilità con CPU e FW ([Pagina 501](#))

Descrizione di SplitRange ([Pagina 502](#))

Parametro di ingresso SplitRange ([Pagina 504](#))

Variabili statiche SplitRange ([Pagina 505](#))

Parametro di uscita SplitRange ([Pagina 505](#))

Parametro ErrorBits ([Pagina 506](#))

9.3 RampFunction

RampFunction

L'istruzione RampFunction limita la velocità di variazione di un segnale. Un gradino del segnale sull'ingresso viene emesso come funzione di rampa del valore di uscita, ad es. per ottenere un comportamento pilota più morbido senza influenzare la risposta ai disturbi.

Ulteriori informazioni

- Compatibilità con CPU e FW ([Pagina 508](#))
- Descrizione di RampFunction ([Pagina 508](#))
- Modo di funzionamento di RampFunction ([Pagina 513](#))
- Parametro di ingresso RampFunction ([Pagina 516](#))
- Parametro di uscita RampFunction ([Pagina 516](#))
- Variabili statiche RampFunction ([Pagina 517](#))
- Parametro ErrorBits ([Pagina 518](#))

9.4 RampSoak

RampSoak

Con quest'istruzione si genera un valore di uscita che segue cronologicamente un profilo configurabile. Ogni punto del profilo è provvisto di un valore di destinazione e di un valore temporale. Durante l'esecuzione del profilo, il valore di destinazione del punto attuale viene raggiunto prima che sia decorso il valore temporale. L'istruzione può essere impiegata ad es. per rendere disponibile un profilo del setpoint per la regolazione di un processo termico.

Ulteriori informazioni

- Compatibilità con CPU e FW ([Pagina 521](#))
- Descrizione di RampSoak ([Pagina 522](#))
- Modo di funzionamento RampSoak ([Pagina 524](#))
- Parametro di ingresso RampSoak ([Pagina 539](#))
- Parametro di uscita RampSoak ([Pagina 540](#))
- Parametro di transito RampSoak ([Pagina 540](#))
- Variabili statiche RampSoak ([Pagina 541](#))
- Parametro ErrorBits ([Pagina 542](#))

9.5 Filter_PT1

Filter_PT1

L'istruzione Filter_PT1 è un elemento di trasmissione proporzionale con un ritardo di prim'ordine. È possibile utilizzare Filter_PT1 come filtro passabasso, elemento di ritardo oppure come blocco di simulazione del processo.

Ulteriori informazioni

Compatibilità con CPU e FW [\(Pagina 546\)](#)

Descrizione del filtro_PT1 [\(Pagina 547\)](#)

Modo di funzionamento Filter_PT1 [\(Pagina 553\)](#)

Parametro di ingresso Filter_PT1 [\(Pagina 555\)](#)

Parametro di uscita Filter_PT1 [\(Pagina 555\)](#)

Variabili Filter_PT1 statiche [\(Pagina 555\)](#)

Parametro ErrorBits [\(Pagina 556\)](#)

9.6 Filter_PT2

Filter_PT2

L'istruzione Filter_PT2 è un elemento di trasmissione proporzionale con un ritardo di secondo ordine. È possibile utilizzare Filter_PT2 come filtro passabasso, elemento di ritardo oppure come blocco di simulazione del processo.

Ulteriori informazioni

Compatibilità con CPU e FW [\(Pagina 559\)](#)

Descrizione del filtro_PT2 [\(Pagina 559\)](#)

Modo di funzionamento Filter_PT2 [\(Pagina 566\)](#)

Parametro di ingresso Filter_PT2 [\(Pagina 568\)](#)

Parametro di uscita Filter_PT2 [\(Pagina 568\)](#)

Variabili statiche Filter_PT2 [\(Pagina 568\)](#)

Descrizione del filtro_PT2 [\(Pagina 559\)](#)

Vedere anche

[Parametro ErrorBits \(Pagina 569\)](#)

9.7 Filter_DT1

Filter_DT1

L'istruzione Filter_DT1 è un regolatore differenziale con un ritardo di prim'ordine. Filter_DT1 può essere utilizzato come filtro passa-alto, regolatore differenziale oppure inserzione di un segnale di disturbo.

Ulteriori informazioni

[Compatibilità con CPU e FW \(Pagina 572\)](#)
[Descrizione del filtro_PT1 \(Pagina 573\)](#)
[Modo di funzionamento Filter_DT1 \(Pagina 580\)](#)
[Parametro di ingresso Filter_DT1 \(Pagina 581\)](#)
[Parametro di uscita Filter_DT1 \(Pagina 582\)](#)
[Variabili Filter_DT1 statiche \(Pagina 582\)](#)
[Parametro ErrorBits \(Pagina 583\)](#)

9.8 Filter_Universal

Filter_Universal

L'istruzione Filter_Universal è un filtro digitale configurabile dell'ordine da 1 a 10. Filter_Universal è utilizzabile come filtro passa-alto, passa-basso, passa-banda o antibanda.

Ulteriori informazioni

[Compatibilità con CPU e FW \(Pagina 586\)](#)
[Descrizione di Filter_Universal \(Pagina 586\)](#)
[Modo di funzionamento Filter_Universal \(Pagina 588\)](#)
[Parametro di ingresso Filter_Universal \(Pagina 599\)](#)
[Parametro di uscita Filter_Universal \(Pagina 599\)](#)
[Variabili statiche Filter_Universal \(Pagina 600\)](#)
[Parametro ErrorBits \(Pagina 601\)](#)

Istruzioni

10.1 PID_Compact

10.1.1 Novità di PID_Compact

PID_Compact V3.0

- **Utilizzo con S7-1200 G2**

Da PID_Compact V3.0 l'istruzione può essere utilizzata anche in un'S7-1200 G2.

Le versioni precedenti dell'istruzione non sono supportate nell'S7-1200 G2.

Se si trasferiscono dei programmi dall'S7-1200 all'S7-1200 G2, alla prima compilazione la versione di PID_Compact viene modificata in V3.0.

Tener conto delle modifiche del comportamento descritte di seguito introdotte tra la versione precedente di PID_Compact sulla CPU 1200 e PID_Compact V3.0.

- **Banda morta**

Se il valore istantaneo è disturbato, è possibile ridurre le oscillazioni inutili del valore di uscita impostando manualmente la larghezza della banda morta.

- **Limitazione attiva del componente integrale e modifica dei limiti del valore iniziale nel funzionamento automatico**

Oltre all'arresto in funzione della direzione, ora il componente I viene limitato attivamente.

Questo consente di modificare i limiti del valore iniziale nel funzionamento automatico.

- **Nuove variabili Config.OutputSelect e Retain.CtrlParams.SetByUser**

Le variabili Config.OutputSelect e Retain.CtrlParams.SetByUser sostituiscono quelle precedenti _Config.OutputSelect e _Retain.CtrlParams.SetByUser, che sono disponibili solo tramite Openness API.

Le variabili Config.OutputSelect e Retain.CtrlParams.SetByUser sono disponibili sia nel blocco dati di istanza che tramite Openness API.

I valori disponibili per le nuove variabili Config.OutputSelect e Retain.CtrlParams.SetByUser vengono acquisiti per le singole istanze dopo il passaggio alla V3. Nel caso delle multiistanze, dopo il passaggio alla V3 le nuove variabili hanno il valore standard. Le rispettive impostazioni devono essere ripristinate manualmente.

Se finora nella propria applicazione Openness sono state utilizzate _Config.OutputSelect o _Retain.CtrlParams.SetByUser, quando si passa a PID_Compact V3 le si deve sostituire con le variabili Config.OutputSelect e Retain.CtrlParams.SetByUser.

PID_Compact V2.4

- **Inizializzazione della componente integrale**

PID_Compact inizializza correttamente la componente integrale se si utilizza OverwriteInitialOutputValue con il senso di regolazione invertito.

Se OverwriteInitialOutputValue era stato utilizzato con il senso di regolazione invertito tenere presente che con PID_Compact V2.4 il segno che precede il valore di uscita cambia.

PID_Compact V2.3

- **Reazione del valore di uscita alla commutazione dal modo di funzionamento "Inattivo" al modo "Funzionamento automatico"**

La nuova opzione IntegralResetMode = 4 è stata inserita e definita come impostazione di default. Con IntegralResetMode = 4, alla commutazione dal modo di funzionamento "Inattivo" al modo "Funzionamento automatico", al componente I viene automaticamente assegnato per default un valore che ad una differenza di regolazione comporti un salto del valore di uscita con segno identico.

- **Inizializzazione del componente I nel funzionamento automatico**

L'inizializzazione del componente integrale può avere luogo nel funzionamento automatico con l'ausilio delle variabili OverwriteInitialOutputValue e PIDCtrl.PIDInit. Ciò semplifica l'impiego di PID_Compact nelle regolazioni con funzione override.

PID_Compact V2.2

- **Impiego con S7-1200**

Da PID_Compact V2.2 l'istruzione con funzionalità V2 può essere utilizzata anche su un S7-1200 a partire dalla versione firmware 4.0.

PID_Compact V2.0

- **Comportamento in caso di errori**

Il comportamento in caso di errori è stato completamente rielaborato. Ora nell'impostazione di default PID_Compact ha un comportamento più fault-tolerant. Questo comportamento viene impostato durante la copia di PID_Compact V1.X da una CPU S7-1200 ad una CPU S7-1500.

ATTENZIONE
L'impianto può subire danneggiamenti. Se si utilizza l'impostazione di default, PID_Compact rimane in funzionamento automatico se vengono superati i limiti del valore istantaneo. In seguito a ciò l'impianto può subire danneggiamenti. Configurare per il sistema regolato un comportamento da adottare in caso di errore che protegga l'impianto da danneggiamenti.

Il parametro Error indica se attualmente è presente un errore. Se l'errore non è più presente, compare Error = FALSE. Il parametro ErrorBits mostra gli errori che si sono

verificati. Con ErrorAck si confermano gli errori e gli avvisi senza dover riavviare il regolatore o cancellare il componente I. Cambiando il modo di funzionamento gli errori che non sono più presenti non vengono più cancellati.

Il comportamento in caso di errori si configura con SetSubstituteOutput e ActivateRecoverMode.

- **Valore di uscita sostitutivo**

È possibile configurare un valore di uscita sostitutivo da emettere in caso di errore.

- **Commutazione dei modi di funzionamento**

Il modo di funzionamento si predefinisce nel parametro di transito Mode e si avvia con un fronte di salita in ModeActivate. La variabile sRet.i_Mode è stata eliminata.

- **Funzionalità multiistanza**

PID_Compact può essere richiamato come DB multiistanza.

- **Comportamento all'avviamento**

Il modo di funzionamento predefinito in Mode viene avviato anche da un fronte di discesa in Reset e con l'avviamento a freddo della CPU se RunModeByStartup = TRUE.

- **Comportamento ENO**

ENO viene impostato in funzione del modo di funzionamento.

Se State = 0, ENO = FALSE.

Se State ≠ 0, ENO = TRUE.

- **Predefinizione del setpoint durante l'ottimizzazione**

L'oscillazione del setpoint consentita durante l'ottimizzazione si configura nella variabile CancelTuningLevel.

- **Campo di valori per i limiti del valore di uscita**

0.0 non deve essere compreso entro i limiti del valore di uscita.

- **Preassegnazione del componente I**

Con le variabili IntegralResetMode e OverwriteInitialOutputValue si stabilisce come preassegnare il componente I alla commutazione del modo di funzionamento da "Inattivo" a "Funzionamento automatico".

- **Inserzione di una grandezza di disturbo**

Nel parametro Disturbance si può inserire una grandezza di disturbo.

- **Predefinizione dei parametri PID**

Le seguenti preimpostazioni sono state modificate:

- Ponderazione del componente P (PWeighting) da 0.0 a 1.0
- Ponderazione del componente D (DWeighting) da 0.0 a 1.0
- Coefficiente per il ritardo di differenziazione (TdFiltRatio) da 0.0 a 0.2

- **Rinomina delle variabili**

Le variabili statiche sono state rinominate compatibilmente con PID_3Step.

PID_Compact V1.2

- **Funzionamento manuale all'avviamento della CPU**
Se all'avvio della CPU ManualEnable = TRUE, PID_Compact si avvia in funzionamento manuale. Non è necessario un fronte di salita in ManualEnable.
- **Ottimizzazione iniziale**
Se si spegne la CPU durante l'ottimizzazione iniziale, l'ottimizzazione iniziale viene riavviata all'accensione della CPU.

PID_Compact V1.1

- **Funzionamento manuale all'avviamento della CPU**
All'avvio della CPU PID_Compact commuta nel funzionamento manuale solo in presenza di un fronte di salita in ManualEnable. Senza fronte di salita PID_Compact si avvia nell'ultimo modo di funzionamento nel quale ManualEnable era FALSE.
- **Comportamento in caso di reset**
Un fronte di salita nel parametro Reset resetta gli errori e gli avvisi e cancella il componente I. Un fronte di discesa nel parametro Reset commuta nell'ultimo modo di funzionamento attivo.
- **Preassegnazione del limite superiore del valore istantaneo**
La preassegnazione di r_Pv_Hlm è stata reimpostata a 120.0.
- **Sorveglianza del tempo di campionamento**
 - Se il tempo di campionamento attuale $\geq 1,5 \times$ valore medio attuale o tempo di campionamento attuale $\leq 0,5 \times$ valore medio attuale, non vengono più emessi messaggi di errore. In funzionamento automatico è ammessa una divergenza del tempo di campionamento maggiore.
 - è compatibile con FW V2.0 o superiori.
- **Accesso alle variabili**
Le seguenti variabili possono essere utilizzate adesso nel programma utente.
 - i_Event_SUT
 - i_Event_TIR
 - r_Ctrl_Ioutv
- **Rimedio**
PID_Compact ora emette impulsi corretti se il minor tempo di inserzione è diverso dal minor tempo di disinserzione.

10.1.2 Compatibilità con CPU e FW

La tabella seguente mostra quali CPU sono compatibili con quale versione di PID_Compact.

CPU	FW	PID_Compact
S7-1200	da V4.2	V2.3 V2.2 V1.2
	V4.0 ... V4.1	V2.2 V1.2
	V3.X	V1.2 V1.1
	V2.X	V1.2 V1.1
S7-1200 G2	da V1.0	V3.0
S7-1500	da V3.1	V3.0 V2.4
	V3.0	V2.4
	Da V2.5 a V2.9	V2.4 V2.3 V2.2 V2.1 V2.0
	V2.0 e V2.1	V2.3 V2.2 V2.1 V2.0
	V1.5 ... V1.8	V2.2 V2.1 V2.0
	V1.1	V2.1 V2.0
	V1.0	V2.0

10.1.3 Tempo di elaborazione della CPU e memoria necessaria di PID_Compact dalla V2

Tempi di elaborazione della CPU

Tempi di elaborazione tipici della CPU dell'oggetto tecnologico PID_Compact dalla versione V2.0 in funzione del tipo di CPU e del modo di funzionamento per CPU standard, F, T e TF.

CPU	FW	Tempi di elaborazione tipici della CPU nel funzionamento automatico	Tempi di elaborazione tipici della CPU per ottimizzazione iniziale e ottimizzazione fine
CPU 1211	≥ V4.0	190 µs	270 µs
CPU 1212			
CPU 1214			

CPU 1215	$\geq V4.0$	190 μs	270 μs
CPU 1217			
CPU 1212 G2	$\geq V1.0$	130 μs	150 μs
CPU 1214 G2			
CPU 1510SP	$\leq V2.9$	65 μs	80 μs
CPU 1511			
CPU 1511C			
CPU 1512C			
CPU 1512SP			
CPU 1513			
CPU 1515		50 μs	65 μs
CPU 1516			
CPU 1517	$\leq V3.1$	8 μs	12 μs
CPU 1518		4 μs	6 μs
CPU 1510SP	V3.0, V3.1	55 μs	70 μs
CPU 1511			
CPU 1511C			
CPU 1512C			
CPU 1512SP			
CPU 1513			
CPU 1514SP			
CPU 1515		50 μs	60 μs
CPU 1516			
CPU 1510SP	$\geq V4.0$	50 μs	60 μs
CPU 1511			
CPU 1511C			
CPU 1512C			
CPU 1512SP			
CPU 1513			
CPU 1514SP			
CPU 1515			
CPU 1516			
CPU 1516T		3 μs	4 μs
CPU 1517		2 μs	3 μs
CPU 1518		1 μs	2 μs

Tempi tipici di elaborazione CPU dell'oggetto tecnologico PID_Compact con versione V2.0 o superiore in funzione del tipo di CPU e del modo di funzionamento per le CPU R/H nello stato di sistema RUN-Redundant.

CPU	FW	Tempi di elaborazione tipici della CPU nel funzionamento automatico	Tempi di elaborazione tipici della CPU per ottimizzazione iniziale e ottimizzazione fine
CPU 1513R	V3.0, V3.1	90 μ s	140 μ s
CPU 1515R		85 μ s	100 μ s
CPU 1513R	$\geq$ V4.0	85 μ s	100 μ s
CPU 1515R			
CPU 1517H		4 μ s	6 μ s
CPU 1518HF			

Memoria necessaria

Memoria necessaria di un DB di istanza dell'oggetto tecnologico PID_Compact dalla versione V2.0.

Memoria necessaria	Memoria necessaria del DB di istanza di PID_Compact V2.x	Memoria necessaria del DB di istanza di PID_Compact V3.x
Memoria di caricamento necessaria	ca. 3700 byte	ca. 3750 byte
Memoria di lavoro necessaria complessivamente	788 byte	802 byte
Memoria di lavoro necessaria a ritenzione	44 byte	52 byte

10.1.4 PID_Compact dalla V2

10.1.4.1 Descrizione di PID_Compact V3

Descrizione

L'istruzione PID_Compact mette a disposizione un regolatore PID con ottimizzazione integrata per organi attuatori con azione proporzionale.

Sono disponibili i seguenti modi di funzionamento:

- Inattivo
- Ottimizzazione iniziale
- Ottimizzazione fine
- Funzionamento automatico
- Funzionamento manuale
- Valore di uscita sostitutivo con controllo errori

Una descrizione dettagliata dei modi di funzionamento è disponibile al paragrafo relativo al parametro State.

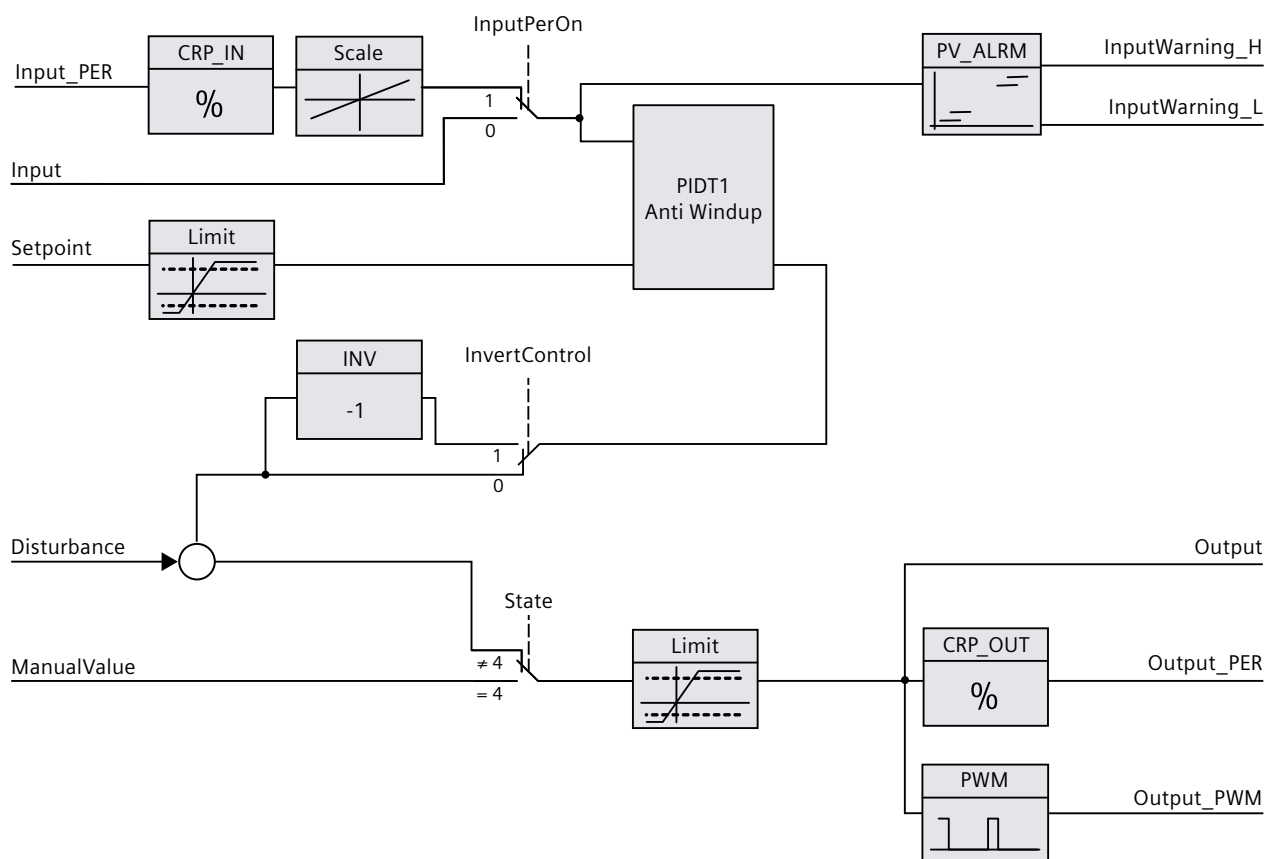
Algoritmo PID

PID_Compact è un regolatore PIDT1 con anti-windup e ponderazione del componente P e D. L'algoritmo PID utilizza la seguente formula (banda morta disattivata):

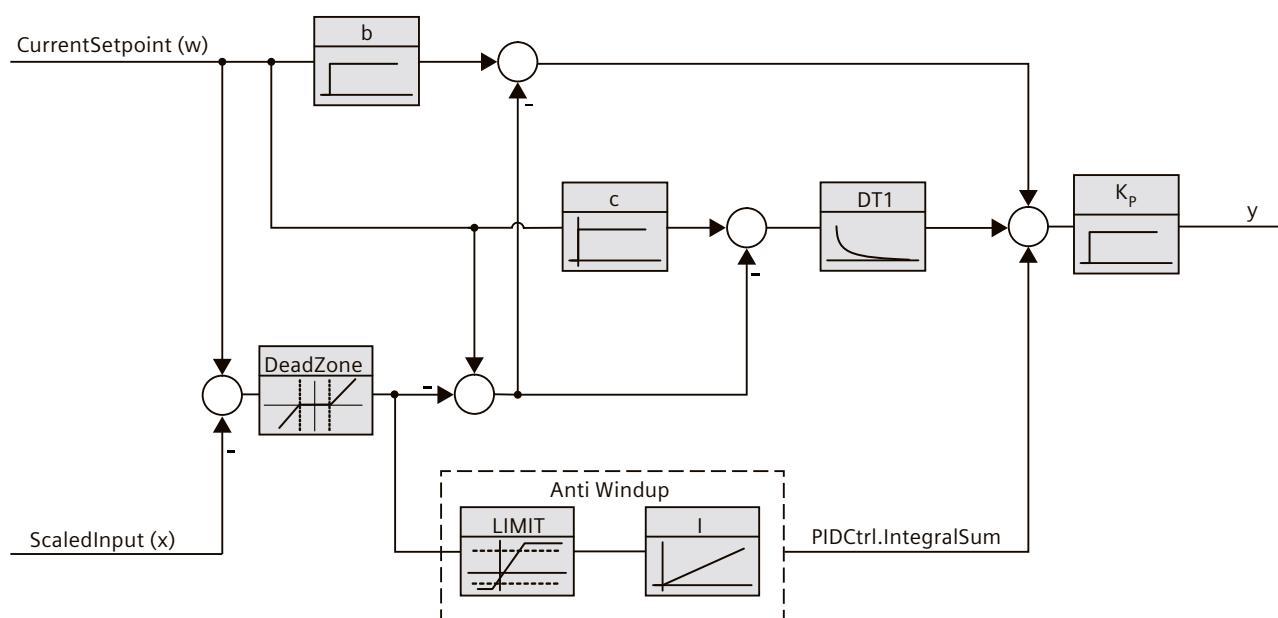
$$y = K_p \left[(b \cdot w - x) + \frac{1}{T_I \cdot s} (w - x) + \frac{T_D \cdot s}{a \cdot T_D \cdot s + 1} (c \cdot w - x) \right]$$

Simbolo	Descrizione	Parametri dell'istruzione PID_Compact
y	Valore di uscita dell'algoritmo PID	-
K _p	Guadagno proporzionale	Retain.CtrlParams.Gain
s	Operatore di Laplace	-
b	Ponderazione del componente P	Retain.CtrlParams.PWeighting
w	Setpoint	CurrentSetpoint
x	Valore istantaneo	ScaledInput
T _I	Tempo di integrazione	Retain.CtrlParams.Ti
T _D	Tempo derivativo	Retain.CtrlParams.Td
a	Coefficiente per il ritardo derivativo (ritardo derivativo T1 = a × T _D)	Retain.CtrlParams.TdFilterRatio
c	Ponderazione del componente D	Retain.CtrlParams.DWeighting
DeadZone	Ampiezza banda morta	Retain.CtrlParams.DeadZone

Schema a blocchi di PID_Compact



Schema a blocchi di PIDT1 con anti-windup



Richiamo

PID_Compact viene richiamato a intervalli costanti nel ciclo di un OB di schedulazione orologio.

Caricamento nel dispositivo

I valori attuali delle variabili a ritenzione vengono aggiornati soltanto se si carica completamente PID_Compact.
Caricamento di un oggetto tecnologico nel dispositivo

Avviamento

All'avviamento della CPU, PID_Compact si avvia nel modo di funzionamento salvato nel parametro di transito Mode. Per commutare al modo di funzionamento "Inattivo" all'avvio, impostare RunModeByStartup = FALSE.

Comportamento in caso di errori

Il comportamento in caso di errori dipende dalle variabili SetSubstituteOutput e ActivateRecoverMode. Se ActivateRecoverMode = TRUE il comportamento dipende anche dall'errore che si è verificato.

SetSubstituteOutput	ActivateRecoverMode	Editor di configurazione > Valore di uscita > Imposta Output su	Comportamento
irrilevante	FALSE	Zero (Inattivo)	Commutazione nel modo di funzionamento "Inattivo" (State = 0) Il valore 0.0 viene trasferito all'organo attuatore.
FALSE	TRUE	Valore di uscita attuale per la durata dell'errore	Commutazione nel modo di funzionamento "valore di uscita sostitutivo con diagnostica" (State = 5) Il valore di uscita attuale viene trasmesso all'organo attuatore per la durata dell'errore.
TRUE	TRUE	Valore di uscita sostitutivo per la durata dell'errore	Commutazione nel modo di funzionamento "valore di uscita sostitutivo con diagnostica" (State = 5) Il valore di SubstituteOutput viene trasmesso all'organo attuatore per la durata dell'errore.

Nel funzionamento manuale PID_Compact utilizza ManualValue come valore di uscita, tranne nel caso in cui ManualValue non sia valido. Se ManualValue non è valido viene utilizzato SubstituteOutput. Se ManualValue e SubstituteOutput non sono validi viene utilizzato Config.OutputLowerLimit.

Il parametro Error indica se attualmente è presente un errore. Se l'errore non è più presente, compare Error = FALSE. Il parametro ErrorBits indica quali errori si sono verificati. ErrorBits viene resettato da un fronte di salita in Reset o ErrorAck.

10.1.4.2 Descrizione di PID_Compact V2

Descrizione

L'istruzione PID_Compact mette a disposizione un regolatore PID con ottimizzazione integrata per organi attuatori con azione proporzionale.

Sono disponibili i seguenti modi di funzionamento:

- Inattivo
- Ottimizzazione iniziale
- Ottimizzazione fine
- Funzionamento automatico
- Funzionamento manuale
- Valore di uscita sostitutivo con controllo errori

Una descrizione dettagliata dei modi di funzionamento è disponibile al paragrafo relativo al parametro State.

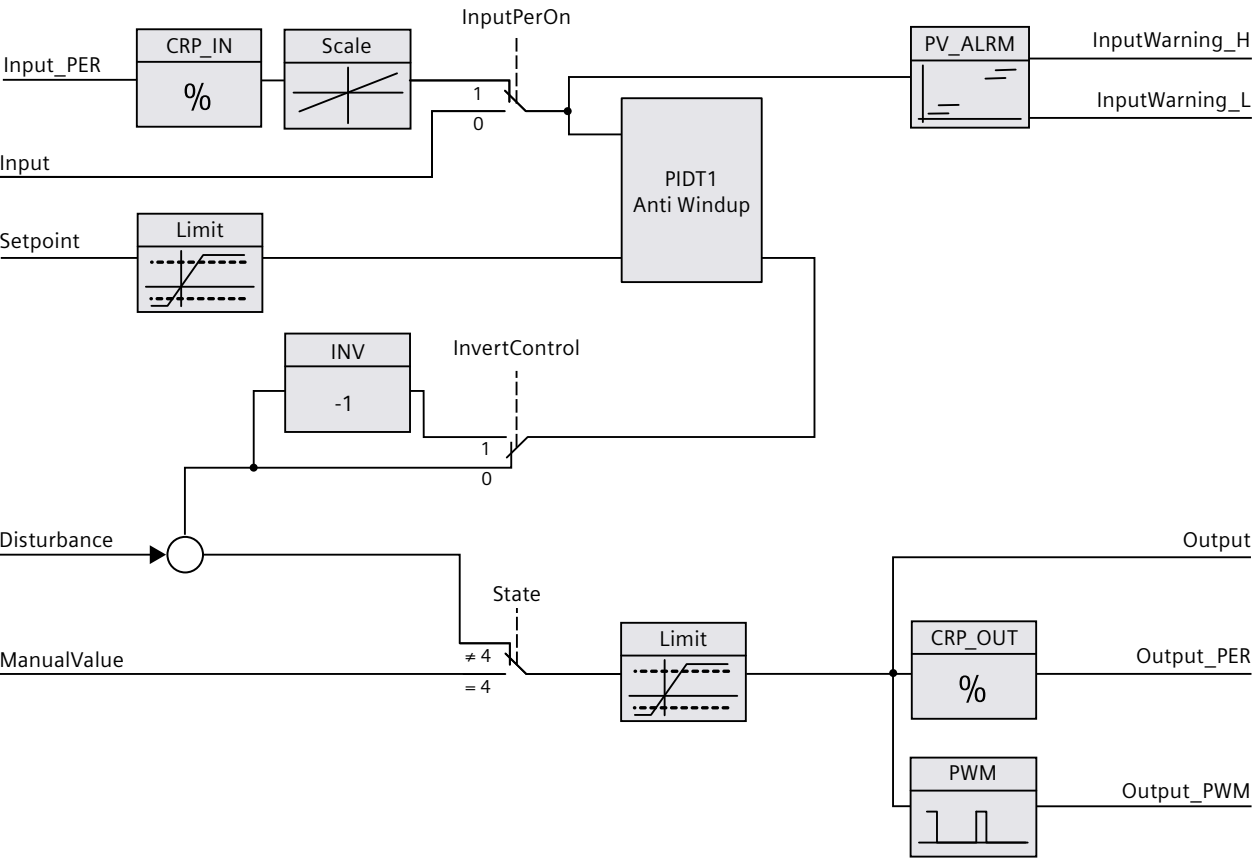
Algoritmo PID

PID_Compact è un regolatore PIDT1 con anti-windup e ponderazione del componente P e D. L'algoritmo PID utilizza la seguente formula:

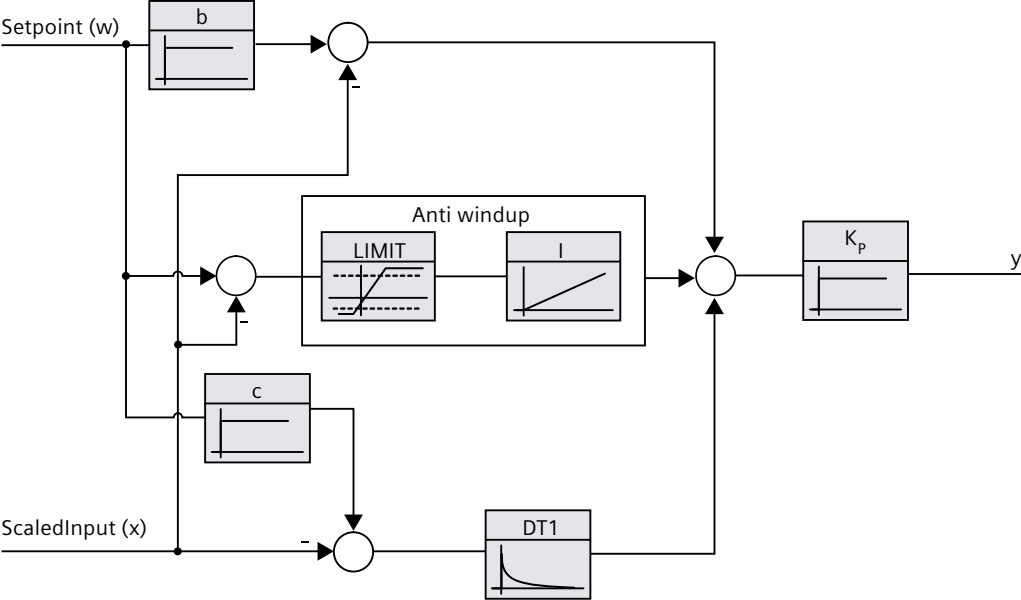
$$y = K_p \left[(b \cdot w - x) + \frac{1}{T_i \cdot s} (w - x) + \frac{T_D \cdot s}{a \cdot T_D \cdot s + 1} (c \cdot w - x) \right]$$

Simbolo	Descrizione	Parametri dell'istruzione PID_Compact
y	Valore di uscita dell'algoritmo PID	-
K _p	Guadagno proporzionale	Retain.CtrlParams.Gain
s	Operatore di Laplace	-
b	Ponderazione del componente P	Retain.CtrlParams.PWeighting
w	Setpoint	CurrentSetpoint
x	Valore istantaneo	ScaledInput
T _i	Tempo di integrazione	Retain.CtrlParams.Ti
T _D	Tempo derivativo	Retain.CtrlParams.Td
a	Coefficiente per il ritardo derivativo (ritardo derivativo T ₁ = a × T _D)	Retain.CtrlParams.TdFiltRatio
c	Ponderazione del componente D	Retain.CtrlParams.DWeighting

Schema a blocchi di PID_Compact



Schema a blocchi di PIDT1 con anti-windup



Richiamo

PID_Compact viene richiamato a intervalli costanti nel ciclo di un OB di schedulazione orologio.

Caricamento nel dispositivo

I valori attuali delle variabili a ritenzione vengono aggiornati soltanto se si carica completamente PID_Compact.

Caricamento degli oggetti tecnologici del dispositivo [\(Pagina 48\)](#)

Avviamento

All'avviamento della CPU, PID_Compact si avvia nel modo di funzionamento salvato nel parametro di transito Mode. Per commutare al modo di funzionamento "Inattivo" all'avvio, impostare RunModeByStartup = FALSE.

Comportamento in caso di errori

Il comportamento in caso di errore dipende dalle variabili SetSubstituteOutput e ActivateRecoverMode. Se ActivateRecoverMode = TRUE il comportamento dipende anche dall'errore che si è verificato.

SetSubstituteOutput	ActivateRecoverMode	Editor di configurazione > Valore di uscita > Imposta Output su	Comportamento
irrilevante	FALSE	Zero (Inattivo)	Commutazione nel modo di funzionamento "Inattivo" (State = 0) Il valore 0.0 viene trasferito all'organo attuatore.
FALSE	TRUE	Valore di uscita attuale per la durata dell'errore	Commutazione nel modo di funzionamento "valore di uscita sostitutivo con diagnostica" (State = 5) Il valore di uscita attuale viene trasmesso all'organo attuatore per la durata dell'errore.
TRUE	TRUE	Valore di uscita sostitutivo per la durata dell'errore	Commutazione nel modo di funzionamento "valore di uscita sostitutivo con diagnostica" (State = 5) Il valore di SubstituteOutput viene trasmesso all'organo attuatore per la durata dell'errore.

Nel funzionamento manuale PID_Compact utilizza ManualValue come valore di uscita, tranne nel caso in cui ManualValue non sia valido. Se ManualValue non è valido viene utilizzato SubstituteOutput. Se ManualValue e SubstituteOutput non sono validi viene utilizzato Config.OutputLowerLimit.

Il parametro Error indica se attualmente è presente un errore. Se l'errore non è più presente, compare Error = FALSE. Il parametro ErrorBits indica quali errori si sono verificati. ErrorBits viene resettato da un fronte di salita in Reset o ErrorAck.

10.1.4.3 Modo di funzionamento di PID_Compact dalla V2

Sorveglianza dei limiti del valore istantaneo

Nelle variabili Config.InputUpperLimit e Config.InputLowerLimit si definisce il limite superiore e inferiore del valore istantaneo. Se il valore istantaneo non rientra in questi limiti si verifica un errore (ErrorBits = 0001h).

Nelle variabili Config.InputUpperWarning e Config.InputLowerWarning si definisce il limite di avviso superiore e inferiore del valore istantaneo. Se il valore istantaneo non rientra in questi limiti viene emesso un avviso (Warning = 0040h) e il parametro di uscita InputWarning_H o InputWarning_L diventa TRUE.

Definizione dei limiti per il setpoint

Nelle variabili Config.SetpointUpperLimit e Config.SetpointLowerLimit si definisce un limite superiore e inferiore per il setpoint. PID_Compact circoscrive automaticamente il setpoint ai limiti del valore istantaneo. È inoltre possibile delimitare il setpoint entro un campo più piccolo. PID_Compact verifica se il campo è compreso entro i limiti del valore istantaneo. Se il setpoint non rientra in questi limiti, il limite superiore o inferiore viene utilizzato come setpoint e il parametro di uscita SetpointLimit_H o SetpointLimit_L diventa TRUE.

Il setpoint viene circoscritto in tutti i modi operativi.

Limitazione del valore di uscita

Nelle variabili Config.OutputUpperLimit e Config.OutputLowerLimit si definisce il limite superiore e inferiore del valore di uscita. Output, ManualValue e SubstituteOutput vengono limitati su questi valori. I limiti del valore di uscita devono essere adeguati al senso di regolazione.

I valori validi per i limiti del valore di uscita dipendono dal valore Output utilizzato.

Output	-100.0 ... 100.0 %
Output_PER	-100.0 ... 100.0 %
Output_PWM	0.0 ... 100.0 %

Deve valere:

OutputUpperLimit > OutputLowerLimit

NOTA

Impiego con due o più attuatori

PID_Compact non si addice all'impiego con due o più attuatori (ad es. in applicazioni di riscaldamento/raffreddamento), in quanto per la realizzazione di un buon comportamento di regolazione, attuatori diversi richiedono anche parametri PID diversi. Per le applicazioni con due attuatori ad azione opposta, utilizzare PID_Temp.

Valore di uscita sostitutivo

PID_Compact in caso di errore può emettere un valore di uscita sostitutivo da predefinire nella variabile SubstituteOutput. Il valore di uscita sostitutivo deve essere compreso entro i limiti del valore di uscita.

Controllo della validità dei segnali

In caso di utilizzo, viene controllata la validità dei valori dei parametri seguenti:

- Setpoint
- Input
- Input_PER
- Disturbance
- ManualValue
- SubstituteOutput
- Output
- Output_PER
- Output_PWM

Sorveglianza del tempo di campionamento PID_Compact

Il tempo di campionamento dovrebbe corrispondere al tempo di ciclo dell'OB richiamante. L'istruzione PID_Compact misura l'intervallo tra due richiami il quale corrisponde al tempo di campionamento attuale. Ad ogni commutazione del modo di funzionamento e al primo avvio viene generato il valore medio dei primi 10 tempi di campionamento. Se il tempo di campionamento attuale si discosta troppo da questo valore medio si verifica un errore (Error = 0800h).

Durante l'ottimizzazione si verifica questo errore se:

- Nuovo valore medio $\geq 1,1 \times$ valore medio precedente
- Nuovo valore medio $\leq 0,9 \times$ valore medio precedente

In funzionamento automatico si verifica questo errore se:

- Nuovo valore medio $\geq 1,5 \times$ valore medio precedente
- Nuovo valore medio $\leq 0,5 \times$ valore medio precedente

Disattivando il controllo del tempo di campionamento (CycleTime.EnMonitoring = FALSE) è possibile richiamare PID_Compact anche nell'OB1. A causa del tempo di campionamento variabile sarà però necessario accettare una qualità della regolazione più scarsa.

Tempo di campionamento dell'algoritmo PID

Poiché il sistema regolato richiede un certo tempo per reagire a una modifica del valore di uscita, è utile non calcolare questo valore in ogni ciclo. Il tempo di campionamento dell'algoritmo PID è il tempo che trascorre tra due calcoli del valore di uscita. Viene determinato durante l'ottimizzazione e arrotondato ad un multiplo del tempo di ciclo. Tutte le altre funzioni di PID_Compact vengono eseguite ad ogni richiamo.

In caso di impiego di Output_PWM, come durata periodo della modulazione ampiezza impulso viene utilizzato il tempo di campionamento dell'algoritmo PID. La precisione del segnale di uscita viene determinata dal rapporto tra tempo di campionamento dell'algoritmo PID e tempo di ciclo dell'OB. Si raccomanda pertanto che il tempo di ciclo non superi un decimo del tempo di campionamento dell'algoritmo PID.

Senso di regolazione

Aumentando il valore di uscita si dovrebbe ottenere un aumento del valore istantaneo. In questo caso si parla di un senso di regolazione normale. Per raffreddamenti e regolazioni di flusso può essere necessario invertire il senso di regolazione. PID_Compact non opera con guadagno proporzionale negativo. Se InvertControl = TRUE, un aumento della differenza di regolazione determina una riduzione del valore di uscita. Il senso di regolazione viene preso in considerazione anche durante l'ottimizzazione iniziale e l'ottimizzazione fine.

10.1.4.4 Parametri di ingresso di PID_Compact dalla V2

I nomi dei seguenti parametri si applicano sia al blocco dati sia all'accesso tramite l'API Openness.

Tabella 10-1

Parametri	Tipo di dati	Preassegnazio- ne	Descrizione
Setpoint	REAL	0.0	Setpoint del regolatore PID nel funzionamento automatico
Input	REAL	0.0	Una variabile del programma utente viene utilizzata come sorgente del valore istantaneo. Per poter utilizzare il parametro Input è necessaria l'impostazione Config.InputPerOn = FALSE.
Input_PER	INT	0	Un ingresso analogico viene utilizzato come sorgente del valore istantaneo. Per poter utilizzare il parametro Input_PER è necessaria l'impostazione Config.InputPerOn = TRUE.
Disturbance	REAL	0.0	Grandezza di disturbo o valore di precomando
ManualEnable	BOOL	FALSE	- Il fronte FALSE -> TRUE attiva il modo "Funzionamento manuale", State = 4, Mode rimane invariato. Finché ManualEnable = TRUE non è possibile modificare il modo di funzionamento attraverso un fronte di salita in ModeActivate e utilizzare la finestra di messa in servizio. - Il fronte TRUE -> FALSE attiva il modo di funzionamento predefinito da Mode.

Parametri	Tipo di dati	Preassegnazio- ne	Descrizione
			Si consiglia di modificare il modo di funzionamento solo tramite ModeActivate.
ManualValue	REAL	0.0	Valore manuale Questo valore viene utilizzato nel funzionamento ma- nuale come valore di uscita. Sono ammessi valori da Config.OutputLowerLimit a Config.OutputUpperLimit
ErrorAck	BOOL	FALSE	• Fronte FALSE -> TRUE ErrorBits e Warning vengono resettati.
Reset	BOOL	FALSE	Esegue un nuovo avvio del regolatore. • Fronte FALSE -> TRUE – Commutazione nel modo di funzionamento "Inattivo" – ErrorBits e Warnings vengono resettati. • Finché Reset = TRUE, – PID_Compact rimane nel modo di funzionamen- to "Inattivo" (State = 0). – il modo di funzionamento non può essere modi- ficato tramite Mode e ModeActivate o ManualEnable. – La finestra di dialogo per la messa in servizio non può essere utilizzata. • Fronte TRUE -> FALSE – Se ManualEnable = FALSE, PID_Compact com- muta nel modo operativo salvato in Mode. – Se Mode = 3, il componente I viene trattato co- me configurato con la variabile IntegralResetMode.
ModeActivate	BOOL	FALSE	• Fronte FALSE -> TRUE PID_Compact passa al modo di funzionamento sal- vato in Mode.

10.1.4.5 Parametri di uscita di PID_Compact dalla V2

I nomi dei seguenti parametri si applicano sia al blocco dati sia all'accesso tramite l'API Openness.

Tabella 10-2

Parameter	Tipo di dati	Preassegnazione	Descrizione
ScaledInput	REAL	0.0	Valore istantaneo riportato in scala
Le uscite "Output", "Output_PER" e "Output_PWM" possono essere utilizzate parallelamente.			
Output	REAL	0.0	Valore di uscita in formato REAL
Output_PER	INT	0	Valore di uscita analogico
Output_PWM	BOOL	FALSE	Valore di uscita con modulazione ad ampiezza d'impulso Il valore di uscita viene calcolato in base a tempi di attivazio- ne e disattivazione variabili.

Parameter	Tipo di dati	Preassegnazione	Descrizione
SetpointLimit_H	BOOL	FALSE	Se SetpointLimit_H = TRUE, è stato raggiunto il limite superiore assoluto del setpoint ($\text{Setpoint} \geq \text{Config.SetpointUpperLimit}$). Il setpoint è limitato a Config.SetpointUpperLimit.
SetpointLimit_L	BOOL	FALSE	Se SetpointLimit_L = TRUE, è stato raggiunto il limite inferiore assoluto del setpoint ($\text{Setpoint} \leq \text{Config.SetpointLowerLimit}$). Il setpoint è limitato a Config.SetpointLowerLimit.
InputWarning_H	BOOL	FALSE	Se InputWarning_H = TRUE il limite superiore di avviso del valore istantaneo è stato raggiunto o superato.
InputWarning_L	BOOL	FALSE	Se InputWarning_L = TRUE il limite inferiore di avviso del valore istantaneo è stato raggiunto o superato per difetto.
State	INT	0	Il parametro State (Pagina 275) indica il modo di funzionamento attuale del regolatore PID. Il modo di funzionamento si modifica con il parametro di ingresso Mode e con un fronte di salita in ModeActivate. - State = 0: Inattivo - State = 1: Ottimizzazione iniziale - State = 2: Ottimizzazione fine - State = 3: Funzionamento automatico - State = 4: Funzionamento manuale - State = 5: Valore di uscita sostitutivo con controllo errori
Error	BOOL	FALSE	Se Error = TRUE in questo ciclo è presente almeno un messaggio di errore.
ErrorBits	DWORD	DW#16#0	Il parametro ErrorBits (Pagina 278) mostra quali messaggi di errore sono presenti. ErrorBits è a ritenzione e viene resettato in presenza di un fronte di salita in Reset o ErrorAck.

10.1.4.6 Parametri di transito di PID_Compact dalla V2

I nomi dei seguenti parametri si applicano sia al blocco dati sia all'accesso tramite l'API Openness.

Tabella 10-3

Parameter	Tipo di dati	Preassegnazione	Descrizione
Mode	INT	4	In Mode si predefinisce il modo di funzionamento al quale deve passare PID_Compact. Sono possibili: - Mode = 0: Inattivo - Mode = 1: Ottimizzazione iniziale - Mode = 2: Ottimizzazione fine - Mode = 3: Funzionamento automatico - Mode = 4: Funzionamento manuale Il modo di funzionamento viene attivato da: - Fronte di salita in ModeActivate - Fronte di discesa in Reset - Fronte di discesa in ManualEnable - Avviamento a freddo della CPU se RunModeByStartup = TRUE Mode è a ritenzione.

Parameter	Tipo di dati	Preassegnazione	Descrizione
			I modi di funzionamento sono descritti dettagliatamente nella sezione Parametri State e Mode a partire da V2 (Pagina 275).

Vedere anche

[Parametri State e Mode a partire da V2 \(Pagina 275\)](#)

10.1.4.7 Variabili statiche di PID_Compact dalla V2

NOTA

Per evitare un comportamento errato del regolatore PID modificare le variabili contrassegnate con ⁽¹⁾ solo nel modo di funzionamento "Inattivo".

Se non diversamente specificato, i nomi delle seguenti variabili si applicano sia al blocco dati sia all'accesso tramite l'API Openness.

Variabile	Tipo di dati	Preassegnazione	Descrizione
IntegralResetMode	INT	fino a V2.2: 1, da V2.3: 4	La variabile Variabile IntegralResetMode dalla V2 (Pagina 283) stabilisce i valori da preassegnare al componente I PIDCtrl.IntegralSum alla commutazione del modo di funzionamento da "Inattivo" a "Funzionamento automatico". Questa impostazione è attiva solo per un ciclo. Sono possibili: - IntegralResetMode = 0: Livella - IntegralResetMode = 1: Elimina - IntegralResetMode = 2: Mantieni - IntegralResetMode = 3: Preassegna - IntegralResetMode = 4: In caso di variazioni del set-point (soltanto per PID_Compact con la versione ≥ 2.3)
OverwriteInitialOutputValue	REAL	0.0	Se una delle seguenti condizioni è soddisfatta, al componente integrale PIDCtrl.IntegralSum viene preassegnato automaticamente un valore tale da far supporre che nel ciclo precedente Output = OverwriteInitialOutputValue ci fosse: - IntegralResetMode = 3 alla commutazione dal modo di funzionamento "Inattivo" al modo "Funzionamento automatico" - IntegralResetMode = 3, Fronte TRUE -> FALSE sul parametro Reset e sul parametro Mode = 3 - PIDCtrl.PIDInit = TRUE nel "Funzionamento automatico" (disponibile dalla PID_Compact versione V. 2.3)

Variabile	Tipo di dati	Preassegnazio- ne	Descrizione
RunModeByStartup	BOOL	TRUE	Dopo il nuovo avviamento della CPU attiva il modo di funzionamento in Mode. Se RunModeByStartup = TRUE, dopo l'avvio della CPU PID_Compact si avvia nel modo di funzionamento salvato in Mode. Se RunModeByStartup = FALSE, dopo l'avvio della CPU PID_Compact resta nel modo di funzionamento "Inattivo".
LoadBackUp	BOOL	FALSE	Se LoadBackUp = TRUE, viene ricaricato l'ultimo set di parametri PID dalla struttura CtrlParamsBackUp. Il set è stato memorizzato prima dell'ultima ottimizzazione. LoadBackUp viene reimpostato automaticamente a FALSE.
PhysicalUnit	INT	0	Unità fisica del valore istantaneo e del setpoint, ad es. °C o °F. PhysicalUnit viene utilizzato per la visualizzazione negli editor e non incide in alcun modo sul comportamento dell'algoritmo di regolazione nella CPU. Se si importa PID_Compact fino alla versione 2.4 tramite l'API Openness, PhysicalUnit viene reimpostato al valore di default. A partire dalla versione 3.0 il valore di PhysicalUnit viene mantenuto durante l'importazione.
PhysicalQuantity	INT	0	Grandezza fisica del valore istantaneo e del setpoint, ad es. la temperatura. PhysicalQuantity viene utilizzato per la visualizzazione negli editor e non incide in alcun modo sul comportamento dell'algoritmo di regolazione nella CPU. Durante l'importazione di PID_Compact fino alla versione 2.4 tramite l'API Openness, PhysicalQuantity viene riportato al valore di default. A partire dalla versione 3.0 il valore di PhysicalQuantity viene mantenuto durante l'importazione.
ActivateRecoverMode	BOOL	TRUE	La Variabile ActivateRecoverMode dalla V2 (Pagina 280) determina il comportamento in caso di errore.
Warning	DWORD	0	Variabile Warning dalla V2 (Pagina 282) mostra gli avvisi da Reset = TRUE oppure ErrorAck = TRUE. Warning è a ritenzione.
Progress	REAL	0.0	Avanzamento della fase di ottimizzazione attuale in percentuale (0.0 - 100.0)
CurrentSetpoint	REAL	0.0	CurrentSetpoint visualizza sempre il setpoint attivo attualmente. Questo valore viene congelato durante l'ottimizzazione.
CancelTuningLevel	REAL	10.0	Oscillazione del setpoint ammessa durante l'ottimizzazione. L'ottimizzazione viene interrotta solo se: - Setpoint > CurrentSetpoint + CancelTuningLevel oppure - Setpoint < CurrentSetpoint - CancelTuningLevel

Variabile	Tipo di dati	Preassegnazione	Descrizione
SubstituteOutput	REAL	0.0	Valore di uscita sostitutivo Se sono soddisfatte le seguenti condizioni, viene utilizzato come valore di uscita il valore sostitutivo. - Nel funzionamento automatico sono presenti uno o più errori nei quali è attivo ActivateRecoverMode. - SetSubstituteOutput = TRUE - ActivateRecoverMode = TRUE Config.OutputUpperLimit $\geq$ SubstituteOutput $\geq$ Config.OutputLowerLimit
SetSubstituteOutput	BOOL	TRUE	Selezione del valore iniziale, finché è presente un errore (State = 5) - Se SetSubstituteOutput = TRUE e ActivateRecoverMode = TRUE, viene emesso il valore di uscita sostitutivo configurato SubstituteOutput fino a quando è presente un errore. - Se SetSubstituteOutput = FALSE e ActivateRecoverMode = TRUE l'organo attuatore resta sul valore di uscita attuale finché è presente un errore. - Se ActivateRecoverMode = FALSE, SetSubstituteOutput non è effettivo. - Se SubstituteOutput non è valido (ErrorBits = 16#0002_0000), il valore di uscita sostitutivo non può essere emesso. In questo caso viene utilizzato come valore di uscita il limite inferiore del valore di uscita (Config.OutputLowerLimit).
Config.InputPerOn ⁽¹⁾	BOOL	TRUE	Se InputPerOn = TRUE viene utilizzato il parametro Input_PER per il rilevamento del valore istantaneo. Se InputPerOn = FALSE viene utilizzato il parametro Input.
Config.InvertControl ⁽¹⁾	BOOL	FALSE	Inversione del senso di regolazione Se InvertControl = TRUE, un aumento della differenza di regolazione determina una riduzione del valore di uscita.
Config.OutputSelect	INT	0	Selezione del valore di uscita (dalla versione 3.0): - OutputSelect = 0: Output_PER (analog) - OutputSelect = 1: Output - OutputSelect = 2: Output_PWM Config.OutputSelect viene utilizzato per la progettazione del regolatore in TIA Portal e non influisce sul calcolo dei valori di uscita nella CPU.
_Config.OutputSelect	INT	0	Selezione del valore di uscita (fino alla versione 2.4): - OutputSelect = 0: Output_PER (analogico) - OutputSelect = 1: Output - OutputSelect = 2: Output_PWM _Config.OutputSelect viene utilizzato per la progettazione del regolatore in TIA Portal e non influisce sul calcolo dei valori di uscita nella CPU. _Config.OutputSelect non è disponibile nel blocco dati e può essere progettato solo nell'editor di configurazione o tramite l'API Openness. Con l'importazione di PID_Compact tramite l'API Openness _Config.OutputSelect viene ripristinato il valore di default.

Variabile	Tipo di dati	Preassegnazio- ne	Descrizione
Config.InputUpperLimit ⁽¹⁾	REAL	120.0	Limite superiore del valore istantaneo Input e Input_PER vengono controllati per verificare il rispetto di tale limite. Se il limite viene superato viene emesso un messaggio di errore, la reazione a questo errore dipende da ActivateRecoverMode. Nell'ingresso di periferia il valore istantaneo può superare al massimo del 18% il campo normale (campo di sovracomando). Con l'impiego dell'ingresso di periferia con preimpostazione del valore limite e della graduazione di scala del valore istantaneo, il superamento del limite non è pertanto possibile. All'avvio della prima ottimizzazione viene eseguito un controllo sulla base della differenza tra limite superiore e limite inferiore del valore istantaneo per accertare se la distanza tra setpoint e valore istantaneo soddisfa i requisiti richiesti. InputUpperLimit > InputLowerLimit
Config.InputLowerLimit ⁽¹⁾	REAL	0.0	Limite inferiore del valore istantaneo Input e Input_PER vengono controllati per verificare il rispetto di tale limite. Se il limite viene superato verso il basso, viene emesso un messaggio di errore; la reazione a questo errore dipende da ActivateRecoverMode. InputLowerLimit < InputUpperLimit
Config.InputUpperWarning ⁽¹⁾	REAL	3.402822e+38	Limite superiore di avviso del valore istantaneo Input e Input_PER vengono controllati per verificare il rispetto di tale limite. Se il limite viene superato verso l'alto, viene emesso un avviso nel parametro. Se si configura InputUpperWarning al di fuori dei limiti del valore istantaneo, viene utilizzato il limite superiore assoluto configurato per il valore istantaneo come limite superiore di avviso. Se si configura InputUpperWarning entro i limiti del valore istantaneo questo valore viene utilizzato come limite superiore di avviso. InputUpperWarning > InputLowerWarning
Config.InputLowerWarning ⁽¹⁾	REAL	-3.402822e+38	Limite inferiore di avviso del valore istantaneo Input e Input_PER vengono controllati per verificare il rispetto di tale limite. Se il limite viene superato verso il basso, viene emesso un avviso nel parametro Warning. Se si configura InputLowerWarning al di fuori dei limiti del valore istantaneo, il limite inferiore assoluto configurato per il valore istantaneo viene utilizzato come limite inferiore di avviso. Se si configura InputLowerWarning entro i limiti del valore istantaneo questo valore viene utilizzato come limite inferiore di avviso. InputLowerWarning < InputUpperWarning
Config.OutputUpperLimit	REAL	100.0	Limite superiore del valore di uscita Per i dettagli vedi OutputLowerLimit 100.0 ≥ OutputUpperLimit > OutputLowerLimit

Variabile	Tipo di dati	Preassegnazio- ne	Descrizione
Config.OutputLowerLimit	REAL	0.0	Limite inferiore del valore di uscita Per Output e Output_PER vale il campo di valori compreso tra -100.0 e +100.0, incluso lo zero. Con -100.0 si ha Output_PER = -27648; con +100.0 Output_PER = 27648. Per Output_PWM vale il campo di valori da 0.0 a +100.0. I limiti del valore di uscita devono essere adeguati al senso di regolazione. A partire dalla versione 3.0 PID_Compact consente di modificare i limiti del valore di uscita nel funzionamento automatico. Fino alla versione 2.4 tali limiti possono essere modificati solo modo di funzionamento inattivo o manuale. OutputLowerLimit < OutputUpperLimit
Config.SetpointUpperLimit ⁽¹⁾	REAL	3.402822e+38	Limite superiore del setpoint Setpoint viene controllato per verificare il rispetto di questo limite. Se il limite viene superato verso l'alto, viene emesso un avviso nel parametro Warning. Se si configura SetpointUpperLimit al di fuori dei limiti del valore istantaneo, il limite superiore assoluto configurato per il valore istantaneo viene utilizzato come limite superiore del setpoint. Se si configura SetpointUpperLimit entro i limiti del valore istantaneo questo valore viene utilizzato come limite superiore del setpoint. SetpointUpperLimit > SetpointLowerLimit
Config.SetpointLowerLimit ⁽¹⁾	REAL	-3.402822e+38	Limite inferiore del setpoint Setpoint viene controllato per verificare il rispetto di questo limite. Se il limite viene superato verso il basso, viene emesso un avviso nel parametro Warning. Se si configura SetpointLowerLimit al di fuori dei limiti del valore istantaneo, il limite inferiore assoluto configurato per il valore istantaneo viene utilizzato come limite inferiore del setpoint. Se si configura SetpointLowerLimit entro i limiti del valore istantaneo questo valore viene utilizzato come limite inferiore del setpoint. SetpointLowerLimit < SetpointUpperLimit
Config.MinimumOnTime ⁽¹⁾	REAL	0.0	Il tempo minimo di attivazione della modulazione di larghezza di impulso in secondi viene arrotondato a MinimumOnTime = nxCycleTime.Value 100000.0 ≥ MinimumOnTime ≥ 0.0
Config.MinimumOffTime ⁽¹⁾	REAL	0.0	Il tempo minimo di disattivazione della modulazione di larghezza di impulso in secondi viene arrotondato a MinimumOffTime = nxCycleTime.Value 100000.0 ≥ MinimumOffTime ≥ 0.0
Config.InputScaling.UpperPointIn ⁽¹⁾	REAL	27648.0	Scala Input_PER superiore Input_PER viene messo in scala in base alle due coppie di valori UpperPointOut, UpperPointIn e LowerPointOut, LowerPointIn. Efficace soltanto se Input_PER viene utilizzato per la rilevazione del valore istantaneo (Config.InputPerOn = TRUE). UpperPointIn > LowerPointIn

Variabile	Tipo di dati	Preassegnazio- ne	Descrizione
Config.InputScaling.LowerPointIn ⁽¹⁾	REAL	0.0	Scala Input_PER inferiore Input_PER viene messo in scala in base alle due coppie di valori UpperPointOut, UpperPointIn e LowerPointOut, LowerPointIn. Efficace soltanto se Input_PER viene utilizzato per la rilevazione del valore istantaneo (Config.InputPerOn = TRUE). LowerPointIn < UpperPointIn
Config.InputScaling.UpperPointOut ⁽¹⁾	REAL	100.0	Valore istantaneo superiore riportato in scala Input_PER viene messo in scala in base alle due coppie di valori UpperPointOut, UpperPointIn e LowerPointOut, LowerPointIn. Efficace soltanto se Input_PER viene utilizzato per la rilevazione del valore istantaneo (Config.InputPerOn = TRUE). UpperPointOut > LowerPointOut
Config.InputScaling.LowerPointOut ⁽¹⁾	REAL	0.0	Valore istantaneo inferiore riportato in scala Input_PER viene messo in scala in base alle due coppie di valori UpperPointOut, UpperPointIn e LowerPointOut, LowerPointIn. Efficace soltanto se Input_PER viene utilizzato per la rilevazione del valore istantaneo (Config.InputPerOn = TRUE). LowerPointOut < UpperPointOut
CycleTime.StartEstimation	BOOL	TRUE	Se CycleTime.EnEstimation = TRUE, CycleTime.StartEstimation = TRUE avvia il rilevamento automatico del tempo di campionamento PID_Compact (tempo di ciclo dell'OB richiama-nte). Al termine dell'operazione di misura viene impostato CycleTime.StartEstimation = FALSE .
CycleTime.EnEstimation	BOOL	TRUE	Se CycleTime.EnEstimation = TRUE, il tempo di campionamento PID_Compact viene calcolato automaticamente. Se CycleTime.EnEstimation = FALSE, il tempo di campionamento PID_Compact non viene calcolato automaticamente e si deve configurare manualmente il corretto CycleTime.Value.
CycleTime.EnMonitoring	BOOL	TRUE	Se CycleTime.EnMonitoring = FALSE il tempo di campionamento PID_Compact non viene controllato. Se PID_Compact non viene eseguito entro il tempo di campionamento, non viene emesso alcun errore (ErrorBits=16#0000_0800) e PID_Compact non reagisce come con ActivateRecoverMode.
CycleTime.Value ⁽¹⁾	REAL	0.1	Tempo di campionamento PID_Compact (tempo di ciclo dell'OB richiama-nte) in secondi CycleTime.Value viene rilevata automaticamente e corrisponde in genere al tempo di ciclo dell'OB richiama-nte.
I valori della struttura CtrlParamsBackUp possono essere caricati con LoadBackUp = TRUE.			
CtrlParamsBackUp.SetByUser	BOOL	FALSE	Valore memorizzato di Retain.CtrlParams.SetByUser (dalla versione 3.0)
CtrlParamsBackUp.Gain	REAL	1.0	Guadagno proporzionale memorizzato

Variabile	Tipo di dati	Preassegnazio- ne	Descrizione
CtrlParamsBackUp.Ti	REAL	20.0	Tempo di integrazione memorizzato in secondi
CtrlParamsBackUp.Td	REAL	0.0	Tempo derivativo memorizzato in secondi
CtrlParamsBackUp.TdFiltRatio	REAL	0.2	Coefficiente del ritardo derivativo memorizzato
CtrlParamsBackUp.PWeighting	REAL	1.0	Fattore di ponderazione del componente P memorizzato
CtrlParamsBackUp.DWeighting	REAL	1.0	Fattore di ponderazione del componente D memorizzato
CtrlParamsBackUp.Cycle	REAL	1.0	Tempo di campionamento algoritmo PID salvato in secondi
CtrlParamsBackUp.DeadZone	REAL	0.0	Ampiezza della banda morta memorizzata (dalla versione 3.0)
PIDSelfTune.SUT.CalculateParams	BOOL	FALSE	Le proprietà del sistema regolato vengono memorizzate durante l'ottimizzazione. Se SUT.CalculateParams = TRUE, in base a queste proprietà vengono ricalcolati i parametri per l'ottimizzazione iniziale. Ciò permette di modificare il metodo di calcolo dei parametri senza dover ripetere l'ottimizzazione. Al termine del calcolo SUT.CalculateParams viene impostato su FALSE.
PIDSelfTune.SUT.TuneRule	INT	0	Calcolo dei parametri durante l'ottimizzazione iniziale secondo un metodo: - SUT.TuneRule = 0: PID secondo Chien, Hrones e Reswick - SUT.TuneRule = 1: PI secondo Chien, Hrones e Reswick
PIDSelfTune.SUT.State	INT	0	La variabile SUT.State mostra la fase attuale dell'ottimizzazione iniziale: - State = 0: Inizializzazione dell'ottimizzazione iniziale - State = 100: Calcolo della divergenza standard - State = 200: Rilevazione del punto di flesso - State = 9900: Ottimizzazione iniziale riuscita - State = 1: Ottimizzazione iniziale non riuscita
PIDSelfTune.TIR.RunIn	BOOL	FALSE	Con la variabile RunIn è possibile stabilire che un'ottimizzazione fine può essere eseguita anche senza un'ottimizzazione iniziale. RunIn = FALSE Se l'ottimizzazione fine viene avviata dal modo di funzionamento inattivo o manuale, si avvia un'ottimizzazione iniziale. Se i presupposti per l'ottimizzazione iniziale non sono soddisfatti, PID_Compact si comporta come con RunIn = TRUE. Se l'ottimizzazione fine viene avviata dal funzionamento automatico, viene eseguita la regolazione sul setpoint con i parametri PID disponibili. Solo in seguito viene avviata l'ottimizzazione fine. Se non è possibile eseguire un'ottimizzazione iniziale PID_Compact passa al modo di funzionamento dal quale è stata avviata l'ottimizzazione.

Variabile	Tipo di dati	Preassegnazione	Descrizione
			RunIn = TRUE L'ottimizzazione iniziale viene ignorata. PID_Compact tenta di raggiungere il setpoint con il valore di uscita minimo o massimo. Ciò può causare una maggiore sovraoscillazione. L'ottimizzazione fine si avvia infine automaticamente. Al termine dell'ottimizzazione fine RunIn viene impostato su FALSE.
PIDSelfTune.TIR.CalculateParams	BOOL	FALSE	Le proprietà del sistema regolato vengono memorizzate durante l'ottimizzazione. Se TIR.CalculateParams = TRUE, in base a queste proprietà vengono ricalcolati i parametri per l'ottimizzazione fine. Ciò permette di modificare il metodo di calcolo dei parametri senza dover ripetere l'ottimizzazione. Al termine del calcolo TIR.CalculateParams viene impostato su FALSE.
PIDSelfTune.TIR.TuneRule	INT	0	Calcolo dei parametri durante l'ottimizzazione fine secondo un metodo: - TIR.TuneRule = 0: PID automatico - TIR.TuneRule = 1: PID veloce (comportamento di regolazione più veloce con ampiezze del valore di uscita più elevate rispetto a TIR.TuneRule = 2) - TIR.TuneRule = 2: PID lento (comportamento di regolazione più lento con ampiezze del valore di uscita più basse rispetto a TIR.TuneRule = 1) - TIR.TuneRule = 3: PID secondo Ziegler-Nichols - TIR.TuneRule = 4: PI secondo Ziegler-Nichols - TIR.TuneRule = 5: P secondo Ziegler-Nichols Affinché il calcolo dei parametri PID con TIR.CalculateParams e TIR.TuneRule = 0, 1 o 2 possa essere ripetuto, anche l'ottimizzazione fine precedente deve essere stata eseguita con TIR.TuneRule = 0, 1 o 2. In caso contrario viene utilizzato TIR.TuneRule = 3. Il nuovo calcolo del parametro PID con TIR.CalculateParams e TIR.TuneRule = 3, 4 o 5 è sempre possibile.
PIDSelfTune.TIR.State	INT	0	La variabile TIR.State mostra la fase attuale dell'"Ottimizzazione fine": - State = -100 Non è possibile eseguire l'ottimizzazione fine. Viene prima eseguita un'ottimizzazione iniziale. - State = 0: Inizializzazione dell'ottimizzazione fine - State = 200: Calcolo della divergenza standard - State = 300: Tentativo di raggiungere il setpoint - State = 400: Tentativi di raggiungere il setpoint con i parametri PID disponibili (se l'ottimizzazione iniziale è riuscita) - State = 500: Rilevazione dell'oscillazione e calcolo dei parametri - State = 9900: Ottimizzazione fine riuscita - State = 1: Ottimizzazione fine non riuscita
PIDCtrl.IntegralSum ⁽¹⁾	REAL	0.0	Componente I attuale

Variabile	Tipo di dati	Preassegnazio- ne	Descrizione
PIDCtrl.PIDInit	BOOL	FALSE	PIDCtrl.PIDInit è disponibile da PID_Compact, versione V. 2.3. Se nel "Funzionamento automatico" PIDCtrl.PIDInit = TRUE, al componente integrale PIDCtrl.IntegralSum viene preassegnato automaticamente un valore tale da far supporre che nel ciclo precedente ci fosse $Output = OverwriteInitialOutputValue$. Questo parametro può essere utilizzato per una Regolazione override con PID_Compact dalla V2 (Pagina 101).
Retain.CtrlParams.SetByUser ⁽¹⁾	BOOL	FALSE	Attivazione dell'inserimento manuale dei parametri PID (dalla versione 3.0) Se Retain.CtrlParams.SetByUser = TRUE, i parametri PID sono editabili. Retain.CtrlParams.SetByUser viene utilizzato per la progettazione del regolatore in TIA Portal e non incide in alcun modo sul comportamento dell'algoritmo di regolazione nella CPU. SetByUser è a ritenzione.
_Retain.CtrlParams.SetByUser ⁽¹⁾	BOOL	FALSE	Attivazione dell'inserimento manuale dei parametri PID (fino alla versione 2.4) Se _Retain.CtrlParams.SetByUser = TRUE, i parametri PID sono editabili. _Retain.CtrlParams.SetByUser viene utilizzato per la progettazione del regolatore in TIA Portal e non incide in alcun modo sul comportamento dell'algoritmo di regolazione nella CPU. _Retain.CtrlParams.SetByUser non è disponibile nel blocco dati e può essere progettato solo nell'editor di configurazione o tramite l'API Openness. Con l'importazione di PID_Compact tramite l'API Openness _Retain.CtrlParams.SetByUser viene ripristinato il valore di default.
Retain.CtrlParams.Gain ⁽¹⁾	REAL	1.0	Guadagno proporzionale attivo PID_Compact non funziona con un guadagno proporzionale negativo. Per invertire il senso di regolazione utilizzare la variabile Config.InvertControl. Gain è a ritenzione. $Gain \geq 0.0$
Retain.CtrlParams.Ti ⁽¹⁾	REAL	20.0	- CtrlParams.Ti > 0.0: Tempo di integrazione attivo in secondi - CtrlParams.Ti = 0.0: Il componente I è disattivato Ti è a ritenzione. $100000.0 \geq Ti \geq 0.0$
Retain.CtrlParams.Td ⁽¹⁾	REAL	0.0	- CtrlParams.Td > 0.0: Tempo derivativo attivo in secondi - CtrlParams.Td = 0.0: Il componente D è disattivato Td è a ritenzione. $100000.0 \geq Td \geq 0.0$

Variabile	Tipo di dati	Preassegnazione	Descrizione
Retain.CtrlParams.TdFiltRatio ⁽¹⁾	REAL	0.2	Coefficiente del ritardo derivativo attivo L'effetto del componente D viene ritardato dal coefficiente per il ritardo derivativo. Ritardo derivativo = Tempo derivativo × Coefficiente ritardo derivativo 0.0: Il componente D ha effetto solo per un ciclo e quindi è quasi inattivo. 0.5: Questo valore ha dato buoni risultati nelle operazioni con i sistemi regolati con una costante temporale dominante. > 1.0: Maggiore è il coefficiente, maggiore sarà il ritardo dell'azione del componente D. TdFiltRatio è a ritenzione. TdFiltRatio ≥ 0.0
Retain.CtrlParams.PWeighting ⁽¹⁾	REAL	1.0	Ponderazione del componente P attiva In caso di variazioni del setpoint è possibile ridurre il componente P. Sono adeguati valori da 0.0 a 1.0. 1.0: componente P pienamente attivo in caso di variazione del setpoint 0.0: componente P non attivo in caso di variazione del setpoint In caso di variazioni del valore istantaneo il componente P è sempre pienamente attivo. PWeighting è a ritenzione. 1.0 ≥ PWeighting ≥ 0.0
Retain.CtrlParams.DWeighting ⁽¹⁾	REAL	1.0	Ponderazione del componente D attiva In caso di variazioni del setpoint è possibile ridurre il componente D. Sono adeguati valori da 0.0 a 1.0. 1.0: componente D pienamente attivo in caso di variazione del setpoint 0.0: componente D non attivo in caso di variazione del setpoint In caso di variazioni del valore istantaneo il componente D è sempre pienamente attivo. DWeighting è a ritenzione. 1.0 ≥ DWeighting ≥ 0.0
Retain.CtrlParams.Cycle ⁽¹⁾	REAL	1.0	Tempo di campionamento attivo dell'algoritmo PID in secondi CtrlParams.Cycle viene rilevato durante l'ottimizzazione e arrotondato a un multiplo intero di CycleTime.Value. CtrlParams.Cycle viene utilizzato come durata periodo della modulazione ampiezza impulso. Cycle è a ritenzione. 100000.0 ≥ Cycle > 0.0
Retain.CtrlParams.DeadZone ⁽¹⁾	REAL	0.0	Ampiezza di banda morta attiva CtrlParams.DeadZone è disponibile dalla versione 3.0 di PID_Compact. Se CtrlParams.DeadZone = 0.0 la banda morta è disattivata. CtrlParams.DeadZone non viene impostato automaticamente né adeguato durante l'ottimizzazione.

Variabile	Tipo di dati	Preassegnazio- ne	Descrizione
			CtrlParams.DeadZone deve essere configurato manualmente in modo corretto. Con la banda morta attivata può verificarsi una differenza di regolazione permanente (differenza tra setpoint e valore istantaneo). Questo può avere conseguenze negative sull'esecuzione dell'ottimizzazione fine. DeadZone è a ritenzione. DeadZone ≥ 0.0

Vedere anche

[Variabile ActivateRecoverMode dalla V2 \(Pagina 280\)](#)

[Variabile Warning dalla V2 \(Pagina 282\)](#)

[Caricamento degli oggetti tecnologici del dispositivo \(Pagina 48\)](#)

10.1.4.8 Modifiche dell'interfaccia PID_Compact a partire da V2

La seguente tabella indica le modifiche dell'interfaccia dell'istruzione PID_Compact.

PID_Compact V1	PID_Compact a partire da V2	Modifica
Input_PER	Input_PER	Tipo di dati da Word a Int
	Disturbance	Nuovo
	ErrorAck	Nuovo
	ModeActivate	Nuovo
Output_PER	Output_PER	Tipo di dati da Word a Int
Error	ErrorBits	Rinominato
	Error	Nuovo
	Mode	Nuovo
sb_RunModeByStartup	RunModeByStartup	Funzione
	IntegralResetMode	
	OverwriteInitialOutputValue	Nuovo
	SetSubstituteOutput	Nuovo
	CancelTuningLevel	Nuovo
	SubstituteOutput	Nuovo

La tabella seguente mostra quali variabili sono state rinominate:

PID_Compact V1.x	PID_Compact a partire da V2
sb_GetCycleTime	CycleTime.StartEstimation
sb_EnCyclEstimation	CycleTime.EnEstimation
sb_EnCyclMonitoring	CycleTime.EnMonitoring
sb_RunModeByStartup	RunModeByStartup
si_Unit	PhysicalUnit

PID_Compact V1.x	PID_Compact a partire da V2
si_Type	PhysicalQuantity
sd_Warning	Warning
sBackUp.r_Gain	CtrlParamsBackUp.Gain
sBackUp.r_Ti	CtrlParamsBackUp.Ti
sBackUp.r_Td	CtrlParamsBackUp.Td
sBackUp.r_A	CtrlParamsBackUp.TdFiltRatio
sBackUp.r_B	CtrlParamsBackUp.PWeighting
sBackUp.r_C	CtrlParamsBackUp.DWeighting
sBackUp.r_Cycle	CtrlParamsBackUp.Cycle
sPid_Calc.r_Cycle	CycleTime.Value
sPid_Calc.b_RunIn	PIDSelfTune.TIR.RunIn
sPid_Calc.b_CalcParamSUT	PIDSelfTune.SUT.CalculateParams
sPid_Calc.b_CalcParamTIR	PIDSelfTune.TIR.CalculateParams
sPid_Calc.i_CtrlTypeSUT	PIDSelfTune.SUT.TuneRule
sPid_Calc.i_CtrlTypeTIR	PIDSelfTune.TIR.TuneRule
sPid_Calc.r_Progress	Progress
sPid_Cmpt.r_Sp_Hlm	Config.SetpointUpperLimit
sPid_Cmpt.r_Sp_Llm	Config.SetpointLowerLimit
sPid_Cmpt.r_Pv_Norm_IN_1	Config.InputScaling.LowerPointIn
sPid_Cmpt.r_Pv_Norm_IN_2	Config.InputScaling.UpperPointIn
sPid_Cmpt.r_Pv_Norm_OUT_1	Config.InputScaling.LowerPointOut
sPid_Cmpt.r_Pv_Norm_OUT_2	Config.InputScaling.UpperPointOut
sPid_Cmpt.r_Lmn_Hlm	Config.OutputUpperLimit
sPid_Cmpt.r_Lmn_Llm	Config.OutputLowerLimit
sPid_Cmpt.b_Input_PER_On	Config.InputPerOn
sPid_Cmpt.b_LoadBackUp	LoadBackUp
sPid_Cmpt.b_InvCtrl	Config.InvertControl
sPid_Cmpt.r_Lmn_Pwm_PPTm	Config.MinimumOnTime
sPid_Cmpt.r_Lmn_Pwm_PBTm	Config.MinimumOffTime
sPid_Cmpt.r_Pv_Hlm	Config.InputUpperLimit
sPid_Cmpt.r_Pv_Llm	Config.InputLowerLimit
sPid_Cmpt.r_Pv_HWrn	Config.InputUpperWarning
sPid_Cmpt.r_Pv_LWrn	Config.InputLowerWarning
sParamCalc.i_Event_SUT	PIDSelfTune.SUT.State
sParamCalc.i_Event_TIR	PIDSelfTune.TIR.State
sRet.i_Mode	sRet.i_Mode è stato eliminato. Il modo di funzionamento viene commutato da Mode e ModeActivate.
sRet.r_Ctrl_Gain	Retain.CtrlParams.Gain
sRet.r_Ctrl_Ti	Retain.CtrlParams.Ti

PID_Compact V1.x	PID_Compact a partire da V2
sRet.r_Ctrl_Td	Retain.CtrlParams.Td
sRet.r_Ctrl_A	Retain.CtrlParams.TdFiltRatio
sRet.r_Ctrl_B	Retain.CtrlParams.PWeighting
sRet.r_Ctrl_C	Retain.CtrlParams.DWeighting
sRet.r_Ctrl_Cycle	Retain.CtrlParams.Cycle

10.1.4.9 Parametri State e Mode a partire da V2

Relazione tra i parametri

Il parametro State indica il modo di funzionamento attuale del regolatore PID. Non è possibile modificarlo.

Con un fronte di salita in ModeActivate PID_Compact passa al modo di funzionamento salvato nel parametro di transito Mode.

Se la CPU si accende o passa da STOP a RUN, PID_Compact si avvia nel modo di funzionamento salvato in Mode. Per lasciare PID_Compact nel modo di funzionamento "Inattivo" impostare RunModeByStartup = FALSE.

Significato dei valori

State / Mode	Descrizione del modo di funzionamento
0	Inattivo Nella modalità "Inattivo" viene sempre emesso il valore 0.0, indipendentemente da Config.OutputUpperLimit e Config.OutputLowerLimit. La modulazione di larghezza di impulso è disattivata.
1	Ottimizzazione iniziale L'ottimizzazione iniziale rileva la risposta del processo a un gradino del valore di uscita e cerca il punto di flesso. Dalla pendenza max. e dal tempo morto del sistema regolato vengono calcolati i parametri PID. I parametri PID migliori si ottengono eseguendo l'ottimizzazione iniziale e l'ottimizzazione fine. Requisiti per l'ottimizzazione iniziale: • Modo di funzionamento Inattivo (State = 0), funzionamento manuale (State = 4) o automatico (State = 3) • ManualEnable = FALSE • Reset = FALSE • Il valore istantaneo non deve essere troppo vicino al setpoint. $\text{Setpoint} - \text{Input} > 0.3 * \text{Config.InputUpperLimit} - \text{Config.InputLowerLimit}$ e $\text{Setpoint} - \text{Input} > 0.5 * \text{Setpoint}$ • Il setpoint e il valore istantaneo sono compresi entro i limiti configurati. Più stabile è il valore istantaneo, più facilmente e con maggiore precisione si possono determinare i parametri PID. Un rumore del valore istantaneo è accettabile fintanto che l'incremento del valore è significativamente maggiore rispetto al rumore. Il setpoint viene congelato nella variabile CurrentSetpoint. L'ottimizzazione viene interrotta se: • $\text{Setpoint} > \text{CurrentSetpoint} + \text{CancelTuningLevel}$ o • $\text{Setpoint} < \text{CurrentSetpoint} - \text{CancelTuningLevel}$ Prima che i parametri PID vengano ricalcolati, vengono salvati e possono essere riattivati con LoadBackUp. Se l'ottimizzazione iniziale si conclude correttamente si passa al funzionamento automatico, in caso contrario la commutazione del modo di funzionamento dipende da ActivateRecoverMode. La fase dell'ottimizzazione iniziale viene visualizzata con PIDSelfTune.SUT.State.

State / Mode	Descrizione del modo di funzionamento
	Per avviare l'ottimizzazione iniziale dal funzionamento automatico si consiglia di eseguire la modifica richiesta del setpoint contemporaneamente al fronte di salita in ModeActivate. Se il setpoint viene modificato prima dell'avvio dell'ottimizzazione iniziale, il valore di uscita viene adattato di conseguenza nel funzionamento automatico, causando una modifica del valore istantaneo. Questo può avere ripercussioni negative sulla successiva esecuzione dell'ottimizzazione iniziale o impedirne l'avvio.
2	ottimizzazione fine L'ottimizzazione fine genera un'oscillazione costante e limitata del valore istantaneo. I parametri PID vengono ricalcolati partendo dall'ampiezza e dalla frequenza dell'oscillazione. I parametri PID risultanti dall'ottimizzazione fine mostrano in genere un comportamento pilota e di disturbo migliore rispetto ai parametri PID dell'ottimizzazione iniziale. I parametri PID migliori si ottengono eseguendo l'ottimizzazione iniziale e l'ottimizzazione fine. PID_Compact tenta di generare automaticamente un'oscillazione che sia maggiore rispetto al fruscio del valore istantaneo. L'ottimizzazione fine viene influenzata solo minimamente dalla stabilità del valore istantaneo. Il setpoint viene congelato nella variabile CurrentSetpoint. L'ottimizzazione viene interrotta se: - Setpoint > CurrentSetpoint + CancelTuningLevel - o - Setpoint < CurrentSetpoint - CancelTuningLevel Prima che i parametri PID vengano ricalcolati, vengono salvati e possono essere riattivati con LoadBackUp. Requisiti per l'ottimizzazione fine: - Non sono previste anomalie. - Il setpoint e il valore istantaneo sono compresi entro i limiti configurati. - ManualEnable = FALSE - Reset = FALSE - Tipo di funzionamento automatico (State = 3), inattivo (State = 0) o manuale (State = 4) rispettivamente nelle modalità descritte di seguito: - Funzionamento automatico (State = 3) Per migliorare i parametri PID esistenti mediante un'ottimizzazione, avviare l'ottimizzazione fine dal modo di funzionamento automatico. PID_Compact esegue la regolazione con i parametri PID esistenti finché il circuito di regolazione non si assesta e finché non sono soddisfatti i requisiti per un'ottimizzazione fine. Solo in seguito viene avviata l'ottimizzazione fine. - Funzionamento Inattivo (State = 0) o manuale (State = 4) Se sono soddisfatti i presupposti dell'ottimizzazione iniziale, viene avviata un'ottimizzazione iniziale. Con i parametri PID rilevati viene quindi effettuata una regolazione finché il circuito di regolazione è assestato e sono soddisfatti i requisiti per l'ottimizzazione fine. Se il valore istantaneo per un'ottimizzazione iniziale si trova già troppo vicino al setpoint o se PIDSelfTune.TIR.RunIn = TRUE, si tenta di raggiungere il setpoint con il valore di uscita minimo o massimo. Ciò può causare una maggiore sovralongazione. Solo in seguito viene avviata l'ottimizzazione fine. Se l'ottimizzazione fine si conclude correttamente il regolatore passa al funzionamento automatico, in caso contrario la commutazione del modo di funzionamento dipende da ActivateRecoverMode. La fase di "Ottimizzazione fine" viene visualizzata con PIDSelfTune.TIR.State.
3	Funzionamento automatico In funzionamento automatico PID_Compact regola il sistema regolato secondo i parametri preimpostati. Se è soddisfatta una delle seguenti condizioni si passa al funzionamento automatico: - Ottimizzazione iniziale conclusa correttamente - ottimizzazione fine conclusa correttamente - Modifica del parametro di transito Mode al valore 3 e un fronte di salita in ModeActivate. La commutazione dal funzionamento automatico al funzionamento manuale è bumpless solo nell'editor di messa in servizio. In funzionamento automatico viene tenuta in considerazione la variabile ActivateRecoverMode.

State / Mode	Descrizione del modo di funzionamento
4	Funzionamento manuale Nel funzionamento manuale l'utente imposta manualmente un valore di uscita nel parametro ManualValue. Il modo di funzionamento si può attivare anche con ManualEnable = TRUE. Si consiglia di modificare i modi di funzionamento solo tramite Mode e ModeActivate. La commutazione dal funzionamento manuale al funzionamento automatico è bumpless. Il funzionamento manuale è possibile anche in presenza di un errore.
5	Valore di uscita sostitutivo con controllo errori L'algoritmo di regolazione è disattivato. La variabile SetSubstituteOutput stabilisce quale valore di uscita venga emesso in questo modo di funzionamento. - SetSubstituteOutput = FALSE: ultimo valore di uscita valido - SetSubstituteOutput = TRUE: Valore di uscita sostitutivo Questa modalità di funzionamento non può essere attivata con Mode = 5 . Viene attivata in caso di errore invece del modo di funzionamento "Inattivo" se sono soddisfatte tutte le seguenti condizioni: - Funzionamento automatico (Mode = 3) - ActivateRecoverMode = TRUE - Si sono verificati uno o più errori nei quali è attiva ActivateRecoverMode. Non appena l'errore viene eliminato PID_Compact torna al funzionamento automatico.

Comportamento ENO

Se State = 0, ENO = FALSE.

Se State ≠ 0, ENO = TRUE.

Commutazione automatica del modo di funzionamento durante la messa in servizio

Al termine dell'ottimizzazione iniziale e dell'ottimizzazione fine senza errori viene attivato il funzionamento automatico. La tabella seguente mostra come cambiano Mode e State durante un'ottimizzazione iniziale corretta.

N° di ciclo	Mode	State	Operazione
0	4	4	Imposta Mode = 1
1	1	4	Imposta ModeActivate = TRUE
1	4	1	Il valore di State viene salvato in Mode L'ottimizzazione iniziale viene avviata
n	4	1	Ottimizzazione iniziale riuscita
n	3	3	Il funzionamento automatico viene avviato

In caso di errore PID_Compact cambia automaticamente modo di funzionamento. La tabella seguente mostra come cambiano Mode e State durante un'ottimizzazione iniziale scorretta.

N° di ciclo	Mode	State	Operazione
0	4	4	Imposta Mode = 1
1	1	4	Imposta ModeActivate = TRUE

N° di ciclo	Mode	State	Operazione
1	4	1	Il valore di State viene salvato in Mode L'ottimizzazione iniziale viene avviata
n	4	1	Ottimizzazione iniziale interrotta
n	4	4	Il funzionamento manuale viene avviato

Se ActivateRecoverMode = TRUE viene attivato il modo di funzionamento salvato in Mode. All'avvio dell'ottimizzazione iniziale o fine PID_Compact ha salvato il valore di State nel parametro di transito Mode. PID_Compact passa quindi al modo di funzionamento dal quale è stata avviata l'ottimizzazione.

Se ActivateRecoverMode = FALSE si passa al modo di funzionamento "Inattivo".

Vedere anche

[Parametri di uscita di PID_Compact dalla V2 \(Pagina 269\)](#)

10.1.4.10 Parametro ErrorBits dalla V2

Se si verificano diversi errori contemporaneamente, i valori di ErrorBits vengono visualizzati in un'addizione binaria. Se ad es. viene visualizzato ErrorBits = 16#0000_0003 significa che sono presenti contemporaneamente gli errori 16#0000_0001 e 16#0000_0002.

PID_Compact utilizza ManualValue come valore di uscita in funzionamento manuale. L'unica eccezione è Errorbits = 16#0001_0000.

ErrorBits (DW#16#...)	Descrizione
0000_0000	Non sono presenti errori.
0000_0001	Il parametro "Input" non rientra nei limiti del valore istantaneo. - Input > Config.InputUpperLimit o - Input < Config.InputLowerLimit Se prima che si verificasse l'errore era attivo il funzionamento automatico e ActivateRecoverMode = TRUE, PID_Compact resta in funzionamento automatico. Se prima che si verificasse l'errore era attivo il modo di funzionamento ottimizzazione iniziale o ottimizzazione fine e ActivateRecoverMode = TRUE, PID_Compact passa al modo di funzionamento salvato in Mode.
0000_0002	Valore del parametro "Input_PER" non valido. Controllare l'eventuale presenza di un errore sull'ingresso analogico. Se prima che si verificasse l'errore era attivo il funzionamento automatico e ActivateRecoverMode = TRUE, PID_Compact emette il valore di uscita sostitutivo configurato. Non appena l'errore viene eliminato PID_Compact torna al funzionamento automatico. Se prima che si verificasse l'errore era attivo il modo di funzionamento ottimizzazione iniziale o ottimizzazione fine e ActivateRecoverMode = TRUE, PID_Compact passa al modo di funzionamento salvato in Mode.
0000_0004	Errore durante l'ottimizzazione fine. Non è stato possibile mantenere l'oscillazione del valore istantaneo. Se ActivateRecoverMode = TRUE prima che si verificasse l'errore, PID_Compact interrompe l'ottimizzazione e passa al modo di funzionamento salvato in Mode.
0000_0008	Errore all'avvio dell'ottimizzazione iniziale. Il valore istantaneo è troppo vicino al setpoint. Avviare l'ottimizzazione fine. Se ActivateRecoverMode = TRUE prima che si verificasse l'errore, PID_Compact interrompe l'ottimizzazione e passa al modo di funzionamento salvato in Mode.
0000_0010	Il setpoint è stato modificato durante l'ottimizzazione. Nella variabile CancelTuningLevel è possibile impostare l'oscillazione ammessa del setpoint. Se ActivateRecoverMode = TRUE prima che si verificasse l'errore, PID_Compact interrompe l'ottimizzazione e passa al modo di funzionamento salvato in Mode.

ErrorBits (DW#16#...)	Descrizione
0000_0020	L'ottimizzazione iniziale non è consentita durante l'ottimizzazione fine. Se ActivateRecoverMode = TRUE prima che si verificasse l'errore, PID_Compact resta nel modo di funzionamento ottimizzazione fine.
0000_0080	Errore nell'ottimizzazione iniziale. I limiti del valore di uscita non sono configurati correttamente e il valore istantaneo non reagisce come previsto. Verificare l'esistenza dei seguenti presupposti: • Che i limiti del valore di uscita siano configurati correttamente e siano compatibili con il senso di regolazione. • Che sia possibile una modifica del valore di uscita in modo che il valore istantaneo si avvicini al setpoint. Che il valore di uscita non venga limitato ancor prima dell'avvio dell'ottimizzazione iniziale per effetto del limite del valore di uscita corrispondente. Esempio: nella direzione di regolazione normale, e con un valore istantaneo che si colloca al di sotto del setpoint, il valore di uscita non deve aver raggiunto il limite superiore ancor prima dell'avvio dell'ottimizzazione iniziale. • Che il valore istantaneo non presenti forti oscillazione prima dell'avvio dell'ottimizzazione iniziale. Per avviare l'ottimizzazione iniziale dal funzionamento automatico si consiglia di eseguire la modifica richiesta del setpoint contemporaneamente al fronte di salita in ModeActivate. In questo modo si evita che il valore di uscita raggiunga la limitazione tra la modifica al setpoint e l'avvio dell'ottimizzazione. In via alternativa è possibile raggiungere questo scopo anche effettuando l'avvio nel funzionamento manuale o nel modo di funzionamento "Inattivo". Se prima che si verificasse l'errore ActivateRecoverMode = TRUE, PID_Compact interrompe l'ottimizzazione e passa al modo di funzionamento memorizzato in Mode.
0000_0100	Un errore durante l'ottimizzazione fine ha causato parametri non validi. Se ActivateRecoverMode = TRUE prima che si verificasse l'errore, PID_Compact interrompe l'ottimizzazione e passa al modo di funzionamento salvato in Mode.
0000_0200	Valore del parametro "Input" non valido: il formato numerico del valore non è valido. Se prima che si verificasse l'errore era attivo il funzionamento automatico e ActivateRecoverMode = TRUE, PID_Compact emette il valore di uscita sostitutivo configurato. Non appena l'errore viene eliminato PID_Compact torna al funzionamento automatico. Se prima che si verificasse l'errore era attivo il modo di funzionamento ottimizzazione iniziale o ottimizzazione fine e ActivateRecoverMode = TRUE, PID_Compact passa al modo di funzionamento salvato in Mode.
0000_0400	Calcolo del valore di uscita non riuscito. Controllare i parametri PID. Se prima che si verificasse l'errore era attivo il funzionamento automatico e ActivateRecoverMode = TRUE, PID_Compact emette il valore di uscita sostitutivo configurato. Non appena l'errore viene eliminato PID_Compact torna al funzionamento automatico. Se prima che si verificasse l'errore era attivo il modo di funzionamento ottimizzazione iniziale o ottimizzazione fine e ActivateRecoverMode = TRUE, PID_Compact passa al modo di funzionamento salvato in Mode.
0000_0800	Errore del tempo di campionamento: PID_Compact non viene richiamato entro il tempo di campionamento dell'OB di schedulazione orologio. Si raccomanda di richiamare PID_Compact in un OB di schedulazione orologio senza condizioni e di eseguire la relativa attivazione o disattivazione tramite il modo di funzionamento nel parametro Mode. Richiami condizionati o il richiamo nell'OB1 possono ripercuotersi negativamente sulla qualità della regolazione. La sorveglianza del tempo di campionamento può essere disattivata con CycleTime.EnMonitoring = FALSE. Se prima che si verificasse l'errore era attivo il funzionamento automatico e ActivateRecoverMode = TRUE, PID_Compact resta in funzionamento automatico. Se prima che si verificasse l'errore era attivo il modo di funzionamento ottimizzazione iniziale o ottimizzazione fine e ActivateRecoverMode = TRUE, PID_Compact passa al modo di funzionamento salvato in Mode. Se quest'errore si è verificato nella simulazione con PLCSIM, osservare le avvertenze riportate alla sezione Simulazione di PID_Compact dalla V2 con PLCSIM (Pagina 105).

ErrorBits (DW#16#...)	Descrizione
0000_1000	Valore del parametro "Setpoint" non valido: il formato numerico del valore non è valido. Se prima che si verificasse l'errore era attivo il funzionamento automatico e ActivateRecoverMode = TRUE, PID_Compact emette il valore di uscita sostitutivo configurato. Non appena l'errore viene eliminato PID_Compact torna al funzionamento automatico. Se prima che si verificasse l'errore era attivo il modo di funzionamento ottimizzazione iniziale o ottimizzazione fine e ActivateRecoverMode = TRUE, PID_Compact passa al modo di funzionamento salvato in Mode.
0001_0000	Valore del parametro ManualValue non valido. Il formato numerico del valore non è valido. Se prima che si verificasse l'errore ActivateRecoverMode = TRUE, PID_Compact utilizza SubstituteOutput come valore di uscita. Non appena si predefinisce un valore valido in ManualValue, PID_Compact lo utilizza come valore di uscita.
0002_0000	Valore della variabile SubstituteOutput non valido. Il formato numerico del valore non è valido. PID_Compact utilizza il limite inferiore del valore di uscita come valore di uscita. Se prima che si verificasse l'errore era attivo il funzionamento automatico, ActivateRecoverMode = TRUE e l'errore non è più presente, PID_Compact torna al funzionamento automatico.
0004_0000	Valore del parametro Disturbance non valido. Il formato numerico del valore non è valido. Se prima che si verificasse l'errore era attivo il funzionamento automatico e ActivateRecoverMode = TRUE Disturbance viene impostato a zero. PID_Compact resta in funzionamento automatico. Se prima che si verificasse l'errore era attivo il modo di funzionamento ottimizzazione iniziale o ottimizzazione fine e ActivateRecoverMode = TRUE, PID_Compact passa al modo di funzionamento salvato in Mode. Se Disturbance non ha alcuna influenza sul valore di uscita nella fase attuale, l'ottimizzazione non viene interrotta.

10.1.4.11 Variabile ActivateRecoverMode dalla V2

La variabile ActivateRecoverMode determina il comportamento in caso di errore. Il parametro Error indica se attualmente è presente un errore. Se l'errore non è più presente, compare Error = FALSE. Il parametro ErrorBits mostra gli errori che si sono verificati.

Funzionamento automatico

ATTENZIONE

L'impianto può subire danneggiamenti.

Se ActivateRecoverMode = TRUE, in caso di errore PID_Compact rimane in funzionamento automatico anche se vengono superati i limiti del valore istantaneo. In seguito a ciò l'impianto può subire danneggiamenti.

Configurare per il sistema regolato un comportamento da adottare in caso di errore che protegga l'impianto da danneggiamenti.

ActivateRecover-Mode	Descrizione
FALSE	In caso di errore PID_Compact passa al modo di funzionamento "Inattivo". Il regolatore viene attivato solo in presenza di un fronte di discesa in Reset o di un fronte di salita in ModeActivate.
TRUE	Se si verificano errori frequenti in funzionamento automatico, questa impostazione peggiora il comportamento del regolatore perché PID_Compact commuta tra valore di uscita calcolato e valore di uscita sostitutivo a ogni errore. In questo caso controllare il parametro ErrorBits ed eliminare la causa dell'errore. Se si verificano uno o più dei seguenti errori PID_Compact resta in funzionamento automatico: • 0001h: Il parametro "Input" non rientra nei limiti del valore istantaneo. • 0800h: Errore del tempo di campionamento • 40000h: Valore del parametro Disturbance non valido. Se si verificano uno o più dei seguenti errori, PID_Compact commuta in modo di funzionamento "valore di uscita sostitutivo con controllo errori": • 0002h: Valore del parametro Input_PER non valido. • 0200h: Valore del parametro Input non valido. • 0400h: Calcolo del valore di uscita non riuscito. • 1000h: Valore del parametro Setpoint non valido. Se si verifica l'errore seguente, PID_Compact passa al modo di funzionamento "Valore di uscita sostitutivo con controllo errori" e sposta l'organo attuatore su Config.OutputLowerLimit: • 20000h: Valore della variabile SubstituteOutput non valido. Il formato numerico del valore non è valido. Questo comportamento è indipendente da SetSubstituteOutput. Non appena l'errore viene eliminato PID_Compact torna al funzionamento automatico.

Ottimizzazione iniziale e ottimizzazione fine

ActivateRecover-Mode	Descrizione
FALSE	In caso di errore PID_Compact passa al modo di funzionamento "Inattivo". Il regolatore viene attivato solo in presenza di un fronte di discesa in Reset o di un fronte di salita in ModeActivate.
TRUE	Se si verifica l'errore seguente PID_Compact resta nel modo di funzionamento attivo: • 0020h: L'ottimizzazione iniziale non è consentita durante l'ottimizzazione fine. I seguenti errori vengono ignorati: • 10000h: Valore del parametro ManualValue non valido. • 20000h: Valore della variabile SubstituteOutput non valido. Con tutti gli altri errori PID_Compact interrompe l'ottimizzazione e passa al modo di funzionamento dal quale è stata avviata l'ottimizzazione.

Funzionamento manuale

In funzionamento manuale ActivateRecoverMode è inefficace.

10.1.4.12 Variabile Warning dalla V2

Se sono presenti più avvisi contemporaneamente, i valori della variabile Warning vengono visualizzati in un'addizione binaria. Se ad es. viene visualizzato l'avviso 16#0000_0003 significa che sono presenti contemporaneamente gli avvisi 0000_0001 e 0000_0002.

Warning (DW#16#....)	Descrizione
0000_0000	Non è presente alcun avviso.
0000_0001	Durante l'ottimizzazione iniziale non è stato trovato il punto di flesso.
0000_0004	Il setpoint è stato circoscritto ai limiti impostati.
0000_0008	Per il metodo di calcolo selezionato non sono state definite tutte le proprietà necessarie per il sistema regolato. In cambio sono stati calcolati i parametri PID con il metodo TIR.TuneRule = 3.
0000_0010	Non è stato possibile commutare il modo di funzionamento perché Reset = TRUE o ManualEnable = TRUE.
0000_0020	Il tempo di campionamento dell'algoritmo PID viene circoscritto dal tempo di ciclo dell'OB richiamante. Per ottenere risultati migliori utilizzare tempi di ciclo dell'OB più brevi.
0000_0040	Il valore istantaneo ha superato uno dei suoi limiti di avviso.
0000_0080	Valore di Mode non valido. Il modo di funzionamento non viene commutato.
0000_0100	Il valore manuale è stato circoscritto ai limiti dell'uscita del regolatore.
0000_0200	La regola per l'ottimizzazione indicata non è supportata. Non vengono calcolati parametri PID.
0000_1000	Non è possibile raggiungere il valore di uscita sostitutivo perché non rientra nei limiti del valore di uscita.

I seguenti avvisi vengono cancellati non appena viene eliminata la causa:

- 16#0000_0001
- 16#0000_0004
- 16#0000_0008
- 16#0000_0040
- 16#0000_0100

Tutti gli altri avvisi vengono cancellati in presenza di un fronte di salita in Reset o ErrorAck.

10.1.4.13 Variabile IntegralResetMode dalla V2

La variabile IntegralResetMode descrive le modalità di assegnazione dei valori al componente I PIDCtrl.IntegralSum:

- alla commutazione dal modo di funzionamento "Inattivo" al modo "Funzionamento automatico"
- con il fronte TRUE -> FALSE sui parametri Reset e Mode = 3

Questa impostazione è attiva solo per un ciclo e soltanto se è attivato il componente I (variabile Retain.CtrlParams.Ti > 0.0)

IntegralReset-Mode	Descrizione
0	Livella Al parametro PIDCtrl.IntegralSum viene preassegnato un valore tale da consentire la commutazione continua, in altri termini, il funzionamento automatico si avvia presupponendo un valore di uscita = 0.0 (parametro Output). Non si avrà alcun gradino del valore di uscita indipendentemente dalla differenza di regolazione (setpoint – valore istantaneo).
1	Elimina Se viene utilizzata quest'opzione, si raccomanda di impostare su 1.0 la ponderazione del componente P (Retain.CtrlParams.PWeighting). Il valore di PIDCtrl.IntegralSum viene cancellato. Se è presente una differenza di regolazione, si forma un gradino del valore di uscita. La direzione del salto del valore di uscita dipende dalla ponderazione configurata per il componente P (variabile Retain.CtrlParams.PWeighting) e dalla differenza di regolazione: • Ponderazione del componente P = 1.0: La direzione del salto del valore di uscita e la differenza di regolazione hanno segni identici. Esempio: Se il valore istantaneo è inferiore al setpoint (differenza di regolazione positiva), il valore di uscita salta su un valore positivo. • Ponderazione del componente P < 1.0: In presenza di differenze di regolazione di una certa entità, il salto del valore di uscita e la differenza di regolazione hanno segni identici. Esempio: Se il valore istantaneo è di gran lunga inferiore al setpoint (differenza di regolazione positiva), il valore di uscita salta su un valore positivo. In presenza di lievi differenze di regolazione, il salto del valore di uscita e la differenza di regolazione hanno segni diversi. Esempio: Se il valore istantaneo è leggermente inferiore al setpoint (differenza di regolazione positiva), il valore di uscita salta su un valore negativo. Normalmente questo comportamento non è auspicabile in quanto comporta un incremento temporaneo della differenza di regolazione. Minore è la ponderazione configurata per il componente P, maggiore deve essere la differenza di regolazione al fine di realizzare un salto del valore di uscita con segno identico. Se viene utilizzata quest'opzione, si raccomanda di impostare su 1.0 la ponderazione del componente P (Retain.CtrlParams.PWeighting). Diversamente sussiste il rischio che si verifichi il comportamento indesiderato descritto nel caso delle differenze di regolazione lievi. In via alternativa è possibile utilizzare anche IntegralResetMode = 4. Quest'opzione garantisce segni identici nel salto del valore di uscita e nella differenza di regolazione a prescindere dalla ponderazione configurata per il componente P e dalla differenza di regolazione:

10.1 PID_Compact

IntegralReset-Mode	Descrizione
2	Mantieni Il valore di PIDCtrl.IntegralSum non viene modificato. È possibile impostare un nuovo valore tramite il programma utente.
3	Preassegna Al parametro PIDCtrl.IntegralSumviene preassegnato automaticamente un valore tale da far supporre che nell'ultimo ciclo ci sia stato = Output = OverwriteInitialOutputValue
4	In caso di variazioni del setpoint (soltanto per PID_Compact con la versione ≥ 2.3) Il valore di PIDCtrl.IntegralSum viene preimpostato automaticamente in modo da determinare un salto del valore di uscita analogo a quello di un regolatore PI nel funzionamento automatico in presenza di una transizione del setpoint dal valore istantaneo attuale al setpoint attuale. In presenza di una differenza di regolazione, si forma un salto del valore di uscita. La direzione del salto del valore di uscita e la differenza di regolazione hanno segni identici. Esempio: Se il valore istantaneo è inferiore al setpoint (differenza di regolazione positiva), il valore di uscita salta su un valore positivo. Ciò prescinde dalla ponderazione configurata per il componente P e dalla differenza di regolazione.

Assegnando a IntegralResetMode un valore al di fuori del campo valori valido, PID_Compact assume lo stesso comportamento attivato con la preassegnazione di IntegralResetMode:

- PID_Compact fino a V2.2: IntegralResetMode = 1
- PID_Compact fino a V2.3: IntegralResetMode = 4

Tutte le informazioni sul salto del valore di uscita finora riportate sono riferite al senso di regolazione normale (variabile Config.InvertControl = FALSE). Nel senso di regolazione inverso (Config.InvertControl = TRUE) si avrà un segno opposto nel salto del valore di uscita.

10.1.4.14 Esempio di programma per PID_Compact V2

Nel seguente esempio si procede alla regolazione della temperatura tramite l'oggetto tecnologico dell'istruzione "PID_Compact". La simulazione dei valori della temperatura avviene sulla base di un blocco che riproduce uno stadio di ritardo di terz'ordine (attuatore PT3) I parametri PID dell'oggetto tecnologico vengono creati automaticamente con la prima ottimizzazione.



Memorizzazione dei dati

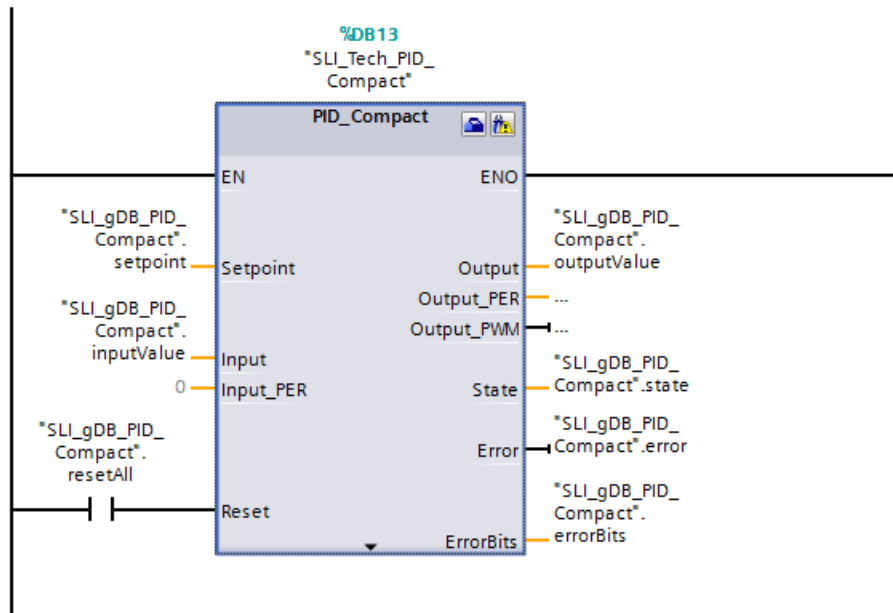
Per la memorizzazione dei dati di interconnessione, creare sette variabili in un blocco dati globale.

SLI_gDB_PID_Compact			
	Name	Data type	Start value
1	Static		
2	setpoint	Real	75.0
3	inputValue	Real	0.0
4	outputValue	Real	0.0
5	state	Int	0
6	error	Bool	false
7	errorBits	DWord	16#0
8	resetAll	Bool	false

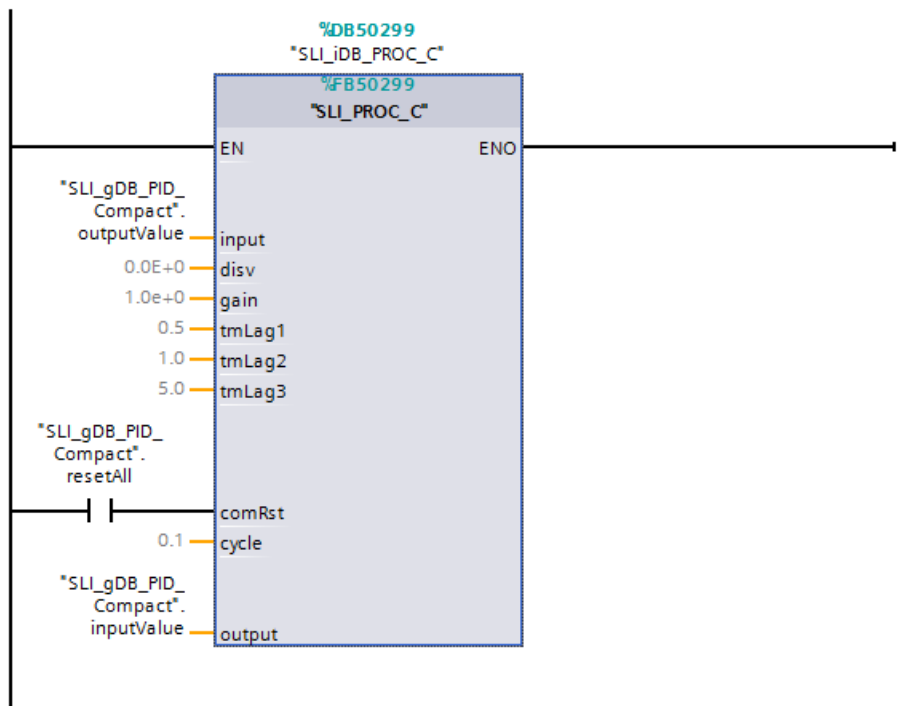
Interconnessione dei parametri

Le seguenti interconnessioni vengono richiamate in un OB di schedulazione orologio.

Segmento 1: I parametri dell'istruzione PID_Compact vengono interconnessi come indicato nel seguito.

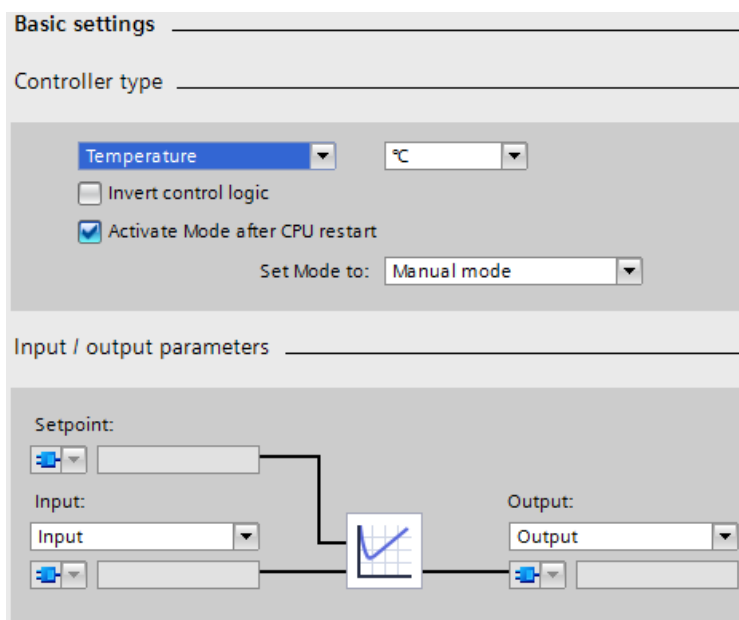


Segmento 2: I parametri del blocco SLI_PROC_C che simula i valori della temperatura vengono interconnessi come indicato nel seguito.



Oggetto tecnologico

La configurazione dell'oggetto tecnologico avviene tramite le proprietà dell'istruzione "PID_Compact" oppure dal percorso Oggetto > Configurazione. Sono di rilievo per l'esempio il modo di regolazione e i parametri di ingresso e uscita. Tramite il modo di regolazione viene eseguita una preselezione dell'unità di misura del valore da regolare. Nell'esempio si ricorre al modo di regolazione "Temperatura" espressa in "°C". I parametri di "PID_Compact" sono già interconnessi con le variabili globali. Di conseguenza è sufficiente l'indicazione sull'utilizzo dei parametri Input e Output.



Procedura di avvio della regolazione

Dopo il caricamento nella CPU PID_Compact si trova nel funzionamento manuale con valore 0.0. Per avviare la regolazione, procedere come indicato nel seguito:

1. Aprire la Messa in servizio dell'oggetto tecnologico "SLI_Tech_PID_Compact".
2. Nell'area "Misura" fare clic sul pulsante "Avvia".



La misura si avvia e PID_Compact può essere attivato.

3. L'ottimizzazione iniziale è stata selezionata.

Nell'area "Modo di ottimizzazione" fare clic sul pulsante "Avvia".

La prima ottimizzazione viene eseguita. I parametri PID vengono adeguati automaticamente nel sistema regolato. Terminata la prima ottimizzazione PID_Compact passa al "funzionamento automatico".

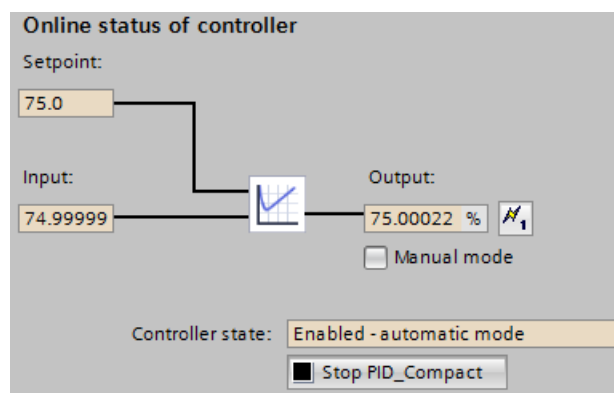
NOTA**Alternativa all'avvio di PID_Compact**

In via alternativa è possibile portare PID_Compact nel funzionamento automatico nell'area "Stato online del regolatore" selezionando i pulsanti "Stop PID_Compact" / "Avvio PID_Compact" senza la prima ottimizzazione. Con quest'operazione il regolatore impiega valori di default per i parametri PID e presenta, per il caso applicativo, un comportamento di regolazione qualitativamente più scadente.

Procedura di arresto della regolazione

Per arrestare e terminare PID_Compact ed il programma, procedere come indicato di seguito:

1. Nell'oggetto tecnologico fare clic su SLI_Tech_PID_Compact nell'area "Stato online del regolatore" selezionando il pulsante "Stop PID_Compact".



L'istruzione "PID_Compact" termina la regolazione ed emette il valore "0.0" come valore regolante.

2. Nell'area "Misura" fare clic sul pulsante "Stop".
3. Per portare il valore istantaneo immediatamente sul valore "0.0", procedere nel modo seguente:

Nel blocco "SLI_OB_PID_Compact" impostare la variabile "resetAll" sul valore "TRUE", quindi sul valore "FALSE".

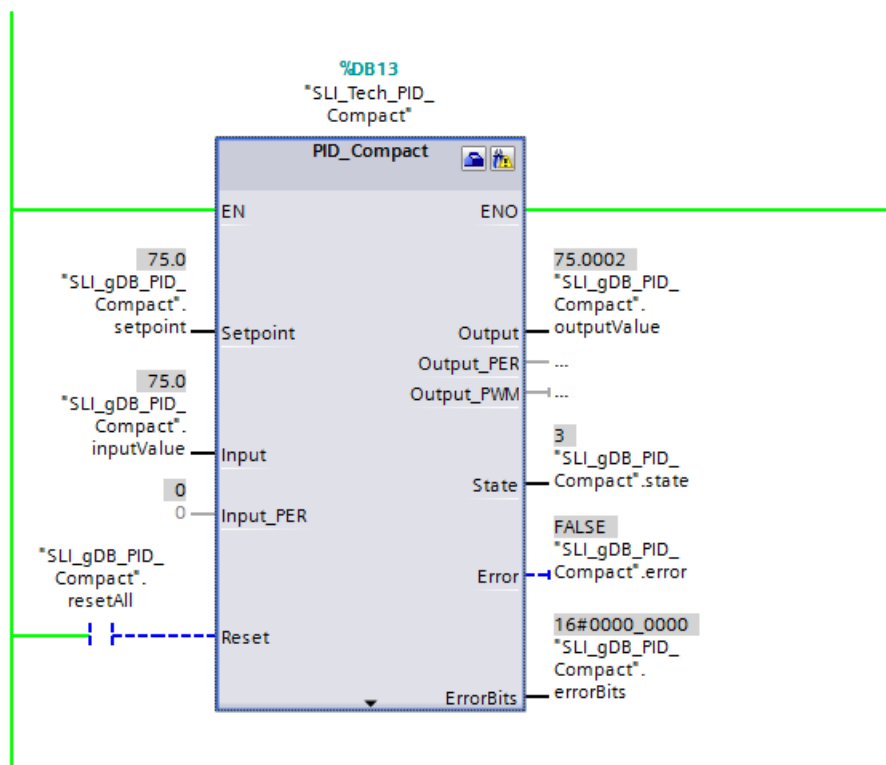
Istruzione PID_Compact

Sul parametro Setpoint ("setpoint") è preimpostato il setpoint per la temperatura da regolare. Se l'istruzione "PID_Compact" è stata avviata dall'oggetto tecnologico, la regolazione viene avviata. L'istruzione "PID_Compact" emette un valore regolante nel parametro di uscita Output ("outputValue"). Dal parametro di ingresso Input ("inputValue") il valore istantaneo della temperatura viene trasmesso all'istruzione "PID_Compact".

In funzione dell'evolversi della differenza tra setpoint ("setpoint") e valore istantaneo ("inputValue") l'istruzione "PID_Compact" adegua il valore regolante ("outputValue"). Questo processo viene ripetuto per consentire al valore istantaneo ("inputValue") di approssimarsi al setpoint ("setpoint") tramite il valore regolante ("outputValue").

Sul parametro di uscita State ("state") viene visualizzato il modo di funzionamento attuale dell'istruzione "PID_Compact". Dopo la prima ottimizzazione (il valore di "state" è "1"), commuta PID_Compact nel funzionamento automatico (il valore è "3").

Il parametro di uscita Error ("error") segnala attualmente l'assenza di errori. In caso di errore il parametro di uscita ErrorBits ("errorBits") fornisce informazioni sul tipo di errore. Al verificarsi di un errore, quest'ultimo può essere confermato nell'oggetto tecnologico, nell'area "Stato dell'ottimizzazione" selezionando il pulsante "ErrorAck".

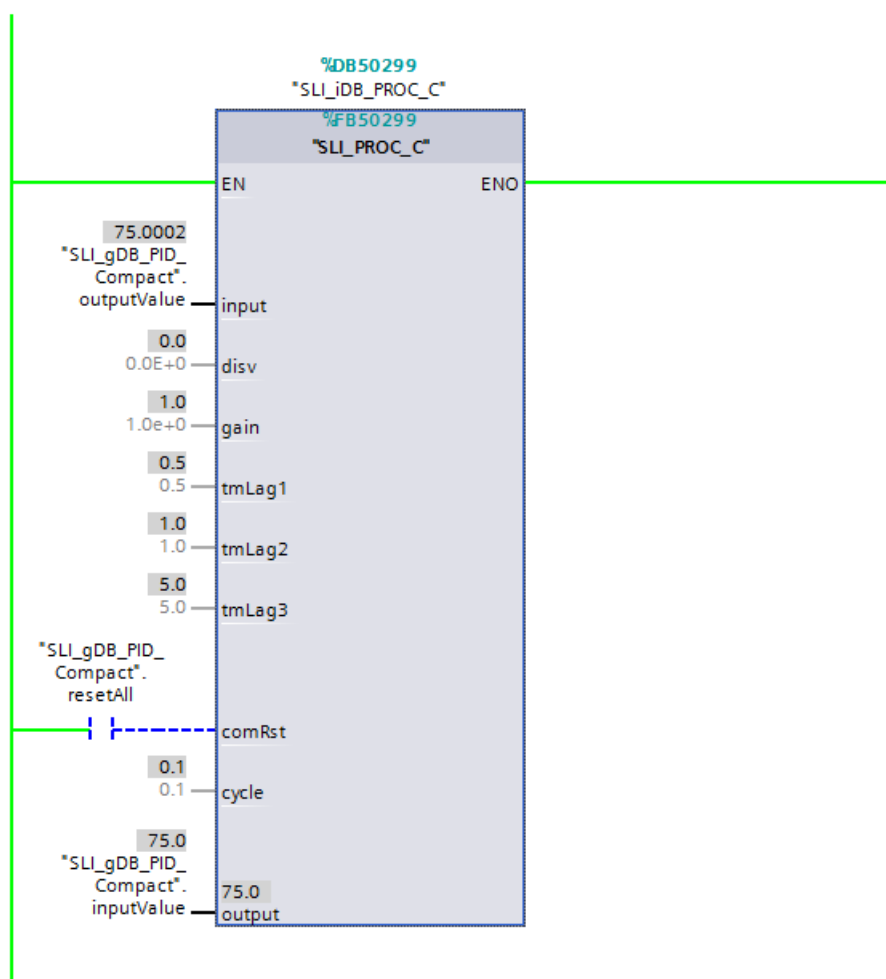


Blocco "SLI_PROC_C"

Il blocco "SLI_PROC_C" simula il valore istantaneo ("inputValue") della temperatura in salita dell'impianto. Il blocco "SLI_PROC_C" ottiene il setpoint del regolatore ("outputValue") e simula il comportamento di temperatura del sistema regolato. La temperatura viene inoltrata come valore istantaneo ("inputValue") nel regolatore.

Una modifica del valore della variabile "resetAll" (del parametro comRst) ha le seguenti ripercussioni:

Parametro comRst ("resetAll")	L'istruzione "PID_Compact" è in corso	L'istruzione PID_Compact è stata arrestata
comRst ("resetAll") rimane sul valor "FALSE"	Il blocco "SLI_PROC_C" emette, sulla base di un valore regolante ("outputValue"), un nuovo valore istantaneo ("inputValue").	Sebbene al blocco "SLI_PROC_C" non venga trasmesso un valore regolante > "0.0", esso continua a fornire un valore istantaneo > "0.0".
comRst ("resetAll"): Modifica di "FALSE" sul valore "TRUE"	Sia il valore regolante ("outputValue") sia il valore istantaneo emesso ("inputValue") vengono resettati su "0.0".	Il valore istantaneo indicato ("inputValue") della temperatura del blocco "SLI_PROC_C" viene resettato a "0.0".
comRst ("resetAll"): Modifica di "TRUE" sul valore "FALSE"	La regolazione della temperatura riprende dall'inizio.	Il valore istantaneo / la temperatura ("inputValue") rimane "0.0".



Codice di programma

Ulteriori informazioni sul codice di programma dell'esempio sopra menzionato si trovano alla voce "Sample Library for Instructions".

10.1.5 PID_Compact V1

10.1.5.1 Descrizione di PID_Compact V1

Descrizione

L'istruzione PID_Compact mette a disposizione un regolatore PID con ottimizzazione integrata per il funzionamento automatico e manuale.

Richiamo

L'istruzione PID_Compact viene richiamata ad intervalli costanti entro il tempo di ciclo dell'OB richiamante (di preferenza in un OB di schedulazione orologio).

Caricamento nel dispositivo

L'aggiornamento dei valori attuali delle variabili a ritenzione, avviene soltanto con il caricamento completo di PID_Compact.

Caricamento degli oggetti tecnologici del dispositivo ([Pagina 48](#))

Avviamento

Quando si avvia la CPU, PID_Compact viene avviato nell'ultimo modo di funzionamento attivo. Per lasciare PID_Compact nel modo di funzionamento "Inattivo" impostare sb_RunModeByStartup = FALSE.

Sorveglianza del tempo di campionamento PID_Compact

Il tempo di campionamento dovrebbe corrispondere al tempo di ciclo dell'OB richiamante. L'istruzione PID_Compact misura l'intervallo tra due richiami. Questo è il tempo di campionamento attuale. Ad ogni commutazione del modo di funzionamento e al primo avvio viene generato il valore medio dei primi 10 tempi di campionamento. Se il tempo di campionamento attuale si discosta troppo dal valore medio si verifica un errore (Error = 0800 hex) e PID_Compact passa al modo di funzionamento "Inattivo".

La commutazione di PID_Compact a partire dalla versione 1.1 nel modo di funzionamento "Inattivo" durante l'ottimizzazione avviene alle seguenti condizioni:

- Nuovo valore medio $\geq 1,1 \times$ valore medio precedente
- Nuovo valore medio $\leq 0,9 \times$ valore medio precedente

La commutazione di PID_Compact a partire dalla versione 1.1 nel modo di funzionamento "Inattivo" in funzionamento automatico avviene alle seguenti condizioni:

- Nuovo valore medio $\geq 1,5 \times$ valore medio precedente
- Nuovo valore medio $\leq 0,5 \times$ valore medio precedente

La commutazione di PID_Compact 1.0 nel modo di funzionamento "Inattivo" durante l'ottimizzazione e il funzionamento automatico avviene alle seguenti condizioni:

- Nuovo valore medio $\geq 1,1 \times$ valore medio precedente
- Nuovo valore medio $\leq 0,9 \times$ valore medio precedente
- Tempo di campionamento attuale $\geq 1,5 \times$ valore medio attuale
- Tempo di campionamento attuale $\leq 0,5 \times$ valore medio attuale

Tempo di campionamento dell'algoritmo PID

Poiché il sistema regolato richiede un certo tempo per reagire a una modifica del valore di uscita, è utile non calcolare questo valore in ogni ciclo. Il tempo di campionamento dell'algoritmo PID è il tempo che trascorre tra due calcoli del valore di uscita. Viene determinato durante l'ottimizzazione e arrotondato ad un multiplo del tempo di ciclo. Tutte le altre funzioni di PID_Compact vengono eseguite ad ogni richiamo.

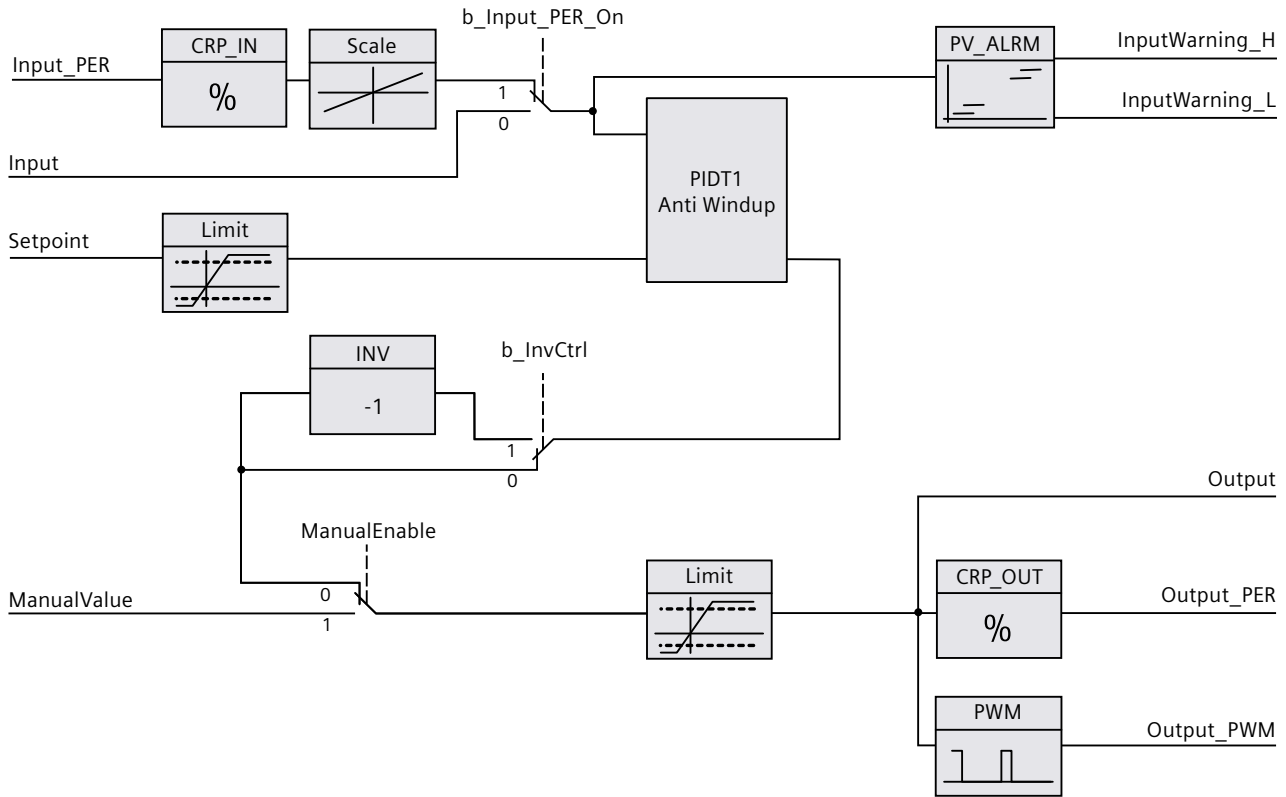
Algoritmo PID

PID_Compact è un regolatore PIDT1 con anti-windup e ponderazione del componente P e D. Il valore di uscita viene calcolato sulla base della seguente formula:

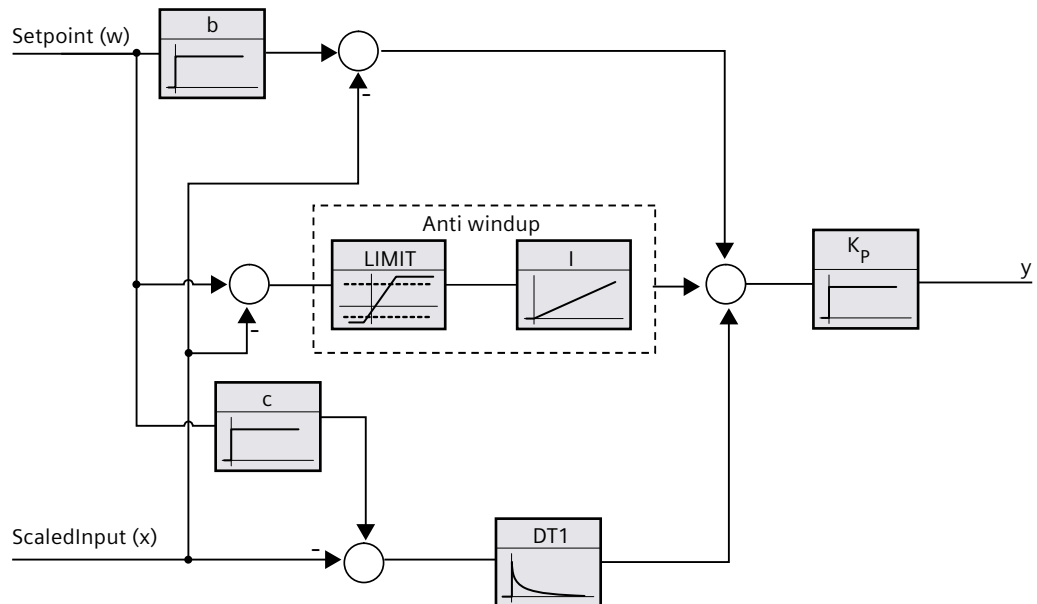
$$y = K_p \left[(b \cdot w - x) + \frac{1}{T_i \cdot s} (w - x) + \frac{T_D \cdot s}{a \cdot T_D \cdot s + 1} (c \cdot w - x) \right]$$

Simbolo	Descrizione
y	Valore di uscita
K_p	Guadagno proporzionale
s	Operatore di Laplace
b	Ponderazione del componente P
w	Setpoint
x	Valore istantaneo
T_i	Tempo di integrazione
a	Coefficiente per il ritardo derivativo ($T_1 = a \times T_D$)
	Tempo derivativo
c	Ponderazione del componente D

Schema a blocchi di PID_Compact



Schema a blocchi di PIDT1 con anti-windup



Comportamento in caso di errori

Quando si verificano degli errori, vengono emessi nel parametro Error e PID_Compact passa al modo di funzionamento "Inattivo". Il parametro Reset resetta nuovamente gli errori.

Senso di regolazione

Aumentando il valore di uscita si dovrebbe ottenere un aumento del valore istantaneo. In questo caso si parla di un senso di regolazione normale. Per raffreddamenti e regolazioni di flusso può essere necessario invertire il senso di regolazione. PID_Compact non funziona con un guadagno proporzionale negativo. Se InvertControl = TRUE, un aumento della differenza di regolazione determina una riduzione del valore di uscita. Il senso di regolazione viene preso in considerazione anche durante l'ottimizzazione iniziale e l'ottimizzazione fine.

Vedere anche

[Tipo di regolazione V1 \(Pagina 107\)](#)

10.1.5.2 Parametri di ingresso PID_Compact V1

Tabella 10-4

Parametri	Tipo di dati	Preassegnazio- ne	Descrizione
Setpoint	REAL	0.0	Setpoint del regolatore PID nel funzionamento automatico
Input	REAL	0.0	Una variabile del programma utente viene utilizzata come sorgente del valore istantaneo. Per utilizzare il parametro Input è necessario impostare sPid_Cmpt.b_Input_PER_On = FALSE.
Input_PER	WORD	W#16#0	Ingresso analogico come sorgente del valore istantaneo Per utilizzare il parametro Input_PER è necessario impostare sPid_Cmpt.b_Input_PER_On = TRUE.
ManualEnable	BOOL	FALSE	- Il fronte FALSE -> TRUE seleziona il modo di funzionamento "Funzionamento manuale", State = 4, sRet.i_Mode rimane invariato. - Il fronte TRUE -> FALSE seleziona l'ultimo modo di funzionamento attivo, State = sRet.i_Mode Se ManualEnable = TRUE un'eventuale modifica di sRet.i_Mode non ha alcuna conseguenza. Solo con il fronte TRUE -> FALSE in ManualEnable la modifica di sRet.i_Mode viene considerata. PID_Compact V1.2 e PID_Compact V1.0 Se all'avvio della CPU ManualEnable = TRUE, PID_Compact si avvia in funzionamento manuale. Non è necessario un fronte di salita (FALSE > TRUE) in ManualEnable. PID_Compact V1.1 All'avvio della CPU PID_Compact commuta nel funzionamento manuale solo in presenza di un fronte di salita (FALSE->TRUE) in ManualEnable. Senza fronte di salita PID_Compact si avvia nell'ultimo modo di funzionamento nel quale ManualEnable era FALSE.
ManualValue	REAL	0.0	Valore manuale Questo valore viene utilizzato nel funzionamento manuale come valore di uscita.
Reset	BOOL	FALSE	Il parametro Reset (Pagina 306) esegue un nuovo avvio del regolatore.

10.1.5.3 Parametri di uscita PID_Compact V1

Tabella 10-5

Parameter	Tipo di dati	Preassegnazione	Descrizione
ScaledInput	REAL	0.0	Emissione del valore istantaneo scalato
Le uscite "Output", "Output_PER" e "Output_PWM" possono essere utilizzate parallelamente.			
Output	REAL	0.0	Valore di uscita in formato REAL
Output_PER	WORD	W#16#0	Valore di uscita analogico
Output_PWM	BOOL	FALSE	Valore di uscita con modulazione ad ampiezza d'impulso Il valore di uscita viene calcolato in base a tempi di attivazione e disattivazione variabili.
SetpointLimit_H	BOOL	FALSE	Se SetpointLimit_H = TRUE è stato raggiunto il limite superiore assoluto del setpoint. Nella CPU il setpoint viene circoscritto al limite superiore assoluto configurato per lo stesso. Come limite superiore del setpoint è preassegnato il limite superiore assoluto configurato per il valore istantaneo. Se si configura sPid_Cmpt.r_Sp_Hlm su un valore compreso nei limiti del valore istantaneo questo valore viene utilizzato come limite superiore del setpoint.
SetpointLimit_L	BOOL	FALSE	Se SetpointLimit_L = TRUE è stato raggiunto il limite inferiore assoluto del setpoint. Nella CPU il setpoint viene circoscritto al limite inferiore assoluto configurato per lo stesso. Come limite inferiore del setpoint è preassegnato il limite inferiore assoluto configurato per il valore istantaneo. Se si configura sPid_Cmpt.r_Sp_Llm su un valore compreso nei limiti del valore istantaneo questo valore viene utilizzato come limite inferiore del setpoint.
InputWarning_H	BOOL	FALSE	Se InputWarning_H = TRUE il limite superiore di avviso del valore istantaneo è stato raggiunto o superato.
InputWarning_L	BOOL	FALSE	Se InputWarning_L = TRUE il limite inferiore di avviso del valore istantaneo è stato raggiunto o superato per difetto.
State	INT	0	Il parametro State (Pagina 302) indica il modo di funzionamento attuale del regolatore PID. per commutare il modo di funzionamento si utilizza la variabile sRet.i_Mode. - State = 0: Inattivo - State = 1: Ottimizzazione iniziale - State = 2: ottimizzazione fine - State = 3: Funzionamento automatico - State = 4: Funzionamento manuale
Error	DWORD	W#16#0	Il parametro Error (Pagina 304) mostra i messaggi di errore. Error = 0000: Senza errori.

10.1.5.4 Variabili statiche di PID_Compact V1

NOTA

Le variabili non specificate non devono essere modificate, perché vengono utilizzate solo internamente.

Per evitare un comportamento errato del regolatore PID modificare le variabili contrassegnate con ⁽¹⁾ solo nel modo di funzionamento "Inattivo". Il modo di funzionamento "Inattivo" può essere attivato forzando la variabile "sRet.i_Mode" sul valore "0".

Tabella 10-6

Variabile	Tipo di dati	Preassegnazione	Descrizione
sb_GetCycleTime	BOOL	TRUE	Se sb_GetCycleTime = TRUE viene avviata la rilevazione automatica del tempo di ciclo. Al termine dell'operazione di misura risulta CycleTime.StartEstimation = FALSE.
sb_EnCyclEstimation	BOOL	TRUE	Se sb_EnCyclEstimation = TRUE viene calcolato il tempo di campionamento PID_Compact.
sb_EnCyclMonitoring	BOOL	TRUE	Se sb_EnCyclMonitoring = FALSE il tempo di campionamento PID_Compact non viene controllato. Se non è possibile eseguire PID_Compact entro il tempo di campionamento, non viene emesso alcun errore 0800 e PID_Compact non passa al modo di funzionamento "Inattivo".
sb_RunModeByStartup	BOOL	TRUE	Dopo il nuovo avviamento della CPU attivare l'ultimo modo di funzionamento Se sb_RunModeByStartup = FALSE, dopo l'avvio della CPU il regolatore rimane inattivo. Se sb_RunModeByStartup = TRUE, dopo l'avvio della CPU il regolatore ritorna all'ultimo modo di funzionamento attivo.
si_Unit	INT	0	Unità fisica del valore istantaneo e del setpoint, ad es. °C o °F.
si_Type	INT	0	Grandezza fisica del valore istantaneo e del setpoint, ad es. la temperatura.
sd_Warning	DWORD	DW#16#0	La variabile sd_warning (Pagina 307) mostra gli avvisi dal reset o dall'ultima commutazione del modo di funzionamento.
sBackUp.r_Gain	REAL	1.0	Guadagno proporzionale memorizzato I valori della struttura sBackUp possono essere ricaricati con sPid_Cmpt.b_LoadBackUp = TRUE.
sBackUp.r_Ti	REAL	20.0	Tempo di integrazione [s] memorizzato
sBackUp.r_Td	REAL	0.0	Tempo derivativo [s] memorizzato
sBackUp.r_A	REAL	0.0	Coefficiente del ritardo derivativo memorizzato
sBackUp.r_B	REAL	0.0	Fattore di ponderazione del componente P memorizzato
sBackUp.r_C	REAL	0.0	Fattore di ponderazione del componente D memorizzato

Variabile	Tipo di dati	Preassegnazione	Descrizione
sBackUp.r_Cycle	REAL	1.0	Tempo di campionamento dell'algoritmo PID memorizzato
sPid_Calc.r_Cycle ⁽¹⁾	REAL	0.1	Tempo di campionamento dell'istruzione PID_Compact r_Cycle viene rilevata automaticamente e corrisponde in genere al tempo di ciclo dell'OB richiamante.
sPid_Calc.b_RunIn	BOOL	FALSE	• b_RunIn = FALSE Se l'ottimizzazione fine viene avviata dal modo di funzionamento inattivo o manuale, si avvia un'ottimizzazione iniziale. Se i presupposti per l'ottimizzazione iniziale non sono soddisfatti, PID_Compact si comporta come con b_RunIn = TRUE. Se l'ottimizzazione fine viene avviata dal funzionamento automatico, viene eseguita la regolazione sul setpoint con i parametri PID disponibili. Solo in seguito viene avviata l'ottimizzazione fine. Se non è possibile eseguire l'ottimizzazione iniziale, PID_Compact passa al modo di funzionamento "Inattivo". • b_RunIn = TRUE L'ottimizzazione iniziale viene ignorata. PID_3Compact tenta di raggiungere il setpoint con il valore di uscita minimo o massimo. Ciò può causare una maggiore sovraoscillazione. L'ottimizzazione fine si avvia infine automaticamente. Al termine dell'ottimizzazione fine b_RunIn viene impostato su FALSE.
sPid_Calc.b_CalcParamSUT	BOOL	FALSE	Se b_CalcParamSUT = TRUE, i parametri per l'ottimizzazione iniziale vengono ricalcolati. Ciò permette di modificare il metodo di calcolo dei parametri senza dover ripetere l'ottimizzazione. Al termine del calcolo b_CalcParamSUT viene impostato su FALSE.
sPid_Calc.b_CalcParamTIR	BOOL	FALSE	Se b_CalcParamTIR = TRUE, i parametri per l'ottimizzazione fine vengono ricalcolati. Ciò permette di modificare il metodo di calcolo dei parametri senza dover ripetere l'ottimizzazione. Al termine del calcolo b_CalcParamTIR viene impostato su FALSE.
sPid_Calc.i_CtrlTypeSUT	INT	0	Calcolo dei parametri durante l'ottimizzazione iniziale secondo un metodo: • i_CtrlTypeSUT = 0: PID secondo Chien, Hrones e Reswick • i_CtrlTypeSUT = 1: PI secondo Chien, Hrones e Reswick
sPid_Calc.i_CtrlTypeTIR	INT	0	Calcolo dei parametri durante l'ottimizzazione fine secondo un metodo: • i_CtrlTypeTIR = 0: PID automatico • i_CtrlTypeTIR = 1: PID veloce (comportamento di regolazione più veloce con ampiezze del valore di uscita più elevate rispetto a i_CtrlTypeTIR = 2) • i_CtrlTypeTIR = 2: PID lento (comportamento di regolazione più lento con ampiezze del valore di uscita più basse rispetto a i_CtrlTypeTIR = 1)

Variabile	Tipo di dati	Preassegnazione	Descrizione
			- i_CtrlTypeTIR = 3: PID secondo Ziegler-Nichols - i_CtrlTypeTIR = 4: PI secondo Ziegler-Nichols - i_CtrlTypeTIR = 5: P secondo Ziegler-Nichols Affinché il calcolo dei parametri PID con b_CalcParamTIR e i_CtrlTypeTIR = 0, 1 o 2 possa essere ripetuto, anche l'ottimizzazione fine precedente deve essere stata eseguita con i_CtrlTypeTIR = 0, 1, o 2. In caso contrario viene utilizzato i_CtrlTypeTIR = 3. Il nuovo calcolo del parametro PID con b_CalcParamTIR e i_CtrlTypeTIR = 3, 4 o 5 è sempre possibile.
sPid_Calc.r_Progress	REAL	0.0	Avanzamento dell'ottimizzazione in percentuale (0,0 - 100,0)
sPid_Cmpt.r_Sp_Hlm ⁽¹⁾	REAL	+3.402822e+38	Limite superiore del setpoint Se si configura sPid_Cmpt.r_Sp_Hlm al di fuori dei limiti del valore istantaneo, il limite superiore assoluto configurato per il valore istantaneo viene utilizzato come limite superiore del setpoint. Se si configura sPid_Cmpt.r_Sp_Hlm entro i limiti del valore istantaneo questo valore viene utilizzato come limite superiore del setpoint.
sPid_Cmpt.r_Sp_Llm ⁽¹⁾	REAL	-3.402822e+38	Limite inferiore del setpoint Se si configura sPid_Cmpt.r_Sp_Llm al di fuori dei limiti del valore istantaneo, il limite inferiore assoluto configurato per il valore istantaneo viene utilizzato come limite inferiore del setpoint. Se si configura sPid_Cmpt.r_Sp_Llm entro i limiti del valore istantaneo questo valore viene utilizzato come limite inferiore del setpoint.
sPid_Cmpt.r_Pv_Norm_IN_1 ⁽¹⁾	REAL	0.0	Scala Input_PER inferiore Sulla base delle due coppie di valori r_Pv_Norm_OUT_1, r_Pv_Norm_IN_1 e r_Pv_Norm_OUT_2, r_Pv_Norm_IN_2 della struttura sPid_Cmpt, Input_PER viene convertito in percentuale.
sPid_Cmpt.r_Pv_Norm_IN_2 ⁽¹⁾	REAL	27648.0	Scala Input_PER superiore Sulla base delle due coppie di valori r_Pv_Norm_OUT_1, r_Pv_Norm_IN_1 e r_Pv_Norm_OUT_2, r_Pv_Norm_IN_2 della struttura sPid_Cmpt, Input_PER viene convertito in percentuale.
sPid_Cmpt.r_Pv_Norm_OUT_1 ⁽¹⁾	REAL	0.0	Valore istantaneo inferiore riportato in scala Sulla base delle due coppie di valori r_Pv_Norm_OUT_1, r_Pv_Norm_IN_1 e r_Pv_Norm_OUT_2, r_Pv_Norm_IN_2 della struttura sPid_Cmpt, Input_PER viene convertito in percentuale.
sPid_Cmpt.r_Pv_Norm_OUT_2 ⁽¹⁾	REAL	100.0	Valore istantaneo superiore riportato in scala Sulla base delle due coppie di valori r_Pv_Norm_OUT_1, r_Pv_Norm_IN_1 e r_Pv_Norm_OUT_2, r_Pv_Norm_IN_2 della struttura sPid_Cmpt, Input_PER viene convertito in percentuale.
sPid_Cmpt.r_Lmn_Hlm ⁽¹⁾	REAL	100.0	Limite superiore del valore di uscita per il parametro di uscita "Output"

10.1 PID_Compact

Variabile	Tipo di dati	Preassegnazione	Descrizione
sPid_Cmpt.r_Lmn_Llm ⁽¹⁾	REAL	0.0	Limite inferiore del valore di uscita per il parametro di uscita "Output"
sPid_Cmpt.b_Input_PER_On ⁽¹⁾	BOOL	TRUE	Se b_Input_PER_On = TRUE viene utilizzato il parametro Input_PER. Se b_Input_PER_On = FALSE viene utilizzato il parametro Input.
sPid_Cmpt.b_LoadBackup	BOOL	FALSE	Attivazione del set di parametri di backup. Se un'ottimizzazione non va a buon fine, impostando questo bit si possono riattivare i parametri PID precedenti.
sPid_Cmpt.b_InvCtrl ⁽¹⁾	BOOL	FALSE	Inversione del senso di regolazione Se b_InvCtrl = TRUE, un aumento della differenza di regolazione determina una riduzione del valore di uscita.
sPid_Cmpt.r_Lmn_Pwm_PPTm ⁽¹⁾	REAL	0.0	Il tempo minimo di attivazione della modulazione di larghezza di impulso in secondi viene arrotondato a $r\_Lmn\_Pwm\_PPTm = r\_Cycle$ o $r\_Lmn\_Pwm\_PPTm = n * r\_Cycle$
sPid_Cmpt.r_Lmn_Pwm_PBTm ⁽¹⁾	REAL	0.0	Il tempo minimo di disattivazione della modulazione di larghezza di impulso in secondi viene arrotondato a $r\_Lmn\_Pwm\_PBTm = r\_Cycle$ o $r\_Lmn\_Pwm\_PBTm = n * r\_Cycle$
sPid_Cmpt.r_Pv_Hlm ⁽¹⁾	REAL	120.0	Limite superiore del valore istantaneo Nell'ingresso di periferia il valore istantaneo può superare al massimo del 18% il campo normale (campo di sovracomando). Non vengono più segnalati errori di superamento del "Limite superiore valore istantaneo". Vengono riconosciuti solo la rottura conduttore e il cortocircuito e PID_Compact passa nel modo di funzionamento "Inattivo". $r\_Pv\_Hlm > r\_Pv\_Llm$
sPid_Cmpt.r_Pv_Llm ⁽¹⁾	REAL	0.0	Limite inferiore del valore istantaneo $r\_Pv\_Llm < r\_Pv\_Hlm$
sPid_Cmpt.r_Pv_HWrn ⁽¹⁾	REAL	+3.402822e+38	Limite superiore di avviso del valore istantaneo Se si configura r_Pv_HWrn al di fuori dei limiti del valore istantaneo, il limite superiore assoluto configurato per il valore istantaneo viene utilizzato come limite superiore di avviso. Se si configura r_Pv_HWrn entro i limiti del valore istantaneo questo valore viene utilizzato come limite superiore di avviso. $r\_Pv\_HWrn > r\_Pv\_LWrn$ $r\_Pv\_HWrn \leq r\_Pv\_Hlm$
sPid_Cmpt.r_Pv_LWrn ⁽¹⁾	REAL	-3.402822e+38	Limite inferiore di avviso del valore istantaneo Se si configura r_Pv_LWrn al di fuori dei limiti del valore istantaneo, il limite inferiore assoluto configurato per il valore istantaneo viene utilizzato come limite inferiore di avviso. Se si configura r_Pv_LWrn entro i limiti del valore istantaneo questo valore viene utilizzato come limite inferiore di avviso. $r\_Pv\_LWrn < r\_Pv\_HWrn$ $r\_Pv\_LWrn \geq r\_Pv\_Lwrn$
sPidCalc.i_Ctrl_IOutv ⁽¹⁾	REAL	0.0	Componente I attuale

Variabile	Tipo di dati	Preassegnazione	Descrizione
sParamCalc.i_Event_SUT	INT	0	La variabile i_Event_SUT (Pagina 308) mostra la fase attuale dell'"Ottimizzazione iniziale":
sParamCalc.i_Event_TIR	INT	0	La variabile i_Event_TIR (Pagina 308) mostra la fase attuale dell'"ottimizzazione fine":
sRet.i_Mode	INT	0	La commutazione del modo di funzionamento è controllata dal fronte. I seguenti modi di funzionamento vengono attivati quando si esegue la commutazione a - i_Mode = 0: modo di funzionamento "Inattivo" (arresto del regolatore) - i_Mode = 1: modo di funzionamento "Ottimizzazione iniziale" - i_Mode = 2: modo di funzionamento "ottimizzazione fine" - i_Mode = 3: modo di funzionamento "Funzionamento automatico" - i_Mode = 4: modo di funzionamento "Funzionamento manuale" i_Mode è a ritenzione.
sRet.r_Ctrl_Gain ⁽¹⁾	REAL	1.0	Guadagno proporzionale attivo Gain è a ritenzione.
sRet.r_Ctrl_Ti ⁽¹⁾	REAL	20.0	- r_Ctrl_Ti > 0.0: Tempo di integrazione attivo - r_Ctrl_Ti = 0.0: Il componente I è disattivato r_Ctrl_Ti è a ritenzione.
sRet.r_Ctrl_Td ⁽¹⁾	REAL	0.0	- r_Ctrl_Td > 0.0: Tempo derivativo attivo - r_Ctrl_Td = 0.0: Il componente D è disattivato r_Ctrl_Td è a ritenzione.
sRet.r_Ctrl_A ⁽¹⁾	REAL	0.0	Coefficiente del ritardo derivativo attivo r_Ctrl_A è a ritenzione.
sRet.r_Ctrl_B ⁽¹⁾	REAL	0.0	Ponderazione del componente P attiva r_Ctrl_B è a ritenzione.
sRet.r_Ctrl_C ⁽¹⁾	REAL	0.0	Ponderazione del componente D attiva r_Ctrl_C è a ritenzione.
sRet.r_Ctrl_Cycle ⁽¹⁾	REAL	1.0	Tempo di campionamento dell'algoritmo PID attivo r_Ctrl_Cycle viene rilevato durante l'ottimizzazione e arrotondato ad un multiplo intero di r_Cycle. r_Ctrl_Cycle viene utilizzato come durata periodo della modulazione ampiezza impulso. r_Ctrl_Cycle è a ritenzione.

Vedere anche

[Caricamento degli oggetti tecnologici del dispositivo \(Pagina 48\)](#)

10.1.5.5 Parametri State e sRet.i_Mode V1

Relazione tra i parametri

Il parametro State mostra il modo di funzionamento attuale del regolatore PID. Non è possibile modificarlo.

Per commutare il modo di funzionamento è necessario modificare la variabile sRet.i_Mode. Ciò vale anche se in sRet.i_Mode è già presente il valore del nuovo modo di funzionamento. Impostare quindi ad es. prima sRet.i_Mode = 0 e successivamente sRet.i_Mode = 3. Se il modo di funzionamento attuale del regolatore permette questa modifica, il parametro State assume il valore della variabile sRet.i_Mode.

Se PID_Compact cambia automaticamente modo di funzionamento, vale: State != sRet.i_Mode.

Esempi:

- Ottimizzazione iniziale riuscita
State = 3 e sRet.i_Mode = 1
- In caso di errore risulta
State = 0 e sRet.i_Mode rimane sul valore finora assunto, ad es. sRet.i_Mode = 3
- ManualEnalbe = TRUE
State = 4 mentre sRet.i_Mode rimane sul valore assunto finora, ad es. sRet.i_Mode = 3

NOTA

Supponiamo ad es. di voler ripetere un'ottimizzazione fine conclusa correttamente senza terminare il funzionamento automatico con i_Mode = 0.

L'impostazione di sRet.i_Mode su un valore non valido per un clock, ad es. 9999, non influisce sul parametro State. Impostando nel clock successivo Mode = 2 è possibile modificare sRet.i_Mode senza dover passare prima al modo di funzionamento "Inattivo".

Significato dei valori

State / sRet.i_Mode	Descrizione del modo di funzionamento
0	Non attivo Il regolatore è disattivato. Prima di un'ottimizzazione iniziale il regolatore si trova nel modo di funzionamento "Inattivo". La commutazione del regolatore PID nel modo "Inattivo" durante il funzionamento avviene al verificarsi di un errore oppure quando nella finestra della messa in servizio è selezionata la funzione "Disattiva regolatore".
1	Ottimizzazione iniziale L'ottimizzazione iniziale rileva la risposta del processo a un gradino del valore di uscita e cerca il punto di flesso. Dall'incremento massimo e dal tempo morto del sistema regolato vengono calcolati i parametri PID ottimali.

State / sRet.i_Mode	Descrizione del modo di funzionamento
	Requisiti per l'ottimizzazione iniziale: - Il regolatore si trova nel modo di funzionamento Inattivo o Funzionamento manuale. - ManualEnable = FALSE - Il valore istantaneo non deve essere troppo vicino al setpoint. $\text{Setpoint} - \text{Input} > 0.3 * \text{sPid\_Cmpt.r\_Pv\_Hlm} - \text{sPid\_Cmpt.r\_Pv\_Llm}$ e $\text{Setpoint} - \text{Input} > 0.5 * \text{Setpoint}$ - Durante l'ottimizzazione iniziale il setpoint non può essere modificato. Maggiore è la stabilità del valore istantaneo, più semplice e precisa sarà la rilevazione dei parametri PID. Un rumore del valore istantaneo è accettabile finché l'incremento del valore è di molto superiore al rumore. Prima che i parametri PID vengano ricalcolati, vengono salvati e possono essere riattivati con sPid_Cmpt.b_LoadBackUp. Se l'ottimizzazione iniziale viene conclusa correttamente si passa al funzionamento automatico, in caso contrario al modo "Inattivo". La fase dell'ottimizzazione iniziale viene visualizzata con Variabile i_Event_SUT V1 (Pagina 308).
2	ottimizzazione fine L'ottimizzazione fine genera un'oscillazione costante e limitata del valore istantaneo. L'ampiezza e la frequenza dell'oscillazione ottimizzano i parametri PID. Le differenze tra la risposta del processo durante l'ottimizzazione iniziale e finale vengono analizzate. A partire dai risultati vengono ricalcolati tutti i parametri PID. I parametri PID risultanti dall'ottimizzazione fine mostrano in genere un comportamento pilota e di disturbo migliore rispetto ai parametri PID dell'ottimizzazione iniziale. PID_Compact tenta di generare automaticamente un'oscillazione che sia maggiore rispetto al fruscio del valore istantaneo. L'ottimizzazione fine viene influenzata solo minimamente dalla stabilità del valore istantaneo. Prima che i parametri PID vengano ricalcolati, vengono salvati e possono essere riattivati con sPid_Cmpt.b_LoadBackUp. Requisiti per l'ottimizzazione fine: - Non sono previste anomalie. - Il setpoint e il valore istantaneo sono compresi entro i limiti configurati. - Durante l'ottimizzazione fine il setpoint non può essere modificato. - ManualEnable = FALSE - Funzionamento automatico (State = 3), inattivo (State = 0) o manuale (State = 4) rispettivamente nelle modalità descritte di seguito: - Funzionamento automatico (State = 3) Per migliorare i parametri PID esistenti mediante un'ottimizzazione, avviare l'ottimizzazione fine dal modo di funzionamento automatico. PID_Compact esegue quindi la regolazione con i parametri PID esistenti finché il circuito di regolazione è assestato e sono soddisfatti i requisiti per l'ottimizzazione fine. Solo in seguito viene avviata l'ottimizzazione fine. - Funzionamento Inattivo (State = 0) o manuale (State = 4) Se sono soddisfatti i presupposti dell'ottimizzazione iniziale, viene avviata un'ottimizzazione iniziale. Viene quindi effettuata la regolazione con i parametri PID rilevati finché il circuito di regolazione è assestato e sono soddisfatti i requisiti per l'ottimizzazione fine. Solo in seguito viene avviata l'ottimizzazione fine. Se non è possibile eseguire l'ottimizzazione iniziale, PID_Compact passa al modo di funzionamento "Inattivo". Se il valore istantaneo per un'ottimizzazione iniziale si trova già troppo vicino al setpoint o se sPid_Calc.b_RunIn = TRUE, si tenta di raggiungere il setpoint con il valore di uscita minimo o massimo. Ciò può causare una maggiore sovraelongazione. Se l'ottimizzazione fine viene conclusa correttamente il regolatore passa al modo "Funzionamento automatico", in caso contrario al modo di funzionamento "Inattivo". La fase di "Ottimizzazione fine" viene visualizzata con Variabile i_Event_TIR V1 (Pagina 308).

State / sRet.i_Mode	Descrizione del modo di funzionamento
3	Funzionamento automatico In funzionamento automatico PID_Compact regola il sistema regolato secondo i parametri preimpostati. Se è soddisfatta una delle seguenti condizioni si passa al funzionamento automatico: • Ottimizzazione iniziale conclusa correttamente • ottimizzazione fine conclusa correttamente • Modifica della variabile sRet.i_Mode sul valore 3. Se la CPU viene attivata o passa da STOP a RUN, PID_Compact si avvia nell'ultimo modo di funzionamento attivo. Per lasciare PID_Compact nel modo di funzionamento "Inattivo" impostare sb_RunModeByStartup = FALSE.
4	Funzionamento manuale Nel funzionamento manuale l'utente imposta manualmente un valore di uscita nel parametro ManualValue. Questo modo di funzionamento viene attivato se sRet.i_Mode = 4 o con fronte di salita in ManualEnable. Se ManualEnable diventa TRUE cambia solo State. sRet.i_Mode rimane sul valore finora assunto. In presenza di un fronte di discesa in ManualEnable PID_Compact torna al modo di funzionamento precedente. La commutazione nel funzionamento automatico avviene in modo regolare.

Vedere anche

[Parametri di uscita PID_Compact V1 \(Pagina 304\)](#)

[Ottimizzazione iniziale V1 \(Pagina 118\)](#)

[Ottimizzazione fine V1 \(Pagina 120\)](#)

[Modo di funzionamento "Funzionamento manuale" V1 \(Pagina 122\)](#)

[Variabile i_Event_SUT V1 \(Pagina 308\)](#)

[Variabile i_Event_TIR V1 \(Pagina 308\)](#)

10.1.5.6 Parametro Error V1

Se si verificano diversi errori contemporaneamente, i valori dei codici di errore vengono visualizzati in un'addizione binaria. Se viene visualizzato il codice di errore 0003, ad es., significa che sono presenti contemporaneamente gli errori 0001 e 0002.

Error (DW#16#...)	Descrizione
0000	Non sono presenti errori.
0001	Il parametro "Input" non rientra nei limiti del valore istantaneo. • Input > sPid_Cmpt.r_Pv_Hlm o • Input < sPid_Cmpt.r_Pv_Llm L'organo attuatore si può spostare solo se l'errore è stato eliminato.
0002	Il parametro "Input_PER" presenta un valore non valido. Controllare l'eventuale presenza di un errore sull'ingresso analogico.
0004	Errore durante l'ottimizzazione fine. Non è stato possibile mantenere l'oscillazione del valore istantaneo.
0008	Errore all'avvio dell'ottimizzazione iniziale. Il valore istantaneo è troppo vicino al setpoint. Avviare l'ottimizzazione fine.
0010	Il setpoint è stato modificato durante l'ottimizzazione.
0020	L'ottimizzazione iniziale non è consentita nel funzionamento automatico e durante l'ottimizzazione fine.

Error (DW#16#...)	Descrizione
0080	Errore nell'ottimizzazione iniziale. I limiti del valore di uscita non sono configurati correttamente e il valore istantaneo non reagisce come previsto. Verificare che i limiti del valore di uscita siano configurati correttamente e che siano compatibili con il senso di regolazione. Accertarsi inoltre che il valore istantaneo non presenti oscillazioni troppo forti prima dell'avvio dell'ottimizzazione iniziale.
0100	Un errore durante l'ottimizzazione ha causato parametri non validi.
0200	Valore del parametro "Input" non valido: il formato numerico del valore non è valido.
0400	Calcolo del valore di uscita non riuscito. Controllare i parametri PID.
0800	Errore del tempo di campionamento: PID_Compact non viene richiamato entro il tempo di campionamento dell'OB di schedulazione orologio. Se quest'errore si è verificato nella simulazione con PLCSIM, osservare le avvertenze riportate alla sezione Simulazione di PID_Compact V1 con PLCSIM (Pagina 123).
1000	Valore del parametro "Setpoint" non valido: il formato numerico del valore non è valido.

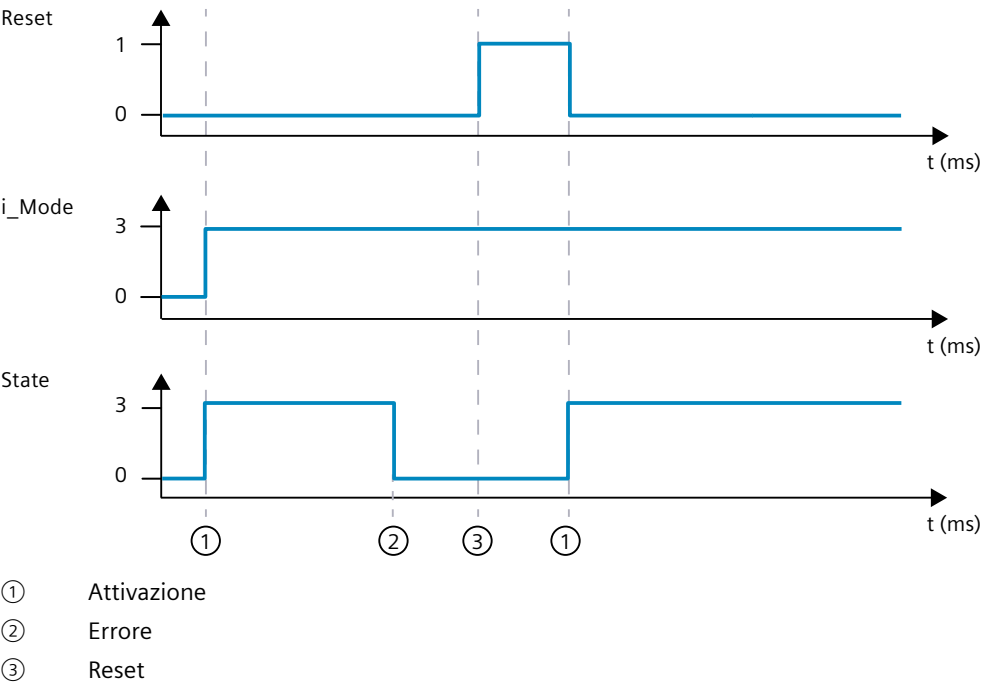
Vedere anche[Parametri di uscita PID_Compact V1 \(Pagina 304\)](#)

10.1.5.7 Parametro Reset V1

Nel caso di Reset = TRUE il comportamento dipende dalla versione dell'istruzione PID_Compact.

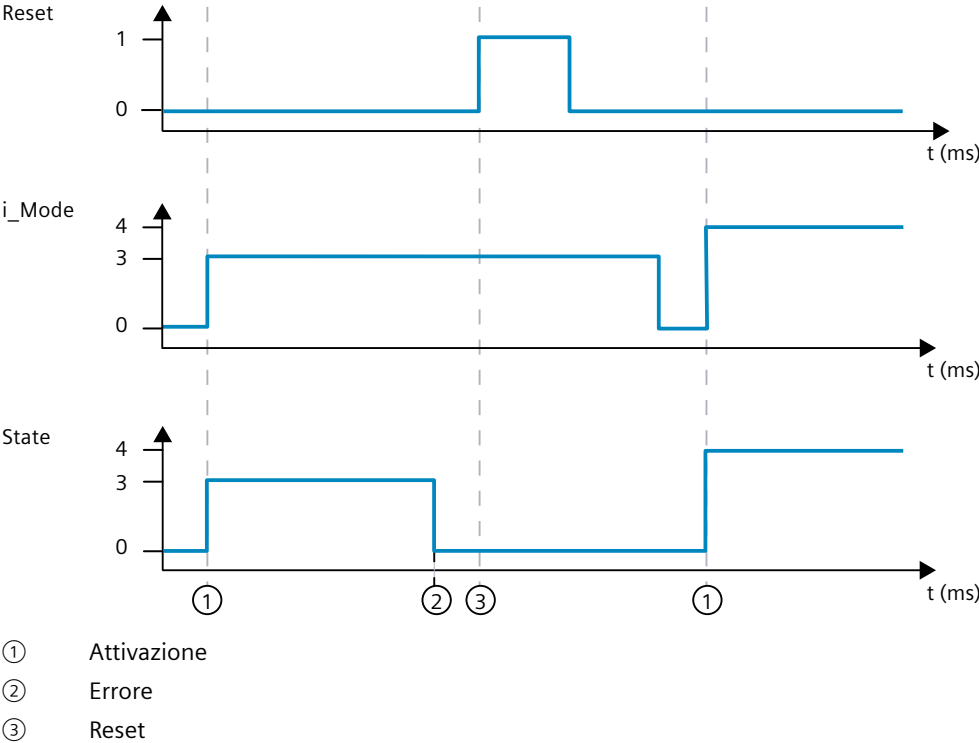
Comportamento in caso di reset di PID_Compact a partire dalla V.1.1

Un fronte di salita nel parametro Reset determina la commutazione nel modo di funzionamento "Inattivo"; gli errori e gli avvisi vengono resettati e il componente I viene cancellato. Un fronte di discesa nel parametro Reset commuta nell'ultimo modo di funzionamento attivo. Se in precedenza era attivo il funzionamento automatico, il componente I viene predefinito in modo da consentire una commutazione bumpless.



Comportamento in caso di reset di PID_Compact V.1.0

Un fronte di salita nel parametro Reset determina la commutazione nel modo di funzionamento "Inattivo"; gli errori e gli avvisi vengono resettati e il componente I viene cancellato. Il regolatore viene riattivato solo tramite un fronte nel parametro i_Mode.



10.1.5.8 Variabile sd_warning V1

Se sono presenti più avvisi contemporaneamente, i valori della variabile sd_warning vengono visualizzati in un'addizione binaria. La visualizzazione ad es. dell'avviso 0003 indica la presenza contemporanea degli avvisi 0001 e 0002.

sd_warning (DW#16#....)	Descrizione
0000	Non è presente alcun avviso.
0001	Durante l'ottimizzazione iniziale non è stato trovato il punto di flesso.
0002	Durante l'ottimizzazione fine l'oscillazione è stata intensificata.
0004	Il setpoint si trovava fuori dei limiti impostati.
0008	Per il metodo di calcolo selezionato non sono state definite tutte le proprietà necessarie per il sistema regolato. Al loro posto sono stati calcolati i parametri PID con il metodo "i_CtrlTypeTIR = 3".
0010	Non è stato possibile commutare il modo di funzionamento perché ManualEnable = TRUE.
0020	Il tempo di campionamento dell'algoritmo PID viene circoscritto dal tempo di ciclo dell'OB richiamante. Per ottenere risultati migliori utilizzare tempi di ciclo dell'OB più brevi.
0040	Il valore istantaneo ha superato uno dei suoi limiti di avviso.

I seguenti avvisi vengono cancellati non appena viene eliminata la causa:

- 0004
- 0020
- 0040

Tutti gli altri avvisi vengono cancellati in presenza di un fronte di salita in Reset.

10.1.5.9 Variabile i_Event_SUT V1

i_Event_SUT	Nome	Descrizione
0	SUT_INIT	Inizializzazione dell'ottimizzazione iniziale
100	SUT_STDABW	Calcolo della divergenza standard
200	SUT_GET_POI	Rilevazione del punto di flesso
9900	SUT_IO	Ottimizzazione iniziale riuscita
1	SUT_NIO	Ottimizzazione iniziale fallita

Vedere anche

[Variabili statiche di PID_Compact V1 \(Pagina 305\)](#)

[Parametri State e sRet.i_Mode V1 \(Pagina 310\)](#)

10.1.5.10 Variabile i_Event_TIR V1

i_Event_TIR	Nome	Descrizione
-100	TIR_FIRST_SUT	Non è possibile eseguire l'ottimizzazione fine. Viene prima eseguita un'ottimizzazione iniziale.
0	TIR_INIT	Inizializzazione dell'ottimizzazione fine
200	TIR_STDABW	Calcolo della divergenza standard
300	TIR_RUN_IN	Tentativo di raggiungere il setpoint
400	TIR_CTRLN	Tentativo di raggiungere il setpoint con i parametri PID esistenti (se l'ottimizzazione iniziale è riuscita)
500	TIR_OSZIL	Rilevazione dell'oscillazione e calcolo dei parametri
9900	TIR_IO	Ottimizzazione fine riuscita
1	TIR_NIO	Ottimizzazione fine fallita

Vedere anche

[Variabili statiche di PID_Compact V1 \(Pagina 305\)](#)

[Parametri State e sRet.i_Mode V1 \(Pagina 310\)](#)

10.2 PID_3Step

10.2.1 Novità di PID_3Step

PID_3Step V2.3

- **Utilizzo con S7-1200 G2**

Da PID_3Step V2.3 l'istruzione può essere utilizzata anche in un S7-1200 G2.
Le versioni precedenti dell'istruzione non sono supportate nell'S7-1200 G2.

- **Controllo della durata corsa**

Da PID_3Step versione 2.3 il controllo e la limitazione della durata corsa con `Config.VirtualActuatorLimit = 0.0` possono essere disattivati.

PID_3Step V2.2

- **Impiego con S7-1200**

Da PID_3Step V2.2 l'istruzione con funzionalità V2 può essere utilizzata anche su un S7-1200 a partire dalla versione firmware 4.0.

PID_3Step V2.0

- **Comportamento in caso di errori**

Il comportamento con `ActivateRecoverMode = TRUE` è stato completamente rielaborato.
Nell'impostazione di default PID_3Step ha un comportamento più fault-tolerant.

ATTENZIONE
L'impianto può subire danneggiamenti. Se si utilizza l'impostazione di default, PID_3Step rimane in funzionamento automatico anche se vengono superati i limiti del valore istantaneo. In seguito a ciò l'impianto può subire danneggiamenti. Configurare per il sistema regolato un comportamento da adottare in caso di errore che protegga l'impianto da danneggiamenti.

Con il parametro di ingresso `ErrorAck` si confermano gli errori e gli avvisi senza dover riavviare il regolatore o cancellare il componente I.

Cambiando il modo di funzionamento gli errori che non sono più presenti non vengono più confermati.

- **Commutazione dei modi di funzionamento**

Il modo di funzionamento si predefinisce nel parametro di transito `Mode` e si avvia con un fronte di salita in `ModeActivate`. La variabile `Retain.Mode` è stata eliminata.

La misura del tempo di regolazione non può più essere avviata con `GetTransitTime.Start`, ma solo con `Mode = 6` e un fronte di salita in `ModeActivate`.

- **Funzionalità multiistanza**

PID_3Step può essere richiamato come DB multiistanza.

- **Comportamento all'avviamento**

Il modo di funzionamento predefinito in Mode viene avviato anche da un fronte di discesa in Reset e con l'avviamento a freddo della CPU se RunModeByStartup = TRUE.

- **Comportamento ENO**

ENO viene impostato in funzione del modo di funzionamento.

Se State = 0, ENO = FALSE.

Se State ≠ 0, ENO = TRUE.

- **Funzionamento manuale**

I parametri di ingresso Manual_UP e Manual_DN non sono più comandati dal fronte. Il funzionamento manuale comandato dal fronte è ancora possibile con le variabili ManualUpInternal e ManualDnInternal.

Nel "Funzionamento manuale senza segnali di riscontro" (Mode = 10) i segnali di riscontro Actuator_H e Actuator_L vengono ignorati benché siano attivi.

- **Predefinizione dei parametri PID**

Le seguenti preimpostazioni sono state modificate:

- Ponderazione del componente P (PWeighting) da 0.0 a 1.0
- Ponderazione del componente D (DWeighting) da 0.0 a 1.0
- Coefficiente per il ritardo di differenziazione (TdFiltRatio) da 0.0 a 0.2

- **Limitazione del tempo di regolazione motore**

Nella variabile Config.VirtualActuatorLimit si configura la % max. del tempo di regolazione del motore in cui l'organo attuatore viene spostato in una direzione.

- **Predefinizione del setpoint durante l'ottimizzazione**

L'oscillazione del setpoint consentita durante l'ottimizzazione si configura nella variabile CancelTuningLevel.

- **Inserzione di una grandezza di disturbo**

Nel parametro Disturbance si può inserire una grandezza di disturbo.

- **Rimedio**

Se i segnali di riscontro non sono attivati (ActuatorEndStopOn = FALSE), Actuator_H e Actuator_L non sono più presi in considerazione per la determinazione di ScaledFeedback.

PID_3Step V1.1

- **Funzionamento manuale all'avviamento della CPU**

Se all'avvio della CPU ManualEnable = TRUE, PID_3Step si avvia in funzionamento manuale. Non è necessario un fronte di salita in ManualEnable.

- **Comportamento in caso di errori**

La variabile ActivateRecoverMode non è più efficace in funzionamento manuale.

- **Rimedio**

La variabile Progress viene resettata una volta eseguita con successo l'ottimizzazione o la misura del tempo di regolazione.

10.2.2 Compatibilità con CPU e FW

La tabella seguente mostra quali CPU sono compatibili con quale versione di PID_3Step.

CPU	FW	PID_3Step
S7-1200	da V4.2	V2.3 V2.2 V1.1
	V4.0 ... V4.1	V2.2 V1.1
	V3.X	V1.1 V1.0
	V2.X	V1.1 V1.0
S7-1200 G2	da V1.0	V2.3
S7-1500	da V3.0	V2.3
	Da V2.0 a V2.9	V2.3 V2.2 V2.1 V2.0
	V1.5 ... V1.8	V2.2 V2.1 V2.0
	V1.1	V2.1 V2.0
	V1.0	V2.0

10.2.3 Tempo di elaborazione della CPU e memoria necessaria di PID_3Step V2.x

Tempi di elaborazione della CPU

Tempi di elaborazione tipici della CPU dell'oggetto tecnologico PID_3Step dalla versione V2.0 in funzione del tipo di CPU e del modo di funzionamento per CPU standard, F, T e TF.

CPU	FW	Tempi di elaborazione tipici della CPU nel funzionamento automatico	Tempi di elaborazione tipici della CPU per ottimizzazione iniziale e ottimizzazione fine
CPU 1211	≥ V4.0	260 μs	310 μs
CPU 1212			
CPU 1214			
CPU 1215			
CPU 1217			
CPU 1212 G2	≥ V1.0	140 μs	170 μs
CPU 1214 G2			
CPU 1510SP	≤ V2.9	80 μs	95 μs
CPU 1511			
CPU 1511C			
CPU 1512C			
CPU 1512SP			
CPU 1513			
CPU 1515		70 μs	80 μs
CPU 1516			
CPU 1517	≤ V3.1	11 μs	15 μs
CPU 1518		5 μs	7 μs
CPU 1510SP	V3.0, V3.1	65 μs	85 μs
CPU 1511			
CPU 1511C			
CPU 1512C			
CPU 1512SP			
CPU 1513			
CPU 1514SP			
CPU 1515		60 μs	75 μs
CPU 1516			
CPU 1510SP	≥ V4.0	60 μs	75 μs
CPU 1511			
CPU 1511C			
CPU 1512C			
CPU 1512SP			
CPU 1513			

CPU 1514SP	≥ V4.0	60 µs	75 µs
CPU 1515			
CPU 1516			
CPU 1516T		3 µs	5 µs
CPU 1517		2 µs	3 µs
CPU 1518		1 µs	2 µs

Tempi tipici di elaborazione CPU dell'oggetto tecnologico PID_3Step con versione V2.0 o superiore in funzione del tipo di CPU e del modo di funzionamento per le CPU R/H nello stato di sistema RUN-Redundant.

CPU	FW	Tempi di elaborazione tipici della CPU nel funzionamento automatico	Tempi di elaborazione tipici della CPU per ottimizzazione iniziale e ottimizzazione fine
CPU 1513R	V3.0, V3.1	110 µs	150 µs
CPU 1515R		100 µs	120 µs
CPU 1513R	≥ V4.0	100 µs	120 µs
CPU 1515R			
CPU 1517H		4 µs	6 µs
CPU 1518HF			

Memoria necessaria

Memoria necessaria di un DB di istanza dell'oggetto tecnologico PID_3Step a partire dalla versione V2.0.

Memoria necessaria	Memoria necessaria di un DB di istanza di PID_3Step V2.x
Memoria di caricamento necessaria	Circa 4400 byte
Memoria di lavoro necessaria complessivamente	1040 byte
Memoria di lavoro necessaria a ritenzione	60 byte

10.2.4 PID_3Step V2

10.2.4.1 Descrizione di PID_3Step V2

Descrizione

L'istruzione PID_3Step permette di configurare un regolatore PID con ottimizzazione automatica delle valvole o degli attuatori con comportamento di integrazione.

Sono disponibili i seguenti modi di funzionamento:

- Inattivo
- Ottimizzazione iniziale
- ottimizzazione fine
- Funzionamento automatico
- Funzionamento manuale
- Approssimazione valore di uscita sostitutivo
- Misura del tempo di regolazione
- Controllo errori
- Approssima valore di uscita sostitutivo con controllo errori
- Funzionamento manuale senza segnali di riscontro

Una descrizione dettagliata dei modi di funzionamento è disponibile al paragrafo relativo al parametro State.

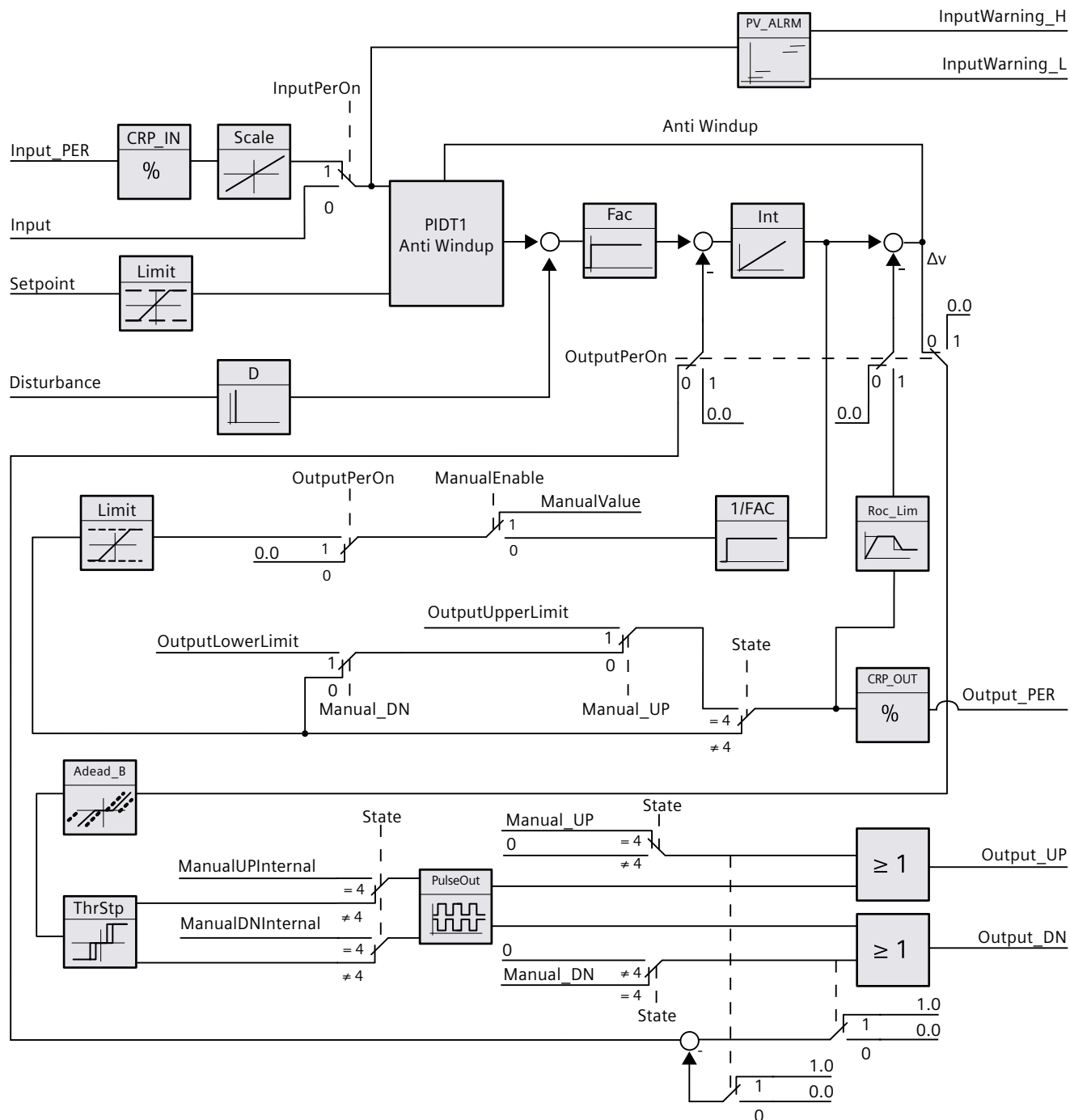
Algoritmo PID

PID_3Step è un regolatore PIDT1 con anti-windup e ponderazione del componente P e D. L'algoritmo PID utilizza la seguente formula:

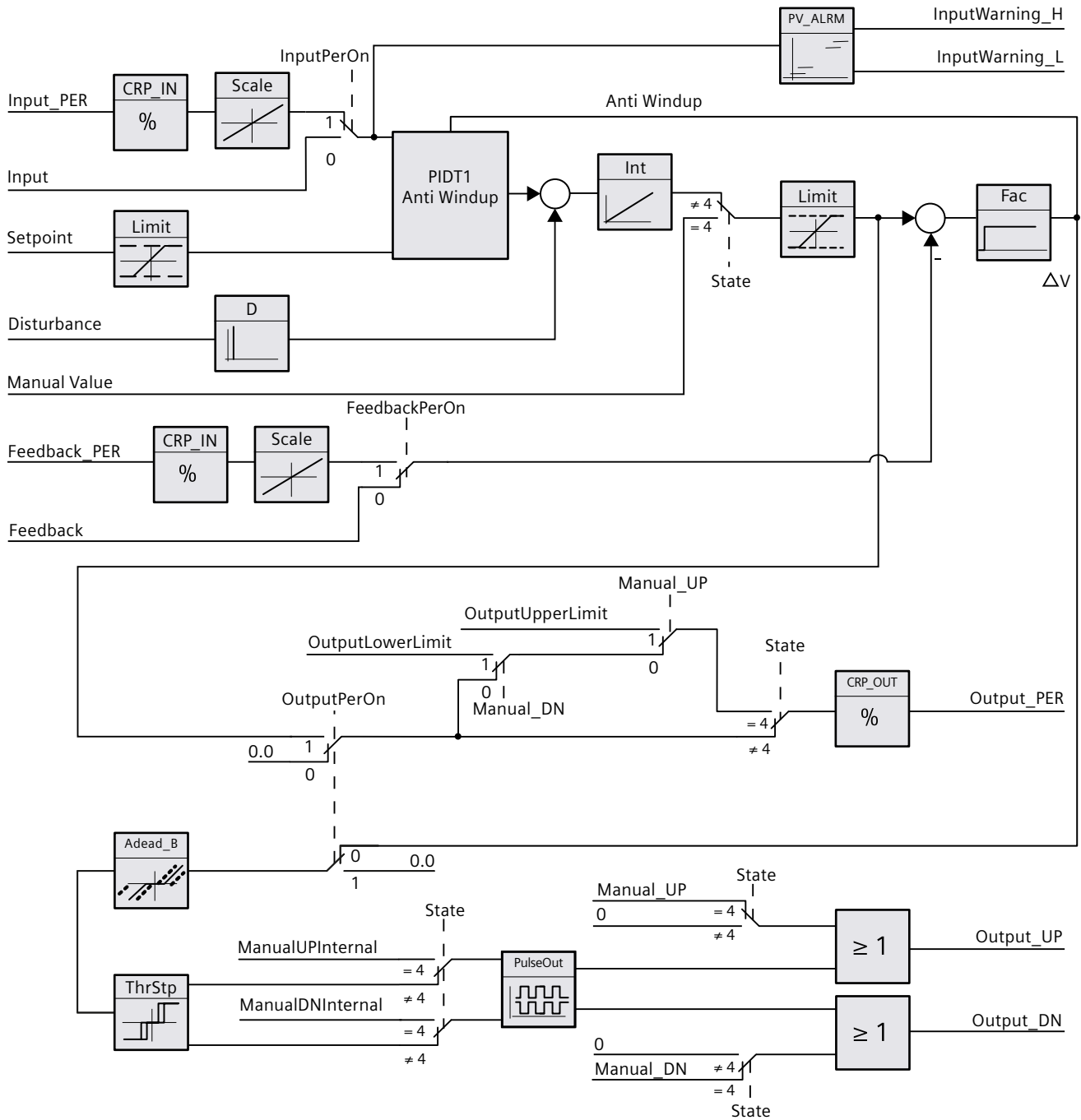
$$\Delta y = K_p \cdot s \cdot \left[(b \cdot w - x) + \frac{1}{T_I \cdot s} (w - x) + \frac{T_D \cdot s}{a \cdot T_D \cdot s + 1} (c \cdot w - x) \right]$$

Δy	Valore di uscita dell'algoritmo PID
K_p	Guadagno proporzionale
s	Operatore di Laplace
b	Ponderazione del componente P
w	Setpoint
x	Valore istantaneo
T_I	Tempo di integrazione
T_D	Tempo derivativo
a	Coefficiente per il ritardo derivativo (ritardo derivativo $T1 = a \times T_D$)
c	Ponderazione del componente D

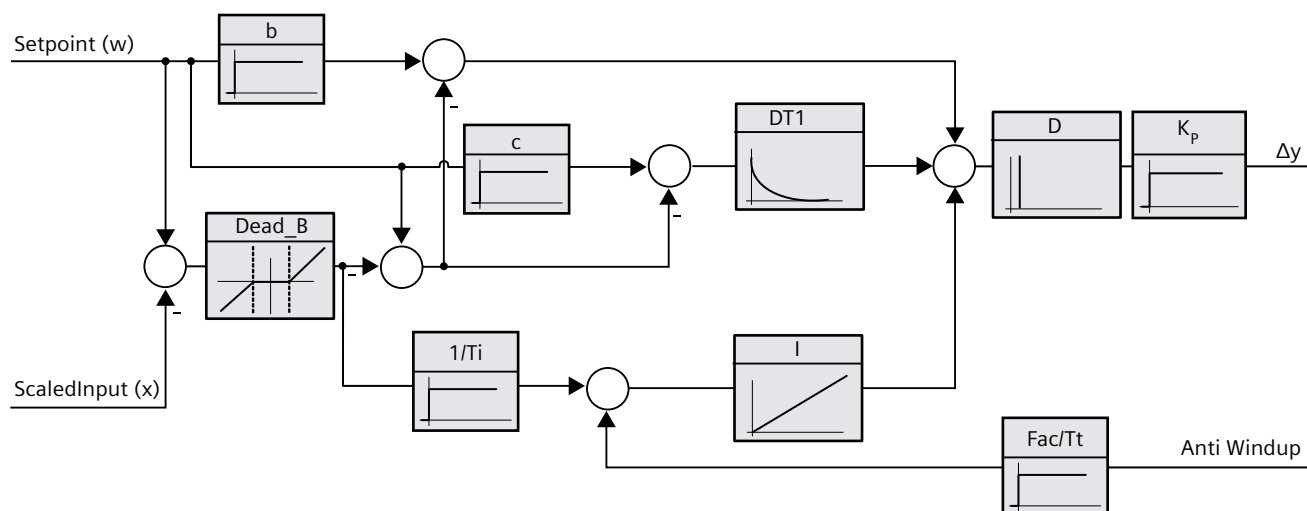
Schema a blocchi senza risposta di posizione



Schema a blocchi con risposta di posizione



Schema a blocchi di PIDT1 con anti-windup



Richiamo

PID_3Step viene richiamato a intervalli costanti nel ciclo di un OB di schedulazione orologio.

Caricamento nel dispositivo

L'aggiornamento dei valori attuali delle variabili a ritenzione, avviene soltanto con il caricamento completo di PID_3Step.

Caricamento degli oggetti tecnologici del dispositivo [\(Pagina 48\)](#)

Avviamento

All'avviamento della CPU PID_3Step si avvia nel modo di funzionamento salvato nel parametro di transito Mode. Per lasciare PID_3Step nel modo di funzionamento "Inattivo" impostare RunModeByStartup = FALSE.

Comportamento in caso di errori

In funzionamento automatico e durante la messa in servizio il comportamento in caso di errori varia in funzione delle variabili ErrorBehaviour e ActivateRecoverMode. In funzionamento manuale il comportamento dipende da ErrorBehaviour e ActivateRecoverMode. Se ActivateRecoverMode = TRUE il comportamento dipende anche dall'errore che si è verificato.

ErrorBehaviour	ActivateRecoverMode	Editor di configurazione > Impostazioni dell'organo attuatore > Imposta Output su	Comportamento
FALSE	FALSE	Valore di uscita attuale	Commutazione nel modo di funzionamento "Inattivo" (State = 0) L'organo attuatore rimane nella posizione attuale.
FALSE	TRUE	Valore di uscita attuale per la durata dell'errore	Commutazione nel modo di funzionamento "Controllo errori" (State = 7) L'organo attuatore rimane nella posizione attuale per la durata dell'errore.
TRUE	FALSE	Valore di uscita sostitutivo	Commutazione nel modo di funzionamento "Approssimazione del valore di uscita sostitutivo" (State = 5) L'organo attuatore viene portato sul valore di uscita sostitutivo configurato. Commutazione nel modo di funzionamento "Inattivo" (State = 0) L'organo attuatore rimane nella posizione attuale.
TRUE	TRUE	Valore di uscita sostitutivo per la durata dell'errore	Commutazione nel modo di funzionamento "Approssimazione del valore di uscita sostitutivo con controllo errori" (State = 8) L'organo attuatore viene portato sul valore di uscita sostitutivo configurato. Commutazione nel modo di funzionamento "Controllo errori" (State = 7)

utilizza ManualValue come valore di uscita in funzionamento manuale, tranne che in presenza dei seguenti errori:

- 2000h: Valore del parametro Feedback_PER non valido.
- 4000h: Valore del parametro Feedback non valido.
- 8000h: Errore nella risposta di posizione digitale.

In questo modo la posizione dell'organo attuatore può essere modificata solo con Manual_UP e Manual_DN, non con ManualValue.

Il parametro Error indica se in questo ciclo si è verificato un errore. Il parametro ErrorBits indica quali errori si sono verificati. ErrorBits viene resettato da un fronte di salita in Reset o ErrorAck.

Vedere anche

[Parametri State e Mode V2 \(Pagina 336\)](#)

[Parametro ErrorBits V2 \(Pagina 341\)](#)

[Configurazione di PID_3Step V2 \(Pagina 125\)](#)

10.2.4.2 Funzionamento di PID_3Step V2

Sorveglianza dei limiti del valore istantaneo

Nelle variabili Config.InputUpperLimit e Config.InputLowerLimit si definisce il limite superiore e inferiore del valore istantaneo. Se il valore istantaneo non rientra in questi limiti si verifica un errore (ErrorBits = 0001h).

Nelle variabili Config.InputUpperWarning e Config.InputLowerWarning si definisce il limite di avviso superiore e inferiore del valore istantaneo. Se il valore istantaneo non rientra in questi limiti viene emesso un avviso (Warning = 0040h) e il parametro di uscita InputWarning_H o InputWarning_L diventa TRUE.

Definizione dei limiti per il setpoint

Nelle variabili Config.SetpointUpperLimit e Config.SetpointLowerLimit si definisce un limite superiore e inferiore per il setpoint. PID_3Step circonda automaticamente il setpoint ai limiti del valore istantaneo. È inoltre possibile delimitare il setpoint entro un campo più piccolo. PID_3Step verifica se il campo è compreso entro i limiti del valore istantaneo. Se il setpoint non rientra in questi limiti, il limite superiore o inferiore viene utilizzato come setpoint e il parametro di uscita SetpointLimit_H o SetpointLimit_L diventa TRUE.

Il setpoint viene circoscritto in tutti i modi operativi.

Limitazione del valore di uscita

Nelle variabili Config.OutputUpperLimit e Config.OutputLowerLimit si definisce il limite superiore e inferiore del valore di uscita. I limiti del valore di uscita devono essere compresi tra il "Riscontro inferiore" e il "Riscontro superiore".

- Riscontro superiore: Config.FeedbackScaling.UpperPointOut
- Riscontro inferiore: Config.FeedbackScaling.LowerPointOut

Deve valere:

$$\text{UpperPointOut} \geq \text{OutputUpperLimit} > \text{OutputLowerLimit} \geq \text{LowerPointOut}$$

I valori validi per "Riscontro superiore" e "Riscontro inferiore" dipendono da:

- FeedbackOn
- FeedbackPerOn
- OutputPerOn

OutputPerOn	FeedbackOn	FeedbackPerOn	LowerPointOut	UpperPointOut
FALSE	FALSE	FALSE	non impostabile (0.0 %)	non impostabile (100.0 %)
FALSE	TRUE	FALSE	-100.0 % oppure 0.0 %	0.0 % oppure +100.0 %
FALSE	TRUE	TRUE	-100.0 % oppure 0.0 %	0.0 % oppure +100.0 %
TRUE	FALSE	FALSE	non impostabile (0.0 %)	non impostabile (100.0 %)
TRUE	TRUE	FALSE	-100.0 % oppure 0.0 %	0.0 % oppure +100.0 %
TRUE	TRUE	TRUE	-100.0 % oppure 0.0 %	0.0 % oppure +100.0 %

Se OutputPerOn = FALSE e FeedbackOn = FALSE, non è possibile delimitare il valore di uscita. Output_UP e Output_DN vengono quindi resettati se Actuator_H = TRUE o Actuator_L = TRUE. Se non sono presenti neppure segnali di riscontro Output_UP e Output_DN vengono resettati dopo un tempo di spostamento pari a $\text{Config.VirtualActuatorLimit} \times \text{Retain.TransitTime}/100$. Da PID_3Step versione 2.3 il controllo e la limitazione della durata corsa con $\text{Config.VirtualActuatorLimit} = 0.0$ possono essere disattivati.

Il valore di uscita corrisponde a 27648 al 100 % e a -27648 a -100 %. PID_3Step deve essere in grado di chiudere completamente la valvola.

NOTA

Impiego con due o più attuatori

PID_3 Step non si addice all'impiego con due o più attuatori (ad es. in applicazioni di riscaldamento/raffreddamento), in quanto per la realizzazione di un buon comportamento di regolazione, attuatori diversi richiedono anche parametri PID diversi.

Valore di uscita sostitutivo

In caso di errore PID_3Step può emettere un valore di uscita sostitutivo e spostare l'organo attuatore in una posizione sicura che viene definita nella variabile SavePosition. Il valore di uscita sostitutivo deve essere compreso entro i limiti del valore di uscita.

Controllo della validità dei segnali

In caso di utilizzo, viene controllata la validità dei valori dei parametri seguenti:

- Setpoint
- Input
- Input_PER
- Input_PER
- Feedback
- Feedback_PER
- Disturbance
- ManualValue

- SavePosition
- Output_PER

Sorveglianza del tempo di campionamento PID_3Step

Il tempo di campionamento dovrebbe corrispondere al tempo di ciclo dell'OB richiamante. L'istruzione PID_3Step misura di volta in volta l'intervallo tra due richiami. Questo è il tempo di campionamento attuale. Ad ogni commutazione del modo di funzionamento e al primo avvio viene generato il valore medio dei primi 10 tempi di campionamento. Se il tempo di campionamento attuale si discosta troppo da questo valore medio si verifica un errore (ErrorBits = 0800h).

Durante l'ottimizzazione si verifica questo errore se:

- Nuovo valore medio $\geq 1,1 \times$ valore medio precedente
- Nuovo valore medio $\leq 0,9 \times$ valore medio precedente

In funzionamento automatico si verifica questo errore se:

- Nuovo valore medio $\geq 1,5 \times$ valore medio precedente
- Nuovo valore medio $\leq 0,5 \times$ valore medio precedente

Disattivando il controllo del tempo di campionamento (CycleTime.EnMonitoring = FALSE) è possibile richiamare PID_3Step anche nell'OB1. A causa del tempo di campionamento variabile sarà però necessario accettare una qualità della regolazione più scarsa.

Tempo di campionamento dell'algoritmo PID

Poiché il sistema regolato richiede un certo tempo per reagire a una modifica del valore di uscita, è utile non calcolare questo valore in ogni ciclo. Il tempo di campionamento dell'algoritmo PID è il tempo che trascorre tra due calcoli del valore di uscita. Viene determinato durante l'ottimizzazione e arrotondato ad un multiplo del tempo di ciclo. Tutte le altre funzioni di PID_3Step vengono eseguite ad ogni richiamo.

Misura del tempo di regolazione motore

Il tempo di regolazione del motore è il tempo in secondi necessario al motore per spostare l'organo attuatore dallo stato chiuso a quello aperto. L'organo attuatore viene spostato in una direzione al massimo per $\text{Config.VirtualActuatorLimit} \times \text{Retain.TransitTime}/100$. Per ottenere un buon risultato di regolazione, PID_3Step richiede un tempo di regolazione del motore il più preciso possibile. I dati contenuti nella documentazione dell'organo attuatore sono valori medi per questo tipo di organi attuatori. Il valore può variare per l'organo attuatore effettivamente utilizzato. Il tempo di regolazione del motore può essere misurato durante la messa in servizio. Durante la misura del tempo di regolazione del motore, i limiti del valore di uscita non vengono considerati. L'organo attuatore può essere traslato fino al riscontro superiore o fino a quello inferiore.

Il tempo di regolazione del motore viene considerato sia nel calcolo del valore di uscita analogico, sia nel calcolo dei valori di uscita digitali. Questo tempo è necessario soprattutto nell'ottimizzazione automatica e nel comportamento anti-windup per un corretto

funzionamento. Configurare pertanto il tempo di regolazione con un valore necessario al motore per spostare l'attuatore dallo stato chiuso a quello aperto.

Se nel processo non è attivo un tempo di regolazione del motore rilevante (ad es. con le valvole magnetiche) in modo da consentire al valore di uscita di agire direttamente e pienamente nel processo, utilizzare in suo luogo PID_Compact.

Senso di regolazione

Aumentando il valore di uscita si dovrebbe ottenere un aumento del valore istantaneo. In questo caso si parla di un senso di regolazione normale. Per raffreddamenti e regolazioni di flusso può essere necessario invertire il senso di regolazione. PID_3Step non opera con guadagno proporzionale negativo. Se InvertControl = TRUE, un aumento della differenza di regolazione determina una riduzione del valore di uscita. Il senso di regolazione viene preso in considerazione anche durante l'ottimizzazione iniziale e l'ottimizzazione fine.

Vedere anche

[Configurazione di PID_3Step V1 \(Pagina 145\)](#)

10.2.4.3 Modifiche dell'interfaccia PID_3Step V2

La seguente tabella indica le modifiche dell'interfaccia dell'istruzione PID_3Step.

PID_3Step V1	PID_3Step V2	Modifica
Input_PER	Input_PER	Tipo di dati da Word a Int
Feedback_PER	Feedback_PER	Tipo di dati da Word a Int
	Disturbance	Nuovo
Manual_UP	Manual_UP	Funzione
Manual_DN	Manual_DN	Funzione
	ErrorAck	Nuovo
	ModeActivate	Nuovo
Output_PER	Output_PER	Tipo di dati da Word a Int
	ManualUPInternal	Nuovo
	ManualDNInternal	Nuovo
	CancelTuningLevel	Nuovo
	VirtualActuatorLimit	Nuovo
Config.Loadbackup	Loadbackup	Rinominato
Config.TransitTime	Retain.TransitTime	Rinominato e inserita ritenzione
GetTransitTime.Start		Sostituita da Mode e ModeActivate.
SUT.CalculateSUTParams	SUT.CalculateParams	Rinominato
SUT.TuneRuleSUT	SUT.TuneRule	Rinominato

PID_3Step V1	PID_3Step V2	Modifica
TIR.CalculateTIRParams	TIR.CalculateParams	Rinominato
TIR.TuneRuleTIR	TIR.TuneRule	Rinominato
Retain.Mode	Mode	Funzione Dichiarazione di Static a parametro di transito

10.2.4.4 Parametri di ingresso di PID_3Step V2

I nomi dei seguenti parametri si applicano sia al blocco dati sia all'accesso tramite l'API Openness.

Tabella 10-7

Parametro	Tipo di dati	Preassegnazione	Descrizione
Setpoint	REAL	0.0	Setpoint del regolatore PID nel funzionamento automatico
Input	REAL	0.0	Una variabile del programma utente viene utilizzata come sorgente del valore istantaneo. Per poter utilizzare il parametro Input è necessaria l'impostazione Config.InputPerOn = FALSE.
Input_PER	INT	0	Un ingresso analogico viene utilizzato come sorgente del valore istantaneo. Per poter utilizzare il parametro Input_PER è necessaria l'impostazione Config.InputPerOn = TRUE.
Actuator_H	BOOL	FALSE	Risposta di posizione digitale della valvola per il riscontro superiore Se Actuator_H = TRUE la valvola è posizionata sul riscontro superiore e non viene più mossa in questa direzione.
Actuator_L	BOOL	FALSE	Risposta di posizione digitale della valvola per il riscontro inferiore Se Actuator_L = TRUE la valvola è posizionata sul riscontro inferiore e non viene più mossa in questa direzione.
Feedback	REAL	0.0	Risposta di posizione della valvola Per poter utilizzare il parametro Feedback è necessaria l'impostazione Config.FeedbackPerOn = FALSE.
Feedback_PER	INT	0	Risposta di posizione analogica di una valvola Per poter utilizzare il parametro Feedback_PER è necessaria l'impostazione Config.FeedbackPerOn = TRUE. Feedback_PER viene riportato in scala sulla base delle variabili: - Config.FeedbackScaling.LowerPointIn - Config.FeedbackScaling.UpperPointIn - Config.FeedbackScaling.LowerPointOut - Config.FeedbackScaling.UpperPointOut
Disturbance	REAL	0.0	Grandezza di disturbo o valore di precomando
ManualEnable	BOOL	FALSE	- Il fronte FALSE -> TRUE attiva il modo "Funzionamento manuale", State = 4, Mode rimane invariato. Finché ManualEnable = TRUE non è possibile modificare il modo di funzionamento attraverso un fronte di salita in ModeActivate e utilizzare la finestra di messa in servizio. - Il fronte TRUE -> FALSE attiva il modo di funzionamento predefinito da Mode.

Parametro	Tipo di dati	Preassegnazione	Descrizione
			Si consiglia di modificare il modo di funzionamento solo tramite ModeActivate.
ManualValue	REAL	0.0	Nel funzionamento manuale viene predefinita la posizione assoluta della valvola. ManualValue viene analizzato solo se si utilizza Output_PER oppure se è disponibile una risposta di posizione.
Manual_UP	BOOL	FALSE	- Manual_UP = TRUE La valvola si apre anche se si utilizza Output_PER o una risposta di posizione. Quando viene raggiunto il riscontro superiore la valvola non viene più spostata. Vedere anche Config.VirtualActuatorLimit - Manual_UP = FALSE Se si utilizza Output_PER o una risposta di posizione la valvola viene portata su ManualValue. In caso contrario la valvola non viene più spostata. Se Manual_UP e Manual_DN sono impostati a TRUE contemporaneamente, la valvola non viene spostata.
Manual_DN	BOOL	FALSE	- Manual_DN = TRUE La valvola si chiude anche se si utilizza Output_PER o una risposta di posizione. Quando viene raggiunto il riscontro inferiore la valvola non viene più spostata. Vedere anche Config.VirtualActuatorLimit - Manual_DN = FALSE Se si utilizza Output_PER o una risposta di posizione la valvola viene portata su ManualValue. In caso contrario la valvola non viene più spostata.
ErrorAck	BOOL	FALSE	Fronte FALSE -> TRUE ErrorBits e Warning vengono resettati.
Reset	BOOL	FALSE	Esegue un nuovo avvio del regolatore. - Fronte FALSE -> TRUE - Commutazione nel modo di funzionamento "Inattivo" - ErrorBits e Warnings vengono resettati. - Fintantoché Reset = TRUE, - PID_3Step rimane nel modo di funzionamento "Inattivo" (State = 0). - il modo di funzionamento non può essere modificato tramite Mode e ModeActivate o ManualEnable. - La finestra di dialogo per la messa in servizio non può essere utilizzata. - Fronte TRUE -> FALSE - Se ManualEnable = FALSE, PID_3Step commuta nel modo operativo salvato in Mode. - Se Mode = 3, la commutazione nel funzionamento automatico avviene bumpless.
ModeActivate	BOOL	FALSE	Fronte FALSE -> TRUE PID_3Step passa al modo di funzionamento salvato in Mode.

10.2.4.5 Parametri di uscita PID_3Step V2

I nomi dei seguenti parametri si applicano sia al blocco dati sia all'accesso tramite l'API Openness.

Tabella 10-8

Parametro	Tipo di dati	Preassegnazio- ne	Descrizione
ScaledInput	REAL	0.0	Valore istantaneo riportato in scala
ScaledFeedback	REAL	0.0	Risposta di posizione riportata in scala Nel caso di un organo attuatore senza risposta di posizione ScaledFeedback mostra la posizione in modo molto impreciso. ScaledFeedback si può utilizzare solo per una valutazione approssimativa della posizione attuale.
Output_UP	BOOL	FALSE	Valore di uscita digitale per l'apertura della valvola Se Config.OutputPerOn = FALSE viene utilizzato il parametro Output_UP.
Output_DN	BOOL	FALSE	Valore di uscita digitale per la chiusura della valvola Se Config.OutputPerOn = FALSE viene utilizzato il parametro Output_DN.
Output_PER	INT	0	Valore di uscita analogico Se Config.OutputPerOn = TRUE viene utilizzato Output_PER. Impiegare Output_PER, se come attuatore viene utilizzata una valvola accessibile da un'uscita analogica e comandata da un segnale continuo, ad es. 0...10V o 4...20mA. Il valore su Output_PER corrisponde alla destinazione della valvola, ad es. Output_PER = 13824, se la valvola deve essere aperta del 50%.
SetpointLimit_H	BOOL	FALSE	Se SetpointLimit_H = TRUE, è stato raggiunto il limite superiore assoluto del setpoint ($\text{Setpoint} \geq \text{Config.SetpointUpperLimit}$). Il setpoint è limitato a Config.SetpointUpperLimit.
SetpointLimit_L	BOOL	FALSE	Se SetpointLimit_L = TRUE, è stato raggiunto il limite inferiore assoluto del setpoint ($\text{Setpoint} \leq \text{Config.SetpointLowerLimit}$). Il setpoint è limitato a Config.SetpointLowerLimit.
InputWarning_H	BOOL	FALSE	Se InputWarning_H = TRUE il limite superiore di avviso del valore istantaneo è stato raggiunto o superato.
InputWarning_L	BOOL	FALSE	Se InputWarning_L = TRUE il limite inferiore di avviso del valore istantaneo è stato raggiunto o superato per difetto.
State	INT	0	Il parametro State (Pagina 336) indica il modo di funzionamento attuale del regolatore PID. Il modo di funzionamento si modifica con il parametro di ingresso Mode e con un fronte di salita in ModeActivate. • State = 0: Inattivo • State = 1: Ottimizzazione iniziale • State = 2: Ottimizzazione fine • State = 3: Funzionamento automatico • State = 4: Funzionamento manuale • State = 5: Approssimazione valore di uscita sostitutivo • State = 6: Misura del tempo di regolazione • State = 7: Controllo errori

Parametro	Tipo di dati	Preassegnazione	Descrizione
			- State = 8: Approssima valore di uscita sostitutivo con controllo errori - State = 10: Funzionamento manuale senza segnali di riscontro
Error	BOOL	FALSE	Se Error = TRUE in questo ciclo è presente almeno un messaggio di errore.
ErrorBits	DWORD	DW#16#0	Il parametro ErrorBits (Pagina 341) mostra quali messaggi di errore sono presenti. ErrorBits è a ritenzione e viene resettato in presenza di un fronte di salita in Reset o ErrorAck.

Vedere anche

[Parametri State e Mode V2 \(Pagina 336\)](#)

[Parametro ErrorBits V2 \(Pagina 341\)](#)

10.2.4.6 Parametri di transito di PID_3Step V2

I nomi dei seguenti parametri si applicano sia al blocco dati sia all'accesso tramite l'API Openness.

Tabella 10-9

Parametro	Tipo di dati	Preassegnazione	Descrizione
Mode	INT	4	In Mode si predefinisce il modo di funzionamento al quale deve passare PID_3Step. Sono possibili: - Mode = 0: Inattivo - Mode = 1: Ottimizzazione iniziale - Mode = 2: Ottimizzazione fine - Mode = 3: Funzionamento automatico - Mode = 4: Funzionamento manuale - Mode = 6: Misura del tempo di regolazione - Mode = 10: Funzionamento manuale senza segnali di riscontro Il modo di funzionamento viene attivato da: - Fronte di salita in ModeActivate - Fronte di discesa in Reset - Fronte di discesa in ManualEnable - Avviamento a freddo della CPU se RunModeByStartup = TRUE Mode è a ritenzione. I modi di funzionamento sono descritti dettagliatamente nella sezione Parametri State e Mode V2 (Pagina 336).

10.2.4.7 Variabili statiche di PID_3Step V2

NOTA

Per evitare un comportamento errato del regolatore PID modificare le variabili contrassegnate con ⁽¹⁾ solo nel modo di funzionamento "Inattivo".

I nomi delle seguenti variabili si applicano sia al blocco dati sia all'accesso tramite l'API Openness.

Variabile	Tipo di dati	Preassegnazione	Descrizione
ManualUpInternal	BOOL	FALSE	Nel funzionamento manuale ogni fronte di salita apre la valvola del 5% del campo di regolazione complessivo oppure per il tempo minimo di regolazione del motore. ManualUpInternal viene analizzato solo se non si utilizza né Output_PER né una risposta di posizione. Questa variabile viene utilizzata nella finestra di messa in servizio.
ManualDnInternal	BOOL	FALSE	Nel funzionamento manuale ogni fronte di salita chiude la valvola del 5% del campo di regolazione complessivo oppure per il tempo minimo di regolazione del motore. ManualDnInternal viene analizzato solo se non si utilizza né Output_PER né una risposta di posizione. Questa variabile viene utilizzata nella finestra di messa in servizio.
ActivateRecoverMode	BOOL	TRUE	La variabile ActivateRecoverMode V2 (Pagina 343) determina il comportamento in caso di errori.
RunModeByStartup	BOOL	TRUE	Dopo il nuovo avviamento della CPU attiva il modo di funzionamento in Mode Se RunModeByStartup = TRUE, dopo l'avvio della CPU PID_3Step si avvia nel modo di funzionamento salvato in Mode. Se RunModeByStartup = FALSE, dopo l'avvio della CPU PID_3Step resta nel modo di funzionamento "Inattivo".
LoadBackUp	BOOL	FALSE	Se LoadBackUp = TRUE viene ricaricato l'ultimo set di parametri PID. Il set è stato memorizzato prima dell'ultima ottimizzazione. LoadBackUp viene reimpostato automaticamente a FALSE.
PhysicalUnit	INT	0	Unità fisica del valore istantaneo e del setpoint, ad es. °C o °F. PhysicalUnit viene utilizzato per la visualizzazione negli editor e non incide in alcun modo sul comportamento dell'algoritmo di regolazione nella CPU. Con l'importazione di PID_3Step tramite l'API Openness PhysicalUnit viene ripristinato il valore di default.
PhysicalQuantity	INT	0	Grandezza fisica del valore istantaneo e del setpoint, ad es. temperatura. PhysicalQuantity viene utilizzato per la visualizzazione negli editor e non incide in alcun modo sul comportamento dell'algoritmo di regolazione nella CPU. Con l'importazione di PID_3Step tramite l'API Openness con PhysicalQuantity viene ripristinato il valore di default.
ErrorBehaviour	BOOL	FALSE	Se ErrorBehaviour = FALSE, in caso di errore la valvola rimane nella posizione attuale e il regolatore passa direttamente al modo di funzionamento "Inattivo" o "Controllo errori". Se ErrorBehaviour = TRUE, in caso di errore l'organo attuatore viene spostato sul valore di uscita sostitutivo e solo successivamente si passa al modo di funzionamento "Inattivo" o "Controllo errori".

Variabile	Tipo di dati	Preassegnazione	Descrizione
			Se si verificano gli errori seguenti non è più possibile portare la valvola a un valore di uscita sostitutivo. 2000h: Valore del parametro Feedback_PER non valido. 4000h: Valore del parametro Feedback non valido. 8000h: Errore nella risposta di posizione digitale. 20000h: Valore della variabile SavePosition non valido.
Warning	DWORD	DW#16#0	La variabile Warning (Pagina 336) mostra gli avvisi da quando Reset = TRUE o ErrorAck = TRUE. Warning è a ritenzione. Gli avvisi ciclici (ad es. avvisi del valore istantaneo) vengono visualizzati finché è presente la rispettiva causa. Una volta eliminata la causa gli avvisi vengono cancellati automaticamente. Gli avvisi non ciclici (ad es. punto di flesso non trovato) rimangono attivi e vengono cancellati come errori.
SavePosition	REAL	0.0	Valore di uscita sostitutivo Se ErrorBehaviour = TRUE, in caso di errore l'organo attuatore viene portato in una posizione sicura per l'impianto. Non appena viene raggiunto il valore di uscita sostitutivo, PID_3Step cambia modo di funzionamento in funzione di ActivateRecoverMode. Il campo dei valori ammesso dipende dalla configurazione. - Config.FeedbackOn = FALSE e Config.OutputPerOn = FALSE: SavePosition = 0.0 oppure 100.0 - Config.FeedbackOn = TRUE o Config.OutputPerOn = TRUE: Config.OutputUpperLimit $\geq$ SavePosition $\geq$ Config.OutputLowerLimit
CurrentSetpoint	REAL	0.0	Setpoint attualmente attivo. Questo valore viene congelato all'avvio dell'ottimizzazione.
CancelTuningLevel	REAL	10.0	Oscillazione del setpoint ammessa durante l'ottimizzazione. L'ottimizzazione viene interrotta solo se: - Setpoint > CurrentSetpoint + CancelTuningLevel oppure - Setpoint < CurrentSetpoint - CancelTuningLevel
Progress	REAL	0.0	Avanzamento dell'ottimizzazione in percentuale (0,0 - 100,0)
Config.InputPerOn ⁽¹⁾	BOOL	TRUE	Se InputPerOn = TRUE viene utilizzato il parametro Input_PER. Se InputPerOn = FALSE viene utilizzato il parametro Input.
Config.OutputPerOn ⁽¹⁾	BOOL	FALSE	Se OutputPerOn = TRUE viene utilizzato il parametro Output_PER. Se OutputPerOn = FALSE vengono utilizzati i parametri Output_UP e Output_DN.
Config.InvertControl ⁽¹⁾	BOOL	FALSE	Inversione del senso di regolazione Se InvertControl = TRUE, un aumento della differenza di regolazione determina una riduzione del valore di uscita.
Config.FeedbackOn ⁽¹⁾	BOOL	FALSE	Se FeedbackOn = FALSE viene simulata una risposta di posizione. Se FeedbackOn = TRUE viene in genere attivata la risposta di posizione.
Config.FeedbackPerOn ⁽¹⁾	BOOL	FALSE	FeedbackPerOn è valido solo se FeedbackOn = TRUE. Se FeedbackPerOn = TRUE viene utilizzato l'ingresso analogico per la risposta di posizione (parametro Feedback_PER). Se FeedbackPerOn = FALSE viene utilizzato il parametro Feedback per la risposta di posizione.
Config.ActuatorEndStopOn ⁽¹⁾	BOOL	FALSE	Se ActuatorEndStopOn = TRUE viene tenuta in considerazione la risposta di posizione digitale Actuator_L e Actuator_H.

Variabile	Tipo di dati	Preassegnazione	Descrizione
Config.InputUpperLimit ⁽¹⁾	REAL	120.0	Limite superiore del valore istantaneo Input e Input_PER vengono controllati per verificare il rispetto di tale limite. Nell'ingresso di periferia il valore istantaneo può superare al massimo del 18% il campo normale (campo di sovracomando). Non vengono più segnalati errori di superamento del "Limite superiore valore istantaneo". Vengono riconosciuti solo la rottura conduttore e il cortocircuito e PID_3Step si comporta come in caso di errore. InputUpperLimit > InputLowerLimit
Config.InputLowerLimit ⁽¹⁾	REAL	0.0	Limite inferiore del valore istantaneo InputLowerLimit < InputUpperLimit
Config.InputUpperWarning ⁽¹⁾	REAL	+3.40282-2e+38	Limite superiore di avviso del valore istantaneo Se si configura InputUpperWarning al di fuori dei limiti del valore istantaneo, viene utilizzato il limite superiore assoluto configurato per il valore istantaneo come limite superiore di avviso. Se si configura InputUpperWarning entro i limiti del valore istantaneo questo valore viene utilizzato come limite superiore di avviso. InputUpperWarning > InputLowerWarning InputUpperWarning ≤ InputUpperLimit
Config.InputLowerWarning ⁽¹⁾	REAL	-3.402822e+38	Limite inferiore di avviso del valore istantaneo Se si configura InputLowerWarning al di fuori dei limiti del valore istantaneo, il limite inferiore assoluto configurato per il valore istantaneo viene utilizzato come limite inferiore di avviso. Se si configura InputLowerWarning entro i limiti del valore istantaneo questo valore viene utilizzato come limite inferiore di avviso. InputLowerWarning < InputUpperWarning InputLowerWarning ≥ InputLowerLimit
Config.OutputUpperLimit ⁽¹⁾	REAL	100.0	Limite superiore del valore di uscita È consentito il campo di valori seguente: UpperPointOut ≥ OutputUpperLimit > OutputLowerLimit Per maggiori dettagli vedere OutputLowerLimit
Config.OutputLowerLimit ⁽¹⁾	REAL	0.0	Limite inferiore del valore di uscita È consentito il campo di valori seguente: OutputUpperLimit > OutputLowerLimit ≥ LowerPointOut Se si utilizza Output_PER un limite del valore di uscita di -100% corrisponde al valore Output_PER = -27648; il 100% corrisponde al valore Output_PER = 27648. Se OutputPerOn = FALSE e FeedbackOn = FALSE, OutputLowerLimit e OutputUpperLimit non vengono analizzati. Output_UP e Output_DN vengono resettati se Actuator_H = TRUE o Actuator_L = TRUE (se ActuatorEndStopOn = TRUE) o dopo una durata corsa di Config.VirtualActuatorLimit * Retain.TransitTime/100 (se ActuatorEndStopOn = FALSE).
Config.SetpointUpperLimit ⁽¹⁾	REAL	+3.40282-2e+38	Limite superiore del setpoint Se si configura SetpointUpperLimit al di fuori dei limiti del valore istantaneo, il limite superiore assoluto configurato per il valore istantaneo viene preassegnato come limite superiore del setpoint. Se si configura SetpointUpperLimit entro i limiti del valore istantaneo questo valore viene utilizzato come limite superiore del setpoint.

Variabile	Tipo di dati	Preassegnazione	Descrizione
Config.SetpointLowerLimit ⁽¹⁾	REAL	-3.402822e+38	Limite inferiore del setpoint Se si configura SetpointLowerLimit al di fuori dei limiti del valore istantaneo, il limite inferiore assoluto configurato per il valore istantaneo viene preassegnato come limite inferiore del setpoint. Se si configura SetpointLowerLimit entro i limiti del valore istantaneo questo valore viene utilizzato come limite inferiore del setpoint.
Config.MinimumOnTime ⁽¹⁾	REAL	0.0	Tempo minimo di attivazione Tempo minimo di attivazione dell'attuatore espresso in secondi. Config.MinimumOnTime è attivo soltanto se Output_UP e Output_DN vengono utilizzati (Config.OutputPerOn = FALSE).
Config.MinimumOffTime ⁽¹⁾	REAL	0.0	Tempo minimo di disattivazione Tempo minimo di disattivazione dell'attuatore espresso in secondi. Config.MinimumOffTime è attivo soltanto se Output_UP e Output_DN vengono utilizzati (Config.OutputPerOn = FALSE).
Config.VirtualActuatorLimit ⁽¹⁾	REAL	150.0	Se le seguenti condizioni sono soddisfatte l'organo attuatore viene spostato al massimo per un tempo pari a VirtualActuatorLimit × Retain.TransitTime/100 in una direzione e viene emesso l'avviso 2000h: • Config.OutputPerOn = FALSE • Config.ActuatorEndStopOn = FALSE • Config.FeedbackOn = FALSE Se Config.OutputPerOn = FALSE e Config.ActuatorEndStopOn = TRUE o Config.FeedbackOn = TRUE viene solo emesso l'avviso 2000h. Se Config.OutputPerOn = TRUE VirtualActuatorLimit non viene considerato. Da PID_3Step versione 2.3 il controllo e la limitazione della durata corsa con Config.VirtualActuatorLimit = 0.0 possono essere disattivati.
Config.InputScaling.UpperPointIn ⁽¹⁾	REAL	27648.0	Scala Input_PER superiore Sulla base delle due coppie di valori UpperPointOut, UpperPointIn e LowerPointOut, LowerPointIn della struttura InputScaling, Input_PER viene convertito in percentuale.
Config.InputScaling.LowerPointIn ⁽¹⁾	REAL	0.0	Scala Input_PER inferiore Sulla base delle due coppie di valori UpperPointOut, UpperPointIn e LowerPointOut, LowerPointIn della struttura InputScaling, Input_PER viene convertito in percentuale.
Config.InputScaling.UpperPointOut ⁽¹⁾	REAL	100.0	Valore istantaneo superiore riportato in scala Sulla base delle due coppie di valori UpperPointOut, UpperPointIn e LowerPointOut, LowerPointIn della struttura InputScaling, Input_PER viene convertito in percentuale.
Config.InputScaling.LowerPointOut ⁽¹⁾	REAL	0.0	Valore istantaneo inferiore riportato in scala Sulla base delle due coppie di valori UpperPointOut, UpperPointIn e LowerPointOut, LowerPointIn della struttura InputScaling, Input_PER viene convertito in percentuale.
Config.FeedbackScaling.UpperPointIn ⁽¹⁾	REAL	27648.0	Scala Feedback_PER superiore Sulla base delle due coppie di valori UpperPointOut, UpperPointIn e LowerPointOut, LowerPointIn della struttura FeedbackScaling, Feedback_PER viene convertito in percentuale.

Variabile	Tipo di dati	Preassegnazione	Descrizione
Config.FeedbackScaling.LowerPointIn ⁽¹⁾	REAL	0.0	Scala Feedback_PER inferiore Sulla base delle due coppie di valori UpperPointOut, UpperPointIn e LowerPointOut, LowerPointIn della struttura FeedbackScaling, Feedback_PER viene convertito in percentuale.
Config.FeedbackScaling.UpperPointOut ⁽¹⁾	REAL	100.0	Riscontro superiore Sulla base delle due coppie di valori UpperPointOut, UpperPointIn e LowerPointOut, LowerPointIn della struttura FeedbackScaling, Feedback_PER viene convertito in percentuale. Il campo dei valori ammesso dipende dalla configurazione. - FeedbackOn = FALSE: UpperPointOut = 100.0 - FeedbackOn = TRUE: UpperPointOut = 100.0 o 0.0 UpperPointOut ≠ LowerPointOut
Config.FeedbackScaling.LowerPointOut ⁽¹⁾	REAL	0.0	Riscontro inferiore Sulla base delle due coppie di valori UpperPointOut, UpperPointIn e LowerPointOut, LowerPointIn della struttura FeedbackScaling, Feedback_PER viene convertito in percentuale. Il campo dei valori ammesso dipende dalla configurazione. - FeedbackOn = FALSE: LowerPointOut = 0.0 - FeedbackOn = TRUE: LowerPointOut = 0.0 o -100.0 LowerPointOut ≠ UpperPointOut
GetTransitTime.InvertDirection	BOOL	FALSE	Se InvertDirection = FALSE per il rilevamento del tempo di regolazione la valvola viene aperta completamente, chiusa e infine riaperta. Se InvertDirection = TRUE, la valvola viene chiusa completamente, aperta e infine richiusa.
GetTransitTime.SelectFeedback	BOOL	FALSE	Se SelectFeedback = TRUE, nella misura del tempo di regolazione viene considerato Feedback_PER o Feedback. Se SelectFeedback = FALSE, nella misura del tempo di regolazione vengono considerati Actuator_H e Actuator_L.
GetTransitTime.State	INT	0	Fase attuale della misura del tempo di regolazione - State = 0: Inattivo - State = 1: Apertura completa della valvola - State = 2: Chiusura completa della valvola - State = 3: impostazione della valvola sulla destinazione (NewOutput) - State = 4: Misurazione del tempo di regolazione terminata correttamente - State = 5: Misurazione del tempo di regolazione interrotta
GetTransitTime.NewOutput	REAL	0.0	Destinazione per la misura del tempo di regolazione con risposta di posizione La destinazione deve essere compresa tra il "Riscontro superiore" e il "Riscontro inferiore". La differenza tra NewOutput e ScaledFeedback deve essere pari almeno al 50 % del campo di regolazione consentito.
CycleTime.StartEstimation	BOOL	TRUE	Se StartEstimation = TRUE viene avviata la misura del tempo di campionamento PID_3Step. Al termine dell'operazione di misura risulta CycleTime.StartEstimation = FALSE.

10.2 PID_3Step

Variabile	Tipo di dati	Preassegnazione	Descrizione
CycleTime.EnEstimation	BOOL	TRUE	Se EnEstimation = TRUE viene calcolato il tempo di campionamento PID_3Step. Se CycleTime.EnEstimation = FALSE, il tempo di campionamento di PID_3Step non viene calcolato e si deve configurare manualmente il CycleTime.Value corretto.
CycleTime.EnMonitoring	BOOL	TRUE	Se EnMonitoring = TRUE viene sorvegliato il tempo di campionamento PID_3Step. Se non è possibile eseguire PID_3Step entro il tempo di campionamento, viene emesso l'errore 0800h e si passa a un altro modo di funzionamento. Il modo di funzionamento a cui il regolatore passa dipende da ActivateRecoverMode e ErrorBehaviour. Se EnMonitoring = FALSE, il tempo di campionamento PID_3Step non viene sorvegliato, l'errore 0800h non viene emesso e non si passa ad un altro modo di funzionamento.
CycleTime.Value ⁽¹⁾	REAL	0.1	Tempo di campionamento PID_3Step in secondi CycleTime.Value viene rilevata automaticamente e corrisponde in genere al tempo di ciclo dell'OB richiamante.
CtrlParamsBackUp.SetByUser	BOOL	FALSE	Valore memorizzato di Retain.CtrlParams.SetByUser I valori della struttura CtrlParamsBackUp possono essere ricaricati con LoadBackUp = TRUE.
CtrlParamsBackUp.Gain	REAL	1.0	Guadagno proporzionale memorizzato
CtrlParamsBackUp.Ti	REAL	20.0	Tempo di integrazione salvato in secondi
CtrlParamsBackUp.Td	REAL	0.0	Tempo derivativo salvato in secondi
CtrlParamsBackUp.TdFiltRatio	REAL	0.2	Coefficiente per il ritardo derivativo memorizzato
CtrlParamsBackUp.PWeighting	REAL	1.0	Ponderazione del componente P memorizzata
CtrlParamsBackUp.DWeighting	REAL	1.0	Ponderazione del componente D memorizzata
CtrlParamsBackUp.Cycle	REAL	1.0	Tempo di campionamento algoritmo PID salvato in secondi
CtrlParamsBackUp.InputDead-Band	REAL	0.0	Ampiezza banda morta della differenza di regolazione memorizzata
PIDSelfTune.SUT.CalculateParams	BOOL	FALSE	Le proprietà del sistema regolato vengono memorizzate durante l'ottimizzazione. Se CalculateParams = TRUE i parametri PID vengono ricalcolati sulla base di queste proprietà. I parametri PID vengono calcolati secondo il metodo impostato in TuneRule. Al termine del calcolo CalculateParams viene impostato su FALSE.
PIDSelfTune.SUT.TuneRule	INT	1	Calcolo dei parametri durante l'ottimizzazione iniziale secondo un metodo: - SUT.TuneRule = 0: PID veloce I (comportamento di regolazione più veloce con ampiezze del valore di uscita più elevate rispetto a SUT.TuneRule = 1) - SUT.TuneRule = 1: PID lento I (comportamento di regolazione più lento con ampiezze del valore di uscita più basse rispetto a SUT.TuneRule = 0) - SUT.TuneRule = 2: Chien, Hrones, Reswick PID - SUT.TuneRule = 3: Chien, Hrones, Reswick PI - SUT.TuneRule = 4: PID veloce II (comportamento di regolazione più veloce con ampiezze del valore di uscita più elevate rispetto a SUT.TuneRule = 5) - SUT.TuneRule = 5: PID lento II (comportamento di regolazione più lento con ampiezze del valore di uscita più basse rispetto a SUT.TuneRule = 4)

Variabile	Tipo di dati	Preassegnazione	Descrizione
			I metodi SUT.TuneRule = 0 e 1 si differenziano soltanto nel calcolo del guadagno proporzionale dai metodi SUT.TuneRule = 4 e 5: Con SUT.TuneRule = 0 e 1, il calcolo del guadagno proporzionale avviene sulla base del tempo di compensazione del processo. Con SUT.TuneRule = 4 e 5 ciò avviene sulla base del tempo di ritardo del processo. SUT.TuneRule = 4 e 5 fornisce un valore più elevato per il guadagno proporzionale e pertanto un comportamento di regolazione più veloce con ampiezze del valore di uscita più elevate rispetto a SUT.TuneRule = 0 e 1.
PIDSelfTune.SUT.State	INT	0	La variabile SUT.State mostra la fase attuale dell'ottimizzazione iniziale: - State = 0: Inizializzazione dell'ottimizzazione iniziale - State = 50: Rilevamento della posizione iniziale senza risposta di posizione - State = 100: Calcolo della divergenza standard - State = 200: Rilevazione del punto di flesso - State = 300: Rilevamento del tempo di stabilizzazione della regolazione - State = 9900: Ottimizzazione iniziale riuscita - State = 1: Ottimizzazione iniziale non riuscita
PIDSelfTune.TIR.RunIn	BOOL	FALSE	Con la variabile RunIn è possibile stabilire che un'ottimizzazione fine può essere eseguita anche senza un'ottimizzazione iniziale. - RunIn = FALSE Se l'ottimizzazione fine viene avviata dal modo di funzionamento inattivo o manuale, si avvia un'ottimizzazione iniziale. Se l'ottimizzazione fine viene avviata dal funzionamento automatico, viene eseguita la regolazione sul setpoint con i parametri PID disponibili. Solo in seguito viene avviata l'ottimizzazione fine. Se non è possibile eseguire un'ottimizzazione iniziale PID_3Step passa al modo di funzionamento dal quale è stata avviata l'ottimizzazione. - RunIn = TRUE L'ottimizzazione iniziale viene ignorata. PID_3Step tenta di raggiungere il setpoint con il valore di uscita minimo o massimo. Ciò può causare una maggiore sovraoscillazione. Solo in seguito viene avviata l'ottimizzazione fine. Al termine dell'ottimizzazione fine RunIn viene impostato su FALSE.
PIDSelfTune.TIR.CalculateParams	BOOL	FALSE	Le proprietà del sistema regolato vengono memorizzate durante l'ottimizzazione. Se CalculateParams = TRUE i parametri PID vengono ricalcolati sulla base di queste proprietà. I parametri PID vengono calcolati secondo il metodo impostato in TuneRule. Al termine del calcolo CalculateParams viene impostato su FALSE.
PIDSelfTune.TIR.TuneRule	INT	0	Calcolo dei parametri durante l'ottimizzazione fine secondo un metodo: - TIR.TuneRule = 0: PID automatico - TIR.TuneRule = 1: PID veloce (comportamento di regolazione più veloce con ampiezze del valore di uscita più elevate rispetto a TIR.TuneRule = 2) - TIR.TuneRule = 2: PID lento (comportamento di regolazione più lento con ampiezze del valore di uscita più basse rispetto a TIR.TuneRule = 1)

Variabile	Tipo di dati	Preassegnazione	Descrizione
			- TIR.TuneRule = 3: PID secondo Ziegler-Nichols - TIR.TuneRule = 4: PI secondo Ziegler-Nichols - TIR.TuneRule = 5: P secondo Ziegler-Nichols Affinché il calcolo dei parametri PIDcon TIR.CalculateParams e TIR.TuneRule = 0, 1 o 2 possa essere ripetuto, anche l'ottimizzazione fine precedente deve essere stata eseguita con TIR.TuneRule = 0, 1, o 2. In caso contrario viene utilizzato TIR.TuneRule = 3. Il nuovo calcolo del parametro PIDcon TIR.CalculateParams e TIR.TuneRule = 3, 4 o 5 è sempre possibile.
PIDSelfTune.TIR.State	INT	0	La variabile TIR.State mostra la fase attuale dell'"Ottimizzazione fine": - State = -100 Non è possibile eseguire l'ottimizzazione fine. Viene prima eseguita un'ottimizzazione iniziale. - State = 0: Inizializzazione dell'ottimizzazione fine - State = 200: Calcolo della divergenza standard - State = 300: Tentativi di raggiungere il setpoint con il valore di uscita massimo o minimo - State = 400: Tentativi di raggiungere il setpoint con i parametri PID disponibili(se l'ottimizzazione iniziale è riuscita) - State = 500: Rilevazione dell'oscillazione e calcolo dei parametri - State = 9900: Ottimizzazione fine riuscita - State = 1: Ottimizzazione fine non riuscita
Retain.TransitTime ⁽¹⁾	REAL	30.0	Tempo di regolazione del motore in secondi Tempo in secondi necessario all'attuatore per spostare la valvola dallo stato chiuso a quello aperto. TransitTime è a ritenzione.
Retain.CtrlParams.SetByUser ⁽¹⁾	BOOL	FALSE	Se SetByUser = FALSE i parametri PID vengono rilevati automaticamente e PID_3Step funziona con una banda morta nel valore di uscita. L'ampiezza della banda morta viene calcolata durante l'ottimizzazione sulla base dello scostamento standard del valore di uscita e memorizzata in Retain.CtrlParams.OutputDeadBand. Se SetByUser = TRUE i parametri PID vengono immessi manualmente e PID_3 Step funziona senza banda morta nel valore di uscita. Retain.CtrlParams.OutputDeadBand = 0.0 SetByUser è a ritenzione.
Retain.CtrlParams.Gain ⁽¹⁾	REAL	1.0	Guadagno proporzionale attivo Per invertire il senso di regolazione utilizzare la variabile Config.InvertControl. Anche valori negativi in Gain invertono il senso di regolazione. Si consiglia di modificare il senso di regolazione solo tramite InvertControl. Se InvertControl = TRUE e Gain < 0.0, anche il senso di regolazione è invertito. Gain è a ritenzione.
Retain.CtrlParams.Ti ⁽¹⁾	REAL	20.0	- Ti > 0.0: Tempo di integrazione attivo in secondi - Ti = 0.0: Il componente I è disattivato Ti è a ritenzione.
Retain.CtrlParams.Td ⁽¹⁾	REAL	0.0	- Td > 0.0: Tempo derivativo attivo in secondi - Td = 0.0: Il componente D è disattivato Td è a ritenzione.

Variabile	Tipo di dati	Preassegnazione	Descrizione
Retain.CtrlParams.TdFiltRatio ⁽¹⁾	REAL	0.2	Coefficiente per il ritardo di differenziazione attivo L'effetto del componente D viene ritardato dal coefficiente per il ritardo derivativo. Ritardo derivativo = Tempo derivativo × Coefficiente ritardo derivativo 0.0: Il componente D ha effetto solo per un ciclo e quindi è quasi inattivo. 0.5: Questo valore ha dato buoni risultati nelle operazioni con i sistemi regolati con una costante temporale dominante. > 1.0: Maggiore è il coefficiente, maggiore sarà il ritardo dell'azione del componente D. TdFiltRatio è a ritenzione.
Retain.CtrlParams.PWeighting ⁽¹⁾	REAL	1.0	Ponderazione del componente P attiva In caso di variazioni del setpoint è possibile ridurre il componente P. Sono adeguati valori da 0.0 a 1.0. 1.0: componente P pienamente attivo in caso di variazione del setpoint 0.0: componente P non attivo in caso di variazione del setpoint In caso di variazioni del valore istantaneo il componente P è sempre pienamente attivo. PWeighting è a ritenzione.
Retain.CtrlParams.DWeighting ⁽¹⁾	REAL	1.0	Ponderazione del componente D attiva In caso di variazioni del setpoint è possibile ridurre il componente D. Sono adeguati valori da 0.0 a 1.0. 1.0: componente D pienamente attivo in caso di variazione del setpoint 0.0: componente D non attivo in caso di variazione del setpoint In caso di variazioni del valore istantaneo il componente D è sempre pienamente attivo. DWeighting è a ritenzione.
Retain.CtrlParams.Cycle ⁽¹⁾	REAL	1.0	Tempo di campionamento attivo dell'algoritmo PID in secondi che viene arrotondato a un multiplo intero del tempo di ciclo dell'OB richiamante. Cycle è a ritenzione.
Retain.CtrlParams.InputDeadBand ⁽¹⁾	REAL	0.0	Ampiezza della banda morta della differenza di regolazione InputDeadBand è a ritenzione.

Vedere anche

[Parametri State e Mode V2 \(Pagina 336\)](#)

[Variabile ActivateRecoverMode V2 \(Pagina 343\)](#)

[Caricamento degli oggetti tecnologici del dispositivo \(Pagina 48\)](#)

10.2.4.8 Parametri State e Mode V2

Relazione tra i parametri

Il parametro State indica il modo di funzionamento attuale del regolatore PID. Il parametro State non è modificabile.

Con un fronte di salita in ModeActivate PID_3Step passa al modo di funzionamento salvato nel parametro di transito Mode.

Se la CPU si accende o passa da STOP a RUN, PID_3Step si avvia nel modo di funzionamento salvato in Mode. Per lasciare PID_3Step nel modo di funzionamento "Inattivo" impostare RunModeByStartup = FALSE.

Significato dei valori

State	Descrizione del modo di funzionamento
0	Inattivo Il regolatore è disattivato e non cambia più la posizione della valvola. La commutazione dal modo di funzionamento Inattivo al funzionamento automatico è bumpless.
1	Ottimizzazione iniziale L'ottimizzazione iniziale rileva la risposta del processo a un impulso del valore di uscita e cerca il punto di flesso. Dalla pendenza max. e dal tempo morto del sistema regolato vengono calcolati i parametri PID. I parametri PID migliori si ottengono eseguendo l'ottimizzazione iniziale e l'ottimizzazione fine. Requisiti per l'ottimizzazione iniziale: • Il tempo di regolazione del motore è stato configurato o misurato. • Modo di funzionamento Inattivo (State = 0), funzionamento manuale (State = 4) o automatico (State = 3) • ManualEnable = FALSE • Reset = FALSE • Il setpoint e il valore istantaneo sono compresi entro i limiti configurati. Più stabile è il valore istantaneo, più facilmente e con maggiore precisione si possono determinare i parametri PID. Un rumore del valore istantaneo è accettabile fintanto che l'incremento del valore è significativamente maggiore rispetto al rumore. Ciò si verifica con la maggiore probabilità nei modi "Inattivo" o "Funzionamento manuale". Il setpoint viene congelato nella variabile CurrentSetpoint. L'ottimizzazione viene interrotta se: • $\text{Setpoint} > \text{CurrentSetpoint} + \text{CancelTuningLevel}$ • $\text{Setpoint} < \text{CurrentSetpoint} - \text{CancelTuningLevel}$ Prima che i parametri PID vengano ricalcolati, vengono salvati e possono essere riattivati con LoadBackUp. Se l'ottimizzazione iniziale si conclude correttamente il regolatore passa al funzionamento automatico, in caso contrario la commutazione del modo di funzionamento dipende da ActivateRecoverMode e ErrorBehaviour. La fase dell'ottimizzazione iniziale viene visualizzata con PIDSelfTune.SUT.State.
2	Ottimizzazione fine L'ottimizzazione fine genera un'oscillazione costante e limitata del valore istantaneo. I parametri PID vengono ricalcolati partendo dall'ampiezza e dalla frequenza dell'oscillazione. I parametri PID risultanti dall'ottimizzazione fine mostrano in genere un comportamento pilota e di disturbo migliore rispetto ai parametri PID dell'ottimizzazione iniziale. I parametri PID migliori si ottengono eseguendo l'ottimizzazione iniziale e l'ottimizzazione fine. PID_3Step tenta di generare automaticamente un'oscillazione che sia maggiore rispetto al fruscio del valore istantaneo. L'ottimizzazione fine viene influenzata solo minimamente dalla stabilità del valore istantaneo. Il setpoint viene congelato nella variabile CurrentSetpoint. L'ottimizzazione viene interrotta se: • $\text{Setpoint} > \text{CurrentSetpoint} + \text{CancelTuningLevel}$ • $\text{Setpoint} < \text{CurrentSetpoint} - \text{CancelTuningLevel}$

State	Descrizione del modo di funzionamento
	Prima dell'ottimizzazione fine i parametri PID vengono salvati e possono essere riattivati con LoadBackUp. Requisiti per l'ottimizzazione fine: • Il tempo di regolazione del motore è stato configurato o misurato. • Il setpoint e il valore istantaneo sono compresi entro i limiti configurati. • ManualEnable = FALSE • Reset = FALSE • Tipo di funzionamento automatico (State = 3), Inattivo (State = 0) o manuale (State = 4) L'ottimizzazione fine si svolge nel modo seguente all'avvio: • Funzionamento automatico (State = 3) Per migliorare i parametri PID esistenti mediante un'ottimizzazione, avviare l'ottimizzazione fine dal modo di funzionamento automatico. PID_3Step esegue la regolazione con i parametri PID esistenti finché il circuito di regolazione non si assesta e finché non sono soddisfatti i requisiti per un'ottimizzazione fine. Solo in seguito viene avviata l'ottimizzazione fine. • Funzionamento Inattivo (State = 0) o manuale (State = 4) Se sono soddisfatti i presupposti dell'ottimizzazione iniziale, viene avviata un'ottimizzazione iniziale. Con i parametri PID rilevati viene quindi effettuata una regolazione finché il circuito di regolazione è assestato e sono soddisfatti i requisiti per l'ottimizzazione fine. Se PIDSelfTune.TIR.RunIn = TRUE l'ottimizzazione iniziale viene ignorata e si tenta di raggiungere il setpoint con il valore di uscita minimo o massimo. Ciò può causare una maggiore sovraelongazione. L'ottimizzazione fine si avvia poi automaticamente. Se l'ottimizzazione fine si conclude correttamente il regolatore passa al funzionamento automatico, in caso contrario la commutazione del modo di funzionamento dipende da ActivateRecoverMode e ErrorBehaviour. La fase dell'ottimizzazione fine viene visualizzata con PIDSelfTune.TIR.State.
3	Funzionamento automatico In funzionamento automatico PID_3Step regola il sistema regolato secondo i parametri preimpostati. Se è soddisfatta una delle seguenti condizioni si passa al funzionamento automatico: • Ottimizzazione iniziale conclusa correttamente • Ottimizzazione fine conclusa correttamente • Modifica del parametro di transito Mode al valore 3 e un fronte di salita in ModeActivate. La commutazione dal funzionamento automatico al funzionamento manuale è bumpless solo nell'editor di messa in servizio. In funzionamento automatico viene tenuta in considerazione la variabile ActivateRecoverMode.
4	Funzionamento manuale Nel funzionamento manuale si preimpostano manualmente i valori di uscita nei parametri Manual_UP e Manual_DN o ManualValue. Nel parametro ErrorBits è indicato se l'organo attuatore può essere spostato sul valore di uscita in caso di errore. Il modo di funzionamento si può attivare anche con ManualEnable = TRUE. Si consiglia di modificare i modi di funzionamento solo tramite Mode e ModeActivate. La commutazione dal funzionamento manuale al funzionamento automatico è bumpless. Il funzionamento manuale è possibile anche in presenza di un errore.
5	Approssimazione valore di uscita sostitutivo Questo modo di funzionamento viene attivato in caso di errore se Errorbehaviour = TRUE e ActivateRecoverMode = FALSE. PID_3Step sposta l'organo attuatore sul valore di uscita sostitutivo e passa quindi al modo di funzionamento "Inattivo".
6	Misura del tempo di regolazione Viene rilevato il tempo impiegato dal motore per aprire completamente la valvola chiusa. Questo modo di funzionamento si attiva con l'impostazione di Mode = 6 e ModeActivate = TRUE. Se per la misura del tempo di regolazione si utilizzano segnali di riscontro, la valvola viene aperta completamente dalla posizione attuale, chiusa completamente e riaperta completamente. Se GetTransitTime.InvertDirection = TRUE si verifica il comportamento inverso. Se per la misura del tempo di regolazione si utilizza una risposta di posizione, l'organo attuatore viene spostato dalla posizione attuale a una di destinazione.

State	Descrizione del modo di funzionamento
	Durante la misura del tempo di regolazione, i limiti del valore di uscita non vengono considerati. L'organo attuatore può essere traslato fino al riscontro superiore o fino a quello inferiore.
7	Controllo errori L'algoritmo di regolazione è disattivato e non modifica più la posizione della valvola. Questo modo di funzionamento viene attivato al posto di "Inattivo" in caso di errore. Devono essere soddisfatte tutte le condizioni seguenti: • Funzionamento automatico (Mode = 3) • Errorbehaviour = FALSE • ActivateRecoverMode = TRUE • Si sono verificati uno o più errori nei quali è attiva ActivateRecoverMode (Pagina 343). Non appena l'errore viene eliminato PID_3Step torna al funzionamento automatico.
8	Approssima valore di uscita sostitutivo con controllo errori Questo modo di funzionamento viene attivato in caso di errore invece del modo "Approssimazione del valore di uscita sostitutivo". PID_3Step sposta l'organo attuatore sul valore di uscita sostitutivo e passa quindi nel modo di funzionamento "Controllo errori". Devono essere soddisfatte tutte le condizioni seguenti: • Funzionamento automatico (Mode = 3) • Errorbehaviour = TRUE • ActivateRecoverMode = TRUE • Si sono verificati uno o più errori nei quali è attiva ActivateRecoverMode (Pagina 343). Non appena l'errore viene eliminato PID_3Step torna al funzionamento automatico.
10	Funzionamento manuale senza segnali di riscontro I segnali di riscontro non vengono considerati benché Config.ActuatorEndStopOn = TRUE. I limiti dei valori di uscita non vengono considerati. A parte questo PID_3Step si comporta esattamente come in funzionamento manuale.

Comportamento ENO

Se State = 0, ENO = FALSE.

Se State ≠ 0, ENO = TRUE.

Commutazione automatica del modo di funzionamento durante la messa in servizio

Al termine dell'ottimizzazione iniziale e dell'ottimizzazione fine senza errori viene attivato il funzionamento automatico. La tabella seguente mostra come cambiano Mode e State durante un'ottimizzazione iniziale corretta.

N° di ciclo	Mode	State	Operazione
0	4	4	Imposta Mode = 1
1	1	4	Imposta ModeActivate = TRUE
1	4	1	Il valore di State viene salvato in Mode L'ottimizzazione iniziale viene avviata
n	4	1	Ottimizzazione iniziale riuscita
n	3	3	Il funzionamento automatico viene avviato

In caso di errore PID_3Step cambia automaticamente modo di funzionamento. La tabella seguente mostra come cambiano Mode e State durante un'ottimizzazione iniziale scorretta.

N° di ciclo	Mode	State	Operazione
0	4	4	Imposta Mode = 1
1	1	4	Imposta ModeActivate = TRUE
1	4	1	Il valore di State viene salvato in Mode L'ottimizzazione iniziale viene avviata
n	4	1	Ottimizzazione iniziale interrotta
n	4	4	Il funzionamento manuale viene avviato

Se ActivateRecoverMode = TRUE viene attivato il modo di funzionamento salvato in Mode. All'avvio della misura del tempo di regolazione, dell'ottimizzazione iniziale o fine PID_3Step ha salvato il valore di State nel parametro di transito Mode. PID_3Step passa quindi al modo di funzionamento dal quale è stata avviata la misura o l'ottimizzazione.

Se ActivateRecoverMode = FALSE viene attivato il modo di funzionamento "Inattivo" o "Approssimazione del valore di uscita sostitutivo".

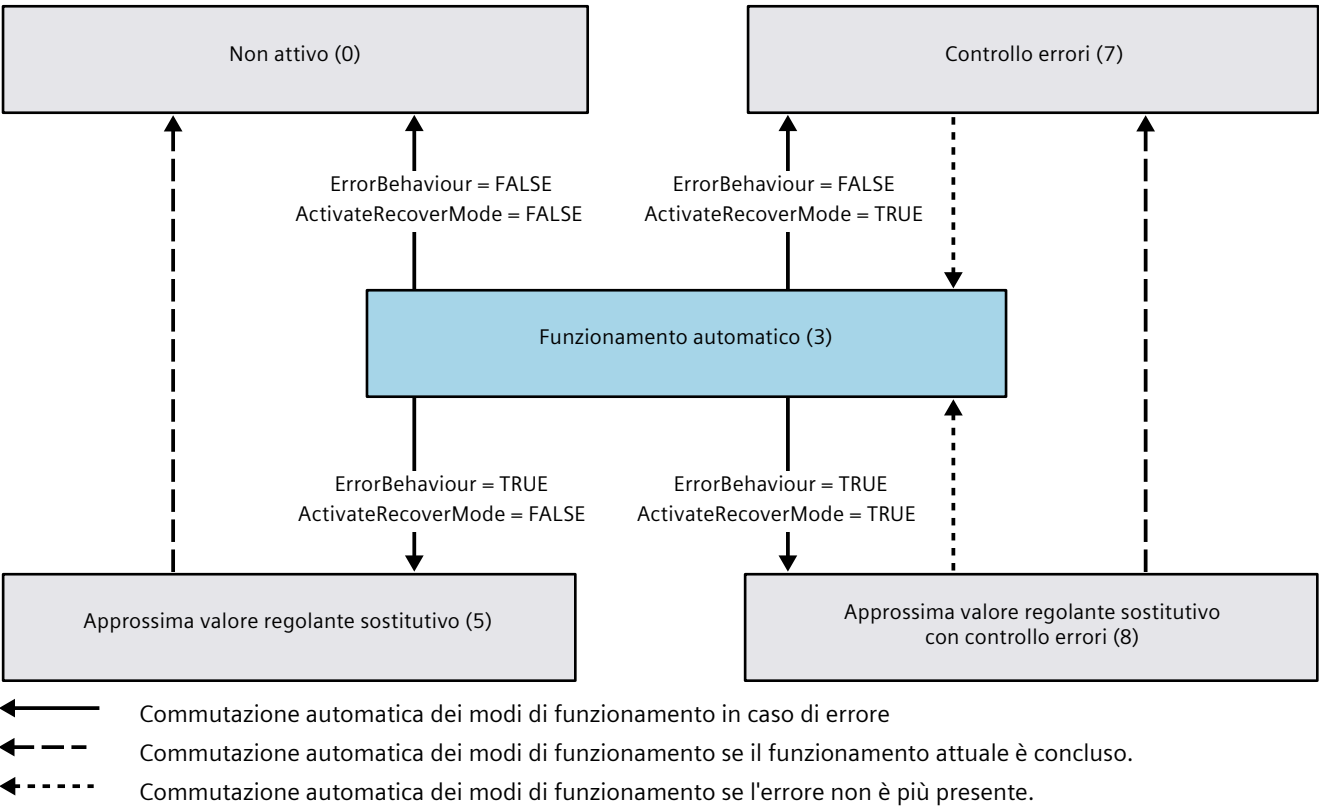
Commutazione automatica dei modi di funzionamento dopo la misura del tempo di regolazione

Se ActivateRecoverMode = TRUE, dopo la misura del tempo di regolazione, viene attivato il modo di funzionamento salvato in Mode.

Se ActivateRecoverMode = FALSE, dopo la misura del tempo di regolazione, ha luogo la commutazione sul modo di funzionamento "Inattivo".

Commutazione automatica del modo di funzionamento in funzionamento automatico

In caso di errore PID_3Step cambia automaticamente modo di funzionamento. Il diagramma seguente mostra l'effetto di ErrorBehaviour e ActivateRecoverMode su questa commutazione dei modi di funzionamento.



Vedere anche

- [Variabile ActivateRecoverMode V2 \(Pagina 343\)](#)
- [Parametro ErrorBits V2 \(Pagina 341\)](#)

10.2.4.9 Parametro ErrorBits V2

Se si verificano diversi errori contemporaneamente, i valori di ErrorBits vengono visualizzati in un'addizione binaria. Se ad es. viene visualizzato ErrorBits = 16#0000_0003 significa che sono presenti contemporaneamente gli errori 16#0000_0001 e 16#0000_0002.

Se è presente una risposta di posizione, PID_3Step utilizza ManualValue come valore di uscita in funzionamento manuale. L'unica eccezione è Errorbits = 16#0001_0000.

ErrorBits (DW#16#...)	Descrizione
0000_0000	Non sono presenti errori.
0000_0001	Il parametro "Input" non rientra nei limiti del valore istantaneo. - Input > Config.InputUpperLimit o - Input < Config.InputLowerLimit Se prima che si verificasse l'errore era attivo il funzionamento automatico e ActivateRecoverMode = TRUE, PID_3Step resta in funzionamento automatico. Se prima che si verificasse l'errore era attivo il modo di funzionamento ottimizzazione iniziale, ottimizzazione fine o misura del tempo di regolazione e ActivateRecoverMode = TRUE, PID_3Step passa al modo di funzionamento salvato in Mode.
0000_0002	Valore del parametro "Input_PER" non valido. Controllare l'eventuale presenza di un errore sull'ingresso analogico. Se prima che si verificasse l'errore era attivo il funzionamento automatico e ActivateRecoverMode = TRUE, PID_3Step passa nel modo di funzionamento "Approssima valore di uscita sostitutivo con controllo errori" o "Controllo errori". Non appena l'errore viene eliminato PID_3Step torna al funzionamento automatico. Se prima che si verificasse l'errore era attivo il modo di funzionamento ottimizzazione iniziale, ottimizzazione fine o misura del tempo di regolazione e ActivateRecoverMode = TRUE, PID_3Step passa al modo di funzionamento salvato in Mode.
0000_0004	Errore durante l'ottimizzazione fine. Non è stato possibile mantenere l'oscillazione del valore istantaneo. Se ActivateRecoverMode = TRUE prima che si verificasse l'errore, PID_3Step interrompe l'ottimizzazione e passa al modo di funzionamento salvato in Mode.
0000_0010	Il setpoint è stato modificato durante l'ottimizzazione. Nella variabile CancelTuningLevel è possibile impostare l'oscillazione ammessa del setpoint. Se ActivateRecoverMode = TRUE prima che si verificasse l'errore, PID_3Step interrompe l'ottimizzazione e passa al modo di funzionamento salvato in Mode.
0000_0020	L'ottimizzazione iniziale non è consentita durante l'ottimizzazione fine. Se ActivateRecoverMode = TRUE prima che si verificasse l'errore, PID_3Step resta nel modo di funzionamento ottimizzazione fine.
0000_0080	Errore nell'ottimizzazione iniziale. I limiti del valore di uscita non sono configurati correttamente e il valore istantaneo non reagisce come previsto. Verificare l'esistenza dei seguenti presupposti: - Che i limiti del valore di uscita siano configurati correttamente e siano compatibili con il senso di regolazione. - Che sia possibile una modifica del valore di uscita in modo che il valore istantaneo si avvicini al setpoint. Che per effetto del limite del valore di uscita corrispondente, il valore di uscita non venga limitato ancor prima dell'avvio dell'ottimizzazione iniziale, e che l'attuatore non abbia ancora raggiunto il riscontro. Esempio: nella direzione di regolazione normale, e con un valore istantaneo che si colloca al di sotto del setpoint, il valore di uscita non deve aver raggiunto il limite superiore, e l'attuatore non deve aver raggiunto il riscontro superiore ancor prima dell'avvio dell'ottimizzazione iniziale. - Che il valore istantaneo non presenti forti oscillazione prima dell'avvio dell'ottimizzazione iniziale. Se ActivateRecoverMode = TRUE prima che si verificasse l'errore, PID_3Step interrompe l'ottimizzazione e passa al modo di funzionamento salvato in Mode.

ErrorBits (DW#16#...)	Descrizione
0000_0100	Un errore durante l'ottimizzazione fine ha causato parametri non validi. Se ActivateRecoverMode = TRUE prima che si verificasse l'errore, PID_3Step interrompe l'ottimizzazione e passa al modo di funzionamento salvato in Mode.
0000_0200	Valore del parametro "Input" non valido: il formato numerico del valore non è valido. Se prima che si verificasse l'errore era attivo il funzionamento automatico e ActivateRecoverMode = TRUE, PID_3Step passa nel modo di funzionamento "Approssima valore di uscita sostitutivo con controllo errori" o "Controllo errori". Non appena l'errore viene eliminato PID_3Step torna al funzionamento automatico. Se prima che si verificasse l'errore era attivo il modo di funzionamento ottimizzazione iniziale, ottimizzazione fine o misura del tempo di regolazione e ActivateRecoverMode = TRUE, PID_3Step passa al modo di funzionamento salvato in Mode.
0000_0400	Calcolo del valore di uscita non riuscito. Controllare i parametri PID. Se prima che si verificasse l'errore era attivo il funzionamento automatico e ActivateRecoverMode = TRUE, PID_3Step passa nel modo di funzionamento "Approssima valore di uscita sostitutivo con controllo errori" o "Controllo errori". Non appena l'errore viene eliminato PID_3Step torna al funzionamento automatico. Se prima che si verificasse l'errore era attivo il modo di funzionamento ottimizzazione iniziale, ottimizzazione fine o misura del tempo di regolazione e ActivateRecoverMode = TRUE, PID_3Step passa al modo di funzionamento salvato in Mode.
0000_0800	Errore del tempo di campionamento: PID_3Step non viene richiamato entro il tempo di campionamento dell'OB di schedulazione orologio. Si raccomanda di richiamare PID_3Step in un OB di schedulazione orologio senza condizioni e di eseguire la relativa attivazione o disattivazione tramite il modo di funzionamento nel parametro Mode. Richiami condizionati o il richiamo nell'OB1 possono ripercuotersi negativamente sulla qualità della regolazione. La sorveglianza del tempo di campionamento può essere disattivata con CycleTime.EnMonitoring = FALSE. Se prima che si verificasse l'errore era attivo il funzionamento automatico e ActivateRecoverMode = TRUE, PID_3Step resta in funzionamento automatico. Se prima che si verificasse l'errore era attivo il modo di funzionamento ottimizzazione iniziale, ottimizzazione fine o misura del tempo di regolazione e ActivateRecoverMode = TRUE, PID_3Step passa al modo di funzionamento salvato in Mode. Se quest'errore si è verificato nella simulazione con PLCSIM, osservare le avvertenze riportate alla sezione Simulazione di PID_3Step V2 con PLCSIM (Pagina 145).
0000_1000	Valore del parametro "Setpoint" non valido: il formato numerico del valore non è valido. Se prima che si verificasse l'errore era attivo il funzionamento automatico e ActivateRecoverMode = TRUE, PID_3Step passa nel modo di funzionamento "Approssima valore di uscita sostitutivo con controllo errori" o "Controllo errori". Non appena l'errore viene eliminato PID_3Step torna al funzionamento automatico. Se prima che si verificasse l'errore era attivo il modo di funzionamento ottimizzazione iniziale, ottimizzazione fine o misura del tempo di regolazione e ActivateRecoverMode = TRUE, PID_3Step passa al modo di funzionamento salvato in Mode.
0000_2000	Valore del parametro Feedback_PER non valido. Controllare l'eventuale presenza di un errore sull'ingresso analogico. L'organo attuatore non può essere posizionato sul valore di uscita sostitutivo e resta fermo sulla posizione attuale. In funzionamento manuale è possibile modificare la posizione dell'organo attuatore solo con Manual_UP e Manual_DN ma non con ManualValue. Se prima che si verificasse l'errore era attivo il funzionamento automatico, ActivateRecoverMode = TRUE e l'errore non è più presente, PID_3Step torna al funzionamento automatico. Se prima che si verificasse l'errore era attivo il modo di funzionamento ottimizzazione iniziale, ottimizzazione fine o misura del tempo di regolazione e ActivateRecoverMode = TRUE, PID_3Step passa al modo di funzionamento salvato in Mode.

ErrorBits (DW#16#...)	Descrizione
0000_4000	Valore del parametro Feedback non valido. Il formato numerico del valore non è valido. L'organo attuatore non può essere posizionato sul valore di uscita sostitutivo e resta fermo sulla posizione attuale. In funzionamento manuale è possibile modificare la posizione dell'organo attuatore solo con Manual_UP e Manual_DN ma non con ManualValue. Se prima che si verificasse l'errore era attivo il funzionamento automatico, ActivateRecoverMode = TRUE e l'errore non è più presente, PID_3Step torna al funzionamento automatico. Se prima che si verificasse l'errore era attivo il modo di funzionamento ottimizzazione iniziale, ottimizzazione fine o misura del tempo di regolazione e ActivateRecoverMode = TRUE, PID_3Step passa al modo di funzionamento salvato in Mode.
0000_8000	Errore nella risposta di posizione digitale. Actuator_H = TRUE e Actuator_L = TRUE. L'organo attuatore non può essere posizionato sul valore di uscita sostitutivo e resta fermo sulla posizione attuale. Il funzionamento manuale non è possibile in questo stato. Per spostare l'organo attuatore da questo stato è necessario disattivare i "Segnali di riscontro organo attuatore" (Config.ActuatorEndStopOn = FALSE) o passare al modo di funzionamento manuale senza segnali di riscontro (Mode = 10). Se prima che si verificasse l'errore era attivo il funzionamento automatico, ActivateRecoverMode = TRUE e l'errore non è più presente, PID_3Step torna al funzionamento automatico. Se prima che si verificasse l'errore era attivo il modo di funzionamento ottimizzazione iniziale, ottimizzazione fine o misura del tempo di regolazione e ActivateRecoverMode = TRUE, PID_3Step passa al modo di funzionamento salvato in Mode.
0001_0000	Valore del parametro ManualValue non valido. Il formato numerico del valore non è valido. L'organo attuatore non può essere posizionato sul valore manuale e resta fermo sulla posizione attuale. Preimpostare un valore valido in ManualValue o spostare l'organo attuatore in funzionamento manuale con Manual_UP e Manual_DN.
0002_0000	Valore della variabile SavePosition non valido. Il formato numerico del valore non è valido. L'organo attuatore non può essere posizionato sul valore di uscita sostitutivo e resta fermo sulla posizione attuale.
0004_0000	Valore del parametro Disturbance non valido. Il formato numerico del valore non è valido. Se prima che si verificasse l'errore era attivo il funzionamento automatico e ActivateRecoverMode = TRUE Disturbance viene impostato a zero. PID_3Step resta in funzionamento automatico. Se prima che si verificasse l'errore era attivo il modo di funzionamento ottimizzazione iniziale o ottimizzazione fine e ActivateRecoverMode = TRUE, PID_3Step passa al modo di funzionamento salvato in Mode. Se Disturbance non ha alcuna influenza sul valore di uscita nella fase attuale, l'ottimizzazione non viene interrotta. Durante la misura del tempo di regolazione l'errore non influisce in alcun modo.

10.2.4.10 Variabile ActivateRecoverMode V2

La variabile ActivateRecoverMode determina il comportamento in caso di errori. Il parametro Error indica se attualmente è presente un errore. Se l'errore non è più presente, compare Error = FALSE. Il parametro ErrorBits mostra gli errori che si sono verificati.

ATTENZIONE
L'impianto può subire danneggiamenti. Se ActivateRecoverMode = TRUE, PID_3Step rimane in funzionamento automatico anche se vengono superati i limiti del valore istantaneo. In seguito a ciò l'impianto può subire danneggiamenti. Configurare per il sistema regolato un comportamento da adottare in caso di errore che protegga l'impianto da danneggiamenti.

Funzionamento automatico

ActivateRecover-Mode	Descrizione
FALSE	In caso di errore PID_3Step commuta nel modo di funzionamento "Inattivo" o "Approssimazione del valore di uscita sostitutivo". Il regolatore viene attivato solo in presenza di un fronte di discesa in Reset o di un fronte di salita in ModeActivate.
TRUE	Se si verificano errori frequenti in funzionamento automatico, questa impostazione peggiora il comportamento del regolatore perché PID_3Step commuta tra valore di uscita calcolato e valore di uscita sostitutivo a ogni errore. In questo caso controllare il parametro ErrorBits ed eliminare la causa dell'errore. Se si verificano uno o più dei seguenti errori PID_3Step resta in funzionamento automatico: • 0001h: Il parametro "Input" non rientra nei limiti del valore istantaneo. • 0800h: Errore del tempo di campionamento • 40000h: Valore del parametro Disturbance non valido. Se si verificano uno o più dei seguenti errori, PID_3Step commuta nel modo di funzionamento "Approssimazione del valore di uscita sostitutivo con controllo errori" o "Controllo errori": • 0002h: Valore del parametro Input_PER non valido. • 0200h: Valore del parametro Input non valido. • 0400h: Calcolo del valore di uscita non riuscito. • 1000h: Valore del parametro Setpoint non valido. Se si verificano uno o più dei seguenti errori, PID_3Step non può più spostare l'organo attuatore: • 2000h: Valore del parametro Feedback_PER non valido. • 4000h: Valore del parametro Feedback non valido. • 8000h: Errore nella risposta di posizione digitale. • 20000h: Valore della variabile SavePosition non valido. Il formato numerico del valore non è valido. Questo comportamento è indipendente da ErrorBehaviour. Non appena l'errore viene eliminato PID_3Step torna al funzionamento automatico.

Ottimizzazione iniziale, ottimizzazione fine e misura del tempo di regolazione

ActivateRecover-Mode	Descrizione
FALSE	In caso di errore PID_3Step commuta nel modo di funzionamento "Inattivo" o "Approssimazione del valore di uscita sostitutivo". Il regolatore viene attivato solo in presenza di un fronte di discesa in Reset o di un fronte di salita in ModeActivate. Dopo la misura del tempo di regolazione, il regolatore commuta sul modo di funzionamento "Inattivo".
TRUE	Se si verifica l'errore seguente PID_3Step resta nel modo di funzionamento attivo: • 0020h: L'ottimizzazione iniziale non è consentita durante l'ottimizzazione fine. I seguenti errori vengono ignorati: • 10000h: Valore del parametro ManualValue non valido. • 20000h: Valore della variabile SavePosition non valido. Con tutti gli altri errori PID_3Step interrompe l'ottimizzazione e passa al modo di funzionamento dal quale è stata avviata l'ottimizzazione.

Funzionamento manuale

In funzionamento manuale ActivateRecoverMode è inefficace.

Vedere anche[Variabili statiche di PID_3Step V2 \(Pagina 335\)](#)[Parametri State e Mode V2 \(Pagina 344\)](#)**10.2.4.11 Variabile Warning V2**

Se si presentano contemporaneamente più avvisi, i loro valori vengono visualizzati in un'addizione binaria. La visualizzazione dell'avviso 16#0000_0005 ad es. indica la presenza contemporanea degli avvisi 16#0000_0001 e 16#0000_0004.

Warning (DW#16#...)	Descrizione
0000_0000	Non è presente alcun avviso.
0000_0001	Durante l'ottimizzazione iniziale non è stato trovato il punto di flesso.
0000_0004	Il setpoint è stato circoscritto ai limiti impostati.
0000_0008	Per il metodo di calcolo selezionato non sono state definite tutte le proprietà necessarie per il sistema regolato. In cambio sono stati calcolati i parametri PID con il metodo TIR.TuneRule = 3.
0000_0010	Non è stato possibile commutare il modo di funzionamento perché Reset = TRUE o ManualEnable = TRUE.
0000_0020	Il tempo di campionamento dell'algoritmo PID viene circoscritto dal tempo di ciclo dell'OB richiamante. Per ottenere risultati migliori utilizzare tempi di ciclo dell'OB più brevi.
0000_0040	Il valore istantaneo ha superato uno dei suoi limiti di avviso.
0000_0080	Valore di Mode non valido. Il modo di funzionamento non viene commutato.
0000_0100	Il valore manuale è stato circoscritto ai limiti dell'uscita del regolatore.
0000_0200	La regola per l'ottimizzazione indicata non è supportata. Non vengono calcolati parametri PID.
0000_0400	Non è possibile misurare il tempo di regolazione perché le impostazioni dell'attuatore non corrispondono al metodo di misura selezionato.
0000_0800	Nella misura del tempo di regolazione la differenza tra la posizione attuale e il nuovo valore di uscita è troppo piccola. Ciò può produrre risultati errati. La differenza tra valore di uscita attuale e predefinito deve essere almeno pari al 50% dell'intero campo di regolazione.
0000_1000	Non è possibile raggiungere il valore di uscita sostitutivo perché non rientra nei limiti del valore di uscita.
0000_2000	L'organo attuatore è stato spostato in una direzione per più di Config.VirtualActuatorLimit x Retain.TransitTime. Controllare se l'organo attuatore ha raggiunto un segnale di riscontro.

I seguenti avvisi vengono cancellati non appena viene eliminata la causa:

- 16#0000_0001
- 16#0000_0004
- 16#0000_0008
- 16#0000_0040
- 16#0000_0100
- 16#0000_2000

Tutti gli altri avvisi vengono cancellati in presenza di un fronte di salita in Reset o ErrorAck.

10.2.5 PID_3Step V1

10.2.5.1 Descrizione PID_3Step V1

Descrizione

L'istruzione PID_3Step permette di configurare un regolatore PID con ottimizzazione automatica delle valvole o degli attuatori con comportamento di integrazione.

Sono disponibili i seguenti modi di funzionamento:

- Inattivo
- Ottimizzazione iniziale
- ottimizzazione fine
- Funzionamento automatico
- Funzionamento manuale
- Approssimazione valore di uscita sostitutivo
- Misura del tempo di regolazione
- Approssima valore di uscita sostitutivo con controllo errori
- Controllo errori

Una descrizione dettagliata dei modi di funzionamento è disponibile al paragrafo relativo al parametro State.

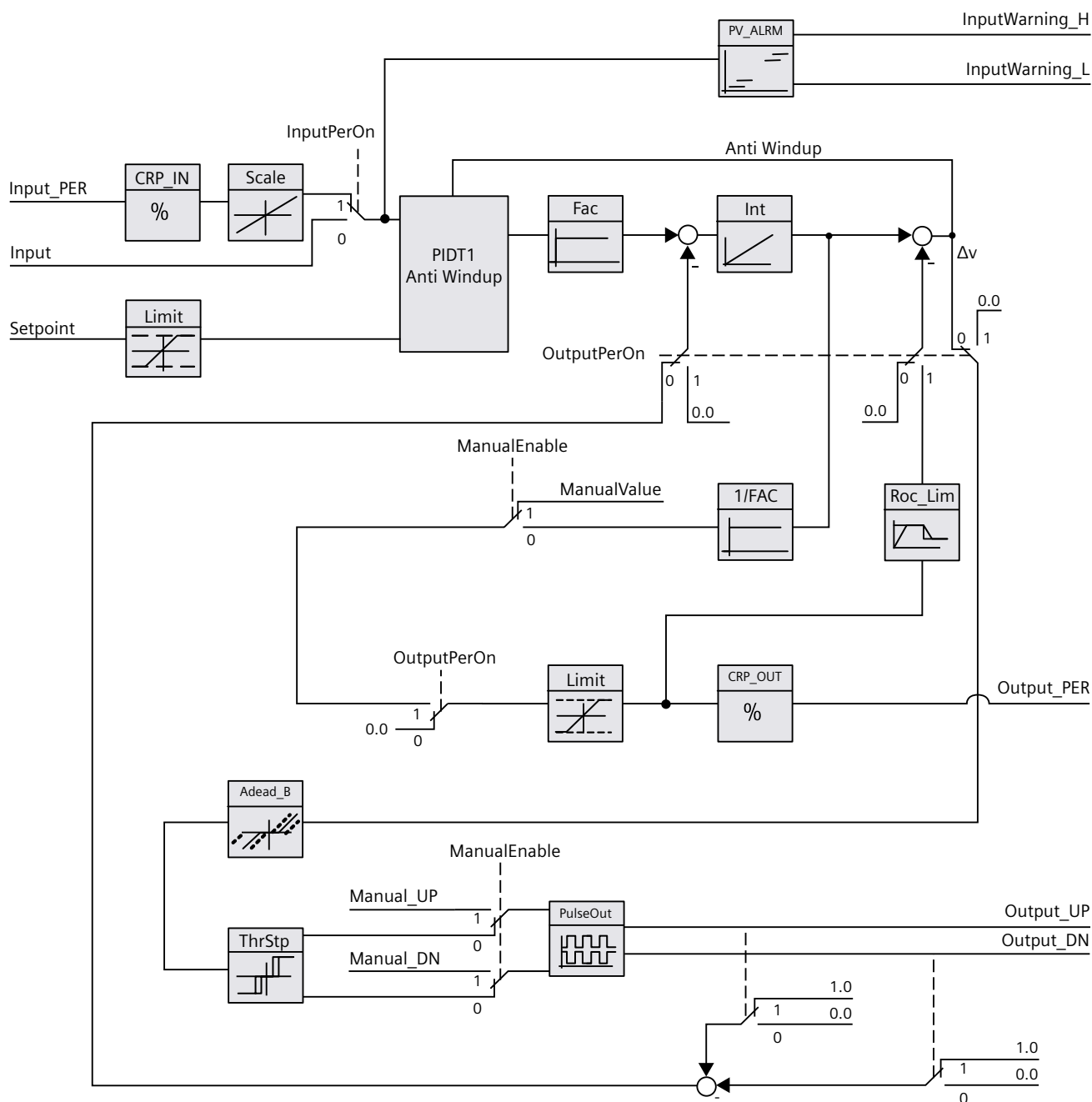
Algoritmo PID

PID_3Step è un regolatore PIDT1 con anti-windup e ponderazione del componente P e D. Il valore di uscita viene calcolato sulla base della seguente formula:

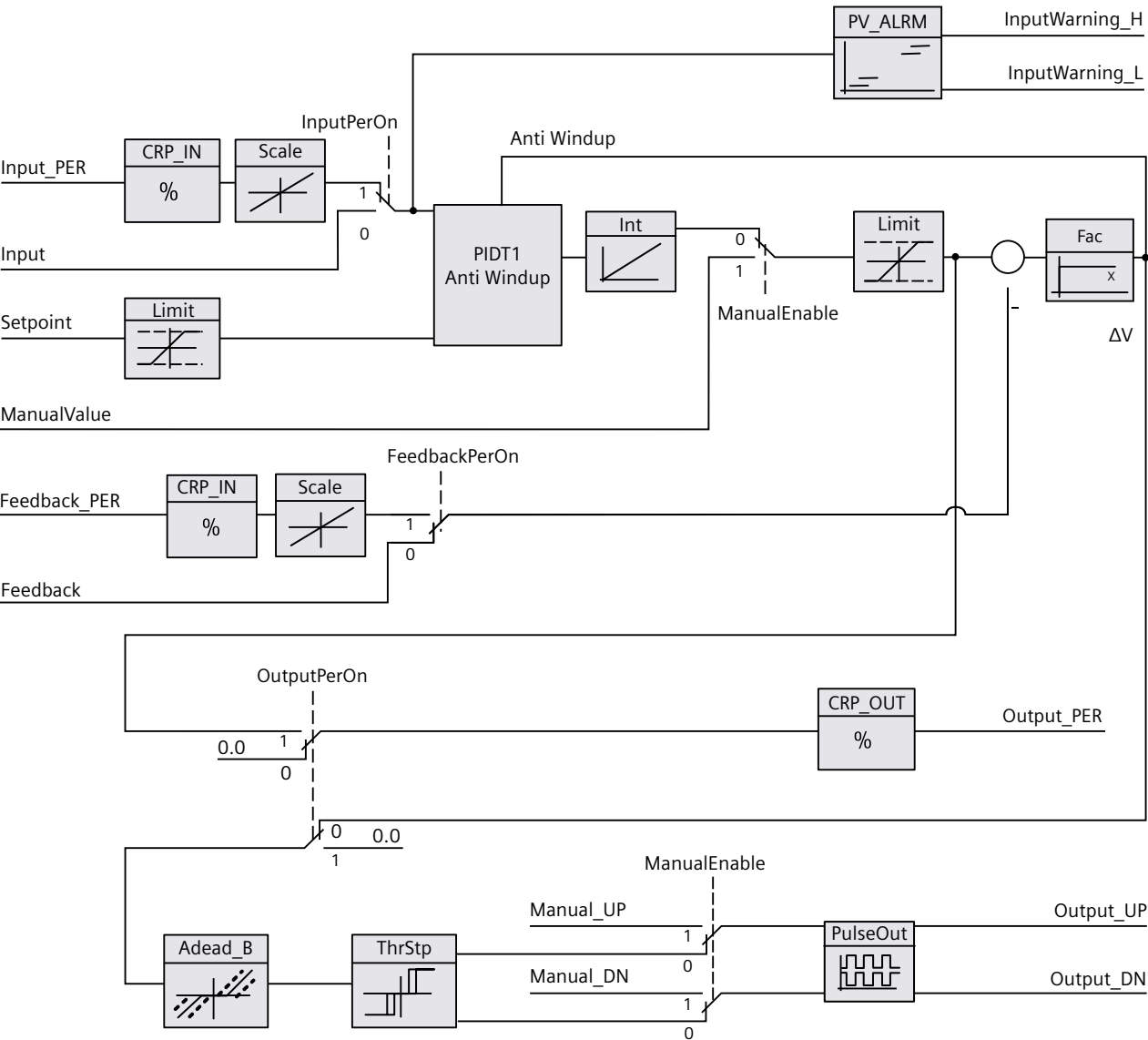
$$\Delta y = K_p \cdot s \cdot \left[(b \cdot w - x) + \frac{1}{T_I \cdot s} (w - x) + \frac{T_D \cdot s}{a \cdot T_D \cdot s + 1} (c \cdot w - x) \right]$$

Δy	Valore di uscita dell'algoritmo PID
K_p	Guadagno proporzionale
s	Operatore di Laplace
b	Ponderazione del componente P
w	Setpoint
x	Valore istantaneo
T_I	Tempo di integrazione
a	Coefficiente per il ritardo derivativo ($T_1 = a \times T_D$)
T_D	Tempo derivativo
c	Ponderazione del componente D

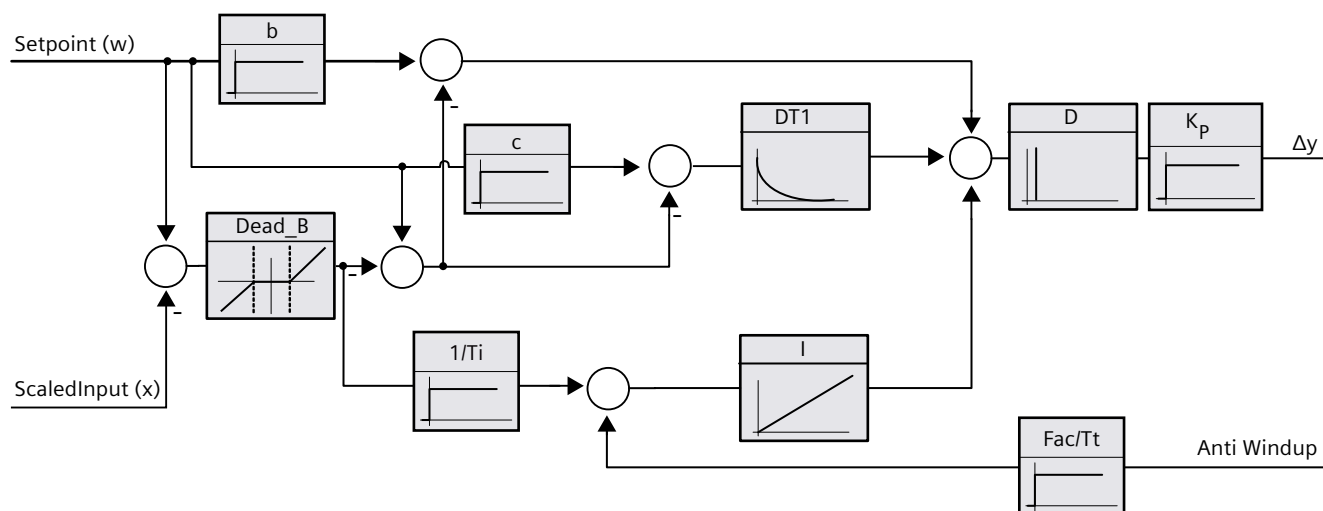
Schema a blocchi senza risposta di posizione



Schema a blocchi con risposta di posizione



Schema a blocchi di PIDT1 con anti-windup



Richiamo

L'istruzione PID_3Step viene richiamata ad intervalli costanti entro il tempo di ciclo dell'OB richiamante (di preferenza in un OB di schedulazione orologio).

Caricamento nel dispositivo

L'aggiornamento dei valori attuali delle variabili a ritenzione, avviene soltanto con il caricamento completo di PID_3Step.

Caricamento degli oggetti tecnologici del dispositivo [\(Pagina 48\)](#)

Avviamento

Quando si avvia la CPU, PID_3Step viene avviato nell'ultimo modo di funzionamento attivo. Per lasciare PID_3Step nel modo di funzionamento "Inattivo" impostare `RunModeByStartup = FALSE`.

Comportamento in caso di errori

Se si verificano degli errori, nel parametro Error viene emesso il rispettivo messaggio. Il comportamento di PID_3Step può essere configurato mediante le variabili ErrorBehaviour e ActivateRecoverMode.

ErrorBehaviour	ActivateRecoverMode	Configurazione impostazione organo attuatore Output viene impostato su	Comportamento
0	FALSE	Valore di uscita attuale	Commutazione nel modo di funzionamento "Inattivo" (Mode = 0)
0	TRUE	Valore di uscita attuale per la durata dell'errore	Commutazione nel modo di funzionamento "Controllo errori" (Mode = 7)
1	FALSE	Valore di uscita sostitutivo	Commutazione nel modo di funzionamento "Approssimazione del valore di uscita sostitutivo" (Mode = 5) Commutazione nel modo di funzionamento "Inattivo" (Mode = 0)
1	TRUE	Valore di uscita sostitutivo per la durata dell'errore	Commutazione nel modo di funzionamento "Approssimazione del valore di uscita sostitutivo con controllo errori" (Mode = 8) Commutazione nel modo di funzionamento "Controllo errori" (Mode = 7)

Il parametro ErrorBits mostra gli errori che si sono verificati.

Vedere anche

[Parametri State e Retain.Mode V1 \(Pagina 364\)](#)

[Parametro ErrorBits V1 \(Pagina 371\)](#)

[Configurazione di PID_3Step V1 \(Pagina 145\)](#)

10.2.5.2 Funzionamento PID_3Step V1

Sorveglianza dei limiti del valore istantaneo

Nelle variabili Config.InputUpperLimit e Config.InputLowerLimit si definisce il limite superiore e inferiore del valore istantaneo. Se il valore istantaneo non rientra in questi limiti si verifica un errore (ErrorBits = 0001hex).

Nelle variabili Config.InputUpperWarning e Config.InputLowerWarning si definisce il limite di avviso superiore e inferiore del valore istantaneo. Se il valore istantaneo non rientra in questi limiti viene emesso un avviso (Warnings = 0040hex) e il parametro di uscita InputWarning_H o InputWarning_L diventa TRUE.

Definizione dei limiti per il setpoint

Nelle variabili Config.SetpointUpperLimit e Config.SetpointLowerLimit si definisce un limite superiore e inferiore per il setpoint. PID_3Step circoscrive automaticamente il setpoint ai limiti del valore istantaneo. È inoltre possibile delimitare il setpoint entro un campo più piccolo. PID_3Step verifica se il campo è compreso entro i limiti del valore istantaneo. Se il setpoint non rientra in questi limiti, il limite superiore o inferiore viene utilizzato come setpoint e il parametro di uscita SetpointLimit_H o SetpointLimit_L diventa TRUE.

Il setpoint viene circoscritto in tutti i modi operativi.

Limitazione del valore di uscita

Nelle variabili Config.OutputUpperLimit e Config.OutputLowerLimit si definisce il limite superiore e inferiore del valore di uscita. I limiti del valore di uscita devono essere compresi tra il "Riscontro inferiore" e il "Riscontro superiore".

- Riscontro superiore: Config.FeedbackScaling.UpperPointOut
- Riscontro inferiore: Config.FeedbackScaling.LowerPointOut

Deve valere:

$\text{UpperPointOut} \geq \text{OutputUpperLimit} > \text{OutputLowerLimit} \geq \text{LowerPointOut}$

I valori validi per "Riscontro superiore" e "Riscontro inferiore" dipendono da:

- FeedbackOn
- FeedbackPerOn
- OutputPerOn

OutputPerOn	FeedbackOn	FeedbackPerOn	LowerPointOut	UpperPointOut
FALSE	FALSE	FALSE	non impostabile (0.0 %)	non impostabile (100.0 %)
FALSE	TRUE	FALSE	-100.0 % oppure 0.0 %	0.0 % oppure +100.0 %
FALSE	TRUE	TRUE	-100.0 % oppure 0.0 %	0.0 % oppure +100.0 %

OutputPerOn	FeedbackOn	FeedbackPerOn	LowerPointOut	UpperPointOut
TRUE	FALSE	FALSE	non impostabile (100.0 %)	non impostabile (100.0 %)
TRUE	TRUE	FALSE	-100.0 % oppure 0.0 %	0.0 % oppure +100.0 %
TRUE	TRUE	TRUE	-100.0 % oppure 0.0 %	0.0 % oppure +100.0 %

Se OutputPerOn = FALSE e FeedbackOn = FALSE non è possibile delimitare il valore di uscita. Le uscite digitali vengono resettate se Actuator_H = TRUE, se Actuator_L = TRUE oppure dopo un tempo di spostamento del 110% del tempo di regolazione del motore.

Il valore di uscita corrisponde a 27648 al 100 % e a -27648 a -100 %. PID_3Step deve essere in grado di chiudere completamente la valvola. Perciò lo zero deve essere compreso nei limiti del valore di uscita.

NOTA

Impiego con due o più attuatori

PID_3 Step non si addice all'impiego con due o più attuatori (ad es. in applicazioni di riscaldamento/raffreddamento), in quanto per la realizzazione di un buon comportamento di regolazione, attuatori diversi richiedono anche parametri PID diversi.

Valore di uscita sostitutivo

In caso di errore PID_3Step può emettere un valore di uscita sostitutivo e spostare l'organo attuatore in una posizione sicura che viene definita nella variabile SavePosition. Il valore di uscita sostitutivo deve essere compreso entro i limiti del valore di uscita.

Controllo della validità dei segnali

Viene controllata la validità dei valori dei parametri seguenti:

- Setpoint
- Input
- Input_PER
- Feedback
- Feedback_PER
- Output

Sorveglianza del tempo di campionamento PID_3Step

Il tempo di campionamento dovrebbe corrispondere al tempo di ciclo dell'OB richiamaente. L'istruzione PID_3Step misura di volta in volta l'intervallo tra due richiami. Questo è il tempo di campionamento attuale. Ad ogni commutazione del modo di funzionamento e al primo avvio viene generato il valore medio dei primi 10 tempi di campionamento. Se il tempo di campionamento attuale si discosta troppo da questo valore medio si verifica un errore (ErrorBits = 0800 hex).

La commutazione di PID_3Step nel modo di funzionamento "Inattivo" durante l'ottimizzazione avviene alle seguenti condizioni:

- Nuovo valore medio $\geq 1,1 \times$ valore medio precedente
- Nuovo valore medio $\leq 0,9 \times$ valore medio precedente

La commutazione di PID_3Step nel modo di funzionamento "Inattivo" durante il funzionamento automatico avviene alle seguenti condizioni:

- Nuovo valore medio $\geq 1,5 \times$ valore medio precedente
- Nuovo valore medio $\leq 0,5 \times$ valore medio precedente

Tempo di campionamento dell'algoritmo PID

Poiché il sistema regolato richiede un certo tempo per reagire a una modifica del valore di uscita, è utile non calcolare questo valore in ogni ciclo. Il tempo di campionamento dell'algoritmo PID è il tempo che trascorre tra due calcoli del valore di uscita. Viene determinato durante l'ottimizzazione e arrotondato ad un multiplo del tempo di ciclo. Tutte le altre funzioni di PID_3Step vengono eseguite ad ogni richiamo.

Misura del tempo di regolazione motore

Il tempo di regolazione del motore è il tempo in secondi necessario al motore per spostare l'organo attuatore dallo stato chiuso a quello aperto. L'organo attuatore viene mosso in una direzione, al massimo per il 110% del tempo di regolazione del motore. Per ottenere un buon risultato di regolazione, PID_3Step richiede un tempo di regolazione del motore il più preciso possibile. I dati contenuti nella documentazione dell'organo attuatore sono valori medi per questo tipo di organi attuatori. Il valore può variare per l'organo attuatore effettivamente utilizzato. Il tempo di regolazione del motore può essere misurato durante la messa in servizio. Durante la misura del tempo di regolazione del motore, i limiti del valore di uscita non vengono considerati. L'organo attuatore può essere traslato fino al riscontro superiore o fino a quello inferiore.

Il tempo di regolazione del motore viene considerato sia nel calcolo del valore di uscita analogico, sia nel calcolo dei valori di uscita digitali. Questo tempo è necessario soprattutto nell'ottimizzazione automatica e nel comportamento anti-windup per un corretto funzionamento. Configurare pertanto il tempo di regolazione con un valore necessario al motore per spostare l'attuatore dallo stato chiuso a quello aperto.

Se nel processo non è attivo un tempo di regolazione del motore rilevante (ad es. con le valvole magnetiche) in modo da consentire al valore di uscita di agire direttamente e pienamente nel processo, utilizzare in suo luogo PID_Compact.

Senso di regolazione

Aumentando il valore di uscita si dovrebbe ottenere un aumento del valore istantaneo. In questo caso si parla di un senso di regolazione normale. Per raffreddamenti e regolazioni di flusso può essere necessario invertire il senso di regolazione. PID_3Step non opera con guadagno proporzionale negativo. Se InvertControl = TRUE, un aumento della differenza di regolazione determina una riduzione del valore di uscita. Il senso di regolazione viene preso in considerazione anche durante l'ottimizzazione iniziale e l'ottimizzazione fine.

Vedere anche

[Configurazione di PID_3Step V1 \(Pagina 145\)](#)

10.2.5.3 Parametro di ingresso PID_3Step V1

Tabella 10-10

Parametro	Tipo di dati	Preassegnazione	Descrizione
Setpoint	REAL	0.0	Setpoint del regolatore PID nel funzionamento automatico
Input	REAL	0.0	Una variabile del programma utente viene utilizzata come sorgente del valore istantaneo. Per poter utilizzare il parametro Input è necessario impostare Config.InputPerOn = FALSE.
Input_PER	WORD	W#16#0	Un ingresso analogico viene utilizzato come sorgente del valore istantaneo. Per poter utilizzare il parametro Input_PER è necessario impostare Config.InputPerOn = TRUE.
Actuator_H	BOOL	FALSE	Risposta di posizione digitale della valvola per il riscontro superiore Se Actuator_H = TRUE la valvola è posizionata sul riscontro superiore e non viene più mossa in questa direzione.
Actuator_L	BOOL	FALSE	Risposta di posizione digitale della valvola per il riscontro inferiore Se Actuator_L = TRUE la valvola è posizionata sul riscontro inferiore e non viene più mossa in questa direzione.
Feedback	REAL	0.0	Risposta di posizione della valvola Per poter utilizzare il parametro Feedback è necessaria l'impostazione Config.FeedbackPerOn = FALSE.
Feedback_PER	WORD	W#16#0	Risposta di posizione analogica di una valvola Per poter utilizzare il parametro Feedback_PER è necessaria l'impostazione Config.FeedbackPerOn = TRUE. Feedback_PER viene riportato in scala sulla base delle variabili: • Config.FeedbackScaling.LowerPointIn • Config.FeedbackScaling.UpperPointIn • Config.FeedbackScaling.LowerPointOut • Config.FeedbackScaling.UpperPointOut

Parametro	Tipo di dati	Preassegnazione	Descrizione
ManualEnable	BOOL	FALSE	- Il fronte FALSE -> TRUE seleziona il modo di funzionamento "Funzionamento manuale", State = 4, Retain.Mode rimane invariato. - Il fronte TRUE -> FALSE seleziona l'ultimo modo di funzionamento attivo. Se ManualEnable = TRUE un'eventuale modifica di Retain.Mode non ha alcuna conseguenza. Solo con il fronte TRUE -> FALSE in ManualEnable la modifica di Retain.Mode viene considerata. PID_3Step V1.1 Se all'avvio della CPU ManualEnable = TRUE, PID_3Step si avvia in funzionamento manuale. Non è necessario un fronte di salita (FALSE > TRUE) in ManualEnable. PID_3Step V1.0 All'avvio della CPU PID_3Step commuta nel funzionamento manuale solo in presenza di un fronte di salita (FALSE->TRUE) in ManualEnable. Senza fronte di salita PID_3Step si avvia nell'ultimo modo di funzionamento nel quale ManualEnable era FALSE.
ManualValue	REAL	0.0	Nel funzionamento manuale viene predefinita la posizione assoluta della valvola. ManualValue viene analizzato solo se si utilizza OutputPer oppure se è disponibile una risposta di posizione.
Manual_UP	BOOL	FALSE	Nel funzionamento manuale ogni fronte di salita apre la valvola del 5% del campo di regolazione complessivo oppure per il tempo minimo di regolazione del motore. Manual_UP viene analizzato solo se non si utilizza Output_PER e non è disponibile una risposta di posizione.
Manual_DN	BOOL	FALSE	Nel funzionamento manuale ogni fronte di salita chiude la valvola del 5% del campo di regolazione complessivo oppure per il tempo minimo di regolazione del motore. Manual_DN viene analizzato solo se non si utilizza Output_PER e non è disponibile una risposta di posizione.
Reset	BOOL	FALSE	Esegue un nuovo avvio del regolatore. - Fronte FALSE -> TRUE - Commutazione nel modo di funzionamento "Inattivo" - ErrorBits e Warning vengono resettati - I valori provvisori della regolazione vengono resettati (i parametri PID vengono mantenuti) - Fronte TRUE -> FALSE - Commutazione nell'ultimo modo di funzionamento attivo - Se era in precedenza attivato il funzionamento automatico, la commutazione nel funzionamento automatico avviene in modo continuo.

10.2.5.4 Parametro di uscita PID_3Step V1

Tabella 10-11

Parametro	Tipo di dati	Preassegnazione	Descrizione
ScaledInput	REAL	0.0	Valore istantaneo riportato in scala
ScaledFeedback	REAL	0.0	Risposta di posizione riportata in scala Nel caso di un organo attuatore senza risposta di posizione ScaledFeedback mostra la posizione in modo molto impreciso. ScaledFeedback si può utilizzare solo per una valutazione approssimativa della posizione attuale.
Output_UP	BOOL	FALSE	Valore di uscita digitale per l'apertura della valvola Se Config.OutputPerOn = FALSE viene utilizzato il parametro Output_UP.
Output_DN	BOOL	FALSE	Valore di uscita digitale per la chiusura della valvola Se Config.OutputPerOn = FALSE viene utilizzato il parametro Output_DN.
Output_PER	WORD	W#16#0	Valore di uscita analogico Se Config.OutputPerOn = TRUE viene utilizzato Output_PER. Impiegare Output_PER, se come attuatore viene utilizzata una valvola accessibile da un'uscita analogica e comandata da un segnale continuo, ad es. 0...10V o 4...20mA. Il valore su Output_PER corrisponde alla destinazione della valvola, ad es. Output_PER = 13824, se la valvola deve essere aperta del 50%.
SetpointLimit_H	BOOL	FALSE	Se SetpointLimit_H = TRUE è stato raggiunto il limite superiore assoluto del setpoint. Nella CPU il setpoint viene circoscritto al limite superiore assoluto configurato per lo stesso. Come limite superiore del setpoint viene preassegnato il limite superiore assoluto configurato per il valore istantaneo. Se si configura Config.SetpointUpperLimit su un valore compreso nei limiti del valore istantaneo questo valore viene utilizzato come limite superiore del setpoint.
SetpointLimit_L	BOOL	FALSE	Se SetpointLimit_L = TRUE è stato raggiunto il limite inferiore assoluto del setpoint. Nella CPU il setpoint viene circoscritto al limite inferiore assoluto configurato. Come limite inferiore del setpoint è preassegnato il limite inferiore assoluto configurato per il valore istantaneo. Se si configura Config.SetpointLowerLimit su un valore compreso nei limiti del valore istantaneo questo valore viene utilizzato come limite inferiore del setpoint.
InputWarning_H	BOOL	FALSE	Se InputWarning_H = TRUE il limite superiore di avviso del valore istantaneo è stato raggiunto o superato.
InputWarning_L	BOOL	FALSE	Se InputWarning_L = TRUE il limite inferiore di avviso del valore istantaneo è stato raggiunto o superato per difetto.
State	INT	0	Il parametro State (Pagina 364) indica il modo di funzionamento attuale del regolatore PID. Per commutare il modo di funzionamento si utilizza la variabile Retain.Mode. • State = 0: Inattivo • State = 1: Ottimizzazione iniziale • State = 2: Ottimizzazione fine • State = 3: Funzionamento automatico • State = 4: Funzionamento manuale

Parametro	Tipo di dati	Preassegnazione	Descrizione
			- State = 5: Approssimazione valore di uscita sostitutivo - State = 6: Misura del tempo di regolazione - State = 7: Controllo errori - State = 8: Approssima valore di uscita sostitutivo con controllo errori
Error	BOOL	FALSE	Se Error = TRUE è presente almeno un messaggio di errore.
ErrorBits	DWORD	DW#16#0	Il parametro ErrorBits (Pagina 371) mostra i messaggi di errore.

Vedere anche

[Parametri State e Retain.Mode V1 \(Pagina 364\)](#)

[Parametro ErrorBits V1 \(Pagina 371\)](#)

10.2.5.5 Variabile statica PID_3Step V1

NOTA

Le variabili non specificate non devono essere modificate, perché vengono utilizzate solo internamente.

Per evitare un comportamento errato del regolatore PID modificare le variabili contrassegnate con ⁽¹⁾ solo nel modo di funzionamento "Inattivo". Il modo di funzionamento "Inattivo" può essere attivato forzando la variabile "Retain.Mode" sul valore "0".

Tabella 10-12

Variabile	Tipo di dati	Preassegnazione	Descrizione
ActivateRecoverMode	BOOL	TRUE	La variabile ActivateRecoverMode (Pagina 373) determina il comportamento in caso di errori.
RunModeByStartup	BOOL	TRUE	Dopo il nuovo avviamento della CPU attivare l'ultimo modo di funzionamento Se RunModeByStartup = TRUE, dopo l'avvio della CPU il regolatore ritorna all'ultimo modo di funzionamento attivo. Se RunModeByStartup = FALSE, dopo l'avvio della CPU il regolatore rimane inattivo.
PhysicalUnit	INT	0	Unità fisica del valore istantaneo e del setpoint, ad es. °C o °F.
PhysicalQuantity	INT	0	Grandezza fisica del valore istantaneo e del setpoint, ad es. temperatura.
ErrorBehaviour	INT	0	Se ErrorBehaviour = 0, in caso di errore la valvola rimane nella posizione attuale e il regolatore passa direttamente al modo di funzionamento "Inattivo" o "Controllo errori". Se ErrorBehaviour = 1, in caso di errore l'organo attuatore viene spostato sul valore di uscita sostitutivo e solo successivamente si passa al modo di funzionamento "Inattivo" o "Controllo errori". Se si verificano gli errori seguenti non è più possibile portare la valvola a un valore di uscita sostitutivo. 2000h: Valore del parametro Feedback_PER non valido. 4000h: Valore del parametro Feedback non valido. 8000h: Errore nella risposta di posizione digitale.

Variabile	Tipo di dati	Preassegnazione	Descrizione
Warning	DWORD	DW#16#0	La variabile Warning (Pagina 364) mostra gli avvisi dal reset o dall'ultima commutazione del modo di funzionamento. Gli avvisi ciclici (ad es. avvisi del valore istantaneo) vengono visualizzati finché è presente la rispettiva causa. Una volta eliminata la causa gli avvisi vengono cancellati automaticamente. Gli avvisi non ciclici (ad es. punto di flesso non trovato) rimangono attivi e vengono cancellati come errori.
SavePosition	REAL	0.0	Valore di uscita sostitutivo Se ErrorBehaviour = 1 in caso di errore l'organo attuatore viene spostato in una posizione sicura per l'impianto e solo successivamente si passa al modo di funzionamento "Inattivo". Il campo dei valori ammesso dipende dalla configurazione. • Config.FeedbackOn = FALSE e Config.OutputPerOn = FALSE: SavePosition = 0.0 oppure 100.0 • Config.FeedbackOn = TRUE o Config.OutputPerOn = TRUE: Config.OutputUpperLimit ≥ SavePosition ≥ Config.OutputLowerLimit
CurrentSetpoint	REAL	0.0	Setpoint attualmente attivo. Questo valore viene congelato all'avvio dell'ottimizzazione.
Progress	REAL	0.0	Avanzamento dell'ottimizzazione in percentuale (0,0 - 100,0)
Config.InputPerOn ⁽¹⁾	BOOL	TRUE	Se InputPerOn = TRUE viene utilizzato il parametro Input_PER. Se InputPerOn = FALSE viene utilizzato il parametro Input.
Config.OutputPerOn ⁽¹⁾	BOOL	FALSE	Se OutputPerOn = TRUE viene utilizzato il parametro Output_PER. Se OutputPerOn = FALSE vengono utilizzati i parametri Output_UP e Output_DN.
Config.LoadBackUp	BOOL	FALSE	Se LoadBackUp = TRUE viene ricaricato l'ultimo set di parametri PID. Il set è stato memorizzato prima dell'ultima ottimizzazione. LoadBackUp viene reimpostato automaticamente a FALSE.
Config.InvertControl ⁽¹⁾	BOOL	FALSE	Inversione del senso di regolazione Se InvertControl = TRUE, un aumento della differenza di regolazione determina una riduzione del valore di uscita.
Config.FeedbackOn ⁽¹⁾	BOOL	FALSE	Se FeedbackOn = FALSE viene simulata una risposta di posizione. Se FeedbackOn = TRUE viene in genere attivata la risposta di posizione.
Config.FeedbackPerOn ⁽¹⁾	BOOL	FALSE	FeedbackPerOn è valido solo se FeedbackOn = TRUE. Se FeedbackPerOn = TRUE viene utilizzato l'ingresso analogico per la risposta di posizione (parametro Feedback_PER). Se FeedbackPerOn = FALSE viene utilizzato il parametro Feedback per la risposta di posizione.
Config.ActuatorEndStopOn ⁽¹⁾	BOOL	FALSE	Se ActuatorEndStopOn = TRUE viene tenuta in considerazione la risposta di posizione digitale Actuator_L e Actuator_H.
Config.InputUpperLimit ⁽¹⁾	REAL	120.0	Limite superiore del valore istantaneo Nell'ingresso di periferia il valore istantaneo può superare al massimo del 18% il campo normale (campo di sovracomando). Non vengono più segnalati errori di superamento del "Limite superiore valore istantaneo". Vengono riconosciuti solo la rottura conduttore e il cortocircuito e PID_3Step si comporta come in caso di errore. InputUpperLimit > InputLowerLimit
Config.InputLowerLimit ⁽¹⁾	REAL	0.0	Limite inferiore del valore istantaneo InputLowerLimit < InputUpperLimit

Variabile	Tipo di dati	Preassegnazione	Descrizione
Config.InputUpperWarning ⁽¹⁾	REAL	+3.40282-2e+38	Limite superiore di avviso del valore istantaneo Se si configura InputUpperWarning al di fuori dei limiti del valore istantaneo, viene utilizzato il limite superiore assoluto configurato per il valore istantaneo come limite superiore di avviso. Se si configura InputUpperWarning entro i limiti del valore istantaneo questo valore viene utilizzato come limite superiore di avviso. $\text{InputUpperWarning} > \text{InputLowerWarning}$ $\text{InputUpperWarning} \leq \text{InputUpperLimit}$
Config.InputLowerWarning ⁽¹⁾	REAL	-3.402822e+38	Limite inferiore di avviso del valore istantaneo Se si configura InputLowerWarning al di fuori dei limiti del valore istantaneo, il limite inferiore assoluto configurato per il valore istantaneo viene utilizzato come limite inferiore di avviso. Se si configura InputLowerWarning entro i limiti del valore istantaneo questo valore viene utilizzato come limite inferiore di avviso. $\text{InputLowerWarning} < \text{InputUpperWarning}$ $\text{InputLowerWarning} \geq \text{InputLowerLimit}$
Config.OutputUpperLimit ⁽¹⁾	REAL	100.0	Limite superiore del valore di uscita È consentito il campo di valori seguente: $\text{UpperPointOut} \geq \text{OutputUpperLimit} > \text{OutputLowerLimit}$ Per maggiori dettagli vedere OutputLowerLimit
Config.OutputLowerLimit ⁽¹⁾	REAL	0.0	Limite inferiore del valore di uscita È consentito il campo di valori seguente: $\text{OutputUpperLimit} > \text{OutputLowerLimit} \geq \text{LowerPointOut}$ Se si utilizza Output_PER un limite del valore di uscita di -100% corrisponde al valore Output_PER = -27648; il 100% corrisponde al valore Output_PER = 27648. Se OutputPerOn = FALSE e FeedbackOn = FALSE, OutputLowerLimit e OutputUpperLimit non vengono analizzati. Output_UP e Output_DN vengono resettati se Actuator_H = TRUE o Actuator_L = TRUE (se ActuatorEndStopOn = TRUE) o dopo una durata corsa di $110\% \cdot \text{Config.TransitTime}$ (se ActuatorEndStopOn = FALSE).
Config.SetpointUpperLimit ⁽¹⁾	REAL	+3.40282-2e+38	Limite superiore del setpoint Se si configura SetpointUpperLimit al di fuori dei limiti del valore istantaneo, il limite superiore assoluto configurato per il valore istantaneo viene preassegnato come limite superiore del setpoint. Se si configura SetpointUpperLimit entro i limiti del valore istantaneo questo valore viene utilizzato come limite superiore del setpoint.
Config.SetpointLowerLimit ⁽¹⁾	REAL	-3.402822e+38	Limite inferiore del setpoint Se si configura SetpointLowerLimit al di fuori dei limiti del valore istantaneo, il limite inferiore assoluto configurato per il valore istantaneo viene preassegnato come limite inferiore del setpoint. Se si configura SetpointLowerLimit entro i limiti del valore istantaneo questo valore viene utilizzato come limite inferiore del setpoint.
Config.MinimumOnTime ⁽¹⁾	REAL	0.0	Tempo minimo di attivazione Tempo minimo di attivazione dell'attuatore espresso in secondi. Config.MinimumOnTime è attivo soltanto se Output_UP e Output_DN vengono utilizzati (Config.OutputPerOn = FALSE).
Config.MinimumOffTime ⁽¹⁾	REAL	0.0	Tempo minimo di disattivazione Tempo minimo di disattivazione dell'attuatore espresso in secondi. Config.MinimumOffTime è attivo soltanto se Output_UP e Output_DN vengono utilizzati (Config.OutputPerOn = FALSE).

Variabile	Tipo di dati	Preassegnazione	Descrizione
Config.TransitTime ⁽¹⁾	REAL	30.0	Tempo di regolazione motore Tempo in secondi necessario all'attuatore per spostare la valvola dallo stato chiuso a quello aperto.
Config.InputScaling.UpperPointIn ⁽¹⁾	REAL	27648.0	Scala Input_PER superiore Sulla base delle due coppie di valori UpperPointOut, UpperPointIn e LowerPointOut, LowerPointIn della struttura InputScaling, Input_PER viene convertito in percentuale.
Config.InputScaling.LowerPointIn ⁽¹⁾	REAL	0.0	Scala Input_PER inferiore Sulla base delle due coppie di valori UpperPointOut, UpperPointIn e LowerPointOut, LowerPointIn della struttura InputScaling, Input_PER viene convertito in percentuale.
Config.InputScaling.UpperPointOut ⁽¹⁾	REAL	100.0	Valore istantaneo superiore riportato in scala Sulla base delle due coppie di valori UpperPointOut, UpperPointIn e LowerPointOut, LowerPointIn della struttura InputScaling, Input_PER viene convertito in percentuale.
Config.InputScaling.LowerPointOut ⁽¹⁾	REAL	0.0	Valore istantaneo inferiore riportato in scala Sulla base delle due coppie di valori UpperPointOut, UpperPointIn e LowerPointOut, LowerPointIn della struttura InputScaling, Input_PER viene convertito in percentuale.
Config.FeedbackScaling.UpperPointIn ⁽¹⁾	REAL	27648.0	Scala Feedback_PER superiore Sulla base delle due coppie di valori UpperPointOut, UpperPointIn e LowerPointOut, LowerPointIn della struttura FeedbackScaling, Feedback_PER viene convertito in percentuale.
Config.FeedbackScaling.LowerPointIn ⁽¹⁾	REAL	0.0	Scala Feedback_PER inferiore Sulla base delle due coppie di valori UpperPointOut, UpperPointIn e LowerPointOut, LowerPointIn della struttura FeedbackScaling, Feedback_PER viene convertito in percentuale.
Config.FeedbackScaling.UpperPointOut ⁽¹⁾	REAL	100.0	Riscontro superiore Sulla base delle due coppie di valori UpperPointOut, UpperPointIn e LowerPointOut, LowerPointIn della struttura FeedbackScaling, Feedback_PER viene convertito in percentuale. Il campo dei valori ammesso dipende dalla configurazione. - FeedbackOn = FALSE: UpperPointOut = 100.0 - FeedbackOn = TRUE: UpperPointOut = 100.0 o 0.0 UpperPointOut ≠ LowerPointOut
Config.FeedbackScaling.LowerPointOut ⁽¹⁾	REAL	0.0	Riscontro inferiore Sulla base delle due coppie di valori UpperPointOut, UpperPointIn e LowerPointOut, LowerPointIn della struttura FeedbackScaling, Feedback_PER viene convertito in percentuale. Il campo dei valori ammesso dipende dalla configurazione. - FeedbackOn = FALSE: LowerPointOut = 0.0 - FeedbackOn = TRUE: LowerPointOut = 0.0 o -100.0 LowerPointOut ≠ UpperPointOut
GetTransitTime.InvertDirection	BOOL	FALSE	Se InvertDirection = FALSE per il rilevamento del tempo di regolazione la valvola viene aperta completamente, chiusa e infine riaperta. Se InvertDirection = TRUE, la valvola viene chiusa completamente, aperta e infine richiusa.

Variabile	Tipo di dati	Preassegnazione	Descrizione
GetTransitTime.SelectFeedback	BOOL	FALSE	Se SelectFeedback = TRUE, nella misura del tempo di regolazione viene considerato Feedback_PER o Feedback. Se SelectFeedback = FALSE, nella misura del tempo di regolazione vengono considerati Actuator_H e Actuator_L.
GetTransitTime.Start	BOOL	FALSE	Se Start = TRUE la misura del tempo di regolazione viene avviata.
GetTransitTime.State	INT	0	Fase attuale della misura del tempo di regolazione - State = 0: Inattivo - State = 1: Apertura completa della valvola - State = 2: Chiusura completa della valvola - State = 3: Impostazione della valvola sulla destinazione (NewOutput) - State = 4: Misurazione del tempo di regolazione terminata correttamente - State = 5: Misurazione del tempo di regolazione interrotta
GetTransitTime.NewOutput	REAL	0.0	Destinazione per la misura del tempo di regolazione con risposta di posizione La destinazione deve essere compresa tra il "Riscontro superiore" e il "Riscontro inferiore". La differenza tra NewOutput e ScaledFeedback deve essere pari almeno al 50 % del campo di regolazione consentito.
CycleTime.StartEstimation	BOOL	TRUE	Se StartEstimation = TRUE viene avviata la misura del tempo di campionamento PID_3Step. Al termine dell'operazione di misura risulta CycleTime.StartEstimation = FALSE.
CycleTime.EnEstimation	BOOL	TRUE	Se EnEstimation = TRUE viene calcolato il tempo di campionamento PID_3Step.
CycleTime.EnMonitoring	BOOL	TRUE	Se EnMonitoring = TRUE viene sorvegliato il tempo di campionamento PID_3Step. Se non è possibile eseguire PID_3Step entro il tempo di campionamento, viene emesso l'errore 0800h e si passa a un altro modo di funzionamento. Il modo di funzionamento a cui il regolatore passa dipende da ActivateRecoverMode e ErrorBehaviour. Se EnMonitoring = FALSE, il tempo di campionamento PID_3Step non viene sorvegliato, l'errore 0800h non viene emesso e non si passa ad un altro modo di funzionamento.
CycleTime.Value ⁽¹⁾	REAL	0.1	Tempo di campionamento PID_3Step in secondi CycleTime.Value viene rilevata automaticamente e corrisponde in genere al tempo di ciclo dell'OB richiamante.
CtrlParamsBackUp.SetByUser	BOOL	FALSE	Valore memorizzato di Retain.CtrlParams.SetByUser I valori della struttura CtrlParamsBackUp possono essere ricaricati con Config.LoadBackUp = TRUE.
CtrlParamsBackUp.Gain	REAL	1.0	Guadagno proporzionale memorizzato
CtrlParamsBackUp.Ti	REAL	20.0	Tempo di integrazione memorizzato
CtrlParamsBackUp.Td	REAL	0.0	Tempo derivativo memorizzato
CtrlParamsBackUp.TdFiltRatio	REAL	0.0	Coefficiente per il ritardo derivativo memorizzato
CtrlParamsBackUp.PWeighting	REAL	0.0	Ponderazione del componente P memorizzata
CtrlParamsBackUp.DWeighting	REAL	0.0	Ponderazione del componente D memorizzata
CtrlParamsBackUp.Cycle	REAL	1.0	Tempo di campionamento dell'algoritmo PID memorizzato
CtrlParamsBackUp.InputDead-Band	REAL	0.0	Ampiezza banda morta della differenza di regolazione memorizzata

Variabile	Tipo di dati	Preassegnazione	Descrizione
PIDSelfTune.SUT.CalculateSUTParams	BOOL	FALSE	Le proprietà del sistema regolato vengono memorizzate durante l'ottimizzazione. Se CalculateSUTParams = TRUE i parametri PID vengono ricalcolati sulla base di queste proprietà. I parametri PID vengono calcolati secondo il metodo impostato in TuneRuleSUT. Al termine del calcolo CalculateSUTParams viene impostato su FALSE.
PIDSelfTune.SUT.TuneRuleSUT	INT	1	Calcolo dei parametri durante l'ottimizzazione iniziale secondo un metodo: - TuneRuleSUT = 0: PID veloce I (comportamento di regolazione più veloce con ampiezze del valore di uscita più elevate rispetto a TuneRuleSUT = 1) - TuneRuleSUT = 1: PID lento I (comportamento di regolazione più lento con ampiezze del valore di uscita più basse rispetto a TuneRuleSUT = 0) - TuneRuleSUT = 2: Chien, Hrones, Reswick PID - TuneRuleSUT = 3: Chien, Hrones, Reswick PI - TuneRuleSUT = 4: PID veloce II (comportamento di regolazione più veloce con ampiezze del valore di uscita più elevate rispetto a TuneRuleSUT = 5) - TuneRuleSUT = 5: PID lento II (comportamento di regolazione più lento con ampiezze del valore di uscita più basse rispetto a TuneRuleSUT = 4) I metodi TuneRuleSUT = 0 e 1 si differenziano soltanto nel calcolo del guadagno proporzionale dai metodi TuneRuleSUT = 4 e 5: Con TuneRuleSUT = 0 e 1, il calcolo del guadagno proporzionale avviene sulla base del tempo di compensazione del processo. Con TuneRuleSUT = 4 e 5 ciò avviene sulla base del tempo di ritardo del processo. TuneRuleSUT = 4 e 5 fornisce un valore più elevato per il guadagno proporzionale e pertanto un comportamento di regolazione più veloce con ampiezze del valore di uscita più elevate rispetto a TuneRuleSUT = 0 e 1.
PIDSelfTune.SUT.State	INT	0	La variabile SUT.State mostra la fase attuale dell'ottimizzazione iniziale:
PIDSelfTune.TIR.RunIn	BOOL	FALSE	- RunIn = FALSE Se l'ottimizzazione fine viene avviata dal modo di funzionamento inattivo o manuale, si avvia un'ottimizzazione iniziale. Se l'ottimizzazione fine viene avviata dal funzionamento automatico, viene eseguita la regolazione sul setpoint con i parametri PID disponibili. Solo in seguito viene avviata l'ottimizzazione fine. Se non è possibile eseguire l'ottimizzazione iniziale, PID_3Step passa al modo di funzionamento "Inattivo". - RunIn = TRUE L'ottimizzazione iniziale viene ignorata. PID_3Step tenta di raggiungere il setpoint con il valore di uscita minimo o massimo. Ciò può causare una maggiore sovraoscillazione. Solo in seguito viene avviata l'ottimizzazione fine. Al termine dell'ottimizzazione fine RunIn viene impostato su FALSE.

Variabile	Tipo di dati	Preassegnazione	Descrizione
PIDSelfTune.TIR.CalculateTIRParams	BOOL	FALSE	Le proprietà del sistema regolato vengono memorizzate durante l'ottimizzazione. Se CalculateTIRParams = TRUE i parametri PID vengono ricalcolati sulla base di queste proprietà. I parametri PID vengono calcolati secondo il metodo impostato in TuneRuleTIR. Al termine del calcolo CalculateTIRParams viene impostato su FALSE.
PIDSelfTune.TIR.TuneRuleTIR	INT	0	Calcolo dei parametri durante l'ottimizzazione fine secondo un metodo: • TuneRuleTIR = 0: PID automatico • TuneRuleTIR = 1: PID veloce (comportamento di regolazione più veloce con ampiezze del valore di uscita più elevate rispetto a TuneRuleTIR = 2) • TuneRuleTIR = 2: PID lento (comportamento di regolazione più lento con ampiezze del valore di uscita più basse rispetto a TuneRuleTIR = 1) • TuneRuleTIR = 3: PID secondo Ziegler-Nichols • TuneRuleTIR = 4: PI secondo Ziegler-Nichols • TuneRuleTIR = 5: P secondo Ziegler-Nichols Affinché il calcolo dei parametri PIDcon CalculateTIRParams e TuneRuleTIR = 0, 1 o 2 possa essere ripetuto, anche l'ottimizzazione fine precedente deve essere stata eseguita con TuneRuleTIR = 0, 1, o 2. In caso contrario viene utilizzato TuneRuleTIR = 3. Il nuovo calcolo del parametro PIDcon CalculateTIRParams e TuneRuleTIR = 3, 4 o 5 è sempre possibile.
PIDSelfTune.TIR.State	INT	0	La variabile TIR.State mostra la fase attuale dell'ottimizzazione fine:
Retain.Mode	INT	0	Quando cambia il valore di Retain.Mode si passa a un altro modo di funzionamento. Il seguente modo di funzionamento viene attivato modificando Mode in: • Mode = 0: Inattivo • Mode = 1: Ottimizzazione iniziale • Mode = 2: Ottimizzazione fine • Mode = 3: Funzionamento automatico • Mode = 4: Funzionamento manuale • Mode = 5: Approssimazione valore di uscita sostitutivo • Mode = 6: Misura del tempo di regolazione • Mode = 7: Controllo errori • Mode = 8: Approssima valore di uscita sostitutivo con controllo errori Mode è a ritenzione.
Retain.CtrlParams.SetByUser ⁽¹⁾	BOOL	FALSE	Se SetByUser = FALSE i parametri PID vengono rilevati automaticamente e PID_3Step funziona con una banda morta nel valore di uscita. L'ampiezza della banda morta viene calcolata durante l'ottimizzazione sulla base dello scostamento standard del valore di uscita e memorizzata in Retain.CtrlParams.OutputDeadBand. Se SetByUser = TRUE i parametri PID vengono immessi manualmente e PID_3 Step funziona senza banda morta nel valore di uscita. Retain.CtrlParams.OutputDeadBand = 0.0 SetByUser è a ritenzione.
Retain.CtrlParams.Gain ⁽¹⁾	REAL	1.0	Guadagno proporzionale attivo Gain è a ritenzione.

Variabile	Tipo di dati	Preassegnazione	Descrizione
Retain.CtrlParams.Ti ⁽¹⁾	REAL	20.0	- Ti > 0.0: Tempo di integrazione attivo - Ti = 0.0: Il componente I è disattivato Ti è a ritenzione.
Retain.CtrlParams.Td ⁽¹⁾	REAL	0.0	- Td > 0.0: Tempo derivativo attivo - Td = 0.0: Il componente D è disattivato Td è a ritenzione.
Retain.CtrlParams.TdFiltRatio ⁽¹⁾	REAL	0.0	Coefficiente per il ritardo di differenziazione attivo TdFiltRatio è a ritenzione.
Retain.CtrlParams.PWeighting ⁽¹⁾	REAL	0.0	Ponderazione del componente P attiva PWeighting è a ritenzione.
Retain.CtrlParams.DWeighting ⁽¹⁾	REAL	0.0	Ponderazione del componente D attiva DWeighting è a ritenzione.
Retain.CtrlParams.Cycle ⁽¹⁾	REAL	1.0	Tempo di campionamento attivo dell'algoritmo PID in secondi che viene arrotondato a un multiplo intero del tempo di ciclo dell'OB richiamante. Cycle è a ritenzione.
Retain.CtrlParams.InputDeadBand ⁽¹⁾	REAL	0.0	Ampiezza della banda morta della differenza di regolazione InputDeadBand è a ritenzione.

Vedere anche

[Parametri State e Retain.Mode V1 \(Pagina 364\)](#)

[Variabile ActivateRecoverMode V1 \(Pagina 373\)](#)

[Caricamento degli oggetti tecnologici del dispositivo \(Pagina 48\)](#)

10.2.5.6 Parametri State e Retain.Mode V1**Relazione tra i parametri**

Il parametro State indica il modo di funzionamento attuale del regolatore PID. Il parametro State non è modificabile.

Per commutare il modo di funzionamento è necessario modificare la variabile Retain.Mode. Ciò vale anche se in Retain.Mode è già presente il valore del nuovo modo di funzionamento. Impostare quindi ad es. prima Retain.Mode = 0 e successivamente Retain.Mode = 3. Se il modo di funzionamento attuale del regolatore permette questa modifica, il parametro State assume il valore della variabile Retain.Mode.

Se PID_3Step cambia automaticamente modo di funzionamento, vale: State != Retain.Mode.

Esempi:

- Ottimizzazione iniziale riuscita
State = 3 e Retain.Mode = 1
- In caso di errore
State = 0 e Retain.Mode rimane sul valore assunto finora, ad es. Retain.Mode = 3

- ManualEnable = TRUE
State = 4 e Retain.Mode rimane sul valore assunto finora, ad es. Retain.Mode = 3

NOTA

Supponiamo ad es. di voler ripetere un'ottimizzazione fine conclusa correttamente senza terminare il funzionamento automatico con Mode = 0.

Se si imposta Retain.Mode su un valore non valido per un clock, ad es. 9999, non si influisce sul parametro State. Impostando nel clock successivo Mode = 2 è possibile generare Retain.Mode senza dover passare prima al modo di funzionamento "Inattivo".

Significato dei valori

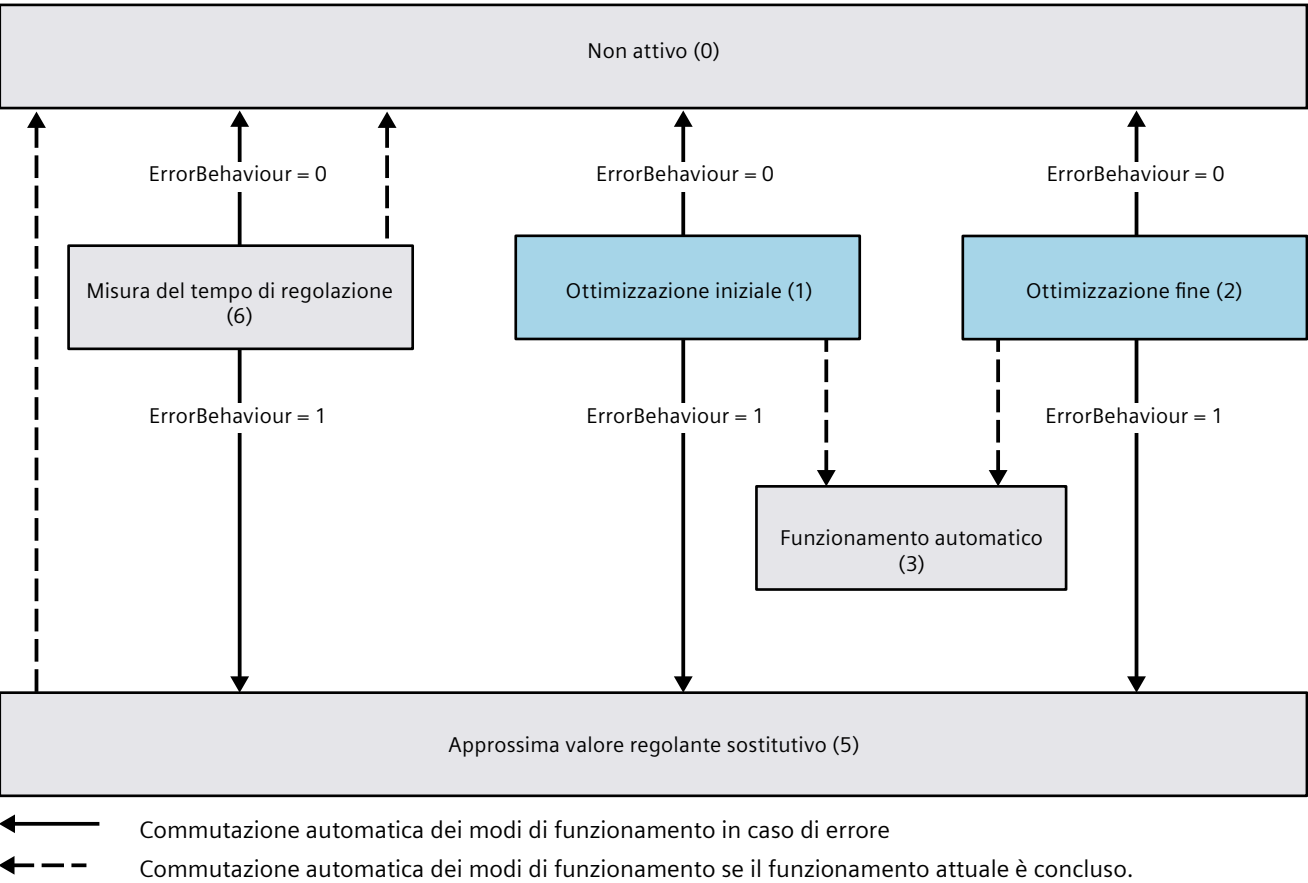
State / Retain.Mode	Descrizione
0	Inattivo Il regolatore è disattivato e non cambia più la posizione della valvola.
1	Ottimizzazione iniziale L'ottimizzazione iniziale rileva la risposta del processo a un impulso del valore di uscita e cerca il punto di flesso. Dalla pendenza max. e dal tempo morto del sistema regolato vengono calcolati i parametri PID ottimali. Requisiti per l'ottimizzazione iniziale: • State = 0 oppure State = 4 • ManualEnable = FALSE • Il tempo di regolazione del motore è stato configurato o misurato. • Il setpoint e il valore istantaneo sono compresi entro i limiti configurati. Più stabile è il valore istantaneo, più facilmente e con maggiore precisione si possono determinare i parametri PID. Un rumore del valore istantaneo è accettabile fintanto che l'incremento del valore è significativamente maggiore rispetto al rumore. Prima che i parametri PID vengano ricalcolati, vengono salvati e possono essere riattivati con Config.LoadBackUp. Il setpoint viene congelato nella variabile CurrentSetpoint. Se l'ottimizzazione iniziale viene conclusa correttamente si passa al funzionamento automatico, in caso contrario al modo "Inattivo". Viene visualizzata la fase dell'ottimizzazione iniziale con la variabile SUT.State.
2	Ottimizzazione fine L'ottimizzazione fine genera un'oscillazione costante e limitata del valore istantaneo. I parametri PID vengono ottimizzati partendo dall'ampiezza e dalla frequenza dell'oscillazione. Le differenze tra la risposta del processo durante l'ottimizzazione iniziale e fine vengono analizzate. A partire dai risultati vengono ricalcolati tutti i parametri PID. I parametri PID risultanti dall'ottimizzazione fine mostrano in genere un comportamento pilota e di disturbo migliore rispetto ai parametri PID dell'ottimizzazione iniziale. PID_3Step tenta di generare automaticamente un'oscillazione che sia maggiore rispetto al fruscio del valore istantaneo. L'ottimizzazione fine viene influenzata solo minimamente dalla stabilità del valore istantaneo. Prima dell'ottimizzazione fine i parametri PID vengono salvati e possono essere riattivati con Config.LoadBackUp. Il setpoint viene congelato nella variabile CurrentSetpoint. Requisiti per l'ottimizzazione fine: • Il tempo di regolazione del motore è stato configurato o misurato. • Il setpoint e il valore istantaneo sono compresi entro i limiti configurati. • ManualEnable = FALSE • Tipo di funzionamento automatico (State = 3), Inattivo (State = 0) o manuale (State = 4)

State / Retain.Mode	Descrizione
	L'ottimizzazione fine si svolge nel modo seguente all'avvio: - Funzionamento automatico (State = 3) Per migliorare i parametri PID esistenti mediante un'ottimizzazione, avviare l'ottimizzazione fine dal modo di funzionamento automatico. PID_3Step esegue la regolazione con i parametri PID esistenti finché il circuito di regolazione non si assesta e finché non sono soddisfatti i requisiti per un'ottimizzazione fine. Solo in seguito viene avviata l'ottimizzazione fine. - Funzionamento Inattivo (State = 0) o manuale (State = 4) Viene sempre avviata per prima un'ottimizzazione iniziale. Con i parametri PID rilevati viene quindi effettuata una regolazione finché il circuito di regolazione è assestato e sono soddisfatti i requisiti per l'ottimizzazione fine. Se PIDSelfTune.TIR.RunIn = TRUE l'ottimizzazione iniziale viene ignorata e si tenta di raggiungere il setpoint con il valore di uscita minimo o massimo. Ciò può causare una maggiore sovraelongazione. L'ottimizzazione fine si avvia poi automaticamente. Se l'ottimizzazione fine si conclude correttamente il regolatore passa al modo "Funzionamento automatico", in caso contrario al modo di funzionamento "Inattivo". Viene visualizzata la fase dell'ottimizzazione fine con la variabile TIR.State.
3	Funzionamento automatico In funzionamento automatico PID_3Step regola il sistema regolato secondo i parametri preimpostati. Se è soddisfatta una delle seguenti condizioni si passa al funzionamento automatico: - Ottimizzazione iniziale conclusa correttamente - Ottimizzazione fine conclusa correttamente - Modifica della variabile Retain.Mode sul valore 3. Se la CPU viene attivata o passa da STOP a RUN, PID_3Step viene avviato nell'ultimo modo di funzionamento attivo. Per lasciare PID_3Step nel modo di funzionamento "Inattivo" impostare RunModeByStartup = FALSE. In funzionamento automatico viene tenuta in considerazione la variabile ActivateRecoverMode.
4	Funzionamento manuale Nel funzionamento manuale si preimpostano manualmente i valori di uscita nei parametri Manual_UP e Manual_DN o ManualValue. Nel parametro ErrorBits è indicato se l'organo attuatore può essere spostato sul valore di uscita in caso di errore. Questo modo di funzionamento viene attivato se Retain.Mode = 4 o con fronte di salita in ManualEnable. Se ManualEnable diventa TRUE cambia solo State. Retain.Mode rimane sul valore finora assunto. In presenza di un fronte di discesa in ManualEnable PID_3Step torna al modo di funzionamento precedente. La commutazione nel funzionamento automatico è bumpless. PID_3Step V1.1 In caso di errore il funzionamento manuale è sempre possibile. PID_3Step V1.0 In caso di errore il funzionamento manuale dipende dalla variabile ActivateRecoverMode.
5	Approssimazione valore di uscita sostitutivo Questo modo di funzionamento viene attivato in caso di errore o con Reset = TRUE se Errorbehaviour = 1 e ActivateRecoverMode = FALSE. PID_3Step sposta l'organo attuatore sul valore di uscita sostitutivo e passa quindi al modo di funzionamento "Inattivo".
6	Misura del tempo di regolazione Viene rilevato il tempo impiegato dal motore per aprire completamente la valvola chiusa. Questo modo di funzionamento si attiva se GetTransitTime.Start = TRUE. Se per la misura del tempo di regolazione si utilizzano segnali di riscontro, la valvola viene aperta completamente dalla posizione attuale, chiusa completamente e riaperta completamente. Se GetTransitTime.InvertDirection = TRUE si verifica il comportamento inverso. Se per la misura del tempo di regolazione si utilizza una risposta di posizione, l'organo attuatore viene spostato dalla posizione attuale a una di destinazione. Durante la misura del tempo di regolazione, i limiti del valore di uscita non vengono considerati. L'organo attuatore può essere traslato fino al riscontro superiore o fino a quello inferiore.

State / Retain.Mode	Descrizione
7	Controllo errori L'algoritmo di regolazione è disattivato e non modifica più la posizione della valvola. Questo modo di funzionamento viene attivato al posto di "Inattivo" in caso di errore. Devono essere soddisfatte tutte le condizioni seguenti: • Mode = 3 (funzionamento automatico) • Errorbehaviour = 0 • ActivateRecoverMode = TRUE • Si sono verificati uno o più errori nei quali è attiva ActivateRecoverMode (Pagina 373). Non appena l'errore viene eliminato PID_3Step torna al funzionamento automatico.
8	Approssima valore di uscita sostitutivo con controllo errori Questo modo di funzionamento viene attivato in caso di errore invece del modo "Approssimazione del valore di uscita sostitutivo". PID_3Step sposta l'organo attuatore sul valore di uscita sostitutivo e passa quindi nel modo di funzionamento "Controllo errori". Devono essere soddisfatte tutte le condizioni seguenti: • Mode = 3 (funzionamento automatico) • Errorbehaviour = 1 • ActivateRecoverMode = TRUE • Si sono verificati uno o più errori nei quali è attiva ActivateRecoverMode (Pagina 373). Non appena l'errore viene eliminato PID_3Step torna al funzionamento automatico.

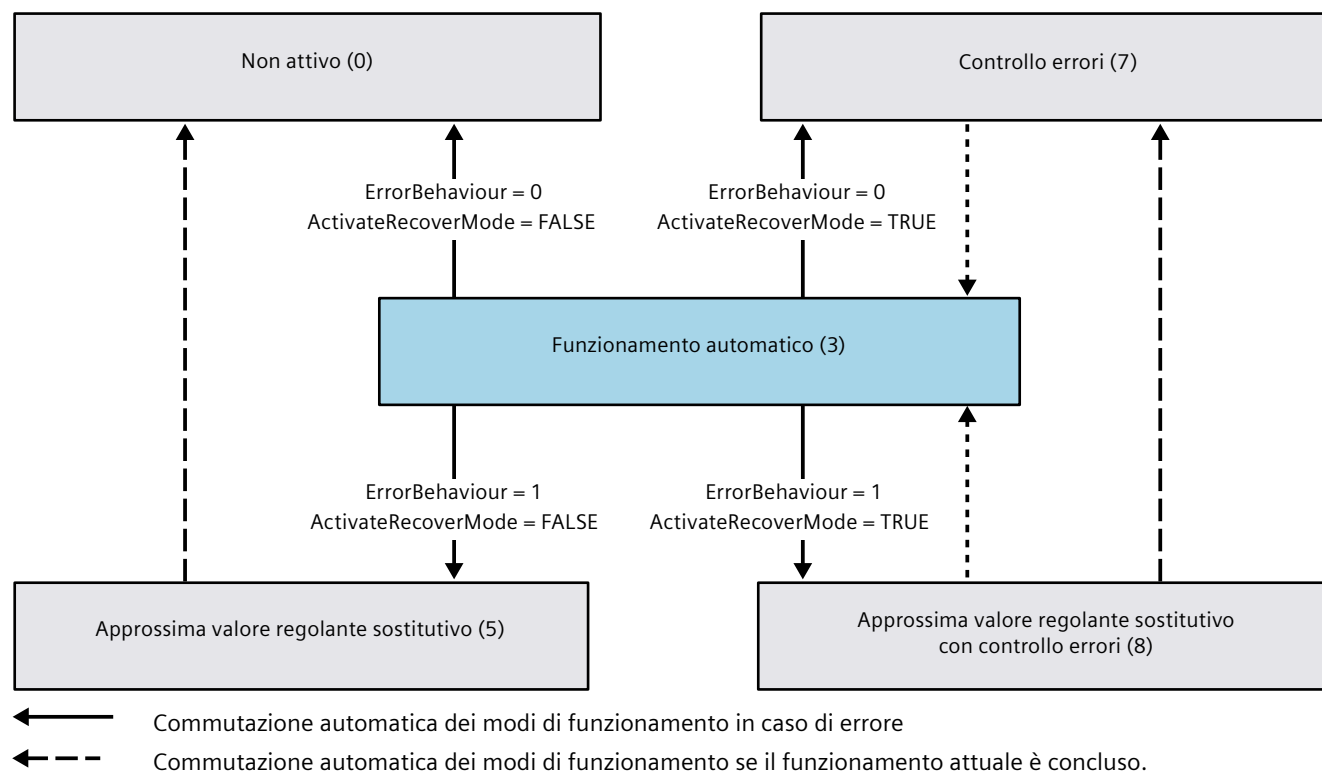
Commutazione automatica del modo di funzionamento durante la messa in servizio

In caso di errore PID_3Step cambia automaticamente modo di funzionamento. Il diagramma seguente mostra l'effetto di ErrorBehaviour sulla commutazione dei modi di funzionamento Misura del tempo di regolazione, Ottimizzazione iniziale e Ottimizzazione fine.



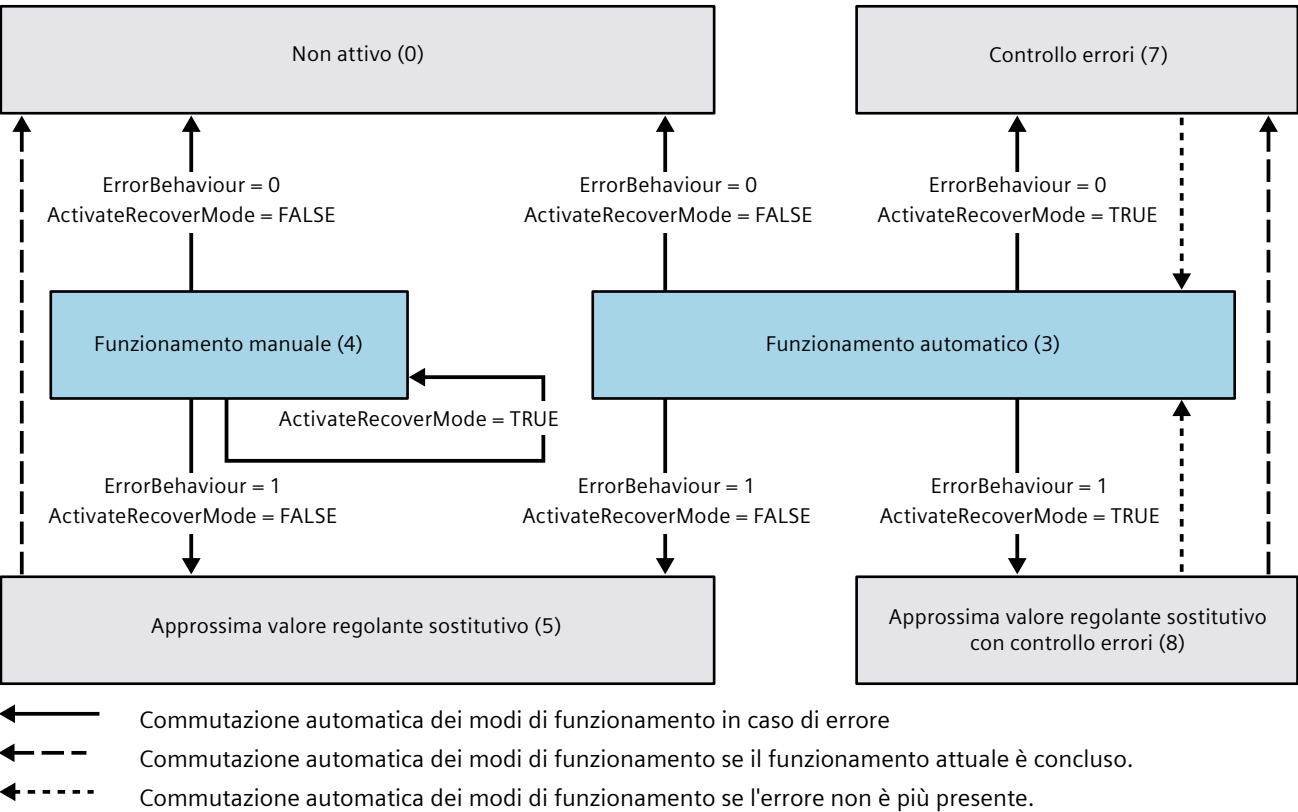
Commutazione automatica del modo di funzionamento in funzionamento automatico (PID_3Step V1.1)

In caso di errore PID_3Step cambia automaticamente modo di funzionamento. Il diagramma seguente mostra l'effetto di ErrorBehaviour e ActivateRecoverMode su questa commutazione dei modi di funzionamento.



Commutazione automatica nel modo di funzionamento automatico e manuale (PID_3Step V1.0)

In caso di errore PID_3Step cambia automaticamente modo di funzionamento. Il diagramma seguente mostra l'effetto di ErrorBehaviour e ActivateRecoverMode su questa commutazione dei modi di funzionamento.



Vedere anche

[Variabile ActivateRecoverMode V1 \(Pagina 373\)](#)

[Parametro ErrorBits V1 \(Pagina 371\)](#)

10.2.5.7 Parametro ErrorBits V1

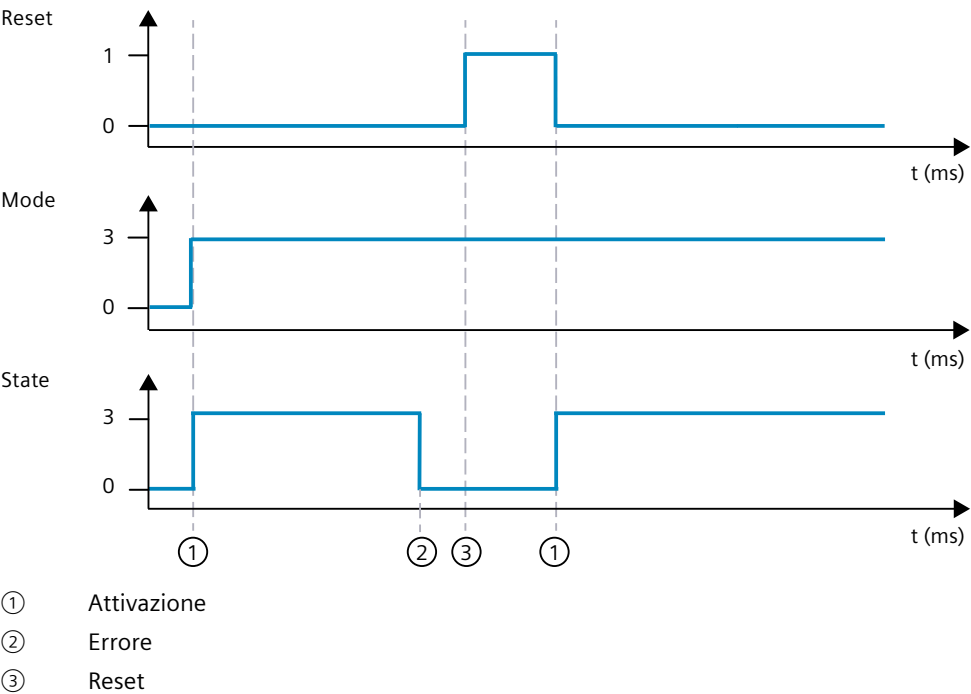
Se si verificano diversi errori contemporaneamente, i valori dei codici di errore vengono visualizzati in un'addizione binaria. Se viene visualizzato il codice di errore 0003, ad es., significa che sono presenti contemporaneamente gli errori 0001 e 0002.

ErrorBits (DW#16#...)	Descrizione
0000	Senza errori.
0001	Il parametro "Input" non rientra nei limiti del valore istantaneo. • Input > Config.InputUpperLimit o • Input < Config.InputLowerLimit Se ActivateRecoverMode = TRUE e ErrorBehaviour = 1, l'organo attuatore viene spostato sul valore di uscita sostitutivo. Se ActivateRecoverMode = TRUE e ErrorBehaviour = 0, l'organo attuatore rimane nella posizione attuale. Se ActivateRecoverMode = FALSE l'organo attuatore rimane nella posizione attuale. PID_3Step V1.1 In funzionamento manuale è possibile spostare l'organo attuatore. PID_3Step V1.0 Il funzionamento manuale non è possibile in questo stato. L'organo attuatore si può spostare solo dopo aver eliminato l'errore.
0002	Valore del parametro "Input_PER" non valido. Controllare l'eventuale presenza di un errore sull'ingresso analogico. Se prima che si verificasse l'errore era attivo il funzionamento automatico, ActivateRecoverMode = TRUE e l'errore non è più presente, PID_3Step torna al funzionamento automatico.
0004	Errore durante l'ottimizzazione fine. Non è stato possibile mantenere l'oscillazione del valore istantaneo.
0020	L'ottimizzazione iniziale non è consentita nel funzionamento automatico e durante l'ottimizzazione fine.
0080	Errore nell'ottimizzazione iniziale. I limiti del valore di uscita non sono configurati correttamente e il valore istantaneo non reagisce come previsto. Verificare che i limiti del valore di uscita siano configurati correttamente e che siano compatibili con il senso di regolazione. Accertarsi inoltre che il valore istantaneo non presenti oscillazioni troppo forti prima dell'avvio dell'ottimizzazione iniziale.
0100	Un errore durante l'ottimizzazione fine ha causato parametri non validi.
0200	Valore del parametro "Input" non valido: il formato numerico del valore non è valido. Se prima che si verificasse l'errore era attivo il funzionamento automatico, ActivateRecoverMode = TRUE e l'errore non è più presente, PID_3Step torna al funzionamento automatico.
0400	Calcolo del valore di uscita non riuscito. Controllare i parametri PID.
0800	Errore del tempo di campionamento: PID_3Step non viene richiamato entro il tempo di campionamento dell'OB di schedulazione orologio. Se prima che si verificasse l'errore era attivo il funzionamento automatico, ActivateRecoverMode = TRUE e l'errore non è più presente, PID_3Step torna al funzionamento automatico. Se quest'errore si è verificato nella simulazione con PLCSIM, osservare le avvertenze riportate alla sezione Simulazione di PID_3Step V1 con PLCSIM (Pagina 165).
1000	Valore del parametro "Setpoint" non valido: il formato numerico del valore non è valido. Se prima che si verificasse l'errore era attivo il funzionamento automatico, ActivateRecoverMode = TRUE e l'errore non è più presente, PID_3Step torna al funzionamento automatico.

ErrorBits (DW#16#...)	Descrizione
2000	Valore del parametro Feedback_PER non valido. Controllare l'eventuale presenza di un errore sull'ingresso analogico. L'organo attuatore non può essere posizionato sul valore di uscita sostitutivo e resta fermo sulla posizione attuale. Il funzionamento manuale non è possibile in questo stato. Per poter spostare l'attuatore da questo stato è necessario disattivare la risposta di posizione (Config. FeedbackOn = FALSE). Se prima che si verificasse l'errore era attivo il funzionamento automatico, ActivateRecoverMode = TRUE e l'errore non è più presente, PID_3Step torna al funzionamento automatico.
4000	Valore del parametro Feedback non valido. il formato numerico del valore non è valido. L'organo attuatore non può essere posizionato sul valore di uscita sostitutivo e resta fermo sulla posizione attuale. Il funzionamento manuale non è possibile in questo stato. Per poter spostare l'attuatore da questo stato è necessario disattivare la risposta di posizione (Config. FeedbackOn = FALSE). Se prima che si verificasse l'errore era attivo il funzionamento automatico, ActivateRecoverMode = TRUE e l'errore non è più presente, PID_3Step torna al funzionamento automatico.
8000	Errore nella risposta di posizione digitale. Actuator_H = TRUE e Actuator_L = TRUE. L'organo attuatore non può essere posizionato sul valore di uscita sostitutivo e resta fermo sulla posizione attuale. Il funzionamento manuale non è possibile in questo stato. Per poter spostare l'attuatore da questo stato è necessario disattivare "Segnali di riscontro organo attuatore" (Config.ActuatorEndStopOn = FALSE). Se prima che si verificasse l'errore era attivo il funzionamento automatico, ActivateRecoverMode = TRUE e l'errore non è più presente, PID_3Step torna al funzionamento automatico.

10.2.5.8 Parametro Reset V1

Un fronte di salita nel parametro Reset determina la commutazione nel modo di funzionamento "Inattivo"; gli errori e gli avvisi vengono resettati. Un fronte di discesa nel parametro Reset commuta nell'ultimo modo di funzionamento attivo. Se era in precedenza attivato il funzionamento automatico, la commutazione nel funzionamento automatico avviene in modo continuo.



10.2.5.9 Variabile ActivateRecoverMode V1

Il comportamento della variabile ActivateRecoverMode dipende dalla versione dell'istruzione PID_3Step.

Comportamento nella versione 1.1

La variabile ActivateRecoverMode determina il comportamento in caso di errore in funzionamento automatico. Durante l'ottimizzazione iniziale, l'ottimizzazione fine e la misura del tempo di regolazione ActivateRecoverMode non è attiva.

ActivateRecover-Mode	Descrizione
FALSE	In caso di errore PID_3Step commuta nel modo di funzionamento "Inattivo" o "Approssimazione del valore di uscita sostitutivo". Il regolatore viene attivato solo con il reset o con una modifica in Retain.Mode.
TRUE	Se in funzionamento automatico si verificano errori frequenti, questa impostazione peggiora il comportamento del regolatore. In questo caso controllare il parametro ErrorBits ed eliminare la causa dell'errore. Se si verificano uno o più dei seguenti errori, PID_3Step commuta nel modo di funzionamento "Approssimazione del valore di uscita sostitutivo con controllo errori" o "Controllo errori": • 0002h: Valore del parametro Input_PER non valido. • 0200h: Valore del parametro Input non valido. • 0800h: Errore del tempo di campionamento • 1000h: Valore del parametro Setpoint non valido. • 2000h: Valore del parametro Feedback_PER non valido. • 4000h: Valore del parametro Feedback non valido. • 8000h: Errore nella risposta di posizione digitale. Con gli errori 2000h, 4000h e 8000h PID_3Step non può approssimare il valore di uscita sostitutivo configurato. Non appena l'errore viene eliminato PID_3Step torna al funzionamento automatico.

Comportamento nella versione 1.0

La variabile ActivateRecoverMode determina il comportamento in caso di errore in funzionamento automatico e manuale. Durante l'ottimizzazione iniziale, l'ottimizzazione fine e la misura del tempo di regolazione ActivateRecoverMode non è attiva.

ActivateRecover-Mode	Descrizione
FALSE	In caso di errore PID_3Step commuta nel modo di funzionamento "Inattivo" o "Approssimazione del valore di uscita sostitutivo". Il regolatore viene attivato solo con il reset o con una modifica in Retain.Mode.
TRUE	Errori in funzionamento automatico Se in funzionamento automatico si verificano errori frequenti, questa impostazione peggiora il comportamento del regolatore. In questo caso controllare il parametro ErrorBits ed eliminare la causa dell'errore. Se si verificano uno o più dei seguenti errori, PID_3Step commuta nel modo di funzionamento "Approssimazione del valore di uscita sostitutivo con controllo errori" o "Controllo errori": • 0002h: Valore del parametro Input_PER non valido. • 0200h: Valore del parametro Input non valido. • 0800h: Errore del tempo di campionamento • 1000h: Valore del parametro Setpoint non valido. • 2000h: Valore del parametro Feedback_PER non valido. • 4000h: Valore del parametro Feedback non valido. • 8000h: Errore nella risposta di posizione digitale.

ActivateRecover-Mode	Descrizione
	Con gli errori 2000h, 4000h e 8000h PID_3Step non può approssimare il valore di uscita sostitutivo configurato. Non appena l'errore viene eliminato PID_3Step torna al funzionamento automatico. Errori in funzionamento manuale Se si verificano uno o più dei seguenti errori PID_3Step resta in funzionamento manuale: • 0002h: Valore del parametro Input_PER non valido. • 0200h: Valore del parametro Input non valido. • 0800h: Errore del tempo di campionamento • 1000h: Valore del parametro Setpoint non valido. • 2000h: Valore del parametro Feedback_PER non valido. • 4000h: Valore del parametro Feedback non valido. • 8000h: Errore nella risposta di posizione digitale. Con gli errori 2000h, 4000h e 8000h non è possibile portare la valvola in una posizione adeguata.

Vedere anche

[Variabile statica PID_3Step V1 \(Pagina 365\)](#)

[Parametri State e Retain.Mode V1 \(Pagina 372\)](#)

10.2.5.10 Variabile Warning V1

Se si presentano contemporaneamente più avvisi, i loro valori vengono visualizzati in un'addizione binaria. La visualizzazione ad es. dell'avviso 0003 indica la presenza contemporanea degli avvisi 0001 e 0002.

Warning (DW#16#...)	Descrizione
0000	Non è presente alcun avviso.
0001	Durante l'ottimizzazione iniziale non è stato trovato il punto di flesso.
0002	Durante l'ottimizzazione fine l'oscillazione è stata intensificata.
0004	Il setpoint è stato circoscritto ai limiti impostati.
0008	Per il metodo di calcolo selezionato non sono state definite tutte le proprietà necessarie per il sistema regolato. Al loro posto sono stati calcolati i parametri PID con il metodo TuneRuleTIR = 3.
0010	Non è stato possibile commutare il modo di funzionamento perché ManualEnable = TRUE.
0020	Il tempo di campionamento dell'algoritmo PID viene circoscritto dal tempo di ciclo dell'OB richiamante. Per ottenere risultati migliori utilizzare tempi di ciclo dell'OB più brevi.
0040	Il valore istantaneo ha superato uno dei suoi limiti di avviso.
0080	Valore di Retain.Mode non valido. Il modo di funzionamento non viene commutato.
0100	Il valore manuale è stato circoscritto ai limiti dell'uscita del regolatore.
0200	La regola utilizzata per l'ottimizzazione non produce alcun risultato corretto o non viene supportata.
0400	Il metodo "Misura del tempo di regolazione" selezionato non è adatto all'organo attuatore. Non è possibile misurare il tempo di regolazione perché le impostazioni dell'attuatore non corrispondono al metodo di misura selezionato.
0800	Nella misura del tempo di regolazione la differenza tra la posizione attuale e il nuovo valore di uscita è troppo piccola. Ciò può produrre risultati errati. La differenza tra valore di uscita attuale e predefinito deve essere almeno pari al 50% dell'intero campo di regolazione.
1000	Non è possibile raggiungere il valore di uscita sostitutivo perché non rientra nei limiti del valore di uscita.

I seguenti avvisi vengono cancellati non appena viene eliminata la causa:

- 0004
- 0020
- 0040
- 0100

Tutti gli altri avvisi vengono cancellati in presenza di un fronte di salita in Reset.

10.2.5.11 Variabile SUT.State V1

SUT.State	Nome	Descrizione
0	SUT_INIT	Inizializzazione dell'ottimizzazione iniziale
50	SUT_TPDN	Rilevamento della posizione iniziale senza risposta di posizione
100	SUT_STDABW	Calcolo della divergenza standard
200	SUT_GET_POI	Rilevazione del punto di flesso
300	SUT_GET_RISETM	Rilevamento del tempo di stabilizzazione della regolazione
9900	SUT_IO	Ottimizzazione iniziale riuscita
1	SUT_NIO	Ottimizzazione iniziale fallita

10.2.5.12 Variabile TIR.State V1

TIR.State	Nome	Descrizione
-100	TIR_FIRST_SUT	Non è possibile eseguire l'ottimizzazione fine. Viene prima eseguita un'ottimizzazione iniziale.
0	TIR_INIT	Inizializzazione dell'ottimizzazione fine
200	TIR_STDABW	Calcolo della divergenza standard
300	TIR_RUN_IN	Tentativi di raggiungere il setpoint con il valore di uscita massimo o minimo
400	TIR_CTRLN	Tentativo di raggiungere il setpoint con i parametri PID esistenti (se l'ottimizzazione iniziale è riuscita)
500	TIR_OSZIL	Rilevazione dell'oscillazione e calcolo dei parametri
9900	TIR_IO	ottimizzazione fine riuscita
1	TIR_NIO	ottimizzazione fine fallita

10.3 PID_Temp

10.3.1 Novità PID_Temp

PID_Temp V1.1

- **Utilizzo con S7-1200 G2**

Da PID_Temp V1.1 l'istruzione può essere utilizzata anche in un'S7-1200 G2.

Le versioni precedenti dell'istruzione non sono supportate nell'S7-1200 G2.

- **Reazione del valore di uscita alla commutazione dal modo di funzionamento "Inattivo" al modo "Funzionamento automatico"**

La nuova opzione IntegralResetMode = 4 è stata inserita e definita come impostazione di default. Con IntegralResetMode = 4 alla commutazione dal modo di funzionamento "Inattivo" al modo "Funzionamento automatico", al componente I viene automaticamente assegnato per default un valore che ad una differenza di regolazione comporti un salto del valore di uscita PID con segno identico.

- **Inizializzazione del componente I nel funzionamento automatico**

L'inizializzazione del componente integrale può avere luogo nel funzionamento automatico con l'ausilio delle variabili OverwriteInitialOutputValue e PIDCtrl.PIDInit. Ciò semplifica l'impiego di PID_Temp nelle regolazioni con funzione override.

10.3.2 Compatibilità con CPU e FW

La tabella seguente mostra quali CPU sono compatibili con quale versione di PID_Temp.

CPU	FW	PID_Temp
S7-1200	da V4.2	V1.1 V1.0
	V4.1	V1.0
S7-1200 G2	da V1.0	V1.1
S7-1500	da V3.0	V1.1
	Da V2.0 a V2.9	V1.1 V1.0
	V1.7 fino a V1.8	V1.0

10.3.3 Tempo di elaborazione della CPU e memoria necessaria di PID_Temp V1

Tempi di elaborazione della CPU

Tempi di elaborazione tipici della CPU dell'oggetto tecnologico PID_Temp dalla versione V1.0 in funzione del tipo di CPU e del modo di funzionamento per CPU standard, F, T e TF.

CPU	FW	Tempi di elaborazione tipici della CPU nel funzionamento automatico	Tempi di elaborazione tipici della CPU per ottimizzazione iniziale e ottimizzazione fine
CPU 1211	≥ V4.1	290 μs	450 μs
CPU 1212			
CPU 1214			
CPU 1215			
CPU 1217			
CPU 1212 G2	≥ V1.0	150 μs	210 μs
CPU 1214 G2			
CPU 1510SP	≤ V2.9	85 μs	140 μs
CPU 1511			
CPU 1511C			
CPU 1512C			
CPU 1512SP			
CPU 1513			
CPU 1515		80 μs	110 μs
CPU 1516			
CPU 1517	≤ V3.1	12 μs	18 μs
CPU 1518		6 μs	9 μs
CPU 1510SP	V3.0, V3.1	75 μs	110 μs
CPU 1511			
CPU 1511C			
CPU 1512C			
CPU 1512SP			
CPU 1513			
CPU 1514SP			
CPU 1515		70 μs	95 μs
CPU 1516			
CPU 1510SP	≥ V4.0	70 μs	95 μs
CPU 1511			
CPU 1511C			
CPU 1512C			
CPU 1512SP			
CPU 1513			

CPU 1514SP	≥ V4.0	70 µs	95 µs
CPU 1515			
CPU 1516			
CPU 1516T		4 µs	6 µs
CPU 1517		3 µs	5 µs
CPU 1518		2 µs	3 µs

Tempi tipici di elaborazione CPU dell'oggetto tecnologico PID_Temp con versione V1.0 o superiore in funzione del tipo di CPU e del modo di funzionamento per le CPU R/H nello stato di sistema RUN-Redundant.

CPU	FW	Tempi di elaborazione tipici della CPU nel funzionamento automatico	Tempi di elaborazione tipici della CPU per ottimizzazione iniziale e ottimizzazione fine
CPU 1513R	V3.0, V3.1	110 µs	160 µs
CPU 1515R		100 µs	140 µs
CPU 1513R	≥ V4.0	100 µs	140 µs
CPU 1515R			
CPU 1517H		5 µs	7 µs
CPU 1518HF			

Memoria necessaria

Memoria necessaria di un DB di istanza dell'oggetto tecnologico PID_Temp a partire dalla versione V1.0.

Memoria necessaria	Memoria necessaria di un DB di istanza di PID_Temp V1
Memoria di caricamento necessaria	Circa 4700 byte
Memoria di lavoro necessaria complessivamente	1280 byte
Memoria di lavoro necessaria a ritenzione	100 byte

10.3.4 PID_Temp

10.3.4.1 Descrizione di PID_Temp

Descrizione

L'istruzione PID_Temp mette a disposizione un regolatore PID con ottimizzazione integrata per processi di temperatura. PID_Temp si addice per applicazioni di solo riscaldamento oppure di riscaldamento/raffreddamento.

Sono disponibili i seguenti modi di funzionamento:

- Inattivo
- Ottimizzazione iniziale
- Ottimizzazione fine
- Funzionamento automatico
- Funzionamento manuale
- Valore di uscita sostitutivo con controllo errori

Una descrizione dettagliata dei modi di funzionamento è disponibile al paragrafo relativo al parametro State.

Algoritmo PID

PID_Temp è un regolatore PIDT1 con anti-windup e ponderazione del componente P e D. L'algoritmo PID utilizza la seguente formula (la zona di regolazione e la banda morta sono disattivate):

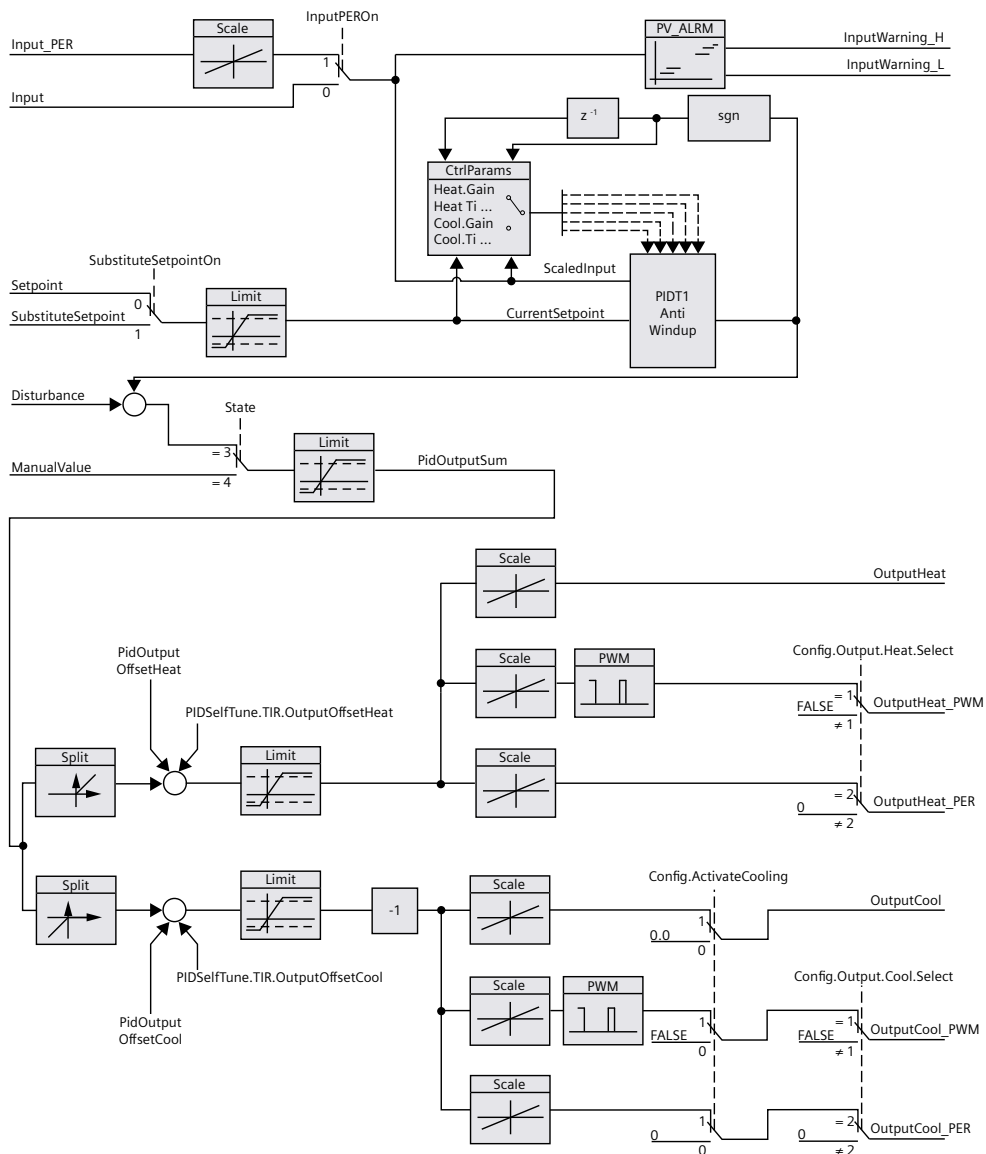
$$y = K_p \left[(b \cdot w - x) + \frac{1}{T_i \cdot s} (w - x) + \frac{T_d \cdot s}{a \cdot T_d \cdot s + 1} (c \cdot w - x) \right]$$

La tabella sottostante illustra il significato dei simboli impiegati nella formula e nelle figure successive.

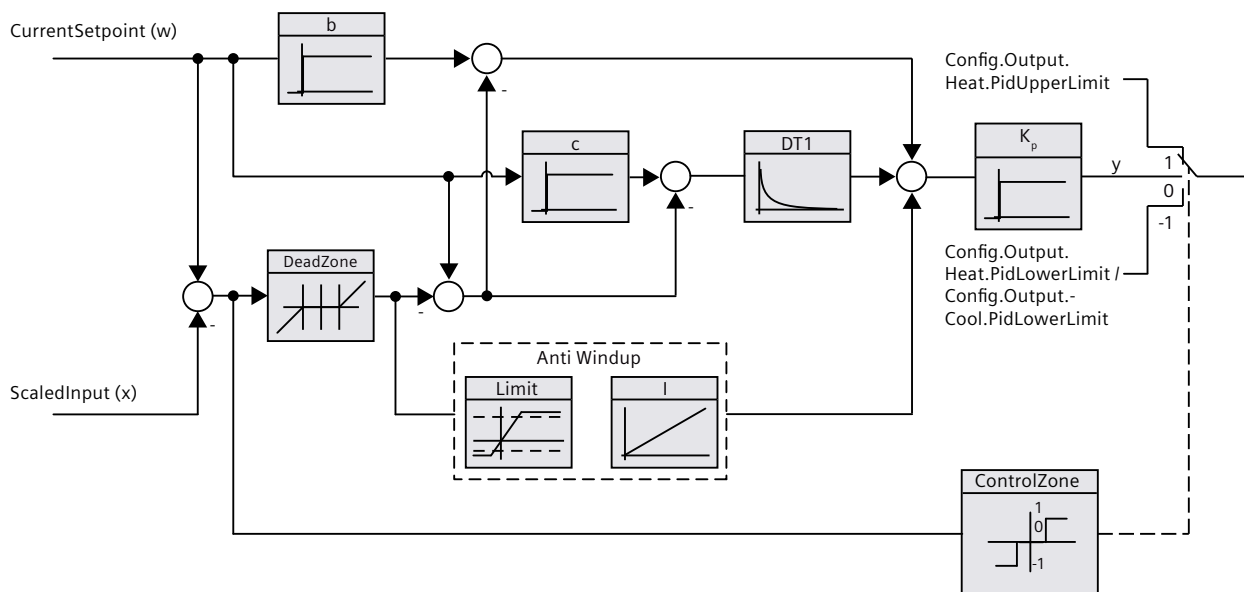
Simbolo	Descrizione	Parametri dell'istruzione PID_Temp
y	Valore di uscita dell'algoritmo PID	-
K _p	Guadagno proporzionale	Retain.CtrlParams.Heat.Gain Retain.CtrlParams.Cool.Gain CoolFactor
s	Operatore di Laplace	-
b	Ponderazione del componente P	Retain.CtrlParams.Heat.PWeighting Retain.CtrlParams.Cool.PWeighting
w	Setpoint	CurrentSetpoint
x	Valore istantaneo	ScaledInput
T _i	Tempo di integrazione	Retain.CtrlParams.Heat.Ti Retain.CtrlParams.Cool.Ti

Simbolo	Descrizione	Parametri dell'istruzione PID_Temp
T_D	Tempo derivativo	Retain.CtrlParams.Heat.Td Retain.CtrlParams.Cool.Td
a	Coefficiente per il ritardo derivativo (ritardo derivativo $T1 = a \times T_D$)	Retain.CtrlParams.Heat.TdFiltRatio Retain.CtrlParams.Cool.TdFiltRatio
c	Ponderazione del componente D	Retain.CtrlParams.Heat.DWeighting Retain.CtrlParams.Cool.DWeighting
DeadZone	Ampiezza banda morta	Retain.CtrlParams.Heat.DeadZone Retain.CtrlParams.Cool.DeadZone
ControlZone	Ampiezza della zona di regolazione	Retain.CtrlParams.Heat.ControlZone Retain.CtrlParams.Cool.ControlZone

Schema a blocchi di PID_Temp



Schema a blocchi di PIDT1 con anti-windup



Richiamo

PID_Temp viene richiamato a intervalli costanti nel ciclo di un OB di schedulazione orologio.

Caricamento nel dispositivo

L'aggiornamento dei valori attuali delle variabili a ritenzione, avviene soltanto con il caricamento completo di PID_Temp .

Caricare un oggetto tecnologico nel dispositivo [\(Pagina 48\)](#)

Avviamento

All'avviamento della CPU PID_Temp si avvia nel modo di funzionamento salvato nel parametro di transito Mode. Per commutare al modo di funzionamento "Inattivo" durante la fase di avvio, impostare `RunModeByStartup = FALSE`.

Comportamento in caso di errori

Il comportamento in caso di errori dipende dalle variabili SetSubstituteOutput e ActivateRecoverMode. Con ActivateRecoverMode = TRUE il comportamento è inoltre in funzione dell'errore verificatosi.

SetSubstituteOutput	ActivateRecoverMode	Editor di configurazione > Impostazioni di base uscita > Imposta PidOutputSum su	Comportamento
irrelevante	FALSE	Zero (Inattivo)	Commutazione nel modo di funzionamento "Inattivo" (State = 0) Il valore di uscita dell'algoritmo PID e tutte le uscite per riscaldamento e raffreddamento vengono impostati su 0. La graduazione di scala delle uscite per riscaldamento e raffreddamento non è attiva.
FALSE	TRUE	Valore attuale per la durata dell'errore	Commutazione nel modo di funzionamento "valore di uscita sostitutivo con diagnostica" (State = 5) Il valore di uscita attuale viene trasmesso all'organo attuatore per la durata dell'errore.
TRUE	TRUE	Valore di uscita sostitutivo per la durata dell'errore	Commutazione nel modo di funzionamento "valore di uscita sostitutivo con diagnostica" (State = 5) Il valore di SubstituteOutput viene trasmesso all'organo attuatore per la durata dell'errore.

PID_Temp utilizza in funzionamento manuale ManualValue come valore di uscita, tranne se ManualValue non è valido

- Se ManualValue non è valido viene utilizzato SubstituteOutput.
- Se ManualValue e SubstituteOutput non sono validi viene utilizzato Config.Output.Heat.PidLowerLimit.

Il parametro Error indica se attualmente è presente un errore. Se l'errore non è più presente, compare Error = FALSE. Il parametro ErrorBits indica quali errori si sono verificati. ErrorBits viene resettato da un fronte di salita in Reset o ErrorAck.

10.3.4.2 Funzionamento di PID_Temp

Sorveglianza dei limiti del valore istantaneo

Nelle variabili Config.InputUpperLimit e Config.InputLowerLimit si definisce il limite superiore e inferiore del valore istantaneo. Se il valore istantaneo non rientra in questi limiti si verifica un errore (ErrorBits = 0000001h).

Nelle variabili Config.InputUpperWarning e Config.InputLowerWarning si definisce il limite di avviso superiore e inferiore del valore istantaneo. Se il valore istantaneo non rientra in questi limiti viene emesso un avviso (Warning = 0000040h) e il parametro di uscita InputWarning_H o InputWarning_L diventa TRUE.

Definizione dei limiti per il setpoint

Nelle variabili Config.SetpointUpperLimit e Config.SetpointLowerLimit si definisce il limite superiore e inferiore del setpoint. PID_Temp circonda automaticamente il setpoint sui limiti del valore istantaneo. Il setpoint può essere limitato su un campo inferiore. PID_Temp controlla se questo campo si trova all'interno dei limiti del valore istantaneo. Se il setpoint non rientra in questi limiti, il limite superiore o inferiore viene utilizzato come setpoint e il parametro di uscita SetpointLimit_H o SetpointLimit_L diventa TRUE.

Il setpoint viene circoscritto in tutti i modi operativi.

Setpoint sostitutivo

Nelle variabili SubstituteSetpoint è possibile indicare un setpoint sostitutivo e procedere alla relativa attivazione con SubstituteSetpointOn = TRUE. In questo modo è possibile ad es. indicare direttamente il setpoint di un regolatore slave in cascata senza modificare il programma utente. I limiti fissati per il setpoint valgono anche per il setpoint sostitutivo.

Riscaldamento e raffreddamento

Con il valore predefinito, PID_Temp utilizza soltanto le uscite per il riscaldamento (OutputHeat, OutputHeat_PWM, OutputHeat_PER). Il valore di uscita dell'algoritmo PID (PidOutputSum) viene riportato in scala ed emesso sulle uscite per il riscaldamento. Se debbano essere calcolati OutputHeat_PWM o OutputHeat_PER, viene stabilito con Config.Output.Heat.Select. OutputHeat viene sempre calcolato.

Config.ActivateCooling = TRUE consente di attivare anche le uscite per il raffreddamento (OutputCool, OutputCool_PWM, OutputCool_PER). I valori positivi di uscita dell'algoritmo PID (PidOutputSum) vengono riportati in scala ed emessi sulle uscite per il riscaldamento. I valori negativi di uscita dell'algoritmo PID vengono riportati in scala ed emessi sulle uscite per il raffreddamento. Se debbano essere calcolati OutputCool_PWM o OutputCool_PER, viene stabilito con Config.Output.Cool.Select. OutputCool viene sempre calcolato.

Per il calcolo del valore di uscita PID con il raffreddamento attivo possono essere seguiti due procedimenti:

- Fattore di raffreddamento (Config.AdvancedCooling = FALSE):

Il calcolo del valore di uscita per il raffreddamento impiega i parametri PID di riscaldamento considerando il fattore di raffreddamento configurabile Config.CoolFactor. Questo metodo si addice quando l'organo attuatore del raffreddamento e quello del riscaldamento presentano lo stesso comportamento temporale ma guadagni diversi. Selezionando questo metodo l'ottimizzazione iniziale e l'ottimizzazione fine nonché il set di parametri PID per il raffreddamento non sono disponibili. Possono essere eseguite soltanto le ottimizzazioni per il riscaldamento.

- Commutazione parametri PID Config.AdvancedCooling = TRUE

Il calcolo del valore di uscita per il raffreddamento avviene da un set di parametri PID proprio. Sulla base del valore di uscita calcolato e della differenza di regolazione, l'algoritmo PID stabilisce se i parametri PID debbano essere impiegati per il riscaldamento o per il raffreddamento. Questo metodo si addice quando l'attuatore di raffreddamento e quello di riscaldamento presentano un comportamento temporale e un guadagno diversi. L'ottimizzazione iniziale e l'ottimizzazione fine per il raffreddamento sono disponibili soltanto con questo metodo.

Limiti e graduazione di scala del valore di uscita

Il valore di uscita PID (PidOutputSum) viene calcolato automaticamente, a seconda del modo di funzionamento, dall'algoritmo PID oppure predefinito dal valore manuale (ManualValue) o dal valore di uscita sostitutivo configurato (SubstituteOutput).

Il valore di uscita PID viene limitato in funzione della configurazione:

- Se il raffreddamento è disattivato (Config.ActivateCooling = FALSE), vale come limite superiore Config.Output.Heat.PidUpperLimit e come limite inferiore Config.Output.Heat.PidLowerLimit.
- Se il raffreddamento è attivato (Config.ActivateCooling = TRUE), vale come limite superiore Config.Output.Heat.PidUpperLimit e come limite inferiore Config.Output.Cool.PidLowerLimit.

Il valore di uscita PID viene riportato in scala ed emesso sulle uscite per il riscaldamento ed il raffreddamento. La graduazione di scala può essere predefinita separatamente per ciascuna uscita e stabilita rispettivamente da due coppie di valori nelle strutture Config.Output.Heat e Config.Output.Cool:

Uscita	Coppia di valori	Parametro
OutputHeat	Coppia di valori 1	Limite superiore valore di uscita PID (riscaldamento) Config.Output.Heat.PidUpperLimit, Valore superiore di uscita riportato in scala (riscaldamento) Config.Output.Heat.UpperScaling
	Coppia di valori 2	Limite inferiore valore di uscita PID (riscaldamento) Config.Output.Heat.PidLowerLimit, Valore inferiore di uscita riportato in scala (riscaldamento) Config.Output.Heat.LowerScaling
OutputHeat_PWM	Coppia di valori 1	Limite superiore valore di uscita PID (riscaldamento) Config.Output.Heat.PidUpperLimit, Valore superiore di uscita PWM riportato in scala (riscaldamento) Config.Output.Heat.PwmUpperScaling
	Coppia di valori 2	Limite inferiore valore di uscita PID (riscaldamento) Config.Output.Heat.PidLowerLimit, Valore inferiore di uscita PWM riportato in scala (riscaldamento) Config.Output.Heat.PwmLowerScaling
OutputHeat_PER	Coppia di valori 1	Limite superiore valore di uscita PID (riscaldamento) Config.Output.Heat.PidUpperLimit, Valore superiore analogico di uscita riportato in scala (riscaldamento) Config.Output.Heat.PerUpperScaling
	Coppia di valori 2	Limite inferiore valore di uscita PID (riscaldamento) Config.Output.Heat.PidLowerLimit, Valore inferiore analogico di uscita riportato in scala (riscaldamento) Config.Output.Heat.PerLowerScaling
OutputCool	Coppia di valori 1	Limite inferiore valore di uscita PID (raffreddamento) Config.Output.Cool.PidLowerLimit, Valore superiore di uscita riportato in scala (raffreddamento) Config.Output.Cool.UpperScaling
	Coppia di valori 2	Limite superiore valore di uscita PID (raffreddamento) Config.Output.Cool.PidUpperLimit, Valore inferiore di uscita riportato in scala (raffreddamento) Config.Output.Cool.LowerScaling
OutputCool_PWM	Coppia di valori 1	Limite inferiore valore di uscita PID (raffreddamento) Config.Output.Cool.PidLowerLimit, Valore superiore di uscita PWM riportato in scala (raffreddamento) Config.Output.Cool.PwmUpperScaling

Se è attivato il raffreddamento (Config.ActivateCooling = TRUE), Config.Output.Heat.PidLowerLimit deve avere il valore 0.0.

Config.Output.Cool.PidUpperLimit deve sempre avere il valore 0.0.

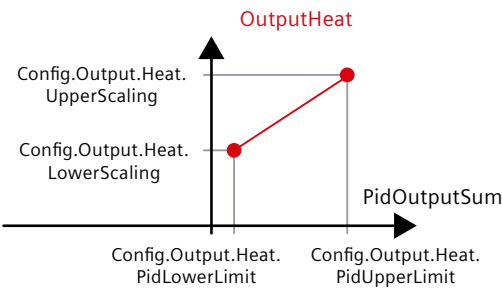
Uscita	Coppia di valori	Parametro
OutputCool_PWM	Coppia di valori 2	Limite superiore valore di uscita PID (raffreddamento) Config.Output.Cool.PidUpperLimit, Valore inferiore di uscita PWM riportato in scala (raffreddamento) Config.Output.Cool.PwmLowerScaling
OutputCool_PER	Coppia di valori 1	Limite inferiore valore di uscita PID (raffreddamento) Config.Output.Cool.PidLowerLimit, Valore superiore di uscita analogico riportato in scala (raffreddamento) Config.Output.Cool.PerUpperScaling
	Coppia di valori 2	Limite superiore valore di uscita PID (raffreddamento) Config.Output.Cool.PidUpperLimit, Valore inferiore di uscita analogico riportato in scala (raffreddamento) Config.Output.Cool.PerLowerScaling

Se è attivato il raffreddamento (Config.ActivateCooling = TRUE), Config.Output.Heat.PidLowerLimit deve avere il valore 0.0.

Config.Output.Cool.PidUpperLimit deve sempre avere il valore 0.0.

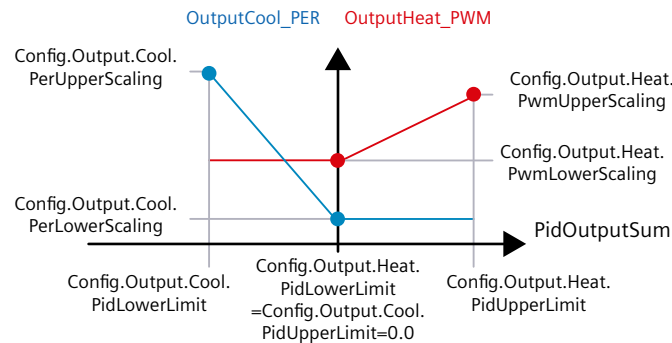
Esempio:

Graduazione di scala con impiego dell'uscita OutputHeat (raffreddamento disattivato; Config.Output.Heat.PidLowerLimit può essere diverso da 0.0):



Esempio:

Graduazione di scala con impiego delle uscite OutputHeat_PWM e OutputCool_PER (raffreddamento attivato; Config.Output.Heat.PidLowerLimit deve essere 0.0):



Ad eccezione del modo di funzionamento "Inattivo", il valore di un'uscita si colloca sempre tra il valore superiore e il valore inferiore di uscita riportati in scala, ad es. nel caso di OutputHeat,

il valore si trova sempre tra Config.Output.Heat.UpperScaling e Config.Output.Heat.LowerScaling.

In caso di limitazione del valore sulla rispettiva uscita, devono essere adeguati anche questi valori della graduazione di scala.

Regolazione in cascata

PID_Temp fornisce un supporto in caso di impiego nella regolazione in cascata (vedere: Creazione del programma [\(Pagina 203\)](#)).

Valore di uscita sostitutivo

PID_Temp in caso di errore può emettere un valore di uscita sostitutivo da predefinire nella variabile SubstituteOutput. Il valore di uscita sostitutivo deve essere compreso entro i limiti del valore di uscita PID. I valori sulle uscite per riscaldamento e raffreddamento, risultanti dal valore di uscita sostitutivo, derivano dalla graduazione di scala configurata dell'uscita.

Controllo della validità dei segnali

In caso di utilizzo, viene controllata la validità dei valori dei parametri seguenti:

- Setpoint
- SubstituteSetpoint
- Input
- Input_PER
- Disturbance
- ManualValue
- SubstituteOutput
- Parametri PID nelle strutture Retain.CtrlParams.Heat e Retain.CtrlParams.Cool.

Controllo del tempo di campionamento PID_Temp

Il tempo di campionamento corrisponde, in condizioni ideali, al tempo di ciclo dell'OB di schedulazione orologio richiamante. L'istruzione PID_Temp misura l'intervallo tra due richiami, il quale corrisponde al tempo di campionamento attuale. Ad ogni commutazione del modo di funzionamento e al primo avvio viene generato il valore medio dei primi 10 tempi di campionamento. Se il tempo di campionamento attuale si discosta troppo da questo valore medio si verifica un errore (Error = 0000800h).

Durante l'ottimizzazione si verifica questo errore se:

- Nuovo valore medio $\geq 1,1 \times$ valore medio precedente
- Nuovo valore medio $\leq 0,9 \times$ valore medio precedente

In funzionamento automatico si verifica questo errore se:

- Nuovo valore medio $\geq 1,5 \times$ valore medio precedente
- Nuovo valore medio $\leq 0,5 \times$ valore medio precedente

Disattivando il controllo del tempo di campionamento (CycleTime.EnMonitoring = FALSE) è possibile richiamare PID_Temp anche nell'OB1. A causa del tempo di campionamento variabile sarà però necessario accettare una qualità della regolazione più scarsa.

Tempo di campionamento dell'algoritmo PID

Poiché il sistema regolato richiede un certo tempo per reagire a una modifica del valore di uscita, è utile non calcolare questo valore in ogni ciclo. Il tempo di campionamento dell'algoritmo PID è il tempo che decorre tra due calcoli del valore di uscita. Questo tempo viene rilevato durante l'ottimizzazione e arrotondato ad un multiplo del tempo di ciclo dell'OB di schedulazione orologio (tempo di campionamento PID_Temp). Tutte le altre funzioni di PID_Temp vengono eseguite ad ogni richiamo.

Se il raffreddamento e la commutazione dei parametri PID sono attivati, PID_Temp impiega, per il riscaldamento ed il raffreddamento, rispettivamente un tempo di campionamento proprio dell'algoritmo PID. In tutte le altre configurazioni viene impiegato soltanto il tempo di campionamento dell'algoritmo PID per il riscaldamento.

In caso di impiego di OutputHeat_PWM e OutputCool_PWM, come durata periodo della modulazione ampiezza impulso viene utilizzato il tempo di campionamento dell'algoritmo PID. La precisione del segnale di uscita viene determinata dal rapporto tra tempo di campionamento dell'algoritmo PID e tempo di ciclo dell'OB. Il tempo di ciclo non dovrebbe superare un decimo del tempo di campionamento dell'algoritmo PID.

Se con l'impiego di OutputHeat_PWM o OutputCool_PWM, il tempo di campionamento dell'algoritmo PID e pertanto la durata periodo della modulazione ampiezza impulso, fossero particolarmente lunghi è possibile, per migliorare il livellamento del valore istantaneo, indicare, divergendo dai valori normalmente previsti, una durata periodo più breve nei parametri Config.Output.Heat.PwmPeriode e Config.Output.Cool.PwmPeriode

Senso di regolazione

PID_Temp può essere impiegato per applicazioni di riscaldamento oppure di riscaldamento/raffreddamento; in questi casi esso opera in modo fisso con normale senso di regolazione.

Incrementando il valore di uscita PID (PidOutputSum) il valore istantaneo dovrebbe venire a sua volta incremento. I valori sulle uscite per riscaldamento e raffreddamento, risultanti dal valore di uscita PID, derivano dalla graduazione di scala configurata dell'uscita.

L'inversione del senso di regolazione o il guadagno proporzionale negativo non vengono supportati.

Se l'applicazione necessita soltanto di un valore di uscita il cui incremento deve provocare una riduzione del valore istantaneo (ad es. regolazioni di flusso), è possibile impiegare PID_Compact con senso di regolazione invertito.

10.3.4.3 Parametri di ingresso di PID_Temp

I nomi dei seguenti parametri si applicano sia al blocco dati sia all'accesso tramite l'API Openness.

Parametro	Tipo di dati	Preassegnazione	Descrizione
Setpoint	REAL	0.0	Setpoint del regolatore PID nel funzionamento automatico Campo di valori ammesso: $\text{Config.SetpointUpperLimit} \geq \text{Setpoint} \geq \text{Config.SetpointLowerLimit}$ $\text{Config.InputUpperLimit} \geq \text{Setpoint} \geq \text{Config.InputLowerLimit}$
Input	REAL	0.0	Una variabile del programma utente viene utilizzata come sorgente del valore istantaneo. Per utilizzare il parametro Input è necessaria l'impostazione $\text{Config.InputPerOn} = \text{FALSE}$.
Input_PER	INT	0	Un ingresso analogico viene utilizzato come sorgente del valore istantaneo. Per utilizzare il parametro Input_PER è necessaria l'impostazione $\text{Config.InputPerOn} = \text{TRUE}$.
Disturbance	REAL	0.0	Grandezza di disturbo o Valore di precomando
ManualEnable	BOOL	FALSE	- Il fronte FALSE -> TRUE attiva il modo "Funzionamento manuale", State = 4, Mode rimane invariato. - Finché ManualEnable = TRUE non è possibile modificare il modo di funzionamento attraverso un fronte di salita in ModeActivate e utilizzare la finestra di messa in servizio. - Il fronte TRUE -> FALSE attiva il modo di funzionamento predefinito da Mode. Si consiglia di modificare il modo di funzionamento solo tramite Mode e ModeActivate.
ManualValue	REAL	0.0	Valore manuale Questo valore viene utilizzato nel funzionamento manuale come valore di uscita PID (PidOutputSum). I valori sulle uscite per riscaldamento e raffreddamento, risultanti da questo valore manuale, derivano dalla graduazione di scala configurata dell'uscita (struttura Config.Output.Heat e Config.Output.Cool). Per i regolatori con uscita di raffreddamento attiva ($\text{Config.ActivateCooling} = \text{TRUE}$) indicare: - un valore manuale positivo, per emettere il valore sulle uscite per il riscaldamento - un valore manuale negativo, per emettere il valore sulle uscite per il raffreddamento Il campo dei valori ammesso dipende dalla configurazione. - Uscita di raffreddamento disattivata ($\text{Config.ActivateCooling} = \text{FALSE}$): $\text{Config.Output.Heat.PidUpperLimit} \geq \text{ManualValue} \geq \text{Config.Output.Heat.PidLowerLimit}$ - Uscita di raffreddamento attivata ($\text{Config.ActivateCooling} = \text{TRUE}$): $\text{Config.Output.Heat.PidUpperLimit} \geq \text{ManualValue} \geq \text{Config.Output.Cool.PidLowerLimit}$

10.3 PID_Temp

Parametro	Tipo di dati	Preassegnazione	Descrizione
ErrorAck	BOOL	FALSE	Fronte FALSE -> TRUE ErrorBits e Warning vengono resettati.
Reset	BOOL	FALSE	Esegue un nuovo avvio del regolatore. - Fronte FALSE -> TRUE - Commutazione nel modo di funzionamento "Inattivo" - ErrorBits e Warning vengono resettati. - Fintantoché Reset = TRUE, - PID_Temp rimane nel modo di funzionamento "Inattivo" (State = 0). - il modo di funzionamento non può essere modificato tramite Mode e ModeActivate o ManualEnable - La finestra di dialogo per la messa in servizio non può essere utilizzata. - Fronte TRUE -> FALSE - Se ManualEnable = FALSE, PID_Temp commuta nel modo operativo salvato in Mode. - Se Mode = 3 (funzionamento automatico), il componente I viene trattato come configurato con la variabile IntegralResetMode.
ModeActivate	BOOL	FALSE	Fronte FALSE -> TRUE PID_Temp passa al modo di funzionamento indicato nell'ingresso Mode.

10.3.4.4 Parametri di uscita di PID_Temp

I nomi dei seguenti parametri si applicano sia al blocco dati sia all'accesso tramite l'API Openness.

Parametro	Tipo di dati	Preassegnazione	Descrizione
ScaledInput	REAL	0.0	Valore istantaneo riportato in scala
OutputHeat	REAL	0.0	Valore di uscita riscaldamento nel formato REAL Il Valore di uscita PID (PidOutputSum) viene riportato in scala dalle due coppie di valori Config.Output.Heat.PidUpperLimit, Config.Output.Heat.UpperScaling e Config.Output.Heat.PidLowerLimit, Config.Output.Heat.LowerScaling ed emesso nel formato REAL OutputHeat. viene sempre calcolato OutputHeat.
OutputCool	REAL	0.0	Valore di uscita raffreddamento nel formato REAL Il Valore di uscita PID (PidOutputSum) viene riportato in scala dalle due coppie di valori Config.Output.Cool.PidUpperLimit, Config.Output.Cool.LowerScaling e Config.Output.Cool.PidLowerLimit, Config.Output.Cool.UpperScaling ed emesso nel formato REAL OutputCool. OutputCool viene calcolato soltanto se l'Uscita di raffreddamento è attivata (Config.ActivateCooling = TRUE).
OutputHeat_PER	INT	0	Valore di uscita analogico riscaldamento Il Valore di uscita PID (PidOutputSum) viene riportato in scala dalle due coppie di valori Config.Output.Heat.PidUpperLimit, Config.Output.Heat.PerUpperScaling e Config.Output.Heat.PidLowerLimit, Config.Output.Heat.PerLowerScaling ed emesso come valore analogico in OutputHeat_PER. OutputHeat_PER viene calcolato soltanto se Config.Output.Heat.Select = 2.

Parametro	Tipo di dati	Preassegnazione	Descrizione
OutputCool_PER	INT	0	Valore di uscita analogico raffreddamento Il Valore di uscita PID (PidOutputSum) viene riportato in scala dalle due coppie di valori Config.Output.Cool.PidUpperLimit, Config.Output.Cool.PerLowerScaling e Config.Output.Cool.PidLowerLimit, Config.Output.Cool.PerUpperScaling ed emesso come valore analogico in OutputCool_PER. OutputCool_PER viene calcolato soltanto se l'Uscita di raffreddamento è attivata (Config.ActivateCooling = TRUE) e Config.Output.Cool.Select = 2.
OutputHeat_PWM	BOOL	FALSE	Valore di uscita con modulazione ad ampiezza d'impulso per il riscaldamento Il Valore di uscita PID (PidOutputSum) viene riportato in scala dalle due coppie di valori Config.Output.Heat.PidUpperLimit, Config.Output.Heat.PwmUpperScaling e Config.Output.Heat.PidLowerLimit, Config.Output.Heat.PwmLowerScaling ed emesso come valore con modulazione ampiezza impulso (tempi di attivazione e disattivazione variabili) in OutputHeat_PWM. OutputHeat_PWM viene calcolato soltanto se Config.Output.Heat.Select = 1.
OutputCool_PWM	BOOL	FALSE	Valore di uscita con modulazione ad ampiezza d'impulso per il raffreddamento Il Valore di uscita PID (PidOutputSum) viene riportato in scala dalle due coppie di valori Config.Output.Cool.PidUpperLimit, Config.Output.Cool.PwmLowerScaling e Config.Output.Cool.PidLowerLimit, Config.Output.Cool.PwmUpperScaling ed emesso come valore con modulazione ampiezza impulso (tempi di attivazione e disattivazione variabili) in OutputCool_PWM. OutputCool_PWM viene calcolato soltanto se l'uscita di raffreddamento è attivata (Config.ActivateCooling = TRUE) e Config.Output.Cool.Select = 1.
SetpointLimit_H	BOOL	FALSE	Se SetpointLimit_H = TRUE, è stato raggiunto il limite superiore assoluto del setpoint ($\text{Setpoint} \geq \text{Config.SetpointUpperLimit}$ o $\text{Setpoint} \geq \text{Config.InputUpperLimit}$). Il setpoint viene limitato verso l'alto su un valore minimo costituito da Config.SetpointUpperLimit e Config.InputUpperLimit.
SetpointLimit_L	BOOL	FALSE	Se SetpointLimit_L = TRUE, è stato raggiunto il limite inferiore assoluto del setpoint ($\text{Setpoint} \leq \text{Config.SetpointLowerLimit}$ o $\text{Setpoint} \leq \text{Config.InputLowerLimit}$). Il setpoint viene limitato verso il basso su un valore massimo costituito da Config.SetpointLowerLimit e Config.InputLowerLimit.
InputWarning_H	BOOL	FALSE	Se InputWarning_H = TRUE il limite superiore di avviso del valore istantaneo è stato raggiunto o superato ($\text{ScaledInput} \geq \text{Config.InputUpperWarning}$).
InputWarning_L	BOOL	FALSE	Se InputWarning_L = TRUE il limite inferiore di avviso del valore istantaneo è stato raggiunto o superato verso il basso ($\text{ScaledInput} \leq \text{Config.InputLowerWarning}$).
State	INT	0	Il parametro Parametri State e Mode PID_Temp (Pagina 420) indica il modo di funzionamento attuale del regolatore PID. Il modo di funzionamento si modifica con il parametro di ingresso Mode e con un fronte di salita in ModeActivate. Per l'Ottimizzazione iniziale e l'Ottimizzazione fine definire tramite Heat.EnableTuning e Cool.EnableTuning se l'ottimizzazione debba essere eseguita per il riscaldamento o per il raffreddamento. • State = 0: Inattivo • State = 1: Ottimizzazione iniziale • State = 2: Ottimizzazione fine • State = 3: Funzionamento automatico • State = 4: Funzionamento manuale • State = 5: Valore di uscita sostitutivo con controllo errori

10.3 PID_Temp

Parametro	Tipo di dati	Preassegnazione	Descrizione
Error	BOOL	FALSE	Se Error = TRUE in questo ciclo è presente almeno un messaggio di errore.
ErrorBits	DWORD	DW#16#0	Parametri ErrorBits PID_Temp (Pagina 427) indica i messaggi di errore presenti. ErrorBits è a ritenzione e viene resettato al verificarsi di un fronte di salita in Reset o ErrorAck.

10.3.4.5 Parametri di transito di PID_Temp

I nomi dei seguenti parametri si applicano sia al blocco dati sia all'accesso tramite l'API Openness.

Parametro	Tipo di dati	Preassegnazione	Descrizione
Mode	INT	4	In Mode si predefinisce il modo di funzionamento al quale deve passare PID_Temp. Sono possibili: - Mode = 0: Inattivo - Mode = 1: Ottimizzazione iniziale - Mode = 2: Ottimizzazione fine - Mode = 3: Funzionamento automatico - Mode = 4: Funzionamento manuale Il modo di funzionamento viene attivato da: - Fronte di salita in ModeActivate - Fronte di discesa in Reset - Fronte di discesa in ManualEnable - Avviamento a freddo della CPU se RunModeByStartup = TRUE Per l'Ottimizzazione iniziale e l'Ottimizzazione fine definire tramite Heat.EnableTuning e Cool.EnableTuning se l'ottimizzazione debba essere eseguita per il riscaldamento o per il raffreddamento. Mode è a ritenzione. I modi di funzionamento sono descritti dettagliatamente nella sezione Parametri State e Mode (Pagina 420).
Master	DWORD	DW#16#0	Interfaccia per la regolazione in cascata Se l'istanza PID_Temp viene utilizzata come regolatore slave in una cascata (Config.Cascade.IsSlave = TRUE), assegnare al parametro Master, nel richiamo dell'istruzione, il parametro Slave del regolatore master. Esempio: Richiamo di un regolatore slave "PID_Temp_2" con il regolatore master "PID_Temp_1" in SCL: <pre> "PID_Temp_2"(Master := "PID_Temp_1".Slave, Setpoint := "PID_Temp_1".OutputHeat); </pre> Da quest'interfaccia i regolatori slave scambiano informazioni con il regolatore master sul modo di funzionamento, la limitazione e il setpoint sostitutivo. Tenere presente che il richiamo del regolatore master deve avvenire prima del richiamo del regolatore slave e nello stesso OB di schedulazione orologio. Assegnazione: - Bit da 0 a 15: non configurato - Bit da 16 a 23 – Contatore di limitazione: Un regolatore slave il cui valore di uscita si trovi nei limiti, incrementa questo contatore. Il regolatore master reagisce in funzione del numero di slave (Config.Cascade.CountSlaves) configurati e del modo anti windup (Config.Cascade.AntiWindUpMode) impostato.

Parametro	Tipo di dati	Preassegnazione	Descrizione
			- Bit 24 – Funzionamento automatico del regolatore slave: TRUE, se tutti i regolatori slave si trovano nel Funzionamento automatico - Bit 25 – Setpoint sostitutivo del regolatore slave: TRUE, se un regolatore slave ha attivato il setpoint sostitutivo (SubstituteSetpointOn = TRUE)
Slave	DWORD	DW#16#0	Interfaccia per la regolazione in cascata Da quest'interfaccia i regolatori slave scambiano informazioni con il regolatore master sul modo di funzionamento, la limitazione e il setpoint sostitutivo. Vedere la descrizione dei parametri del master

Vedere anche

[Parametri State e Mode PID_Temp \(Pagina 420\)](#)

[Creazione del programma \(Pagina 203\)](#)

[Regolazione in cascata con PID_Temp \(Pagina 201\)](#)

10.3.4.6 Variabili statiche di PID_Temp

NOTA

Per evitare un comportamento errato del regolatore PID modificare le variabili contrassegnate con ⁽¹⁾ solo nel modo di funzionamento "Inattivo".

I nomi delle seguenti variabili si applicano sia al blocco dati sia all'accesso tramite l'API Openness.

Variabile	Tipo di dati	Preassegnazione	Descrizione
IntegralResetMode	Int	V1.0: 1, dalla V1.1: 4	La variabile Variabile IntegralResetMode (Pagina 435) stabilisce i valori da preassegnare al componente I PIDCtrl.IOutputOld alla commutazione del modo di funzionamento da "Inattivo" a "Funzionamento automatico". Questa impostazione è attiva solo per un ciclo. - IntegralResetMode = 0: Livella - IntegralResetMode = 1: Elimina - IntegralResetMode = 2: Mantiene - IntegralResetMode = 3: Preassegna - IntegralResetMode = 4: in caso di variazioni del setpoint (soltanto per PID_Temp con la versione ≥ 1.1)
OverwriteInitialOutputValue	REAL	0.0	Se una delle seguenti condizioni è soddisfatta, al componente integrale PIDCtrl.IOutputOld viene preassegnato automaticamente un valore tale da far supporre che nel ciclo precedente PIDOutputSum = OverwriteInitialOutputValue ci fosse: - IntegralResetMode = 3 alla commutazione dal modo di funzionamento "Inattivo" al modo "Funzionamento automatico" - IntegralResetMode = 3, Fronte TRUE -> FALSE sul parametro Reset e sul parametro Mode = 3 - PIDCtrl.PIDInit = TRUE nel "Funzionamento automatico" (disponibile dalla PID_Temp versione V. 1.1)

Variabile	Tipo di dati	Preassegnazione	Descrizione
RunModeByStartup	BOOL	TRUE	Dopo il nuovo avviamento della CPU attiva il modo di funzionamento in Mode - Se RunModeByStartup = TRUE, dopo l'avvio della CPU PID_Temp si avvia nel modo di funzionamento salvato in Mode. - Se RunModeByStartup = FALSE, dopo l'avvio della CPU PID_Temp resta nel modo di funzionamento "Inattivo".
LoadBackUp	BOOL	FALSE	Se LoadBackUp = TRUE, viene ricaricato l'ultimo set di parametri PID dalla struttura CtrlParamsBackUp. Il set è stato memorizzato prima dell'ultima ottimizzazione. LoadBackUp viene reimpostato automaticamente su FALSE. L'acquisizione avviene bumpless.
SetSubstituteOutput	BOOL	TRUE	Selezione del valore di uscita, non appena si presenta un errore (State = 5): - Se SetSubstituteOutput = TRUE e ActivateRecoverMode = TRUE, finché è presente un errore, viene emesso, come valore di uscita PID, il valore di uscita sostitutivo configurato SubstituteOutput. - Se SetSubstituteOutput = FALSE e ActivateRecoverMode = TRUE, l'organo attuatore rimane, finché è presente un errore, sul valore di uscita PID attuale. - Se ActivateRecoverMode = FALSE, SetSubstituteOutput non è effettivo. - Se SubstituteOutput non è valido (ErrorBits = 0020000h), il valore di uscita sostitutivo non può essere emesso. In questo caso il limite inferiore del valore di uscita PID per il riscaldamento (Config.Output.Heat.PidLowerLimit) viene emesso come valore di uscita PID.
PhysicalUnit	INT	0	Unità fisica del valore istantaneo e del setpoint, ad es. °C o °F. PhysicalUnit viene utilizzato per la visualizzazione negli editor e non incide in alcun modo sul comportamento dell'algoritmo di regolazione nella CPU. Con l'importazione di PID_Temp tramite l'API Openness PhysicalUnit viene ripristinato il valore di default.
PhysicalQuantity	INT	0	Grandezza fisica del valore istantaneo e del setpoint, ad es. la temperatura. PhysicalQuantity viene utilizzato per la visualizzazione negli editor e non incide in alcun modo sul comportamento dell'algoritmo di regolazione nella CPU. Con l'importazione di PID_Temp tramite l'API Openness PhysicalQuantity viene ripristinato il valore di default.
ActivateRecoverMode	BOOL	TRUE	La variabile ActivateRecoverMode determina il comportamento in caso di errore.
Warning	DWORD	0	La variabile Warning mostra gli avvisi da Reset = TRUE o ErrorAck = TRUE. Warning è a ritenzione.
Progress	REAL	0.0	Avanzamento dell'ottimizzazione attuale in percentuale (0.0 - 100.0)
CurrentSetpoint	REAL	0.0	CurrentSetpoint visualizza sempre il setpoint attuale attivo. Questo valore viene congelato durante l'ottimizzazione.

Variabile	Tipo di dati	Preassegnazione	Descrizione
CancelTuningLevel	REAL	10.0	Oscillazione del setpoint ammessa durante l'ottimizzazione. L'ottimizzazione viene interrotta solo se: - Setpoint > CurrentSetpoint + CancelTuningLevel oppure - Setpoint < CurrentSetpoint - CancelTuningLevel
SubstituteOutput	REAL	0.0	Il valore di uscita sostitutivo viene impiegato come valore di uscita PID fintantoché sono soddisfatte le seguenti condizioni: - Nel funzionamento automatico sono presenti uno o più errori nei quali è attivo ActivateRecoverMode - SetSubstituteOutput = TRUE - ActivateRecoverMode = TRUE I valori sulle uscite per riscaldamento e raffreddamento, risultanti dal valore di uscita sostitutivo, derivano dalla graduazione di scala configurata dell'uscita (strutture Config.Output.Heat e Config.Output.Cool). Per i regolatori con uscita di raffreddamento attiva (Config.ActivateCooling = TRUE) indicare: - un valore di uscita sostitutivo positivo, per emettere il valore sulle uscite per il riscaldamento - un valore di uscita sostitutivo negativo, per emettere il valore sulle uscite per il raffreddamento Il campo dei valori ammesso dipende dalla configurazione. - Uscita di raffreddamento disattivata (Config.ActivateCooling = FALSE): Config.Output.Heat.PidUpperLimit ≥ SubstituteOutput ≥ Config.Output.Heat.PidLowerLimit - Uscita di raffreddamento attivata (Config.ActivateCooling = TRUE): Config.Output.Heat.PidUpperLimit ≥ SubstituteOutput ≥ Config.Output.Cool.PidLowerLimit
PidOutputSum	REAL	0.0	Valore di uscita PID PidOutputSum visualizza il valore di uscita dell'algoritmo PID. A seconda del modo di funzionamento, questo valore viene calcolato automaticamente oppure predefinito dal valore manuale o dal valore di uscita sostitutivo configurato. I valori sulle uscite per riscaldamento e raffreddamento, risultanti dal valore di uscita PID, derivano dalla graduazione di scala configurata dell'uscita (strutture Config.Output.Heat e Config.Output.Cool). PidOutputSum viene limitato in funzione della configurazione. - Uscita di raffreddamento disattivata (Config.ActivateCooling = FALSE): Config.Output.Heat.PidUpperLimit ≥ PidOutputSum ≥ Config.Output.Heat.PidLowerLimit - Uscita di raffreddamento attivata (Config.ActivateCooling = TRUE): Config.Output.Heat.PidUpperLimit ≥ PidOutputSum ≥ Config.Output.Cool.PidLowerLimit

Variabile	Tipo di dati	Preassegnazione	Descrizione
PidOutputOffsetHeat	REAL	0.0	Offset del valore di uscita PID per il riscaldamento PidOutputOffsetHeat viene addizionato al valore risultante da PidOutputSum per il ramo di riscaldamento. Predefinire un valore positivo per PidOutputOffsetHeat al fine di ottenere un offset positivo sulle uscite per il riscaldamento. I valori risultanti sulle uscite per il riscaldamento derivano dalla graduazione di scala configurata dell'uscita (struttura Config.Output.Heat). Questo offset può essere impiegato per organi attuatore che richiedono un valore minimo fisso, ad es. ventole con un numero di giri minimo.
PidOutputOffsetCool	REAL	0.0	Offset del valore di uscita PID per il raffreddamento PidOutputOffsetCool viene addizionato al valore risultante da PidOutputSum per il ramo di raffreddamento. Predefinire un valore negativo per PidOutputOffsetCool al fine di ottenere un offset positivo sulle uscite per il raffreddamento. I valori risultanti sulle uscite per il raffreddamento derivano dalla graduazione di scala configurata dell'uscita (struttura Config.Output.Cool). Questo offset può essere impiegato per organi attuatore che richiedono un valore minimo fisso, ad es. ventole con un numero di giri minimo.
SubstituteSetpointOn	BOOL	FALSE	Attiva il Setpoint sostitutivo come setpoint del regolatore. - FALSE = viene utilizzato il parametro Setpoint. - TRUE = il parametro SubstituteSetpoint viene utilizzato come setpoint SubstituteSetpointOn può essere utilizzato per la predefinizione diretta del setpoint di un regolatore slave in una cascata senza dover modificare il programma utente.
SubstituteSetpoint	REAL	0.0	Setpoint sostitutivo Se SubstituteSetpointOn = TRUE, il setpoint sostitutivo SubstituteSetpoint viene utilizzato come setpoint. Campo di valori ammesso: Config.SetpointUpperLimit $\geq$ SubstituteSetpoint $\geq$ Config.SetpointLowerLimit, Config.InputUpperLimit $\geq$ SubstituteSetpoint $\geq$ Config.InputLowerLimit
DisableCooling	BOOL	FALSE	DisableCooling = TRUE disattiva il ramo di raffreddamento per il regolatore di riscaldamento e raffreddamento (Config.ActivateCooling = TRUE) nel Funzionamento automatico circoscrivendo PidOutputSum su 0.0 come limite inferiore. PidOutputOffsetCool e la graduazione di scala delle uscite per il raffreddamento rimangono attivi. DisableCooling può essere utilizzato per l'ottimizzazione di applicazioni multizona per disattivare provvisoriamente il ramo di raffreddamento finché l'ottimizzazione non si è conclusa in tutti i regolatori. Questo parametro viene impiegato/resettato manualmente dall'utente e non viene resettato automaticamente dall'istruzione PID_Temp.

Variabile	Tipo di dati	Preassegnazione	Descrizione
AllSlaveAutomaticState	BOOL	FALSE	Se l'istanza PID_Temp viene utilizzata come regolatore master di una cascata (Config.Cascade.IsMaster = TRUE), AllSlaveAutomaticState = TRUE indica che tutti i regolatori slave si trovano nel funzionamento automatico. L'ottimizzazione, il funzionamento manuale o automatico del regolatore master possono essere eseguiti correttamente soltanto se tutti i regolatori slave si trovano nel funzionamento automatico. AllSlaveAutomaticState viene rilevato soltanto se i regolatori master e slave vengono interconnessi tramite i parametri master e slave. Per dettagli vedere i parametri master.
NoSlaveSubstituteSetpoint	BOOL	FALSE	Se l'istanza PID_Temp viene utilizzata come regolatore master di una cascata (Config.Cascade.IsMaster = TRUE), NoSlaveSubstituteSetpoint = TRUE indica che nessun regolatore slave ha attivato il setpoint sostitutivo. L'ottimizzazione, il funzionamento manuale o automatico del regolatore master possono essere eseguiti correttamente soltanto se nessun regolatore slave ha attivato il setpoint sostitutivo. NoSlaveSubstituteSetpoint viene rilevato soltanto se i regolatori master e slave vengono interconnessi tramite i parametri master e slave. Per dettagli vedere i parametri master.
Heat.EnableTuning	BOOL	TRUE	Abilitazione dell'ottimizzazione per il riscaldamento Heat.EnableTuning deve essere impostato per le seguenti ottimizzazioni (contemporaneamente o prima dell'avvio con Mode e ModeActivate): • Ottimizzazione iniziale riscaldamento • Ottimizzazione iniziale riscaldamento e raffreddamento • Ottimizzazione fine riscaldamento Questo parametro non viene resettato automaticamente dall'istruzione PID_Temp.
Cool.EnableTuning	BOOL	FALSE	Abilitazione dell'ottimizzazione per il raffreddamento Cool.EnableTuning deve essere impostato per le seguenti ottimizzazioni (contemporaneamente o prima dell'avvio con Mode e ModeActivate): • Ottimizzazione iniziale raffreddamento • Ottimizzazione iniziale riscaldamento e raffreddamento • Ottimizzazione fine raffreddamento Attivo soltanto se l'uscita di raffreddamento e la commutazione dei parametri PID sono attivi ("Config.ActivateCooling" = TRUE e "Config.AdvancedCooling" = TRUE). Questo parametro non viene resettato automaticamente dall'istruzione PID_Temp.
Config.InputPerOn ⁽¹⁾	BOOL	TRUE	Se InputPerOn = TRUE, viene utilizzato il parametro Input_PER per la rilevazione del valore istantaneo. Se InputPerOn = FALSE, viene utilizzato il parametro Input.

Variabile	Tipo di dati	Preassegnazione	Descrizione
Config.InputUpperLimit ⁽¹⁾	REAL	120.0	Limite superiore del valore istantaneo Input e Input_PER vengono controllati per verificare il rispetto di tale limite. Se il limite viene superato, viene emesso un messaggio di errore; la reazione a quest'errore è in funzione di ActivateRecoverMode. Nell'ingresso di periferia il valore istantaneo può superare al massimo del 18% il campo nominale (campo di sovracomando). Con l'impiego dell'ingresso di periferia con preimpostazione del valore limite e della graduazione di scala del valore istantaneo, il superamento del limite non è pertanto possibile. All'avvio della prima ottimizzazione, sulla base della differenza tra limite superiore e limite inferiore del valore istantaneo, viene eseguito un controllo volto ad appurare se la distanza tra setpoint e valore istantaneo soddisfa i requisiti richiesti. InputUpperLimit > InputLowerLimit
Config.InputLowerLimit ⁽¹⁾	REAL	0.0	Limite inferiore del valore istantaneo Input e Input_PER vengono controllati per verificare il rispetto di tale limite. Se il limite viene superato verso il basso, viene emesso un messaggio di errore; la reazione a quest'errore è in funzione di ActivateRecoverMode. InputLowerLimit < InputUpperLimit
Config.InputUpperWarning ⁽¹⁾	REAL	3.402822e+38	Limite superiore di avviso del valore istantaneo Input e Input_PER vengono controllati per verificare il rispetto di tale limite. Se il limite viene superato verso l'alto, nel parametro Warning viene emesso un avviso. - Se si configura InputUpperWarning al di fuori dei limiti del valore istantaneo, viene utilizzato il limite superiore assoluto configurato per il valore istantaneo come limite superiore di avviso. - Se si configura InputUpperWarning entro i limiti del valore istantaneo questo valore viene utilizzato come limite superiore di avviso. InputUpperWarning > InputLowerWarning
Config.InputLowerWarning ⁽¹⁾	REAL	-3.402822e+38	Limite inferiore di avviso del valore istantaneo Input e Input_PER vengono controllati per verificare il rispetto di tale limite. Se il limite viene superato verso il basso, nel parametro Warning viene emesso un avviso. - Se si configura InputLowerWarning al di fuori dei limiti del valore istantaneo, il limite inferiore assoluto configurato per il valore istantaneo viene utilizzato come limite inferiore di avviso. - Se si configura InputLowerWarning entro i limiti del valore istantaneo questo valore viene utilizzato come limite inferiore di avviso. InputLowerWarning < InputUpperWarning
Config.SetpointUpperLimit ⁽¹⁾	REAL	3.402822e+38	Limite superiore del setpoint

Variabile	Tipo di dati	Preassegnazione	Descrizione
			Setpoint e SubstituteSetpoint vengono controllati per verificare il rispetto di tale limite. Se il limite viene superato verso l'alto, nel parametro Warning viene emesso un avviso. - Se si configura SetpointUpperLimit al di fuori dei limiti del valore istantaneo, il limite superiore assoluto configurato per il valore istantaneo viene utilizzato come limite superiore del setpoint. - Se si configura SetpointUpperLimit entro i limiti del valore istantaneo questo valore viene utilizzato come limite superiore del setpoint. SetpointUpperLimit > SetpointLowerLimit
Config.SetpointLowerLimit ⁽¹⁾	REAL	-3.402822e+38	Limite inferiore del setpoint Setpoint e SubstituteSetpoint vengono controllati per verificare il rispetto di tale limite. Se il limite viene superato verso il basso, nel parametro Warning viene emesso un avviso. - Se si configura SetpointLowerLimit al di fuori dei limiti del valore istantaneo, il limite inferiore assoluto configurato per il valore istantaneo viene utilizzato come limite inferiore del setpoint. - Se si configura SetpointLowerLimit entro i limiti del valore istantaneo questo valore viene utilizzato come limite inferiore del setpoint. SetpointLowerLimit < SetpointUpperLimit
Config.ActivateCooling ⁽¹⁾	BOOL	FALSE	Uscita di raffreddamento attivata - Config.ActivateCooling = FALSE Vengono utilizzate soltanto le uscite per il riscaldamento. - Config.ActivateCooling = TRUE Vengono utilizzate le uscite per il riscaldamento ed il raffreddamento. In caso di impiego dell'uscita di raffreddamento, il regolatore non deve essere configurato come regolatore master (Config.Cascade.IsMaster deve essere FALSE) .
Config.AdvancedCooling ⁽¹⁾	BOOL	TRUE	Metodo per riscaldamento/raffreddamento - Fattore di raffreddamento (Config.AdvancedCooling = FALSE) Il calcolo del valore di uscita per il raffreddamento impiega i parametri PID di riscaldamento (struttura Retain.CtrlParams.Heat) considerando il fattore di raffreddamento configurabile Config.CoolFactor. Questo metodo si addice quando l'organo attuatore del raffreddamento e quello del riscaldamento presentano lo stesso comportamento temporale ma guadagni diversi. L'ottimizzazione iniziale e l'ottimizzazione fine per il raffreddamento con questo metodo non sono disponibili. Possono essere eseguite soltanto le ottimizzazioni per il riscaldamento. - Commutazione parametri PID (Config.AdvancedCooling = TRUE) Il calcolo del valore di uscita per il raffreddamento avviene con un set di parametri PID proprio (struttura Retain.CtrlParams.Cool). Questo metodo si addice quando l'attuatore di raffreddamento e quello di riscaldamento presentano un comportamento temporale e un guadagno diversi.

Variabile	Tipo di dati	Preassegnazione	Descrizione
			L'ottimizzazione iniziale e l'ottimizzazione fine per il raffreddamento sono disponibili soltanto con questo metodo (Mode = 1 o 2, Cool.EnableTuning = TRUE). Config.AdvancedCooling è attiva soltanto se l'uscita di raffreddamento è attivata (Config.ActivateCooling = TRUE).
Config.CoolFactor ⁽¹⁾	REAL	1.0	Fattore di raffreddamento Se Config.AdvancedCooling = FALSE, Config.CoolFactor viene considerato come fattore nel calcolo del valore di uscita per il raffreddamento. In questo modo, per l'organo attuatore di riscaldamento e di raffreddamento, possono essere considerati guadagni diversi. Config.CoolFactor non vengono impostati automaticamente o adeguati durante l'ottimizzazione. Config.CoolFactor deve essere configurato manualmente e in modo corretto con il rapporto "Guadagno dell'organo attuatore del riscaldamento / Guadagno dell'organo attuatore del raffreddamento" Esempio: Config.CoolFactor = 2.0 sta ad indicare che il guadagno dell'organo attuatore del riscaldamento è il doppio di quello dell'organo attuatore del raffreddamento. Config.CoolFactor è efficace soltanto se l'uscita di raffreddamento è attivata (Config.ActivateCooling = TRUE) e se il fattore di raffreddamento è stato selezionato come metodo di riscaldamento/raffreddamento (Config.AdvancedCooling = FALSE). Config.CoolFactor > 0.0
Config.InputScaling.UpperPointIn ⁽¹⁾	REAL	27648.0	Scala Input_PER superiore Sulla base delle due coppie di valori UpperPointOut, UpperPointIn e LowerPointOut, LowerPointIn Input_PER viene riportato in scala. Efficace soltanto se Input_PER viene utilizzato per la rilevazione del valore istantaneo (Config.InputPerOn = TRUE). UpperPointIn > LowerPointIn
Config.InputScaling.LowerPointIn ⁽¹⁾	REAL	0.0	Scala Input_PER inferiore Sulla base delle due coppie di valori UpperPointOut, UpperPointIn e LowerPointOut, LowerPointIn Input_PER viene riportato in scala. Efficace soltanto se Input_PER viene utilizzato per la rilevazione del valore istantaneo (Config.InputPerOn = TRUE). LowerPointIn < UpperPointIn
Config.InputScaling.UpperPointOut ⁽¹⁾	REAL	100.0	Valore istantaneo superiore riportato in scala Sulla base delle due coppie di valori UpperPointOut, UpperPointIn e LowerPointOut, LowerPointIn Input_PER viene riportato in scala. Efficace soltanto se Input_PER viene utilizzato per la rilevazione del valore istantaneo (Config.InputPerOn = TRUE). UpperPointOut > LowerPointOut
Config.InputScaling.LowerPointOut ⁽¹⁾	REAL	0.0	Valore istantaneo inferiore riportato in scala Sulla base delle due coppie di valori UpperPointOut, UpperPointIn e LowerPointOut, LowerPointIn Input_PER viene riportato in scala. Efficace soltanto se Input_PER viene utilizzato per la rilevazione del valore istantaneo (Config.InputPerOn = TRUE). LowerPointOut < UpperPointOut

Variabile	Tipo di dati	Preassegnazione	Descrizione
Config.Output.Heat.Select ⁽¹⁾	INT	1	Selezione del valore di uscita per il riscaldamento Config.Output.Heat.Select indica le uscite utilizzate per il riscaldamento: - Heat.Select = 0 - OutputHeat non utilizzato - vengono utilizzati Heat.Select = 1 - OutputHeat e OutputHeat_PWM - vengono utilizzati Heat.Select = 2 -OutputHeat e OutputHeat_PER Le uscite non utilizzate non vengono calcolate e rimangono sul valore di default.
Config.Output.Heat.PwmPeriode ⁽¹⁾	REAL	0.0	Durata periodo della Modulazione ampiezza impulso (PWM) per il riscaldamento (uscita OutputHeat_PWM) in secondi: - Heat.PwmPeriode = 0.0 Il tempo di campionamento dell'algoritmo PID per il riscaldamento (Retain.CtrlParams.Heat.Cycle) viene utilizzato come durata periodo di PWM. - Heat.PwmPeriode > 0.0 Il valore viene arrotondato ad un multiplo intero del tempo di campionamento PID_Temp (CycleTime.Value) e utilizzato come durata periodo di PWM. Con un tempo di campionamento piuttosto lungo dell'algoritmo PID, quest'impostazione consente di migliorare il livellamento del valore istantaneo. Il valore deve soddisfare le seguenti condizioni: - Heat.PwmPeriode ≤ Retain.CtrlParams.Heat.Cycle, - Heat.PwmPeriode > Config.Output.Heat.MinimumOnTime - Heat.PwmPeriode > Config.Output.Heat.MinimumOffTime
Config.Output.Heat.PidUpperLimit ⁽¹⁾	REAL	100.0	Limite superiore del valore di uscita PID per il riscaldamento Il valore di uscita PID (PidOutputSum) viene circoscritto su questo limite superiore. Heat.PidUpperLimit costituisce, insieme ai seguenti parametri, una coppia di valori per la graduazione di scala del valore di uscita PID (PidOutputSum) sulle uscite per il riscaldamento: - Heat.UpperScaling per OutputHeat - Heat.PwmUpperScaling per OutputHeat_PWM - Heat.PerUpperScaling per OutputHeat_PER In caso di limitazione del valore sulla rispettiva uscita, devono essere adeguati anche questi valori di scala. Heat.PidUpperLimit > Heat.PidLowerLimit
Config.Output.Heat.PidLowerLimit ⁽¹⁾	REAL	0.0	Limite inferiore del valore di uscita PID per il riscaldamento Nei regolatori nei quali l'uscita di raffreddamento è disattivata (Config.ActivateCooling = FALSE), il valore di uscita PID (PidOutputSum) viene circoscritto su questo limite inferiore. Nei regolatori con uscita di raffreddamento attiva (Config.ActivateCooling = TRUE), il valore deve essere 0.0. Heat.PidLowerLimit costituisce, insieme ai seguenti parametri, una coppia di valori per la graduazione di scala del valore di uscita PID (PidOutputSum) sulle uscite per il riscaldamento: - Heat.LowerScaling per OutputHeat - Heat.PwmLowerScaling per OutputHeat_PWM - Heat.PerLowerScaling per OutputHeat_PER

Variabile	Tipo di dati	Preassegnazione	Descrizione
			In caso di limitazione del valore sulla rispettiva uscita, devono essere adeguati anche questi valori di scala. Il campo dei valori ammesso dipende dalla configurazione. - Uscita di raffreddamento disattivata (Config.ActivateCooling = FALSE): Heat.PidLowerLimit < Heat.PidUpperLimit - Uscita di raffreddamento attivata (Config.ActivateCooling = TRUE): Heat.PidLowerLimit = 0.0
Config.Output.Heat.UpperScaling ⁽¹⁾	REAL	100.0	Valore di uscita superiore riportato in scala per il riscaldamento Heat.UpperScaling e Heat.PidUpperLimit formano una coppia di valori per la graduazione di scala del valore di uscita PID (PidOutputSum) sul valore di uscita per il riscaldamento (OutputHeat). Il valore di OutputHeat si colloca sempre tra Heat.UpperScaling e Heat.LowerScaling. Heat.UpperScaling ≠ Heat.LowerScaling
Config.Output.Heat.LowerScaling ⁽¹⁾	REAL	0.0	Valore di uscita inferiore riportato in scala per il riscaldamento Heat.LowerScaling e Heat.PidLowerLimit formano una coppia di valori per la graduazione di scala del valore di uscita PID (PidOutputSum) sul valore di uscita per il riscaldamento (OutputHeat). Il valore di OutputHeat si colloca sempre tra Heat.UpperScaling e Heat.LowerScaling. Heat.UpperScaling ≠ Heat.LowerScaling
Config.Output.Heat.PwmUpperScaling ⁽¹⁾	REAL	100.0	Valore superiore di uscita PWM riportato in scala per il riscaldamento Heat.PwmUpperScaling e Heat.PidUpperLimit formano una coppia di valori per la graduazione di scala del valore di uscita PID (PidOutputSum) sul valore di uscita con modulazione ampiezza impulsi per il riscaldamento (OutputHeat_PWM). Il valore di OutputHeat_PWM si colloca sempre tra Heat.PwmUpperScaling e Heat.PwmLowerScaling. Heat.PwmUpperScaling è efficace soltanto se come uscita per il riscaldamento è stato selezionato OutputHeat_PWM (Heat.Select = 1) 100.0 ≥ Heat.PwmUpperScaling ≥ 0.0 Heat.PwmUpperScaling ≠ Heat.PwmLowerScaling
Config.Output.Heat.PwmLowerScaling ⁽¹⁾	REAL	0.0	Valore inferiore di uscita PWM riportato in scala per il riscaldamento Heat.PwmLowerScaling e Heat.PidLowerLimit formano una coppia di valori per la graduazione di scala del valore di uscita PID (PidOutputSum) sul valore di uscita con modulazione ampiezza impulsi per il riscaldamento (OutputHeat_PWM). Il valore di OutputHeat_PWM si colloca sempre tra Heat.PwmUpperScaling e Heat.PwmLowerScaling. Heat.PwmLowerScaling è efficace soltanto se come uscita per il riscaldamento è stato selezionato OutputHeat_PWM (Heat.Select = 1) 100.0 ≥ Heat.PwmLowerScaling ≥ 0.0 Heat.PwmUpperScaling ≠ Heat.PwmLowerScaling

Variabile	Tipo di dati	Preassegnazione	Descrizione
Config.Output.Heat.PerUpperScaling ⁽¹⁾	REAL	27648.0	Valore di uscita analogico superiore riportato in scala per il riscaldamento Heat.PerUpperScaling e Heat.PidUpperLimit formano una coppia di valori per la graduazione di scala del valore di uscita PID (PidOutputSum) sul valore di uscita analogico per il riscaldamento (OutputHeat_PER). Il valore di OutputHeat_PER si colloca sempre tra Heat.PerUpperScaling e Heat.PerLowerScaling. Heat.PerUpperScaling è efficace soltanto se come uscita per il riscaldamento è stato selezionato OutputHeat_PER (Heat.Select = 2) $32511.0 \geq \text{Heat.PerUpperScaling} \geq -32512.0$ $\text{Heat.PerUpperScaling} \neq \text{Heat.PerLowerScaling}$
Config.Output.Heat.PerLowerScaling ⁽¹⁾	REAL	0.0	Valore di uscita analogico inferiore riportato in scala il riscaldamento Heat.PerLowerScaling e Heat.PidLowerLimit formano una coppia di valori per la graduazione di scala del valore di uscita PID (PidOutputSum) sul valore di uscita analogico per il riscaldamento (OutputHeat_PER). Il valore di OutputHeat_PER si colloca sempre tra Heat.PerUpperScaling e Heat.PerLowerScaling. Heat.PerLowerScaling è efficace soltanto se come uscita per il riscaldamento è stato selezionato OutputHeat_PER (Heat.Select = 2) $32511.0 \geq \text{Heat.PerLowerScaling} \geq -32512.0$ $\text{Heat.PerUpperScaling} \neq \text{Heat.PerLowerScaling}$
Config.Output.Heat.MinimumOnTime ⁽¹⁾	REAL	0.0	Tempo ON minimo della Modulazione ampiezza impulsi per il riscaldamento (uscita OutputHeat_PWM) Un impulso PWM non è mai inferiore a questo valore. Il valore viene arrotondato a: $\text{Heat.MinimumOnTime} = n \times \text{CycleTime.Value}$ Heat.MinimumOnTime è efficace soltanto se come uscita per il riscaldamento è stato selezionato OutputHeat_PWM (Heat.Select = 1)". $100000.0 \geq \text{Heat.MinimumOnTime} \geq 0.0$
Config.Output.Heat.MinimumOffTime ⁽¹⁾	REAL	0.0	Tempo OFF minimo della Modulazione ampiezza impulsi per il riscaldamento (uscita OutputHeat_PWM) Una pausa PWM non è mai inferiore a questo valore. Il valore viene arrotondato a: $\text{Heat.MinimumOffTime} = n \times \text{CycleTime.Value}$ Heat.MinimumOffTime è efficace soltanto se come uscita per il riscaldamento è stato selezionato OutputHeat_PWM (Heat.Select = 1)". $100000.0 \geq \text{Heat.MinimumOffTime} \geq 0.0$
Config.Output.Cool.Select ⁽¹⁾	INT	1	Selezione del valore di uscita per il raffreddamento Config.Output.Cool.Select indica le uscite utilizzate per il raffreddamento: • Cool.Select = 0 - OutputCool non utilizzato • vengono utilizzati Cool.Select = 1 -OutputCoole OutputCool_PWM • vengono utilizzati Cool.Select = 2 - OutputCoole OutputCool_PER Le uscite non utilizzate non vengono calcolate e rimangono sul valore di default.

Variabile	Tipo di dati	Preassegnazione	Descrizione
			Viene calcolato soltanto se l'uscita di raffreddamento è attivata (Config.ActivateCooling = TRUE).
Config.Output.Cool.PwmPeriode ⁽¹⁾	REAL	0.0	Durata periodo della Modulazione ampiezza impulso per il raffreddamento (uscita OutputCool_PWM) in secondi: - Cool.PwmPeriode = 0.0 e Config.AdvancedCooling = FALSE: Il tempo di campionamento dell'algoritmo PID per il riscaldamento (Retain.CtrlParams.Heat.Cycle) viene utilizzato come durata periodo di PWM. - Cool.PwmPeriode = 0.0 e Config.AdvancedCooling = TRUE: Il tempo di campionamento dell'algoritmo PID per il raffreddamento (Retain.CtrlParams.Cool.Cycle) viene utilizzato come durata periodo di PWM. - Cool.PwmPeriode > 0.0: Il valore viene arrotondato ad un multiplo intero del tempo di campionamento PID_Temp (CycleTime.Value) e utilizzato come durata periodo di PWM. Con un tempo di campionamento piuttosto lungo dell'algoritmo PID, quest'impostazione consente di migliorare il livellamento del valore istantaneo. Il valore deve soddisfare le seguenti condizioni: - Cool.PwmPeriode ≤ Retain.CtrlParams.Cool.Cycle oppure Retain.CtrlParams.Heat.Cycle - Cool.PwmPeriode > Config.Output.Cool.MinimumOnTime - Cool.PwmPeriode > Config.Output.Cool.MinimumOffTime È attiva soltanto se l'uscita di raffreddamento è attivata (Config.ActivateCooling = TRUE).
Config.Output.Cool.PidUpperLimit ⁽¹⁾	REAL	0.0	Limite superiore del valore di uscita PID per il raffreddamento. Il valore deve essere 0.0. Cool.PidUpperLimit costituisce, insieme ai seguenti parametri, una coppia di valori per la graduazione di scala del valore di uscita PID (PidOutputSum) sulle uscite per il raffreddamento: - Cool.LowerScaling per OutputCool - Cool.PwmLowerScaling per OutputCool_PWM - Cool.PerLowerScaling per OutputCool_PER Se si intende limitare il valore sulla rispettiva uscita, anche i valori di scala devono essere adeguati di conseguenza. È attiva soltanto se l'uscita di raffreddamento è attivata (Config.ActivateCooling = TRUE). Cool.PidUpperLimit = 0.0
Config.Output.Cool.PidLowerLimit ⁽¹⁾	REAL	-100.0	Limite inferiore del valore di uscita PID per il raffreddamento. Nei regolatori nei quali l'uscita di raffreddamento è attivata (Config.ActivateCooling = TRUE) il valore di uscita PID (PidOutputSum) viene circoscritto su questo limite inferiore. Cool.PidLowerLimit costituisce, insieme ai seguenti parametri, una coppia di valori per la graduazione di scala del valore di uscita PID (PidOutputSum) sulle uscite per il raffreddamento: - Cool.UpperScaling per OutputCool - Cool.PwmUpperScaling per OutputCool_PWM - Cool.PerUpperScaling per OutputCool_PER Se si intende limitare il valore sulla rispettiva uscita, anche i valori di scala devono essere adeguati di conseguenza.

Variabile	Tipo di dati	Preassegnazione	Descrizione
			È attiva soltanto se l'uscita di raffreddamento è attivata (Config.ActivateCooling = TRUE). Cool.PidLowerLimit < Cool.PidUpperLimit
Config.Output.Cool.UpperScaling ⁽¹⁾	REAL	100.0	Valore di uscita superiore riportato in scala il raffreddamento Cool.UpperScaling e Cool.PidLowerLimit formano una coppia di valori per la graduazione di scala del valore di uscita PID (PidOutputSum) sul valore di uscita per il raffreddamento (OutputCool). Il valore di OutputCool si colloca sempre tra Cool.UpperScaling e Cool.LowerScaling. È attiva soltanto se l'uscita di raffreddamento è attivata (Config.ActivateCooling = TRUE). Cool.UpperScaling ≠ Cool.LowerScaling
Config.Output.Cool.LowerScaling ⁽¹⁾	REAL	0.0	Valore di uscita inferiore riportato in scala per il raffreddamento Cool.LowerScaling e Cool.PidUpperLimit formano una coppia di valori per la graduazione di scala del valore di uscita PID (PidOutputSum) sul valore di uscita per il raffreddamento (OutputCool). Il valore di OutputCool si colloca sempre tra Cool.UpperScaling e Cool.LowerScaling. È attiva soltanto se l'uscita di raffreddamento è attivata (Config.ActivateCooling = TRUE). Cool.UpperScaling ≠ Cool.LowerScaling
Config.Output.Cool.PwmUpperScaling ⁽¹⁾	REAL	100.0	Valore di uscita superiore PWM riportato in scala per il raffreddamento Cool.PwmUpperScaling e Cool.PidLowerLimit formano una coppia di valori per la graduazione di scala del valore di uscita PID (PidOutputSum) sul valore di uscita con modulazione di ampiezza impulsi per il raffreddamento (OutputCool_PWM). Il valore di OutputCool_PWM si colloca sempre tra Cool.PwmUpperScaling e Cool.PwmLowerScaling. Cool.PwmUpperScaling è efficace soltanto se l'uscita di raffreddamento è attivata (Config.ActivateCooling = TRUE) e se come uscita per il raffreddamento OutputCool_PWM è stato selezionato (Cool.Select = 1). 100.0 ≥ Cool.PwmUpperScaling ≥ 0.0 Cool.PwmUpperScaling ≠ Cool.PwmLowerScaling
Config.Output.Cool.PwmLowerScaling ⁽¹⁾	REAL	0.0	Valore di uscita inferiore PWM riportato in scala per il raffreddamento Cool.PwmLowerScaling e Cool.PidUpperLimit formano una coppia di valori per la graduazione di scala del valore di uscita PID (PidOutputSum) sul valore di uscita con modulazione di ampiezza impulsi per il raffreddamento (OutputCool_PWM). Il valore di OutputCool_PWM si colloca sempre tra Cool.PwmUpperScaling e Cool.PwmLowerScaling. Cool.PwmLowerScaling è efficace soltanto se l'uscita di raffreddamento è attivata (Config.ActivateCooling = TRUE) e se come uscita per il raffreddamento OutputCool_PWM è stato selezionato (Cool.Select = 1). 100.0 ≥ Cool.PwmLowerScaling ≥ 0.0 Cool.PwmUpperScaling ≠ Cool.PwmLowerScaling

Variabile	Tipo di dati	Preassegnazione	Descrizione
Config.Output.Cool.PerUpperScaling ⁽¹⁾	REAL	27648.0	Valore di uscita analogico superiore riportato in scala il raffreddamento Cool.PerUpperScaling e Cool.PidLowerLimit formano una coppia di valori per la graduazione di scala del valore di uscita PID (PidOutputSum) sul valore di uscita analogico per il raffreddamento (OutputCool_PER). Il valore di OutputCool_PER si colloca sempre tra Cool.PerUpperScaling e Cool.PerLowerScaling. Cool.PerUpperScaling è efficace soltanto se l'uscita di raffreddamento è attivata (Config.ActivateCooling = TRUE) e se come uscita per il raffreddamento OutputCool_PER è stato selezionato (Cool.Select = 2). $32511.0 \geq \text{Cool.PerUpperScaling} \geq -32512.0$ $\text{Cool.PerUpperScaling} \neq \text{Cool.PerLowerScaling}$
Config.Output.Cool.PerLowerScaling ⁽¹⁾	REAL	0.0	Valore di uscita analogico inferiore riportato in scala per il raffreddamento Cool.PerLowerScaling e Cool.PidUpperLimit formano una coppia di valori per la graduazione di scala del valore di uscita PID (PidOutputSum) sul valore di uscita analogico per il raffreddamento (OutputCool_PER). Il valore di OutputCool_PER si colloca sempre tra Cool.PerUpperScaling e Cool.PerLowerScaling. Cool.PerLowerScaling è efficace soltanto se l'uscita di raffreddamento è attivata (Config.ActivateCooling = TRUE) e se come uscita per il raffreddamento OutputCool_PER è stato selezionato (Cool.Select = 2). $32511.0 \geq \text{Cool.PerLowerScaling} \geq -32512.0$ $\text{Cool.PerUpperScaling} \neq \text{Cool.PerLowerScaling}$
Config.Output.Cool.MinimumOnTime ⁽¹⁾	REAL	0.0	Tempo ON minimo della modulazione ampiezza impulsi per il raffreddamento (uscita OutputCool_PWM) Un impulso PWM non è mai inferiore a questo valore. Il valore viene arrotondato a: $\text{Cool.MinimumOnTime} = n \times \text{CycleTime.Value}$ Cool.MinimumOnTime è efficace soltanto se come uscita per il raffreddamento è stato selezionato OutputCool_PWM (Cool.Select = 1). È attiva soltanto se l'uscita di raffreddamento è attivata (Config.ActivateCooling = TRUE). $100000.0 \geq \text{Cool.MinimumOnTime} \geq 0.0$
Config.Output.Cool.MinimumOffTime ⁽¹⁾	REAL	0.0	Tempo OFF minimo della modulazione ampiezza impulsi per il raffreddamento (uscita OutputCool_PWM) Una pausa PWM non è mai inferiore a questo valore. Il valore viene arrotondato a: $\text{Cool.MinimumOffTime} = n \times \text{CycleTime.Value}$ Cool.MinimumOffTime è efficace soltanto se come uscita per il raffreddamento è stato selezionato OutputCool_PWM (Cool.Select = 1). È attiva soltanto se l'uscita di raffreddamento è attivata (Config.ActivateCooling = TRUE). $100000.0 \geq \text{Cool.MinimumOffTime} \geq 0.0$
Se PID_Temp viene utilizzato in una cascata, i regolatori master e slave scambiano informazioni utilizzando i parametri master e slave. L'interconnessione deve essere eseguita dall'utente. Per dettagli vedere i parametri master.			

Variabile	Tipo di dati	Preassegnazione	Descrizione
Config.Cascade.IsMaster ⁽¹⁾	BOOL	FALSE	Il regolatore è master in una cascata e fornisce il setpoint per lo slave. Impostare IsMaster = TRUE, se l'istanza PID_Temp viene impiegata come regolatore master in una cascata. Un regolatore master predefinisce con la relativa uscita il setpoint di un regolatore slave. Un'istanza PID_Temp può essere contemporaneamente regolatore master e regolatore slave. Se il regolatore viene utilizzato come master, l'uscita di raffreddamento deve essere disattivata (Config.ActivateCooling = FALSE).
Config.Cascade.IsSlave ⁽¹⁾	BOOL	FALSE	Il regolatore è slave in una cascata e riceve il setpoint dal master. Impostare IsSlave = TRUE, se l'istanza PID_Temp viene impiegata come regolatore slave in una cascata. Un regolatore slave riceve il setpoint (parametri Setpoint) dall'uscita del suo regolatore master (parametri OutputHeat). Un'istanza PID_Temp può essere contemporaneamente regolatore master e regolatore slave.
Config.Cascade.AntiWindUpMode ⁽¹⁾	INT	1	Comportamento anti windup nella cascata Sono possibili: • Anti windup = 0 La funzionalità anti windup è disattivata. Il regolatore master non reagisce alla limitazione del relativo slave. • Anti windup = 1 Il componente I del regolatore master viene ridotto nella proporzione degli "Slave nella limitazione" rispetto al "Numero di slave" (parametro CountSlaves). Ciò consente di ridurre le ripercussioni della limitazione sul comportamento di regolazione. • Anti windup = 2 Il componente I del regolatore master viene mantenuto fintantoché il regolatore slave si trova nella limitazione. Efficace soltanto se il regolatore viene configurato come master (Config.Cascade.IsMaster = TRUE).
Config.Cascade.CountSlaves ⁽¹⁾	INT	1	Numero di slave subordinati Indicare qui il numero dei regolatori slave direttamente subordinati che ricevono il setpoint da questo regolatore master. Efficace soltanto se il regolatore viene configurato come master (Config.Cascade.IsMaster = TRUE). 255 ≥ CountSlaves ≥ 1
CycleTime.StartEstimation	BOOL	TRUE	Se CycleTime.EnEstimation = TRUE, CycleTime.StartEstimation = TRUE avvia il rilevamento automatico del tempo di campionamento PID_Temp (tempo di ciclo dell'OB richiamato). Al termine dell'operazione di misura viene impostato CycleTime.StartEstimation = FALSE.
CycleTime.EnEstimation	BOOL	TRUE	Se CycleTime.EnEstimation = TRUE, il tempo di campionamento PID_Temp viene rilevato automaticamente. Se CycleTime.EnEstimation = FALSE, il Tempo di campionamento PID_Temp non viene rilevato automaticamente e la configurazione corretta di CycleTime.Value deve essere eseguita manualmente.

10.3 PID_Temp

Variabile	Tipo di dati	Preassegnazione	Descrizione
CycleTime.EnMonitoring	BOOL	TRUE	Se CycleTime.EnMonitoring = FALSE, il Tempo di campionamento PID_Temp non viene controllato. Se PID_Temp non viene eseguito entro il tempo di campionamento, non viene emesso alcun messaggio di errore (ErrorBits=0000800h) e PID_Temp non reagisce conformemente alla configurazione eseguita in ActivateRecoverMode.
CycleTime.Value ⁽¹⁾	REAL	0.1	Tempo di campionamento PID_Temp (tempo di ciclo dell'OB richiamato) in secondi CycleTime.Value viene rilevata automaticamente e corrisponde in genere al tempo di ciclo dell'OB richiamante.
I valori della struttura CtrlParamsBackUp possono essere ricaricati con LoadBackUp.= TRUE			
CtrlParamsBackUp.SetByUser	BOOL	FALSE	Valore memorizzato di Retain.CtrlParams.SetByUser
CtrlParamsBackUp.Heat.Gain	REAL	1.0	Guadagno proporzionale memorizzato per il riscaldamento
CtrlParamsBackUp.Heat.Ti	REAL	20.0	Tempo di integrazione salvato in secondi per il riscaldamento
CtrlParamsBackUp.Heat.Td	REAL	0.0	Tempo derivativo salvato in secondi per il riscaldamento
CtrlParamsBackUp.Heat.TdFiltRatio	REAL	0.2	Coefficiente Ritardo derivativo salvato per il riscaldamento
CtrlParamsBackUp.Heat.PWeighting	REAL	1.0	Ponderazione del componente P memorizzata per il riscaldamento
CtrlParamsBackUp.Heat.DWeighting	REAL	1.0	Ponderazione del componente D memorizzata per il riscaldamento
CtrlParamsBackUp.Heat.Cycle	REAL	1.0	Tempo di campionamento dell'algoritmo PID salvato in secondi per il riscaldamento
CtrlParamsBackUp.Heat.ControlZone	REAL	3.402822e+38	Ampiezza della zona di regolazione memorizzata per il riscaldamento
CtrlParamsBackUp.Heat.DeadZone	REAL	0.0	Ampiezza banda morta memorizzata per il riscaldamento
CtrlParamsBackUp.Cool.Gain	REAL	1.0	Guadagno proporzionale memorizzato per il raffreddamento
CtrlParamsBackUp.Cool.Ti	REAL	20.0	Tempo di integrazione salvato in secondi per il raffreddamento
CtrlParamsBackUp.Cool.Td	REAL	0.0	Tempo derivativo salvato in secondi per il raffreddamento
CtrlParamsBackUp.Cool.TdFiltRatio	REAL	0.2	Coefficiente Ritardo derivativo salvato per il raffreddamento
CtrlParamsBackUp.Cool.PWeighting	REAL	1.0	Fattore di ponderazione del componente P memorizzato per il raffreddamento
CtrlParamsBackUp.Cool.DWeighting	REAL	1.0	Fattore di ponderazione del componente D memorizzato per il raffreddamento
CtrlParamsBackUp.Cool.Cycle	REAL	1.0	Tempo di campionamento dell'algoritmo PID salvato in secondi per il raffreddamento
CtrlParamsBackUp.Cool.ControlZone	REAL	3.402822e+38	Ampiezza della zona di regolazione memorizzato per il raffreddamento
CtrlParamsBackUp.Cool.DeadZone	REAL	0.0	Ampiezza banda morta memorizzata per il raffreddamento
PIDSelfTune.SUT.CalculateParamsHeat	BOOL	FALSE	Le proprietà del ramo di riscaldamento del sistema regolato vengono memorizzate durante l'Ottimizzazione iniziale per il riscaldamento. Se SUT.CalculateParamsHeat = TRUE, i parametri PID per il riscaldamento vengono ricalcolati sulla base di queste proprietà (struttura Retain.CtrlParams.Heat). Ciò consente di modificare il metodo di calcolo dei parametri (parametro PIDSelfTune.SUT.TuneRuleHeat) senza ripetere l'ottimizzazione.

Variabile	Tipo di dati	Preassegnazione	Descrizione
			Al termine del calcolo SUT.CalculateParamsHeat viene impostato su FALSE. Possibile soltanto se l'ottimizzazione iniziale è riuscita (SUT.ProcParHeatOk = TRUE).
PIDSelfTune.SUT.CalculateParamsCool	BOOL	FALSE	Le proprietà del ramo di raffreddamento del sistema regolato vengono memorizzate durante l'ottimizzazione raffreddamento. Se SUT.CalculateParamsCool = TRUE, i parametri PID per il raffreddamento vengono ricalcolati sulla base di queste proprietà (struttura Retain.CtrlParams.Cool). Ciò consente di modificare il metodo di calcolo dei parametri (parametro PIDSelfTune.SUT.TuneRuleCool) senza ripetere l'ottimizzazione. Al termine del calcolo SUT.CalculateParamsCool viene impostato su FALSE. Possibile soltanto se l'ottimizzazione iniziale è riuscita (SUT.ProcParCoolOk = TRUE). Efficace soltanto se Config.ActivateCooling = TRUE e Config.AdvancedCooling = TRUE.
PIDSelfTune.SUT.TuneRuleHeat	INT	2	Metodo di calcolo dei parametri PID nell'Ottimizzazione iniziale riscaldamento Sono possibili: • SUT.TuneRuleHeat = 0: PID secondo CHR • SUT.TuneRuleHeat = 1: PI secondo CHR • SUT.TuneRuleHeat = 2: PID per processi di temperatura secondo CHR (determina un comportamento di regolazione più lento e piuttosto asintomatico con sovraoscillazioni ridotte come SUT.TuneRuleHeat = 0) (CHR = Chien, Hrones e Reswick) Soltanto con SUT.TuneRuleHeat = 2 la zona di regolazione Retain.CtrlParams.Heat.ControlZone viene impostata automaticamente durante l'ottimizzazione iniziale riscaldamento.
PIDSelfTune.SUT.TuneRuleCool	INT	2	Metodo di calcolo dei parametri PID nell'Ottimizzazione iniziale raffreddamento Sono possibili: • SUT.TuneRuleCool = 0: PID secondo CHR • SUT.TuneRuleCool = 1: PI secondo CHR • SUT.TuneRuleCool = 2: PID per processi di temperatura secondo CHR (determina un comportamento di regolazione più lento e piuttosto asintomatico con sovraoscillazioni ridotte come SUT.TuneRuleCool = 0) (CHR = Chien, Hrones e Reswick) Soltanto con SUT.TuneRuleCool = 2 la zona di regolazione Retain.CtrlParams.Cool.ControlZone viene impostata automaticamente durante l'ottimizzazione iniziale raffreddamento. SUT.TuneRuleCool è attivo soltanto se l'uscita di raffreddamento e la commutazione dei parametri PID sono attivi (Config.ActivateCooling = TRUE, Config.AdvancedCooling = TRUE).

Variabile	Tipo di dati	Preassegnazione	Descrizione
PIDSelfTune.SUT.State	INT	0	La variabile SUT.State mostra la fase attuale dell'ottimizzazione iniziale: - State = 0: Inizializzazione dell'ottimizzazione iniziale - State = 100: Calcolo della divergenza standard di riscaldamento - State = 200: Calcolo della divergenza standard di raffreddamento - State = 300: Rilevamento del punto di flesso di riscaldamento - State = 400: Rilevamento del punto di flesso di raffreddamento - State = 500: Dopo il raggiungimento del punto di flesso di riscaldamento, quest'ultimo deve avvenire con il setpoint - State = 600: Dopo il raggiungimento del punto di flesso di raffreddamento, il riscaldamento deve avvenire con il setpoint - State = 700: Confronta l'azione dell'organo attuatore del riscaldamento e di quello del raffreddamento. - State = 800: Riscaldamento e raffreddamento attivati - State = 900: Raffreddamento attivato - State = 1000: Rileva il tempo di ritardo dopo lo spegnimento del riscaldamento - State = 9900: Ottimizzazione iniziale riuscita - State = 1: Ottimizzazione iniziale non riuscita
PIDSelfTune.SUT.ProcParHeatOk	BOOL	FALSE	TRUE: Il calcolo dei parametri di processo per la prima ottimizzazione riscaldamento è riuscito. Questa variabile viene impostata durante l'ottimizzazione e deve soddisfare la condizione TRUE per il calcolo dei parametri PID per il riscaldamento.
PIDSelfTune.SUT.ProcParCoolOk	BOOL	FALSE	TRUE: Il calcolo dei parametri di processo per l'ottimizzazione iniziale raffreddamento è riuscito. Questa variabile viene impostata durante l'ottimizzazione e deve soddisfare la condizione TRUE per il calcolo dei parametri PID per il raffreddamento.
PIDSelfTune.SUT.AdaptDelayTime	INT	0	La variabile AdaptDelayTime stabilisce l'adeguamento del tempo di ritardo del riscaldamento nel punto di lavoro (per le funzioni "Ottimizzazione iniziale riscaldamento" e "Ottimizzazione iniziale riscaldamento e raffreddamento"). Sono possibili: - SUT.AdaptDelayTime = 0: Nessun adeguamento del tempo di ritardo. La fase SUT.State = 1000 viene ignorata. Quest'opzione comporta una più breve durata dell'ottimizzazione rispetto a SUT.AdaptDelayTime = 1. - SUT.AdaptDelayTime = 1: Adeguamento del tempo di ritardo sul setpoint nella fase SUT.State = 1000 tramite spegnimento provvisorio del riscaldamento. Quest'opzione comporta una maggiore durata dell'ottimizzazione rispetto a SUT.AdaptDelayTime = 0. Essa è in grado di migliorare il comportamento di regolazione nel caso in cui il comportamento del processo sia fortemente dipendente dal punto di lavoro (non linearità). Si sconsiglia l'im-

Variabile	Tipo di dati	Preassegnazione	Descrizione
			piego di quest'opzione in applicazioni multizona con forte accoppiamento termico.
PIDSelfTune.SUT.CoolingMode	INT	0	La variabile CoolingMode determina l'uscita del valore regolante per la rilevazione dei parametri di raffreddamento (nella prima ottimizzazione riscaldamento e raffreddamento). Sono possibili: • SUT.CoolingMode = 0: Spegnimento del riscaldamento e accensione del raffreddamento al raggiungimento del setpoint. La fase SUT.State= 700 viene ignorata. Alla fase SUT.State = 500 segue la fase SUT.State = 900. Quest'opzione può migliorare il comportamento di regolazione, se il guadagno dell'organo attuatore di raffreddamento è più basso di quello dell'organo attuatore di riscaldamento. Quest'opzione comporta una più breve durata dell'ottimizzazione rispetto a SUT.CoolingMode = 1 o 2. • SUT.CoolingMode = 1: Accensione del raffreddamento oltre al riscaldamento al raggiungimento del setpoint. La fase SUT.State= 700 viene ignorata. Alla fase SUT.State = 500 segue la fase SUT.State = 800. Quest'opzione può migliorare il comportamento di regolazione, se il guadagno dell'organo attuatore di raffreddamento è più elevato di quello dell'organo attuatore di riscaldamento. • SUT.CoolingMode = 2: Dopo il riscaldamento fino al setpoint, la fase SUT.State = 700 stabilirà automaticamente se il riscaldamento debba essere spento. Alla fase SUT.State = 500 segue la fase SUT.State = 700 ed infine SUT.State = 800 oppure SUT.State = 900. Quest'opzione richiede più tempo delle opzioni 0 e 1.
PIDSelfTune.TIR.RunIn	BOOL	FALSE	La variabile RunIn consente di definire lo svolgimento dell'Ottimizzazione fine all'avvio dal Funzionamento automatico. • RunIn = FALSE Se l'ottimizzazione fine viene avviata dal funzionamento automatico, viene eseguita la regolazione sul setpoint con i parametri PID disponibili (TIR.State = 500 o 600). Solo in seguito viene avviata l'ottimizzazione fine. • RunIn = TRUE PID_Temp cerca di raggiungere il setpoint con il valore di uscita minimo o massimo (TIR.State = 300 o 400). Ciò può causare una maggiore sovraoscillazione. L'ottimizzazione fine si avvia infine automaticamente. Al termine dell'ottimizzazione fine RunIn viene impostato su FALSE. All'avvio dell'ottimizzazione fine dal funzionamento manuale o inattivo, PID_Temp si comporta come descritto in RunIn = TRUE.

Variabile	Tipo di dati	Preassegnazione	Descrizione
PIDSelfTune.TIR.CalculateParamsHeat	BOOL	FALSE	Le proprietà del ramo di riscaldamento del sistema regolato vengono memorizzate durante l'ottimizzazione fine del riscaldamento. Se TIR.CalculateParamsHeat= TRUE, i parametri PID per il riscaldamento vengono ricalcolati sulla base di queste proprietà (struttura Retain.CtrlParams.Heat). Ciò consente di modificare il metodo di calcolo dei parametri (parametro PIDSelfTune.TIR.TuneRuleHeat) senza ripetere l'ottimizzazione. Al termine del calcolo TIR.CalculateParamsHeat viene impostato su FALSE. Possibile soltanto se l'ottimizzazione fine è in precedenza riuscita (TIR.ProcParHeatOk = TRUE).
PIDSelfTune.TIR.CalculateParamsCool	BOOL	FALSE	Le proprietà del ramo di raffreddamento del sistema regolato vengono memorizzate durante l'ottimizzazione fine raffreddamento. Se TIR.CalculateParamsCool= TRUE, i parametri PID per il raffreddamento vengono ricalcolati sulla base di queste proprietà (struttura Retain.CtrlParams.Cool). Ciò consente di modificare il metodo di calcolo dei parametri (parametro PIDSelfTune.TIR.TuneRuleCool) senza ripetere l'ottimizzazione. Al termine del calcolo TIR.CalculateParamsCool viene impostato su FALSE. Possibile soltanto se l'ottimizzazione fine del raffreddamento è in precedenza riuscita (TIR.ProcParCoolOk = TRUE). Efficace soltanto se Config.ActivateCooling = TRUE e Config.AdvancedCooling = TRUE.
PIDSelfTune.TIR.TuneRuleHeat	INT	0	Metodo di calcolo dei parametri PID nell'Ottimizzazione fine riscaldamento Sono possibili: • TIR.TuneRuleHeat = 0: PID automatico • TIR.TuneRuleHeat = 1: PID veloce (comportamento di regolazione più veloce con ampiezze del valore di uscita più elevate rispetto a TIR.TuneRuleHeat = 2) • TIR.TuneRuleHeat = 2: PID lento (comportamento di regolazione più lento con ampiezze del valore di uscita più basse rispetto a TIR.TuneRuleHeat = 1) • TIR.TuneRuleHeat = 3: ZN PID • TIR.TuneRuleHeat = 4: ZN PI • TIR.TuneRuleHeat = 5: ZN P (ZN=Ziegler Nichols) Per eseguire il calcolo dei parametri PID per il riscaldamento con TIR.CalculateParamsHeat e TIR.TuneRuleHeat = 0 1 o 2, anche l'ottimizzazione fine precedente deve essere stata eseguita con TIR.TuneRuleHeat = 0 1 o 2. In caso contrario viene utilizzato TIR.TuneRuleHeat = 3. Il nuovo calcolo dei parametri PID per il riscaldamento con TIR.CalculateParamsHeat e TIR.TuneRuleHeat = 3 4 o 5, è sempre possibile.
PIDSelfTune.TIR.TuneRuleCool	INT	0	Metodo di calcolo dei parametri PID nell'Ottimizzazione fine raffreddamento

Variabile	Tipo di dati	Preassegnazione	Descrizione
			Sono possibili: - TIR.TuneRuleCool = 0: PID automatico - TIR.TuneRuleCool = 1: PID veloce (comportamento di regolazione più veloce con ampiezze del valore di uscita più elevate rispetto a TIR.TuneRuleCool = 2) - TIR.TuneRuleCool = 2: PID lento (comportamento di regolazione più lento con ampiezze del valore di uscita più basse rispetto a TIR.TuneRuleCool = 1) - TIR.TuneRuleCool = 3: ZN PID - TIR.TuneRuleCool = 4: ZN PI - TIR.TuneRuleCool = 5: ZN P (ZN=Ziegler Nichols) Per eseguire il calcolo dei parametri PID per il raffreddamento con TIR.CalculateParamsCool e TIR.TuneRuleCool = 0 1 o 2, anche l'ottimizzazione fine precedente deve essere stata eseguita con TIR.TuneRuleCool = 0 1 o 2. In caso contrario viene utilizzato TIR.TuneRuleCool = 3. Il nuovo calcolo dei parametri PID per il raffreddamento con TIR.CalculateParamsCool e TIR.TuneRuleCool = 3 4 o 5, è sempre possibile. Attivo soltanto se l'uscita di raffreddamento e la commutazione dei parametri PID sono attivi ("ConfigActivateCooling" = TRUE e Config.AdvancedCooling = TRUE).
PIDSelfTune.TIR.State	INT	0	La variabile TIR.State mostra la fase attuale dell'"Ottimizzazione fine": - State = 0: Inizializzazione dell'ottimizzazione fine - State = 100: Calcolo della divergenza standard di riscaldamento - State = 200: Calcolo della divergenza standard di raffreddamento - State = 300: Tentativo di raggiungere il setpoint riscaldamento con la regolazione a due punti per il riscaldamento - State = 400: Tentativo di raggiungere il setpoint raffreddamento con la regolazione a due punti per il raffreddamento - State = 500: Tentativo di raggiungere il setpoint riscaldamento con la regolazione PID - State = 600: Tentativo di raggiungere il setpoint raffreddamento con la regolazione PID - State = 700: Calcolo della divergenza standard di riscaldamento - State = 800: Calcolo della divergenza standard di raffreddamento - State = 900: Rilevamento dell'oscillazione e calcolo dei parametri per il riscaldamento - State = 1000: Rilevamento dell'oscillazione e calcolo dei parametri per il raffreddamento - State = 9900: Ottimizzazione fine riuscita - State = 1: Ottimizzazione fine non riuscita Se l'ottimizzazione fine di un regolatore di riscaldamento (Config.ActivateCooling = TRUE) rimane per un molto tempo su TIR.State = 900 o 1000 e Progress = 1,0 senza cambiare, vedere le istruzioni relative all'offset di ottimizzazione in Parametri State e Mode PID_Temp (Pagina 420).

Variabile	Tipo di dati	Preassegnazione	Descrizione
PIDSelfTune.TIR.ProcParHeatOk	BOOL	FALSE	TRUE: Il calcolo dei parametri di processo per l'Ottimizzazione fine è riuscito. Questa variabile viene impostata durante l'ottimizzazione e deve soddisfare la condizione prevista per il calcolo dei parametri PID per il riscaldamento.
PIDSelfTune.TIR.ProcParCoolOk	BOOL	FALSE	TRUE: Il calcolo dei parametri di processo per l'Ottimizzazione fine per il raffreddamento è riuscito. Questa variabile viene impostata durante l'ottimizzazione e deve soddisfare la condizione prevista per il calcolo dei parametri PID per il raffreddamento.
PIDSelfTune.TIR.OutputOffsetHeat	REAL	0.0	Offset di ottimizzazione riscaldamento del valore di uscita PID TIR.OutputOffsetHeat viene addizionato al valore risultante da PidOutputSum per il ramo di riscaldamento. Predefinire un valore positivo per TIR.OutputOffsetHeat al fine di ottenere un offset positivo sulle uscite del riscaldamento. I valori risultanti sulle uscite per il riscaldamento derivano dalla graduazione di scala configurata dell'uscita (Struktur Config.Output.Heat). Nei regolatori con uscita di raffreddamento attivata e commutazione dei parametri PID (Config.ActivateCooling = TRUE, Config.AdvancedCooling = TRUE) questo offset di ottimizzazione può essere impiegato per l'ottimizzazione fine raffreddamento. Se nel setpoint sul quale deve essere eseguita l'ottimizzazione, le uscite per il raffreddamento non sono attive (PidOutputSum > 0.0), l'ottimizzazione fine raffreddamento non è possibile. In questo caso, prima dell'avvio dell'ottimizzazione, predefinire per quest'ultima un offset positivo per il riscaldamento che sia superiore al valore di uscita PID (PidOutputSum) sul setpoint nello stato stazionario. In questo modo i valori sulle uscite per il riscaldamento vengono incrementati e le uscite per il raffreddamento vengono attivate (PidOutputSum < 0.0). Ciò rende possibile l'ottimizzazione fine raffreddamento. Al termine dell'ottimizzazione fine, TIR.OutputOffsetHeat viene resettato a 0.0. Modifiche di grande entità eseguite in un'unica fase in TIR.OutputOffsetHeat possono comportare temporanee sovraoscillazioni. Config.Output.Heat.PidUpperLimit ≥ PIDSelfTune.TIR.OutputOffsetHeat ≥ Config.Output.Heat.PidLowerLimit
PIDSelfTune.TIR.OutputOffsetCool	REAL	0.0	Offset di ottimizzazione raffreddamento del valore di uscita PID TIR.OutputOffsetCool viene addizionato al valore risultante da PidOutputSum per il ramo di raffreddamento. Predefinire un valore negativo per TIR.OutputOffsetCool al fine di ottenere un offset positivo sulle uscite per il raffreddamento. I valori risultanti sulle uscite per il raffreddamento derivano dalla graduazione di scala configurata dell'uscita (Struktur Config.Output.Cool). Nei regolatori con uscita di raffreddamento attivata (Config.ActivateCooling = TRUE) questo offset di ottimizzazione può essere impiegato per l'ottimizzazione fine riscaldamento. Se nel setpoint sul quale deve essere eseguita l'ottimizzazione, le uscite per il riscaldamento non sono attive (PidOutputSum < 0.0), l'ottimizzazione fine riscaldamento non è possibile. In questo caso, prima dell'avvio dell'ottimizzazione, predefinire

Variabile	Tipo di dati	Preassegnazione	Descrizione
			per quest'ultima un offset negativo per il raffreddamento che sia inferiore al valore di uscita PID (PidOutputSum) sul setpoint nello stato stazionario. In questo modo i valori sulle uscite per il raffreddamento vengono incrementati e le uscite per il riscaldamento vengono attivate (PidOutputSum > 0.0). Ciò rende possibile l'ottimizzazione fine riscaldamento. Al termine dell'ottimizzazione fine, TIR.OutputOffsetCool viene resettato a 0.0. Modifiche di grande entità eseguite in un'unica fase in TIR.OutputOffsetCool possono comportare temporanee sovraoscillazioni. Config.Output.Cool.PidUpperLimit ≥ PIDSelfTune.TIR.OutputOffsetCool ≥ Config.Output.Cool.PidLowerLimit
PIDSelfTune.TIR.WaitForControlln	BOOL	FALSE	Durante l'Ottimizzazione fine attendere il raggiungimento del setpoint Se TIR.WaitForControlln = TRUE, durante l'ottimizzazione fine, nell'intervallo tra il raggiungimento del setpoint (TIR.State = 500 o 600) ed il calcolo della divergenza standard (TIR.State = 700 o 800), si attende fino a quando in TIR.FinishControlln viene predefinito un fronte FALSE -> TRUE. TIR.WaitForControlln può essere impiegato nell'ottimizzazione fine simultanea di più regolatori in applicazioni multizona al fine di sincronizzare l'ottimizzazione delle singole zone. Ciò garantisce il raggiungimento in tutte le zone dei rispettivi valori di setpoint prima dell'avvio dell'ottimizzazione vera e propria. Questa procedura consente inoltre di ridurre le ripercussioni sull'ottimizzazione degli accoppiamenti termici tra le zone. TIR.WaitForControlln è efficace soltanto se l'ottimizzazione fine viene avviata nel funzionamento automatico con PIDSelfTune.TIR.RunIn = FALSE.
PIDSelfTune.TIR.ControllnReady	BOOL	FALSE	Se con un'ottimizzazione fine viene raggiunto il setpoint, viene impostato PID_Temp TIR.ControllnReady = TRUE. Se in precedenza era stato impostato TIR.WaitForControlln = TRUE, PID_Temp attende con ulteriori operazioni di ottimizzazione finché non si verifica un fronte FALSE -> TRUE in TIR.FinishControlln. Se TIR.WaitForControlln = FALSE, PID_Temp continua direttamente l'ottimizzazione e TIR.ControllnReady non è rilevante.
PIDSelfTune.TIR.FinishControlln	BOOL	FALSE	Se TIR.WaitForControlln = TRUE e TIR.ControllnReady = TRUE, un fronte FALSE -> TRUE in TIR.FinishControlln pone fine all'attesa e l'ottimizzazione fine continua.
PIDCtrl.IOutputOld ⁽¹⁾	REAL	0.0	Componente I nell'ultimo ciclo
PIDCtrl.PIDInit	BOOL	FALSE	PIDCtrl.PIDInit è disponibile da PID_Temp, versione V. 1.1. Se nel "Funzionamento automatico" PIDCtrl.PIDInit = TRUE, al componente integrale PIDCtrl.IOutputOld viene preassegnato automaticamente un valore tale da far supporre che nel ciclo precedente ci fosse PidOutputSum = OverwriteInitialOutputValue. Questo parametro può essere utilizzato per una Regolazione override con PID_Temp (Pagina 210).

Variabile	Tipo di dati	Preassegnazione	Descrizione
Retain.CtrlParams.SetByUser ⁽¹⁾	BOOL	FALSE	Attivazione dell'inserimento manuale dei parametri PID Se Retain.CtrlParams.SetByUser = TRUE, i parametri PID sono editabili. Retain.CtrlParams.SetByUser viene utilizzato per la progettazione del regolatore in TIA Portal e non incide in alcun modo sul comportamento dell'algoritmo di regolazione nella CPU. SetByUser è a ritenzione.
Retain.CtrlParams.Heat.Gain ⁽¹⁾	REAL	1.0	Guadagno proporzionale attivo per il riscaldamento Heat.Gain è a ritenzione. $\text{Heat.Gain} \geq 0.0$
Retain.CtrlParams.Heat.Ti ⁽¹⁾	REAL	20.0	Tempo di integrazione attivo in secondi per il riscaldamento Con Heat.CtrlParams.Ti = 0.0 il componente I per il riscaldamento è disattivato. Heat.Ti è a ritenzione. $100000.0 \geq \text{Heat.Ti} \geq 0.0$
Retain.CtrlParams.Heat.Td ⁽¹⁾	REAL	0.0	Tempo derivativo attivo in secondi per il riscaldamento Con Heat.CtrlParams.Td = 0.0 il componente D per il riscaldamento è disattivato. Heat.Td è a ritenzione. $100000.0 \geq \text{Heat.Td} \geq 0.0$
Retain.CtrlParams.Heat.TdFiltRatio ⁽¹⁾	REAL	0.2	Coefficiente Ritardo derivativo attivo per il riscaldamento L'effetto del componente D viene ritardato dal coefficiente per il ritardo derivativo. Ritardo derivativo = Tempo derivativo × Coefficiente ritardo derivativo 0.0: Il componente D ha effetto solo per un ciclo e quindi è quasi inattivo. 0.5: Questo valore ha dato buoni risultati nelle operazioni con i sistemi regolati con una costante temporale dominante. > 1.0: Maggiore è il coefficiente, maggiore sarà il ritardo dell'azione del componente D. Heat.TdFiltRatio è a ritenzione. $\text{Heat.TdFiltRatio} \geq 0.0$
Retain.CtrlParams.Heat.PWeighting ⁽¹⁾	REAL	1.0	Ponderazione attiva del componente P per il riscaldamento In caso di variazioni del setpoint è possibile ridurre il componente P. Sono adeguati valori da 0.0 a 1.0. 1.0: componente P pienamente attivo in caso di variazione del setpoint 0.0: componente P non attivo in caso di variazione del setpoint In caso di variazioni del valore istantaneo il componente P è sempre pienamente attivo. Heat.PWeighting è a ritenzione. $1.0 \geq \text{Heat.PWeighting} \geq 0.0$

Variabile	Tipo di dati	Preassegnazione	Descrizione
Retain.CtrlParams.Heat.DWeighting ⁽¹⁾	REAL	1.0	Ponderazione attiva del componente D per il riscaldamento In caso di variazioni del setpoint è possibile ridurre il componente D. Sono adeguati valori da 0.0 a 1.0. 1.0: componente D pienamente attivo in caso di variazione del setpoint 0.0: componente D non attivo in caso di variazione del setpoint In caso di variazioni del valore istantaneo il componente D è sempre pienamente attivo. Heat.DWeighting è a ritenzione. $1.0 \geq \text{Heat.DWeighting} \geq 0.0$
Retain.CtrlParams.Heat.Cycle ⁽¹⁾	REAL	1.0	Tempo di campionamento attivo dell'algoritmo PID in secondi per il riscaldamento CtrlParams.Heat.Cycle viene rilevato durante l'ottimizzazione e arrotondato a un multiplo intero di CycleTime.Value. Se Config.Output.Heat.PwmPeriode = 0.0, Heat.Cycle viene impiegato come durata periodo della modulazione ampiezza impulsi per il riscaldamento. Se Config.Output.Cool.PwmPeriode = 0.0 e Config.AdvancedCooling = FALSE, Heat.Cycle viene impiegato come durata periodo della modulazione ampiezza impulsi per il raffreddamento. Heat.Cycle è a ritenzione. $100000.0 \geq \text{Heat.Cycle} > 0.0$
Retain.CtrlParams.Heat.ControlZone ⁽¹⁾	REAL	3.402822e+38	Ampiezza della zona di regolazione attiva per il riscaldamento Con Heat.ControlZone = 3.402822e+38 la zona di regolazione per il riscaldamento è disattivata. Heat.ControlZone viene impostato automaticamente soltanto durante l'ottimizzazione iniziale riscaldamento oppure durante l'ottimizzazione iniziale riscaldamento e raffreddamento, se per il metodo è stato selezionato il calcolo dei parametri PIDSelfTune.SUT.TuneRuleHeat = 2. Nei regolatori con uscita di raffreddamento disattivata (Config.ActivateCooling = FALSE) oppure in regolatori con uscita e fattore di raffreddamento attivati (Config.AdvancedCooling = FALSE) la zona di regolazione si trova simmetricamente tra Setpoint - Heat.ControlZone e Setpoint + Heat.ControlZone. Nei regolatori con uscita di raffreddamento attivata e commutazione dei parametri PID (Config.ActivateCooling = TRUE, Config.AdvancedCooling = TRUE), la zona di regolazione si trova tra Setpoint - Heat.ControlZone e Setpoint + Cool.ControlZone. Heat.ControlZone è a ritenzione. Heat.ControlZone > 0.0:
Retain.CtrlParams.Heat.DeadZone ⁽¹⁾	REAL	0.0	Ampiezza banda morta attiva per il riscaldamento (vedere Parametri PID (Pagina 184)) Con Heat.DeadZone = 0.0 la banda morta per il riscaldamento è disattivata. Heat.DeadZone non vengono impostati automaticamente o adeguati durante l'ottimizzazione. Heat.DeadZone deve essere configurato manualmente in modo corretto. Con la banda morta attivata, può instaurarsi una differenza di regolazione permanente (divergenza tra setpoint e valore istantaneo). Questo può avere ripercussioni negative sull'esecuzione dell'ottimizzazione fine.

Variabile	Tipo di dati	Preassegnazione	Descrizione
			Nei regolatori con uscita di raffreddamento disattivata (Config.ActivateCooling = FALSE) oppure in regolatori con uscita e fattore di raffreddamento attivati (Config.AdvancedCooling = FALSE) la banda morta si trova simmetricamente tra Setpoint - Heat.DeadZone e Setpoint + Heat.DeadZone. Nei regolatori con uscita di raffreddamento attivata e commutazione dei parametri PID (Config.ActivateCooling = TRUE, Config.AdvancedCooling = TRUE), la banda morta si trova tra Setpoint - Heat.DeadZone e Setpoint + Cool.DeadZone. Heat.DeadZone è a ritenzione. Heat.DeadZone ≥ 0.0
Retain.CtrlParams.Cool.Gain ⁽¹⁾	REAL	1.0	Guadagno proporzionale attivo per il raffreddamento Cool.Gain è a ritenzione. Efficace soltanto se l'uscita di raffreddamento e la commutazione dei parametri PID sono attivi (Config.ActivateCooling = TRUE, Config.AdvancedCooling = TRUE). Cool.Gain ≥ 0.0
Retain.CtrlParams.Cool.Ti ⁽¹⁾	REAL	20.0	Tempo di integrazione attivo in secondi per il raffreddamento Con Cool.CtrlParams.Ti = 0.0 il componente I per il raffreddamento è disattivato. Cool.Ti è a ritenzione. Efficace soltanto se l'uscita di raffreddamento e la commutazione dei parametri PID sono attivi (Config.ActivateCooling = TRUE, Config.AdvancedCooling = TRUE). 100000.0 $\geq$ Cool.Ti ≥ 0.0
Retain.CtrlParams.Cool.Td ⁽¹⁾	REAL	0.0	Tempo derivativo attivo in secondi per il raffreddamento Con Cool.CtrlParams.Td = 0.0 il componente D per il raffreddamento è disattivato. Cool.Td è a ritenzione. Efficace soltanto se l'uscita di raffreddamento e la commutazione dei parametri PID sono attivi (Config.ActivateCooling = TRUE, Config.AdvancedCooling = TRUE). 100000.0 $\geq$ Cool.Td ≥ 0.0
Retain.CtrlParams.Cool.TdFiltRatio ⁽¹⁾	REAL	0.2	Coefficiente Ritardo derivativo attivato per il raffreddamento L'effetto del componente D viene ritardato dal coefficiente per il ritardo derivativo. Ritardo derivativo = Tempo derivativo $\times$ Coefficiente ritardo derivativo 0.0: Il componente D ha effetto solo per un ciclo e quindi è quasi inattivo. 0.5: Questo valore ha dato buoni risultati nelle operazioni con i sistemi regolati con una costante temporale dominante. > 1.0: Maggiore è il coefficiente, maggiore sarà il ritardo dell'azione del componente D. Cool.TdFiltRatio è a ritenzione. Efficace soltanto se l'uscita di raffreddamento e la commutazione dei parametri PID sono attivi (Config.ActivateCooling = TRUE, Config.AdvancedCooling = TRUE). Cool.TdFiltRatio ≥ 0.0

Variabile	Tipo di dati	Preassegnazione	Descrizione
Retain.CtrlParams.Cool.PWeighting ⁽¹⁾	REAL	1.0	Ponderazione attiva del componente P per il raffreddamento In caso di variazioni del setpoint è possibile ridurre il componente P. Sono adeguati valori da 0.0 a 1.0. 1.0: componente P pienamente attivo in caso di variazione del setpoint 0.0: componente P non attivo in caso di variazione del setpoint In caso di variazioni del valore istantaneo il componente P è sempre pienamente attivo. Cool.PWeighting è a ritenzione. Efficace soltanto se l'uscita di raffreddamento e la commutazione dei parametri PID sono attivi (Config.ActivateCooling = TRUE, Config.AdvancedCooling = TRUE). $1.0 \geq \text{Cool.PWeighting} \geq 0.0$
Retain.CtrlParams.Cool.DWeighting ⁽¹⁾	REAL	1.0	Ponderazione attiva del componente D per il raffreddamento In caso di variazioni del setpoint è possibile ridurre il componente D. Sono adeguati valori da 0.0 a 1.0. 1.0: componente D pienamente attivo in caso di variazione del setpoint 0.0: componente D non attivo in caso di variazione del setpoint In caso di variazioni del valore istantaneo il componente D è sempre pienamente attivo. Cool.DWeighting è a ritenzione. Efficace soltanto se l'uscita di raffreddamento e la commutazione dei parametri PID sono attivi (Config.ActivateCooling = TRUE, Config.AdvancedCooling = TRUE). $1.0 \geq \text{Cool.DWeighting} \geq 0.0$
Retain.CtrlParams.Cool.Cycle ⁽¹⁾	REAL	1.0	Tempo di campionamento attivo dell'algoritmo PID in secondi per il raffreddamento CtrlParams.Cool.Cycle viene rilevato durante l'ottimizzazione e arrotondato a un multiplo intero di CycleTime.Value. Se Config.Output.Cool.PwmPeriode = 0.0 e Config.AdvancedCooling = TRUE, Cool.Cycle viene impiegato come durata periodo della modulazione ampiezza impulsi per il raffreddamento. Se Config.Output.Cool.PwmPeriode = 0.0 e Config.AdvancedCooling = FALSE, Heat.Cycle viene impiegato come durata periodo della modulazione ampiezza impulsi per il raffreddamento. Cool.Cycle è a ritenzione. Efficace soltanto se l'uscita di raffreddamento e la commutazione dei parametri PID sono attivi (Config.ActivateCooling = TRUE, Config.AdvancedCooling = TRUE). $100000.0 \geq \text{Cool.Cycle} > 0.0$
Retain.CtrlParams.Cool.ControlZone ⁽¹⁾	REAL	3.402822e+38	Ampiezza della zona di regolazione attiva per il raffreddamento Con Cool.ControlZone = 3.402822e+38 la zona di regolazione del raffreddamento è disattivata. Cool.ControlZone viene impostato automaticamente soltanto durante l'ottimizzazione iniziale raffreddamento oppure durante l'ottimizzazione iniziale riscaldamento e raffreddamento, se

Variabile	Tipo di dati	Preassegnazione	Descrizione
			per il metodo è stato selezionato il calcolo dei parametri PIDSelfTune.SUT.TuneRuleCool = 2. Cool.ControlZone è a ritenzione. Efficace soltanto se l'uscita di raffreddamento e la commutazione dei parametri PID sono attivi (Config.ActivateCooling = TRUE, Config.AdvancedCooling = TRUE). Cool.ControlZone > 0.0:
Retain.CtrlParams.Cool.DeadZone ⁽¹⁾	REAL	0.0	Ampiezza banda morta attiva per il raffreddamento (vedere Parametri PID (Pagina 184)) Con Cool.DeadZone = 0.0 la banda morta per il raffreddamento è disattivata. Cool.DeadZone non vengono impostati automaticamente o adeguati durante l'ottimizzazione. Cool.DeadZone deve essere configurato manualmente in modo corretto. Con la banda morta attivata, può instaurarsi una differenza di regolazione permanente (divergenza tra setpoint e valore istantaneo). Questo può avere ripercussioni negative sull'esecuzione dell'ottimizzazione fine. Cool.DeadZone è a ritenzione. Efficace soltanto se l'uscita di raffreddamento e la commutazione dei parametri PID sono attivi (Config.ActivateCooling = TRUE, Config.AdvancedCooling = TRUE). Cool.DeadZone ≥ 0.0

Vedere anche

[Variabile ActivateRecoverMode PID_Temp \(Pagina 430\)](#)

[Variabile Warning PID_Temp \(Pagina 431\)](#)

[Regolazione multizona con PID_Temp \(Pagina 207\)](#)

10.3.4.7 Parametri State e Mode PID_Temp**Relazione tra i parametri**

Il parametro State indica il modo di funzionamento attuale del regolatore PID. Non è possibile modificarlo.

Con un fronte di salita in ModeActivate PID_Temp passa al modo di funzionamento salvato nel parametro di transito Mode.

Heat.EnableTuning e Cool.EnableTuning definiscono per l'Ottimizzazione iniziale e l'Ottimizzazione fine se l'ottimizzazione debba essere eseguita per il riscaldamento o per il raffreddamento.

Se la CPU si accende o passa da STOP a RUN, PID_Temp si avvia nel modo di funzionamento salvato in Mode. Per lasciare PID_Temp nel modo di funzionamento "Inattivo" impostare RunModeByStartup = FALSE.

Significato dei valori

State / Mode	Descrizione del modo di funzionamento
0	Inattivo Nel modo di funzionamento "Inattivo" vengono emessi i seguenti valori di uscita: 0.0 come valore di uscita PID (PidOutputSum) 0.0 come valore di uscita per il riscaldamento (OutputHeat) e valore di uscita per il raffreddamento (OutputCool) 0 come valore di uscita analogico per il riscaldamento (OutputHeat_PER) e valore di uscita analogico per il raffreddamento (OutputCool_PER) - FALSE come valore di uscita PWM per il riscaldamento (OutputHeat_PWM) e valore di uscita PWM per il raffreddamento (OutputCool_PWM) Questo comportamento si verifica indipendentemente dai limiti del valore di uscita configurati e dalla scala nelle strutture Config.Output.Heat e Config.Output.Cool.
1	Ottimizzazione iniziale L'ottimizzazione iniziale rileva la risposta del processo a un gradino del valore di uscita e cerca il punto di flesso. Dalla pendenza max. e dal tempo morto del sistema regolato vengono calcolati i parametri PID. I parametri PID migliori si ottengono eseguendo l'ottimizzazione iniziale e l'ottimizzazione fine. PID_Temp consente, in funzione della configurazione, diversi tipi di ottimizzazione iniziale: Ottimizzazione iniziale riscaldamento: Sul valore di uscita del riscaldamento viene emesso un gradino, vengono calcolati i parametri PID per il riscaldamento (struttura Retain.CtrlParams.Heat) e nel funzionamento automatico viene poi eseguita la regolazione sul setpoint. Se il comportamento di processo è fortemente dipendente dal punto di lavoro, PIDSelfTune.SUT.AdaptDelayTime consente l'attivazione di un adeguamento del tempo di ritardo nel setpoint. Ottimizzazione iniziale riscaldamento e raffreddamento: Viene emesso un gradino sul valore di uscita del riscaldamento. Non appena il valore istantaneo si avvicina al setpoint, viene emesso un gradino sul valore di uscita del raffreddamento. Vengono calcolati i parametri PID per il riscaldamento (struttura Retain.CtrlParams.Heat) ed il raffreddamento (struttura Retain.CtrlParams.Cool). Segue la regolazione sul setpoint nel funzionamento automatico. Se il comportamento di processo è fortemente dipendente dal punto di lavoro, PIDSelfTune.SUT.AdaptDelayTime consente l'attivazione di un adeguamento del tempo di ritardo nel setpoint. L'azione dell'organo attuatore del raffreddamento rispetto a quella dell'organo attuatore del riscaldamento, può incidere sulla qualità dell'ottimizzazione determinando se durante quest'ultima le uscite di riscaldamento e raffreddamento possano essere attivate o meno contemporaneamente. Questo comportamento può essere definito in PIDSelfTune.SUT.CoolingMode. Ottimizzazione iniziale raffreddamento: Sul valore di uscita del raffreddamento viene emesso un gradino, vengono calcolati i parametri PID per il raffreddamento (Struktur Retain.CtrlParams.Cool). Segue la regolazione sul setpoint nel funzionamento automatico. L'ottimizzazione dei parametri PID per riscaldamento e raffreddamento garantisce un miglior comportamento di regolazione eseguendo la funzione "Ottimizzazione iniziale riscaldamento" seguita da "Ottimizzazione iniziale raffreddamento" in luogo della sola funzione "Ottimizzazione iniziale riscaldamento e raffreddamento". L'esecuzione dell'ottimizzazione in due fasi richiede tuttavia più tempo. Presupposti generali per l'ottimizzazione iniziale: - L'istruzione PID_Temp deve essere richiamata in un OB di schedulazione orologio. - Modo di funzionamento Inattivo (State = 0), funzionamento manuale (State = 4) o automatico (State = 3) - ManualEnable = FALSE - Reset = FALSE - Il setpoint e il valore istantaneo sono compresi entro i limiti configurati.

State / Mode	Descrizione del modo di funzionamento
	Presupposti per l'ottimizzazione iniziale riscaldamento: - Heat.EnableTuning = TRUE - Cool.EnableTuning = FALSE - Il valore istantaneo non deve essere troppo vicino al setpoint. $\text{Setpoint} - \text{Input} > 0.3 * \text{Config.InputUpperLimit} - \text{Config.InputLowerLimit}$ e $\text{Setpoint} - \text{Input} > 0.5 * \text{Setpoint}$ - Il setpoint è maggiore del valore istantaneo. $\text{Setpoint} > \text{Input}$ Presupposti per l'ottimizzazione iniziale riscaldamento e raffreddamento: - Heat.EnableTuning = TRUE. - Cool.EnableTuning = TRUE - L'uscita di raffreddamento deve essere attivata (Config.ActivateCooling = TRUE). - La commutazione dei parametri PID deve essere attivata (Config.AdvancedCooling = TRUE). - Il valore istantaneo non deve essere troppo vicino al setpoint. $\text{Setpoint} - \text{Input} > 0.3 * \text{Config.InputUpperLimit} - \text{Config.InputLowerLimit}$ e $\text{Setpoint} - \text{Input} > 0.5 * \text{Setpoint}$ - Il setpoint è maggiore del valore istantaneo. $\text{Setpoint} > \text{Input}$ Presupposti per l'ottimizzazione iniziale raffreddamento: - Heat.EnableTuning = FALSE. - Cool.EnableTuning = TRUE. - L'uscita di raffreddamento deve essere attivata (Config.ActivateCooling = TRUE). - La commutazione dei parametri PID deve essere attivata (Config.AdvancedCooling = TRUE). - Una delle funzioni "Ottimizzazione iniziale riscaldamento" oppure "Ottimizzazione iniziale riscaldamento e raffreddamento" deve essere stata eseguita correttamente (PIDSelfTune.SUT.ProcParHeatOk = TRUE), possibilmente nel medesimo setpoint. - Il valore istantaneo deve essere vicino al setpoint. $\text{Setpoint} - \text{Input} < 0.05 * \text{Config.InputUpperLimit} - \text{Config.InputLowerLimit}$ Più stabile è il valore istantaneo, più facilmente e con maggiore precisione si possono determinare i parametri PID. Un rumore del valore istantaneo è accettabile fintanto che l'incremento del valore è significativamente maggiore rispetto al rumore. Ciò si verifica con maggiore probabilità nei modi di funzionamento "Inattivo" oppure "Funzionamento manuale". Il setpoint viene congelato nella variabile CurrentSetpoint. L'ottimizzazione viene interrotta se: $\text{Setpoint} > \text{CurrentSetpoint} + \text{CancelTuningLevel}$ o $\text{Setpoint} < \text{CurrentSetpoint} - \text{CancelTuningLevel}$ Il metodo di calcolo dei parametri PID può essere definito separatamente per riscaldamento e raffreddamento con PIDSelfTune.SUT.TuneRuleHeat e PIDSelfTune.SUT.TuneRuleCool. Prima di essere ricalcolati, i parametri PID vengono salvati nella struttura CtrlParamsBackUp e possono essere riattivati tramite LoadBackUp Se l'ottimizzazione iniziale si conclude correttamente si ha la commutazione al funzionamento automatico. In caso di mancata riuscita dell'ottimizzazione, la commutazione del modo di funzionamento dipende da ActivateRecoverMode. La fase dell'ottimizzazione iniziale viene visualizzata con PIDSelfTune.SUT.State. Per avviare l'ottimizzazione iniziale riscaldamento o l'ottimizzazione iniziale riscaldamento e raffreddamento dal funzionamento automatico si consiglia di eseguire la modifica richiesta del setpoint contemporaneamente al fronte di salita in ModeActivate. Se il setpoint viene modificato prima dell'avvio dell'ottimizzazione iniziale, il valore di uscita viene adattato di conseguenza nel funzionamento automatico, causando una modifica del valore istantaneo. Questo può avere ripercussioni negative sulla successiva esecuzione dell'ottimizzazione iniziale o impedirne l'avvio.

State / Mode	Descrizione del modo di funzionamento
2	ottimizzazione fine L'ottimizzazione fine genera un'oscillazione costante e limitata del valore istantaneo. I parametri PID per il punto di lavoro vengono ottimizzati partendo dall'ampiezza e dalla frequenza dell'oscillazione. I parametri PID risultanti dall'ottimizzazione fine mostrano in genere un comportamento pilota e di disturbo migliore rispetto ai parametri PID dell'ottimizzazione iniziale. I parametri PID migliori si ottengono eseguendo l'ottimizzazione iniziale e l'ottimizzazione fine. PID_Temp cerca di generare automaticamente un'oscillazione che sia maggiore rispetto al fruscio del valore istantaneo. L'ottimizzazione fine viene influenzata solo minimamente dalla stabilità del valore istantaneo. PID_Temp consente, in funzione della configurazione, diversi tipi di ottimizzazione fine: • Ottimizzazione fine riscaldamento: PID_Temp crea, con modifiche periodiche, un'oscillazione sul valore di uscita di riscaldamento del valore istantaneo e calcola i parametri PID per il riscaldamento (Struktur Retain.CtrlParams.Heat). • Ottimizzazione fine raffreddamento: PID_Temp crea, con modifiche periodiche, un'oscillazione sul valore di uscita di raffreddamento del valore istantaneo e calcola i parametri PID per il raffreddamento (Struktur Retain.CtrlParams.Cool). Offset di ottimizzazione provvisorio per regolatori di riscaldamento/raffreddamento Se PID_Temp viene impiegato come regolatore di riscaldamento/raffreddamento (Config.ActivateCooling = TRUE), affinché possa essere creata un'oscillazione del valore istantaneo, e l'ottimizzazione fine possa essere eseguita correttamente, il valore di uscita PID (PidOutputSum) sul setpoint deve soddisfare i seguenti presupposti: • Valore di uscita PID positivo per l'ottimizzazione fine del riscaldamento • Valore di uscita PID negativo per l'ottimizzazione fine del raffreddamento Se questa condizione non viene soddisfatta è possibile predefinire un offset provvisorio per l'ottimizzazione fine che verrà emesso sull'uscita attiva opposta: • Offset per l'uscita di raffreddamento (PIDSelfTune.TIR.OutputOffsetCool) nell'ottimizzazione fine del riscaldamento. Prima dell'avvio dell'ottimizzazione, predefinire per quest'ultima un offset negativo per il raffreddamento che sia inferiore al valore di uscita PID (PidOutputSum) sul setpoint nello stato stazionario. • Offset per l'uscita di riscaldamento (PIDSelfTune.TIR.OutputOffsetHeat) nell'ottimizzazione fine del raffreddamento. Prima dell'avvio dell'ottimizzazione, predefinire per quest'ultima un offset positivo per il riscaldamento che sia superiore al valore di uscita PID (PidOutputSum) sul setpoint nello stato stazionario. L'offset predefinito viene poi compensato dall'algoritmo PID consentendo così al valore istantaneo di mantenersi sul setpoint. Sulla base della grandezza dell'offset è possibile adeguare il valore di uscita PID affinché quest'ultimo soddisfi la condizione in precedenza indicata. Per evitare maggiori sovraoscillazioni del valore istantaneo durante la predefinizione dell'offset, quest'ultimo può essere incrementato anche in più fasi. Se PID_Temp abbandona il modo di funzionamento Ottimizzazione fine, l'offset di ottimizzazione viene resettato. Esempio di predefinizione di un offset per l'ottimizzazione fine del raffreddamento: • Senza Offset: – Setpoint (setpoint) = valore istantaneo (ScaledInput) = 80°C – Valore di uscita PID (PidOutputSum) = 30.0 – Valore di uscita per il riscaldamento (OutputHeat) = 30.0 – Valore di uscita per il raffreddamento (OutputCool) = 0.0 Un'oscillazione del valore istantaneo intorno al setpoint, non può essere creata soltanto con l'uscita di raffreddamento. L'ottimizzazione fine in questo caso non riuscirebbe. • Con predefinizione di un offset per l'uscita di riscaldamento (PIDSelfTune.TIR.OutputOffsetHeat) = 80.0 – Setpoint (Setpoint) = valore istantaneo (ScaledInput) = 80°C – Valore di uscita PID (PidOutputSum) = -50.0

State / Mode	Descrizione del modo di funzionamento
	– Valore di uscita per il riscaldamento (OutputHeat) = 80.0 – Valore di uscita per il raffreddamento (OutputCool) = -50.0 Predefinendo un offset per l'uscita di riscaldamento, l'uscita di raffreddamento può creare soltanto un'oscillazione del valore istantaneo intorno al setpoint. L'ottimizzazione fine può essere così eseguita correttamente. Requisiti generali per l'ottimizzazione fine: • L'istruzione PID_Temp deve essere richiamata in un OB di schedulazione orologio. • Non sono previste anomalie. • Il setpoint e il valore istantaneo sono compresi entro i limiti configurati. • Il circuito di regolazione si è assestato nel punto di lavoro. Il punto di lavoro è raggiunto quando il valore istantaneo corrisponde al setpoint. Con la banda morta attivata, può instaurarsi una differenza di regolazione permanente (divergenza tra setpoint e valore istantaneo). Questo può avere ripercussioni negative sull'esecuzione dell'ottimizzazione fine. • ManualEnable = FALSE • Reset = FALSE • Modo di funzionamento automatico (State = 3), inattivo (State = 0) o manuale (State = 4) Requisiti per l'ottimizzazione fine del riscaldamento: • Heat.EnableTuning = TRUE • Cool.EnableTuning = FALSE • Se PID_Temp è stato configurato come regolatore di riscaldamento/raffreddamento (Config.ActivateCooling = TRUE), sul punto di lavoro sul quale eseguire l'ottimizzazione, deve essere attiva l'uscita di riscaldamento (PidOutputSum > 0.0 (vedere Offset di ottimizzazione)). Presupposti per l'ottimizzazione fine raffreddamento: • Heat.EnableTuning = FALSE • Cool.EnableTuning = TRUE • L'uscita di raffreddamento deve essere attivata (Config.ActivateCooling = TRUE). • La commutazione dei parametri PID deve essere attivata (Config.AdvancedCooling = TRUE). • L'uscita di raffreddamento deve essere attiva sul punto di lavoro sul quale eseguire l'ottimizzazione (PidOutputSum < 0.0 (vedere Offset di ottimizzazione)). Lo svolgimento dell'ottimizzazione fine dipende dal modo di funzionamento nel quale viene avviata: • Funzionamento automatico (State = 3) con PIDSelfTune.TIR.RunIn = FALSE (preassegnazione) Per migliorare i parametri PID esistenti mediante un'ottimizzazione, avviare l'ottimizzazione fine dal modo di funzionamento automatico. PID_Temp esegue la regolazione con i parametri PID esistenti finché il circuito di regolazione non si assesta e finché non sono soddisfatti i requisiti per un'ottimizzazione fine. Solo in seguito viene avviata l'ottimizzazione fine. • Inattivo (State = 0), funzionamento manuale (State = 4) o automatico (State = 3) con PIDSelfTune.TIR.RunIn = TRUE Viene effettuato il tentativo di raggiungere il setpoint con il valore di uscita minimo o massimo: – Riscaldamento con valore di uscita minimo o massimo nell'ottimizzazione fine riscaldamento – Raffreddamento con valore di uscita minimo o massimo nell'ottimizzazione fine raffreddamento. Ciò può causare una maggiore sovraoscillazione. Una volta raggiunto il setpoint, ha inizio l'ottimizzazione fine. Qualora il setpoint non possa essere raggiunto, PID_Temp non interrompe automaticamente l'ottimizzazione. Il setpoint viene congelato nella variabile CurrentSetpoint. L'ottimizzazione viene interrotta se: • Setpoint > CurrentSetpoint + CancelTuningLevel o • Setpoint < CurrentSetpoint - CancelTuningLevel

State / Mode	Descrizione del modo di funzionamento
	Il metodo di calcolo dei parametri PID può essere definito separatamente per riscaldamento e raffreddamento con PIDSelfTune.TIR.TuneRuleHeat e PIDSelfTune.TIR.TuneRuleCool. Prima di essere ricalcolati, i parametri PID vengono salvati nella struttura CtrlParamsBackUp e possono essere riattivati tramite LoadBackUp Dopo l'ottimizzazione fine, il regolatore commuta sul modo di funzionamento automatico. In caso di mancata riuscita dell'ottimizzazione fine, la commutazione del modo di funzionamento dipende da ActivateRecoverMode. La fase di "Ottimizzazione fine" viene visualizzata con PIDSelfTune.TIR.State.
3	Funzionamento automatico In funzionamento automatico PID_Temp regola il sistema regolato secondo i parametri preimpostati. Se è soddisfatta una delle seguenti condizioni si ha il passaggio al funzionamento automatico: • Ottimizzazione iniziale conclusa correttamente • Ottimizzazione fine conclusa correttamente • Modifica del parametro di transito Mode al valore 3 e un fronte di salita in ModeActivate. La commutazione dal funzionamento automatico al funzionamento manuale è bumpless solo nell'editor di messa in servizio. In funzionamento automatico viene tenuta in considerazione la variabile ActivateRecoverMode.
4	Funzionamento manuale Nel funzionamento manuale impostare manualmente un valore di uscita PID nel parametro ManualValue. I valori derivanti da questo valore manuale sulle uscite per riscaldamento e raffreddamento, risultano dalla graduazione di scala dell'uscita configurata. Il modo di funzionamento si può attivare anche con ManualEnable = TRUE. Si consiglia di modificare i modi di funzionamento solo tramite Mode e ModeActivate. La commutazione dal funzionamento manuale al funzionamento automatico è bumpless. Nel funzionamento manuale viene tenuta in considerazione la variabile ActivateRecoverMode.
5	Valore di uscita sostitutivo con controllo errori L'algoritmo di regolazione è disattivato. La variabile SetSubstituteOutput stabilisce quale valore di uscita PID (PidOutputSum) venga emesso in questo modo di funzionamento. • SetSubstituteOutput = FALSE: ultimo valore di uscita PID valido • SetSubstituteOutput = TRUE: Valore di uscita sostitutivo (SubstituteOutput) Questa modalità di funzionamento non può essere attivata con Mode = 5 . Viene attivata in caso di errore invece del modo di funzionamento "Inattivo" se sono soddisfatte tutte le seguenti condizioni: • Funzionamento automatico (State = 3) • ActivateRecoverMode = TRUE • Si sono verificati uno o più errori nei quali è attiva ActivateRecoverMode. Non appena gli errori vengono eliminati PID_Temp torna al funzionamento automatico.

Comportamento ENO

Se State = 0, ENO = FALSE.

Se State ≠ 0, ENO = TRUE.

Commutazione automatica del modo di funzionamento durante la messa in servizio

Al termine dell'ottimizzazione iniziale e dell'ottimizzazione fine senza errori viene attivato il funzionamento automatico. La tabella seguente mostra come cambiano Mode e State durante un'ottimizzazione iniziale corretta.

N° di ciclo	Mode	State	Operazione
0	4	4	Imposta Mode = 1
1	1	4	Imposta ModeActivate = TRUE
1	4	1	Il valore di State viene salvato in Mode L'ottimizzazione iniziale viene avviata
n	4	1	Ottimizzazione iniziale riuscita
n	3	3	Il funzionamento automatico viene avviato

In caso di errore PID_Temp cambia automaticamente modo di funzionamento.

La tabella seguente mostra come cambiano Mode e State durante un'ottimizzazione iniziale scorretta.

N° di ciclo	Mode	State	Operazione
0	4	4	Imposta Mode = 1
1	1	4	Imposta ModeActivate = TRUE
1	4	1	Il valore di State viene salvato in Mode L'ottimizzazione iniziale viene avviata
n	4	1	Ottimizzazione iniziale interrotta
n	4	4	Il funzionamento manuale viene avviato

Se ActivateRecoverMode = TRUE viene attivato il modo di funzionamento salvato in Mode. All'avvio dell'ottimizzazione iniziale o fine, PID_Temp ha salvato il valore di State nel parametro di transitoMode. PID_Temp passa al modo di funzionamento dal quale è stata avviata l'ottimizzazione.

Se ActivateRecoverMode = FALSE si ha il passaggio al modo di funzionamento "Inattivo".

Vedere anche

[Parametri di uscita di PID_Temp \(Pagina 398\)](#)

[Parametri di transito di PID_Temp \(Pagina 400\)](#)

10.3.4.8 Parametri ErrorBits PID_Temp

Se si verificano diversi errori contemporaneamente, i valori di ErrorBits vengono visualizzati in un'addizione binaria. Se ad es. viene visualizzato ErrorBits = 16#0000_0003 significa che sono presenti contemporaneamente gli errori 16#0000_0001 e 16#0000_0002.

ErrorBits (DW#16#...)	Descrizione
0000_0000	Non sono presenti errori.
0000_0001	Il parametro "Input" non rientra nei limiti del valore istantaneo. • Input > Config.InputUpperLimit o • Input < Config.InputLowerLimit Se prima del verificarsi dell'errore era attivo il funzionamento automatico e ActivateRecoverMode = TRUE, PID_Temp rimane in questo modo di funzionamento. Se prima del verificarsi dell'errore era attivo il funzionamento manuale e ActivateRecoverMode = TRUE, PID_Temp rimane in questo modo di funzionamento. Se prima del verificarsi dell'errore era attivo il modo di funzionamento Ottimizzazione iniziale o Ottimizzazione fine e ActivateRecoverMode = TRUE, PID_Temp passa al modo di funzionamento salvato in Mode.
0000_0002	Valore del parametro "Input_PER" non valido. Controllare l'eventuale presenza di un errore sull'ingresso analogico. Se prima del verificarsi dell'errore era attivo il funzionamento automatico e ActivateRecoverMode = TRUE, PID_Temp emette il valore di uscita sostitutivo configurato. Non appena l'errore viene eliminato PID_Temp torna al funzionamento automatico. Se prima del verificarsi dell'errore era attivo il funzionamento manuale e ActivateRecoverMode = TRUE, PID_Temp rimane in questo modo di funzionamento. Se prima del verificarsi dell'errore era attivo il modo di funzionamento Ottimizzazione iniziale o Ottimizzazione fine e ActivateRecoverMode = TRUE, PID_Temp passa al modo di funzionamento salvato in Mode.
0000_0004	Errore durante l'Ottimizzazione fine. Non è stato possibile mantenere l'oscillazione del valore istantaneo. Se PID_Temp viene impiegato come regolatore di riscaldamento/raffreddamento (Config.ActivateCooling = TRUE), affinché possa essere creata un'oscillazione del valore istantaneo e l'ottimizzazione fine possa avvenire correttamente, il valore di uscita PID (PidOutputSum) nel setpoint • deve essere positivo per l'ottimizzazione fine del riscaldamento • negativo per l'ottimizzazione fine del raffreddamento Se queste condizioni non sono soddisfatte utilizzare l'offset di ottimizzazione (variabili PIDSelfTune.TIR.OutputOffsetCool e PIDSelfTune.TIR.OutputOffsetHeat), vedere Ottimizzazione fine (Pagina 195). Se prima del verificarsi dell'errore ActivateRecoverMode = TRUE, PID_Temp interrompe l'ottimizzazione e commuta nel modo di funzionamento salvato in Mode.
0000_0008	Errore all'avvio dell'Ottimizzazione iniziale. Il valore istantaneo è troppo vicino al setpoint o maggiore di quest'ultimo. Avviare l'ottimizzazione fine. Se prima del verificarsi dell'errore ActivateRecoverMode = TRUE, PID_Temp interrompe l'ottimizzazione e commuta nel modo di funzionamento salvato in Mode.
0000_0010	Il setpoint è stato modificato durante l'ottimizzazione. Nella variabile CancelTuningLevel è possibile impostare l'oscillazione ammessa del setpoint. Se prima del verificarsi dell'errore ActivateRecoverMode = TRUE, PID_Temp interrompe l'ottimizzazione e commuta nel modo di funzionamento salvato in Mode.
0000_0020	L'Ottimizzazione iniziale non è consentita durante l'Ottimizzazione fine. Se prima del verificarsi dell'errore ActivateRecoverMode = TRUE, PID_Temp rimane nel modo di funzionamento Ottimizzazione fine.
0000_0040	Errore durante l'Ottimizzazione iniziale. Il raffreddamento non ha ridotto il valore istantaneo. Se prima del verificarsi dell'errore ActivateRecoverMode = TRUE, PID_Temp interrompe l'ottimizzazione e commuta nel modo di funzionamento salvato in Mode.

ErrorBits (DW#16#...)	Descrizione
0000_0100	Un errore durante l'Ottimizzazione fine ha causato parametri non validi. Se prima del verificarsi dell'errore ActivateRecoverMode = TRUE, PID_Temp interrompe l'ottimizzazione e commuta nel modo di funzionamento salvato in Mode.
0000_0200	Valore del parametro "Input" non valido: il formato numerico del valore non è valido. Se prima del verificarsi dell'errore era attivo il funzionamento automatico e ActivateRecoverMode = TRUE, PID_Temp emette il valore di uscita sostitutivo configurato. Non appena l'errore viene eliminato PID_Temp torna al funzionamento automatico. Se prima del verificarsi dell'errore era attivo il funzionamento manuale e ActivateRecoverMode = TRUE, PID_Temp rimane in questo modo di funzionamento. Se prima del verificarsi dell'errore era attivo il modo di funzionamento Ottimizzazione iniziale o Ottimizzazione fine e ActivateRecoverMode = TRUE, PID_Temp passa al modo di funzionamento salvato in Mode.
0000_0400	Calcolo del valore di uscita non riuscito. Controllare i parametri PID. Se prima del verificarsi dell'errore era attivo il funzionamento automatico e ActivateRecoverMode = TRUE, PID_Temp emette il valore di uscita sostitutivo configurato. Non appena l'errore viene eliminato PID_Temp torna al funzionamento automatico. Se prima del verificarsi dell'errore era attivo il modo di funzionamento Ottimizzazione iniziale o Ottimizzazione fine e ActivateRecoverMode = TRUE, PID_Temp passa al modo di funzionamento salvato in Mode.
0000_0800	Errore del tempo di campionamento: PID_Temp non viene richiamato entro il tempo di campionamento dell'OB di schedulazione orologio. Si raccomanda di richiamare PID_Temp in un OB di schedulazione orologio senza condizioni e di eseguire la relativa attivazione o disattivazione tramite il modo di funzionamento nel parametro Mode. Richiami condizionati o il richiamo nell'OB1 possono ripercuotersi negativamente sulla qualità della regolazione. La sorveglianza del tempo di campionamento può essere disattivata con CycleTime.EnMonitoring = FALSE. Se prima del verificarsi dell'errore era attivo il funzionamento automatico e ActivateRecoverMode = TRUE, PID_Temp rimane in questo modo di funzionamento. Se prima del verificarsi dell'errore era attivo il funzionamento manuale e ActivateRecoverMode = TRUE, PID_Temp rimane in questo modo di funzionamento. Se prima del verificarsi dell'errore era attivo il modo di funzionamento Ottimizzazione iniziale o Ottimizzazione fine e ActivateRecoverMode = TRUE, PID_Temp passa al modo di funzionamento salvato in Mode. Se quest'errore si è verificato nella simulazione con PLCSIM, osservare le avvertenze riportate alla sezione Simulazione di PID_Temp con PLCSIM (Pagina 215).
0000_1000	Valore non valido nel parametro "Setpoint" oppure "SubstituteSetpoint": il formato numerico del valore non è valido. Se prima del verificarsi dell'errore era attivo il funzionamento automatico e ActivateRecoverMode = TRUE, PID_Temp emette il valore di uscita sostitutivo configurato. Non appena l'errore viene eliminato PID_Temp torna al funzionamento automatico. Se prima del verificarsi dell'errore era attivo il funzionamento manuale e ActivateRecoverMode = TRUE, PID_Temp rimane in questo modo di funzionamento. Se prima del verificarsi dell'errore era attivo il modo di funzionamento Ottimizzazione iniziale o Ottimizzazione fine e ActivateRecoverMode = TRUE, PID_Temp passa al modo di funzionamento salvato in Mode.
0001_0000	Valore del parametro ManualValue non valido. Il formato numerico del valore non è valido. Se prima del verificarsi dell'errore ActivateRecoverMode = TRUE, PID_Temp rimane nel modo di funzionamento manuale e impiega SubstituteOutput come valore di uscita PID. Non appena viene predefinito un valore valido in ManualValue, PID_Temp lo utilizza come valore di uscita PID.
0002_0000	Valore della variabile SubstituteOutput non valido. Il formato numerico del valore non è valido. PID_Temp rimane nel modo di funzionamento "Valore di uscita sostitutivo con controllo errori" o nel funzionamento manuale e impiega, come valore di uscita PID, il limite inferiore del valore di uscita PID per il riscaldamento (Config.Output.Heat.PidLowerLimit). Non appena viene predefinito un valore valido in SubstituteOutput, PID_Temp lo utilizza come valore di uscita PID.

ErrorBits (DW#16#...)	Descrizione
0004_0000	Valore del parametro Disturbance non valido. Il formato numerico del valore non è valido. Se prima del verificarsi dell'errore era attivo il funzionamento automatico e ActivateRecoverMode = TRUE, Disturbance viene impostato su zero. PID_Temp rimane nel funzionamento automatico. Se prima del verificarsi dell'errore era attivo il modo di funzionamento ottimizzazione iniziale o ottimizzazione fine e ActivateRecoverMode = TRUE, PID_Temp passa al modo di funzionamento salvato in Mode. Se Disturbance non ha alcuna influenza sul valore di uscita nella fase attuale, l'ottimizzazione non viene interrotta.
0020_0000	Errore nel master nella cascata: Gli Slaves non si trovano nel funzionamento automatico oppure hanno attivato il setpoint sostitutivo e impediscono l'ottimizzazione del master. Se prima del verificarsi dell'errore ActivateRecoverMode = TRUE, PID_Temp interrompe l'ottimizzazione e commuta nel modo di funzionamento salvato in Mode.
0040_0000	Fintantoché il raffreddamento è attivo, l'ottimizzazione iniziale riscaldamento non è consentita. Se prima del verificarsi dell'errore ActivateRecoverMode = TRUE, PID_Temp interrompe l'ottimizzazione e commuta nel modo di funzionamento salvato in Mode.
0080_0000	Per l'avvio dell'Ottimizzazione iniziale raffreddamento il valore istantaneo deve essere vicino al setpoint. Se prima del verificarsi dell'errore ActivateRecoverMode = TRUE, PID_Temp interrompe l'ottimizzazione e commuta nel modo di funzionamento salvato in Mode.
0100_0000	Errore all'avvio dell'ottimizzazione: Heat.EnableTuning e Cool.EnableTuning non sono stati impostati o non si addicono alla configurazione. Se prima del verificarsi dell'errore ActivateRecoverMode = TRUE, PID_Temp interrompe l'ottimizzazione e commuta nel modo di funzionamento salvato in Mode.
0200_0000	L'Ottimizzazione iniziale raffreddamento presuppone che l'Ottimizzazione iniziale riscaldamento sia riuscita. Se prima del verificarsi dell'errore ActivateRecoverMode = TRUE, PID_Temp interrompe l'ottimizzazione e commuta nel modo di funzionamento salvato in Mode.
0400_0000	Errore all'avvio dell'ottimizzazione fine: Heat.EnableTuning e Cool.EnableTuning non possono essere impostati contemporaneamente. Se prima del verificarsi dell'errore ActivateRecoverMode = TRUE, PID_Temp interrompe l'ottimizzazione e commuta nel modo di funzionamento salvato in Mode.
0800_0000	Un errore durante il calcolo dei parametri PID ha causato parametri non validi. I parametri non validi vengono respinti e i parametri PID originali vengono mantenuti invariati. Se quest'errore si presenta durante l'ottimizzazione iniziale, verificare l'esistenza dei seguenti presupposti: - Ottimizzazione iniziale riscaldamento oppure Ottimizzazione iniziale riscaldamento e raffreddamento. Che il valore di uscita PID non venga limitato dal limite superiore per il riscaldamento già prima dell'avvio dell'ottimizzazione iniziale. - Ottimizzazione iniziale raffreddamento: Che il valore di uscita PID non venga limitato dal limite inferiore per il raffreddamento già prima dell'avvio dell'ottimizzazione iniziale. Per avviare l'ottimizzazione iniziale riscaldamento o l'ottimizzazione iniziale riscaldamento e raffreddamento dal funzionamento automatico, si consiglia di eseguire la modifica richiesta del setpoint contemporaneamente al fronte di salita in ModeActivate. In questo modo si evita che il valore di uscita PID raggiunga la limitazione tra la modifica al setpoint e l'avvio dell'ottimizzazione iniziale. In via alternativa è possibile raggiungere questo scopo anche effettuando l'avvio nel funzionamento manuale o nel modo di funzionamento "Inattivo". Si distinguono i casi seguenti: - Se prima del verificarsi dell'errore era attivo il funzionamento automatico e ActivateRecoverMode = TRUE, PID_Temp rimane in questo modo di funzionamento. - Se prima del verificarsi dell'errore era attivo il funzionamento manuale e ActivateRecoverMode = TRUE, PID_Temp rimane in questo modo di funzionamento. - Se prima del verificarsi dell'errore era attivo il modo di funzionamento Ottimizzazione iniziale oppure ottimizzazione fine e ActivateRecoverMode = TRUE, passa al modo di funzionamento salvato in Mode.

10.3.4.9 Variabile ActivateRecoverMode PID_Temp

La variabile ActivateRecoverMode determina il comportamento in caso di errore. Il parametro Error indica se attualmente è presente un errore. Se l'errore è stato eliminato, viene visualizzato Error = FALSE. Il parametro ErrorBits mostra gli errori che si sono verificati.

Funzionamento automatico e funzionamento manuale

ATTENZIONE

L'impianto può subire danneggiamenti.

Se ActivateRecoverMode = TRUE, in presenza di errori PID_Temp rimane nel funzionamento automatico o manuale anche in caso di superamento dei limiti del valore istantaneo.

In seguito a ciò l'impianto può subire danneggiamenti.

Configurare per il sistema regolato un comportamento da adottare in caso di errore che protegga l'impianto da danneggiamenti.

ActivateRecover-Mode	Descrizione
FALSE	In caso di errore PID_Temp passa al modo di funzionamento "Inattivo". Il regolatore viene attivato solo al verificarsi di un fronte di discesa in Reset o di un fronte di salita in ModeActivate.
TRUE	Funzionamento automatico Se si verificano errori frequenti in funzionamento automatico, questa impostazione peggiora il comportamento del regolatore perché PID_Temp commuta tra valore di uscita calcolato e valore di uscita sostitutivo a ogni errore. In questo caso controllare il parametro ErrorBits ed eliminare la causa dell'errore. Se prima del verificarsi dell'errore era attivo il funzionamento automatico e si verificano uno o più dei seguenti errori, PID_Temp rimane in questo modo di funzionamento. • 0000001h: Il parametro "Input" non rientra nei limiti del valore istantaneo. • 0000800h: Errore del tempo di campionamento • 0040000h: Valore del parametro Disturbance non valido. • 8000000h: Errore durante il calcolo dei parametri PID Se prima del verificarsi dell'errore era attivo il funzionamento automatico e si verificano uno o più dei seguenti errori, PID_Temp commuta nel modo di funzionamento "Valore di uscita sostitutivo con controllo errori": • 0000002h: Valore del parametro Input_PER non valido. • 0000200h: Valore del parametro Input non valido. • 0000400h: Calcolo del valore di uscita non riuscito. • 0001000h: Valore non valido nel parametro Setpoint oppure SubstituteSetpoint. Non appena gli errori vengono eliminati PID_Temp torna al funzionamento automatico. Se nel modo di funzionamento "Valore di uscita sostitutivo con controllo errori" si verifica il seguente errore, fino a quando quest'ultimo non sarà eliminato, PID_Temp imposta il valore di uscita PID su Config.Output.Heat.PidLowerLimit: • 0020000h: Valore della variabile SubstituteOutput non valido. Il formato numerico del valore non è valido. Questo comportamento è indipendente da SetSubstituteOutput.

ActivateRecover-Mode	Descrizione
TRUE	Funzionamento manuale Se prima del verificarsi dell'errore era attivo il funzionamento manuale e si verificano uno o più errori, PID_Temp rimane in questo modo di funzionamento. Se nel funzionamento manuale si verifica il seguente errore, fino a quando quest'ultimo non sarà eliminato, PID_Temp imposta il valore di uscita PID su SubstituteOutput:: 0010000h: Valore del parametro ManualValue non valido. Il formato numerico del valore non è valido. Se nel funzionamento manuale è presente l'errore 0010000h e si verifica inoltre l'errore, fino a quando quest'ultimo non sarà eliminato, PID_Temp imposta il valore di uscita PID su Config.Output.Heat.PidLowerLimit: 0020000h: Valore della variabile SubstituteOutput non valido. Il formato numerico del valore non è valido. Questo comportamento è indipendente da SetSubstituteOutput.

Ottimizzazione iniziale e ottimizzazione fine

ActivateRecover-Mode	Descrizione
FALSE	In caso di errore PID_Temp passa al modo di funzionamento "Inattivo". Il regolatore viene attivato solo al verificarsi di un fronte di discesa in Reset o di un fronte di salita in ModeActivate.
TRUE	Se si verifica l'errore seguente PID_Temp rimane nel modo di funzionamento attivo: 0000020h: L'ottimizzazione iniziale non è consentita durante l'ottimizzazione fine. I seguenti errori vengono ignorati: 0010000h: Valore del parametro ManualValue non valido. 0020000h: Valore della variabile SubstituteOutput non valido. Con tutti gli altri errori PID_Temp interrompe l'ottimizzazione e passa al modo di funzionamento dal quale è stata avviata l'ottimizzazione.

10.3.4.10 Variabile Warning PID_Temp

Se sono presenti più avvisi contemporaneamente, i valori della variabile Warning vengono visualizzati in un'addizione binaria. La visualizzazione dell'avviso 16#0000_0003 ad es. indica la presenza contemporanea degli avvisi 16#0000_0001 e 16#0000_0002.

Warning (DW#16#....)	Descrizione
0000_0000	Non è presente alcun avviso.
0000_0001	Durante l'ottimizzazione iniziale non è stato trovato il punto di flesso.
0000_0004	Il setpoint è stato circoscritto ai limiti impostati.
0000_0008	Per il metodo di calcolo selezionato non sono state definite tutte le proprietà necessarie per il sistema regolato. In via sostitutiva i parametri PID sono stati calcolati con il metodo TIR.TuneRuleHeat = 3 e TIR.TuneRuleCool = 3.
0000_0010	Non è stato possibile commutare il modo di funzionamento perché Reset = TRUE o ManualEnable = TRUE.
0000_0020	Il tempo di campionamento dell'algoritmo PID viene circoscritto dal tempo di ciclo dell'OB richiamaente. Per ottenere risultati migliori utilizzare tempi di ciclo dell'OB più brevi.
0000_0040	Il valore istantaneo ha superato uno dei suoi limiti di avviso.
0000_0080	Valore di Mode non valido. Il modo di funzionamento non viene commutato.

Warning (DW#16#....)	Descrizione
0000_0100	Il valore manuale è stato circoscritto ai limiti del valore di uscita PID.
0000_0200	La regola per l'ottimizzazione indicata non è supportata. Non vengono calcolati parametri PID.
0000_1000	Non è possibile raggiungere il valore di uscita sostitutivo perché non rientra nei limiti del valore di uscita.
0000_4000	La selezione del valore di uscita effettuata per riscaldamento e raffreddamento non viene supportata. Viene utilizzata soltanto l'uscita OutputHeat e OutputCool.
0000_8000	Valore di PIDSelfTune.SUT.AdaptDelayTime non valido. Viene utilizzato il valore di default 0.
0001_0000	Valore di PIDSelfTune.SUT.CoolingMode non valido. Viene utilizzato il valore di default 0.
0002_0000	L'attivazione del raffreddamento (variabile Config.ActivateCooling) non viene supportata in un regolatore utilizzato come master (variabile Config.Cascade.IsMaster). PID_Temp opera come regolatore di riscaldamento. Impostare su FALSE la variabile Config.ActivateCooling.
0004_0000	Valore non valido in Retain.CtrlParams.Heat.Gain, Retain.CtrlParams.Cool.Gain oppure Config.CoolFactor. PID_Temp supporta soltanto valori positivi per il guadagno proporzionale (riscaldamento e raffreddamento) e per il fattore di raffreddamento. Con il valore di uscita PID 0.0, il funzionamento automatico rimane attivo. Il componente integrale viene arrestato.

I seguenti avvisi vengono cancellati non appena ne viene eliminata la causa o alla ripetizione dell'operazione con parametri validi:

- 16#0000_0001
- 16#0000_0004
- 16#0000_0008
- 16#0000_0040
- 16#0000_0100

Tutti gli altri avvisi vengono cancellati in presenza di un fronte di salita in Reset o ErrorAck.

10.3.4.11 Variabile PwmPeriode

Se con l'impiego di OutputHeat_PWM o OutputCool_PWM, il tempo di campionamento dell'algoritmo PID (Retain.CtrlParams.Heat.Cycle o Retain.CtrlParams.Heat.Cycle) e pertanto la durata periodo della modulazione ampiezza impulso, fossero particolarmente lunghi è possibile, per migliorare il livellamento del valore istantaneo, indicare, divergendo dai valori normalmente previsti, una durata periodo più breve nei parametri Config.Output.Heat.PwmPeriode o Config.Output.Cool.PwmPeriode.

Durata periodo della modulazione ampiezza impulsi in OutputHeat_PWM

Durata periodo di PWM sull'uscita OutputHeat_PWM in funzione di Config.Output.Heat.PwmPeriode:

- Heat.PwmPeriode = 0.0 (default)
Il tempo di campionamento dell'algoritmo PID per il riscaldamento (Retain.CtrlParams.Heat.Cycle) viene utilizzato come durata periodo di PWM.
- Heat.PwmPeriode > 0.0
Il valore viene arrotondato ad un multiplo intero del tempo di campionamento PID_Temp (CycleTime.Value) e utilizzato come durata periodo di PWM.
Il valore deve soddisfare le seguenti condizioni:
 - Heat.PwmPeriode ≤ Retain.CtrlParams.Heat.Cycle
 - Heat.PwmPeriode > Config.Output.Heat.MinimumOnTime
 - Heat.PwmPeriode > Config.Output.Heat.MinimumOffTime

Durata periodo della modulazione ampiezza impulsi in OutputCool_PWM

Durata periodo di PWM sull'uscita OutputCool_PWM in funzione di Config.Output.Cool.PwmPeriode e del metodo di riscaldamento/raffreddamento:

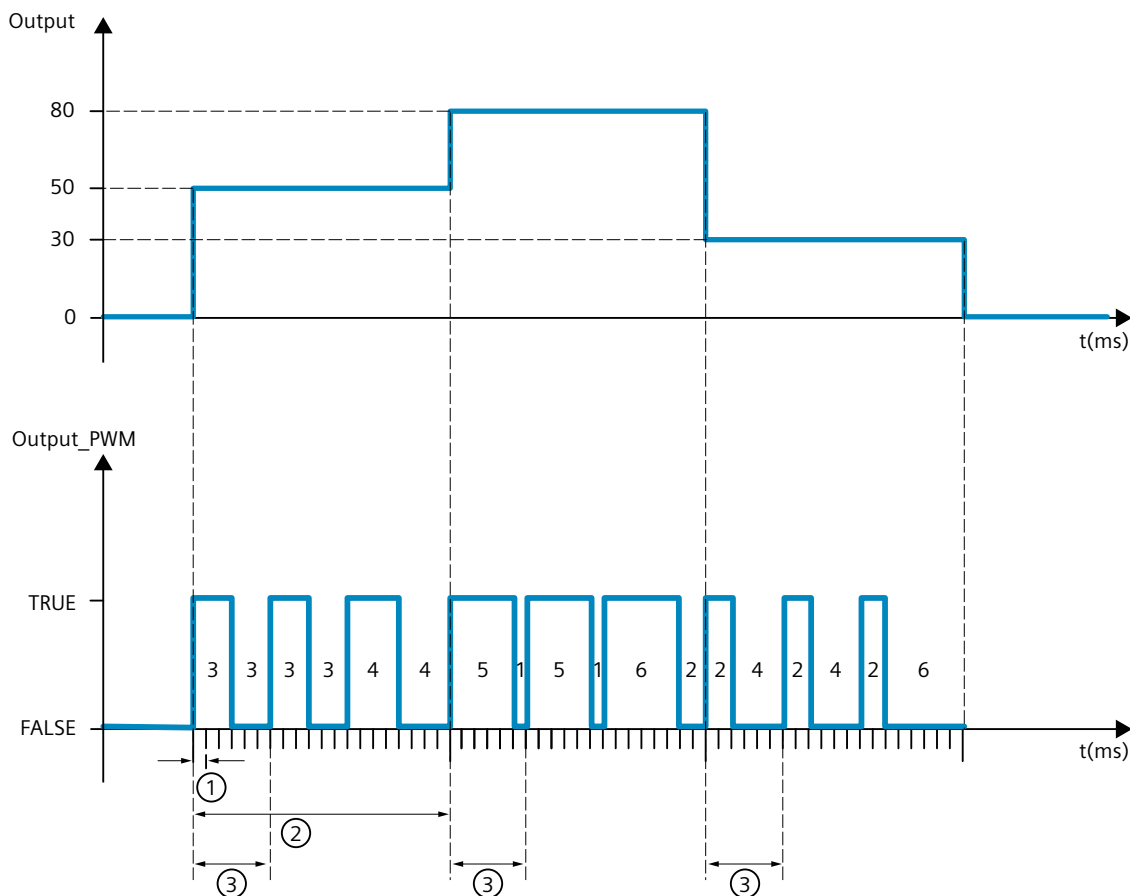
- Cool.PwmPeriode = 0.0 e fattore di raffreddamento (Config.AdvancedCooling = FALSE):
Il tempo di campionamento dell'algoritmo PID per il riscaldamento (Retain.CtrlParams.Heat.Cycle) viene utilizzato come durata periodo di PWM.
- Cool.PwmPeriode = 0.0 e commutazione dei parametri PID (Config.AdvancedCooling = TRUE):
Il tempo di campionamento dell'algoritmo PID per il raffreddamento (Retain.CtrlParams.Cool.Cycle) viene utilizzato come durata periodo di PWM.
- Cool.PwmPeriode > 0.0:
Il valore viene arrotondato ad un multiplo intero del tempo di campionamento PID_Temp (CycleTime.Value) e utilizzato come durata periodo di PWM.
Il valore deve soddisfare le seguenti condizioni:
 - Cool.PwmPeriode ≤ Retain.CtrlParams.Cool.Cycle oppure Retain.CtrlParams.Heat.Cycle
 - Cool.PwmPeriode > Config.Output.Cool.MinimumOnTime
 - Cool.PwmPeriode > Config.Output.Cool.MinimumOffTime

Config.Output.Cool.PwmPeriode è efficace soltanto se l'uscita di raffreddamento è attivata (Config.ActivateCooling =TRUE).

In caso di impiego di PwmPeriode, la precisione del segnale di uscita PWM viene determinata dal rapporto di PwmPeriode rispetto al tempo di campionamento PID_Temp (tempo di ciclo dell'OB). PwmPeriode deve essere almeno dieci volte maggiore del tempo di campionamento PID_Temp.

Se il tempo di campionamento dell'algoritmo PID non è un multiplo intero di PwmPeriode, ogni ultimo periodo di PWM all'interno del tempo di campionamento dell'algoritmo PID viene prorogato in modo corrispondente.

Esempio di OutputHeat_PWM



- ① Tempo di campionamento PID_Temp = 100.0 ms (tempo di ciclo dell'OB di schedulazione orologio richiamante, variabile CycleTime.Value)
- ② Tempo di campionamento algoritmo PID = 2000.0 ms (variabile Retain.CtrlParams.Heat.Cycle)
- ③ Durata periodo di PWM per il riscaldamento = 600.0 ms (variabile Config.Output.Heat.PwmPeriode)

10.3.4.12 Variabile IntegralResetMode

La variabile IntegralResetMode descrive le modalità di assegnazione dei valori al componente I PIDCtrl.IOutputOld:

- alla commutazione dal modo di funzionamento "Inattivo" al modo "Funzionamento automatico"
- con il fronte TRUE -> FALSE sui parametri Reset e Mode = 3

Questa impostazione è attiva solo per un ciclo e soltanto se è attivato il componente I (variabili Retain.CtrlParams.Heat.Ti e Retain.CtrlParams.Cool.Ti > 0.0)

IntegralReset-Mode	Descrizione
0	Livella Al parametro PIDCtrl.IOutputOld viene preassegnato un valore tale da consentire la commutazione continua, in altri termini, il funzionamento automatico si avvia presupponendo un valore di uscita = 0.0 (parametro PidOutputSum). Non si avrà alcun gradino del valore di uscita indipendentemente dalla differenza di regolazione (setpoint – valore istantaneo).
1	Elimina Se viene utilizzata quest'opzione, si raccomanda di impostare su 1.0 la ponderazione del componente P (variabili Retain.CtrlParams.Heat.PWeighting e Retain.CtrlParams.Cool.PWeighting) Il valore di PIDCtrl.IOutputOld viene cancellato. Se è presente una differenza di regolazione, si forma un gradino del valore di uscita PID. La direzione del salto del valore di uscita dipende dalla ponderazione attivata per il componente P (variabili Retain.CtrlParams.Heat.PWeighting e Retain.CtrlParams.Cool.PWeighting) e dalla differenza di regolazione: • ponderazione attiva del componente P = 1.0: La direzione del salto del valore di uscita e la differenza di regolazione hanno segni identici. Esempio: Se il valore istantaneo è inferiore al setpoint (differenza di regolazione positiva), il valore di uscita PID salta su un valore positivo. • ponderazione attiva del componente P < 1.0: In presenza di differenze di regolazione di una certa entità, il salto del valore di uscita PID e la differenza di regolazione hanno segni identici. Esempio: Se il valore istantaneo è di gran lunga inferiore al setpoint (differenza di regolazione positiva), il valore di uscita PID salta su un valore positivo. In presenza di lievi differenze di regolazione, il salto del valore di uscita PID e la differenza di regolazione hanno segni diversi. Esempio: Se il valore istantaneo è leggermente inferiore al setpoint (differenza di regolazione positiva), il valore di uscita PID salta su un valore negativo. Normalmente questo comportamento non è auspicabile in quanto comporta un incremento temporaneo della differenza di regolazione. Minore è la ponderazione configurata per il componente P, maggiore deve essere la differenza di regolazione al fine di realizzare un salto del valore di uscita PID con segno identico. Se viene utilizzata quest'opzione, si raccomanda di impostare su 1.0 la ponderazione del componente P (variabili Retain.CtrlParams.Heat.PWeighting e Retain.CtrlParams.Cool.PWeighting) Diversamente sussiste il rischio che si verifichi il comportamento indesiderato descritto nel caso delle differenze di regolazione lievi. In via alternativa è possibile utilizzare anche IntegralResetMode = 4. Quest'opzione garantisce segni identici nel salto del valore di uscita PID e nella differenza di regolazione a prescindere dalla ponderazione configurata per il componente P e dalla differenza di regolazione:

IntegralReset-Mode	Descrizione
2	Mantieni Il valore di PIDCtrl.IOutputOld non viene modificato. È possibile impostare un nuovo valore tramite il programma utente.
3	Preassegna Al parametro PIDCtrl.IOutputOld viene preassegnato automaticamente un valore tale da far supporre che nell'ultimo ciclo ci sia stato = PidOutputSum = OverwriteInitialOutputValue
4	In caso di variazioni del setpoint (soltanto per PID_Temp con la versione ≥ 1.1) Il valore di PIDCtrl.IOutputOld viene preimpostato automaticamente in modo da determinare un salto del valore di uscita PID analogo a quello di un regolatore PI nel funzionamento automatico in presenza di una transizione del setpoint dal valore istantaneo attuale al setpoint attuale. Se è presente una differenza di regolazione, si forma un gradino del valore di uscita PID. La direzione del salto del valore di uscita PID e la differenza di regolazione hanno segni identici. Esempio: Se il valore istantaneo è inferiore al setpoint (differenza di regolazione positiva), il valore di uscita PID salta su un valore positivo. Ciò prescinde dalla ponderazione configurata per il componente P e dalla differenza di regolazione.

Assegnando a IntegralResetMode un valore al di fuori del campo valori valido, PID_Temp assume lo stesso comportamento attivato con la preassegnazione di IntegralResetMode:

- PID_Temp da V1.0: IntegralResetMode = 1
- PID_Temp da V1.1: IntegralResetMode = 4

10.4 Funzioni di base PID

10.4.1 CONT_C

10.4.1.1 Descrizione di CONT_C

L'istruzione CONT_C consente la regolazione di processi tecnici con grandezze di ingresso e di uscita continue nei sistemi di automazione SIMATIC S7. Mediante parametrizzazione è possibile attivare o disattivare le funzioni parziali del regolatore PID per adattarlo al sistema regolato. Oltre alle funzioni nel ramo del setpoint e del valore istantaneo, questa istruzione realizza un regolatore PID completo con uscita continua per grandezze regolanti e possibilità di influenzare manualmente il valore regolante.

Impiego

Il regolatore può essere utilizzato come regolatore PID a valore fisso in regolazioni singole o multiloop come regolatore in cascata, di miscela o di rapporto. Il funzionamento si basa sull'algoritmo di regolazione PID del regolatore digitale con segnale di uscita analogico, eventualmente completato da un generatore di impulsi per la generazione di segnali di uscita modulati alla durata dell'impulso per regolazioni a 2 o 3 punti con organi attuatori proporzionali.

Richiamo

L'istruzione CONT_C dispone di una routine di inizializzazione che viene eseguita se è impostato il parametro di ingresso COM_RST = TRUE. Durante l'inizializzazione l'integratore viene impostato sul valore di inizializzazione I_ITVAL. Tutte le altre uscite di segnale vengono impostate a zero. Dopo l'esecuzione della routine di inizializzazione, COM_RST = FALSE deve essere impostato.

Il calcolo dei valori nei blocchi di regolazione è corretto solo se il blocco viene richiamato a intervalli regolari. Perciò è opportuno richiamare i blocchi di regolazione in un OB di schedulazione orologio (OB 30-OB 38). Il tempo di campionamento va indicato nel parametro CYCLE.

Informazioni di errore

Il blocco non analizza la parola di segnalazione di errore RET_VAL.

10.4.1.2 Funzionamento di CONT_C

Ramo del setpoint

Il setpoint viene immesso nell'ingresso SP_INT in formato in virgola mobile.

Ramo del valore istantaneo

Il valore istantaneo può essere letto in formato periferia e in virgola mobile. La funzione CRP_IN converte il valore di periferia PV_PER in un formato in virgola mobile compreso tra -100 e +100% secondo la regola seguente:

Uscita di CRP_IN = $PV\_PER * 100 / 27648$

La funzione PV_NORM normalizza l'uscita di CRP_IN secondo la regola seguente:

Uscita di PV_NORM = (uscita di CRP_IN) * PV_FAC + PV_OFF

A PV_FAC è preassegnato 1 e a PV_OFF 0.

Calcolo della differenza di regolazione

La differenza tra setpoint e valore istantaneo rappresenta la differenza di regolazione. Per sopprimere una piccola oscillazione continua causata dalla quantizzazione delle grandezze regolanti (ad es. nel caso di una modulazione dell'ampiezza degli impulsi con PULSEGEN), la differenza di regolazione viene condotta attraverso una banda morta (DEADBAND). Se DEADB_W = 0 la banda morta è disattivata.

Algoritmo PID

L'algoritmo PID utilizza un algoritmo di posizione. I componenti proporzionale, integrale (INT) e differenziale (DIF) sono collegati in parallelo e possono essere attivati o disattivati singolarmente. In questo modo è possibile parametrizzare i regolatori P, PI, PD e PID. Sono comunque possibili anche semplici regolatori I.

Elaborazione del valore manuale

È possibile commutare tra funzionamento manuale e funzionamento automatico. Nel funzionamento manuale la grandezza regolante viene collegata a un valore manuale.

L'integratore (INT) viene impostato internamente su LMN - LMN_P - DISV mentre il derivatore (DIF) viene azzerato e sincronizzato internamente. La commutazione al funzionamento automatico avviene così in modo continuo.

Elaborazione del valore regolante

Con la funzione LMNLIMIT il valore regolante viene limitato a valori predefinitibili. Il superamento dei limiti da parte della grandezza di ingresso viene indicato da bit di segnalazione.

La funzione LMN_NORM normalizza l'uscita di LMNLIMIT secondo la regola seguente:

$$\text{LMN} = (\text{uscita di LMNLIMIT}) * \text{LMN\_FAC} + \text{LMN\_OFF}$$

A LMN_FAC è preassegnato 1 e a LMN_OFF 0.

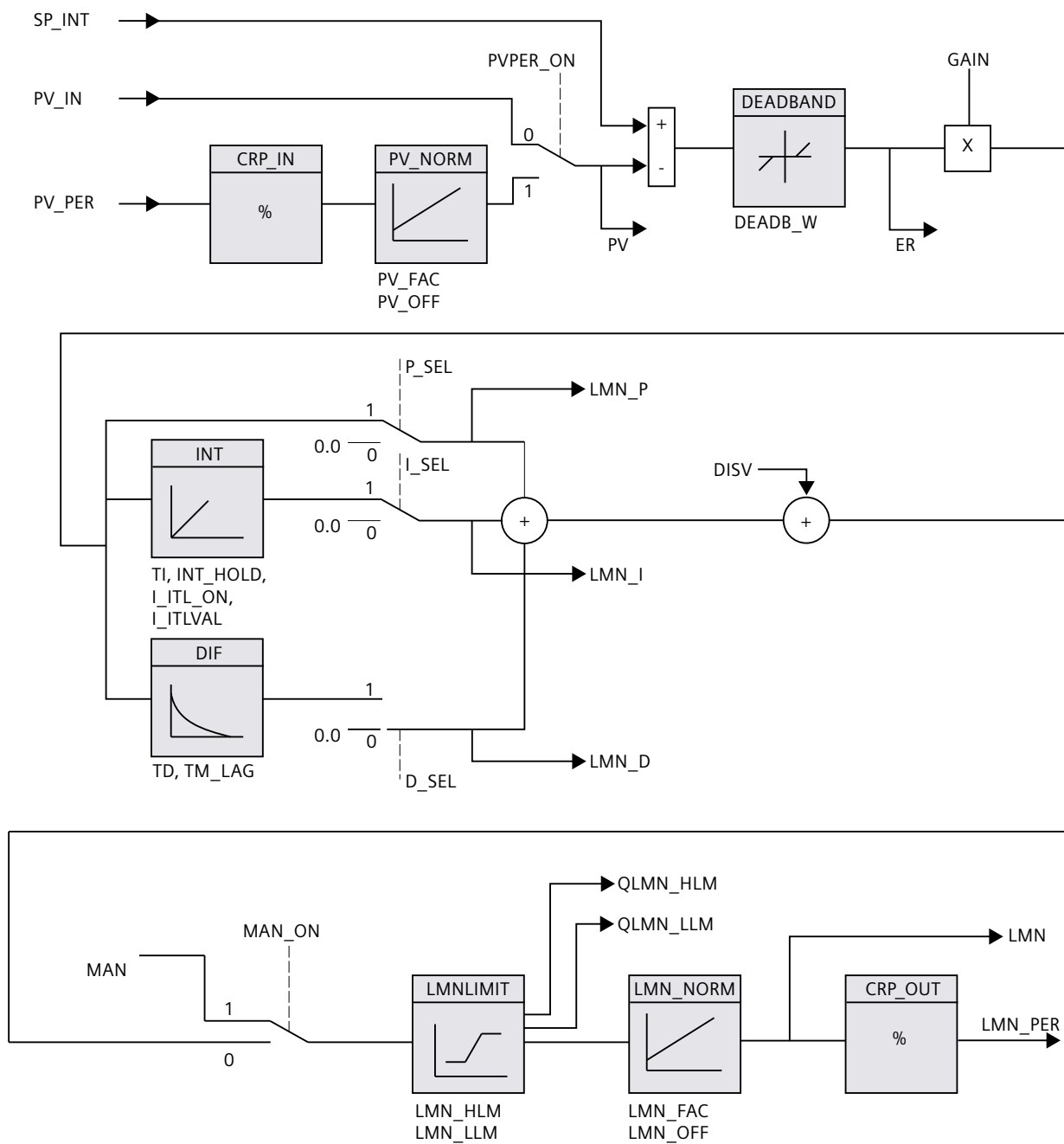
Il valore regolante è disponibile anche in formato periferia. La funzione CRP_OUT converte il valore in virgola mobile LMN in un valore di periferia secondo la regola seguente:

$$\text{LMN\_PER} = \text{LMN} * 27648 / 100$$

Inserzione di un segnale di disturbo

Nell'ingresso DISV è possibile inserire un segnale di disturbo.

10.4.1.3 Schema a blocchi di CONT_C



10.4.1.4 Parametri di ingresso di CONT_C

I nomi dei seguenti parametri si applicano sia al blocco dati sia all'accesso tramite l'API Openness.

Tabella 10-13

Parametri	Tipo di dati	Preas-segnazione	Descrizione
COM_RST	BOOL	FALSE	L'istruzione dispone di una routine di inizializzazione che viene elaborata quando è impostato l'ingresso "Nuovo avviamento".
MAN_ON	BOOL	TRUE	Se è impostato l'ingresso "Inserire funzionamento manuale" il circuito di regolazione è interrotto. Come valore regolante viene predefinito un valore manuale.
PVPER_ON	BOOL	FALSE	Per leggere il valore istantaneo dalla periferia è necessario interconnettere l'ingresso PV_PER con la periferia e impostare l'ingresso "Attivazione valore istantaneo dalla periferia".
P_SEL	BOOL	TRUE	Nell'algoritmo PID è possibile attivare e disattivare singolarmente i componenti PID. Il componente P è attivo se è impostato l'ingresso "Attiva componente P".
I_SEL	BOOL	TRUE	Nell'algoritmo PID è possibile attivare e disattivare singolarmente i componenti PID. Il componente I è attivo se è impostato l'ingresso "Attiva componente I".
INT_HOLD	BOOL	FALSE	L'uscita dell'integratore può essere congelata. Perciò deve essere impostato l'ingresso "Congela componente I".
I_ITL_ON	BOOL	FALSE	L'uscita dell'integratore può essere impostata sull'ingresso I_ITLVAL. Perciò deve essere impostato l'ingresso "Imposta componente I".
D_SEL	BOOL	FALSE	Nell'algoritmo PID è possibile attivare e disattivare singolarmente i componenti PID. Il componente D è attivo se è impostato l'ingresso "Attiva componente D".
CYCLE	TIME	T#1s	Il tempo che trascorre tra i richiami dei blocchi deve essere costante. L'ingresso "Tempo di campionamento" indica il tempo tra i richiami dei blocchi. CYCLE >= 1ms
SP_INT	REAL	0.0	L'ingresso "Setpoint interno" permette di predefinire un setpoint. Sono ammessi valori da -100 a 100% o una grandezza fisica 1).
PV_IN	REAL	0.0	Nell'ingresso "Ingresso valore istantaneo" è possibile parametrizzare un valore di messa in funzione o interconnettere un valore istantaneo esterno in virgola mobile. Sono ammessi valori da -100 a 100% o una grandezza fisica 1).
PV_PER	WORD	W#16#0000	Il valore istantaneo in formato periferia viene interconnesso con il regolatore nell'ingresso "Valore istantaneo periferia".
MAN	REAL	0.0	L'ingresso "Valore manuale" consente di predefinire un valore manuale mediante la funzione di servizio e supervisione. Sono ammessi valori da -100 a 100% o una grandezza fisica 2).
GAIN	REAL	2.0	L'ingresso "Coefficiente proporzionale" indica il guadagno del regolatore.
TI	TIME	T#20s	L'ingresso "Tempo di integrazione" determina il comportamento temporale dell'integratore. TI >= CYCLE
TD	TIME	T#10s	L'ingresso "Tempo derivativo" determina il comportamento temporale del differenziatore. TD >= CYCLE
TM_LAG	TIME	T#2s	Tempo di ritardo del componente D L'algoritmo del componente D comprende un ritardo che può essere parametrizzato nell'ingresso "Tempo di ritardo del componente D". TM_LAG >= CYCLE/2

Parametri	Tipo di dati	Preas-segnazione	Descrizione
DEADB_W	REAL	0.0	La differenza di regolazione viene condotta attraverso una banda morta. L'ingresso "Ampiezza banda morta" determina le dimensioni della banda morta. DEADB_W >= 0.0 (%) o una grandezza fisica 1)
LMN_HLM	REAL	100.0	Il valore regolante viene sempre limitato da un valore minimo e uno massimo. L'ingresso "Valore regolante, limite superiore" indica il limite superiore. Sono ammessi valori reali a partire da LMN_LLM (%) o una grandezza fisica 2).
LMN_LLM	REAL	0.0	Il valore regolante viene sempre limitato da un valore minimo e uno massimo. L'ingresso "Valore regolante, limite inferiore" indica il limite inferiore. Sono ammessi valori reali fino a LMN_HLM (%) o una grandezza fisica 2).
PV_FAC	REAL	1.0	L'ingresso "Fattore del valore istantaneo" viene moltiplicato per il valore istantaneo. Questo ingresso consente l'adeguamento del campo del valore istantaneo.
PV_OFF	REAL	0.0	L'ingresso "Offset del valore istantaneo" viene sommato al valore istantaneo. Questo ingresso consente l'adeguamento del campo del valore istantaneo.
LMN_FAC	REAL	1.0	L'ingresso "Fattore del valore regolante" viene moltiplicato per il valore regolante. Questo ingresso consente l'adeguamento del campo del valore regolante.
LMN_OFF	REAL	0.0	L'ingresso "Offset del valore regolante" viene sommato al valore regolante. Questo ingresso consente l'adeguamento del campo del valore regolante.
I_ITLVAL	REAL	0.0	L'uscita dell'integratore può essere impostata nell'ingresso I_ITL_ON. Nell'ingresso "Valore di inizializzazione per il componente I" si trova il valore di inizializzazione. Sono ammessi valori da -100.0 a 100.0 (%) o una grandezza fisica 2).
DISV	REAL	0.0	Per l'inserzione di un segnale di disturbo, il segnale viene interconnesso con l'ingresso "Segnale di disturbo". Sono ammessi valori da -100.0 a 100.0 (%) o una grandezza fisica 2).

1) Parametro nel ramo del setpoint/valore istantaneo con uguale unità

2) Parametro nel ramo del valore regolante con uguale unità

10.4.1.5 Parametri di uscita di CONT_C

I nomi dei seguenti parametri si applicano sia al blocco dati sia all'accesso tramite l'API Openness.

Tabella 10-14

Parametri	Tipo di dati	Preas-segnazione	Descrizione
LMN	REAL	0.0	Nell'uscita "Valore regolante" viene emesso il valore regolante effettivo in virgola mobile.
LMN_PER	WORD	W#16#0000	Il valore regolante in formato periferia viene interconnesso con il regolatore nell'uscita "Valore regolante periferia".
QLMN_HLM	BOOL	FALSE	Il valore regolante viene sempre limitato da un valore minimo e uno massimo. L'uscita "Attivazione del limite superiore del valore regolante" segnala il raggiungimento del limite superiore.
QLMN_LLM	BOOL	FALSE	Il valore regolante viene sempre limitato da un valore minimo e uno massimo. L'uscita "Attivazione del limite inferiore del valore regolante" segnala il raggiungimento del limite inferiore.
LMN_P	REAL	0.0	L'uscita "Componente P" contiene il componente proporzionale della grandezza regolante.

Parametri	Tipo di dati	Preas-segnazione	Descrizione
LMN_I	REAL	0.0	L'uscita "Componente I" contiene il componente integrale della grandezza regolante.
LMN_D	REAL	0.0	L'uscita "Componente D" contiene il componente differenziale della grandezza regolante.
PV	REAL	0.0	Nell'uscita "Valore istantaneo" viene emesso il valore istantaneo effettivo.
ER	REAL	0.0	Nell'uscita "Differenza di regolazione" viene emessa la differenza di regolazione effettiva.

10.4.2 CONT_S

10.4.2.1 Descrizione di CONT_S

L'istruzione CONT_S consente la regolazione di processi tecnici con segnali di uscita binari del valore regolante per organi attuatori integranti nei sistemi di automazione SIMATIC S7. Mediante parametrizzazione è possibile attivare o disattivare le funzioni parziali del regolatore a passi PI, adattandole così al sistema regolato. Oltre alle funzioni nel ramo del valore istantaneo, questa istruzione realizza un regolatore PI completo con uscita binaria del valore regolante e possibilità di influenzare manualmente il valore regolante. Il regolatore a passi funziona senza risposta di posizione.

Impiego

Il regolatore può essere utilizzato come regolatore PI a valore fisso o in circuiti di regolazione subordinati per regolazioni in cascata, di miscela o di rapporto, ma non come regolatore pilota. Il funzionamento si basa sull'algoritmo di regolazione PI del regolatore digitale e viene integrato con gli elementi funzionali per la generazione del segnale di uscita binario a partire dal segnale di regolazione analogico.

Richiamo

L'istruzione CONT_S dispone di una routine di inizializzazione che viene eseguita se è impostato il parametro di ingresso COM_RST = TRUE. Tutte le uscite di segnale vengono impostate a zero. Dopo l'esecuzione della routine di inizializzazione, COM_RST = FALSE deve essere impostato.

Il calcolo dei valori nei blocchi di regolazione è corretto solo se il blocco viene richiamato a intervalli regolari. Perciò è opportuno richiamare i blocchi di regolazione in un OB di schedulazione orologio (OB 30-OB 38). Il tempo di campionamento va indicato nel parametro CYCLE.

Informazioni di errore

Il blocco non analizza la parola di segnalazione di errore RET_VAL.

10.4.2.2 Funzionamento di CONT_S

Ramo del setpoint

Il setpoint viene immesso nell'ingresso SP_INT in formato in virgola mobile.

Ramo del valore istantaneo

Il valore istantaneo può essere letto in formato periferia e in virgola mobile. La funzione CRP_IN converte il valore di periferia PV_PER in un formato in virgola mobile compreso tra -100 e +100% secondo la regola seguente:

Uscita di CRP_IN = $PV\_PER * 100 / 27648$

La funzione PV_NORM normalizza l'uscita di CRP_IN secondo la regola seguente:

Uscita di PV_NORM = (uscita di CRP_IN) * PV_FAC + PV_OFF

A PV_FAC è preassegnato 1 e a PV_OFF 0.

Calcolo della differenza di regolazione

La differenza tra setpoint e valore istantaneo rappresenta la differenza di regolazione. Per sopprimere una piccola oscillazione continua dovuta alla quantizzazione delle grandezze regolanti (risoluzione del valore regolante limitata dalla valvola di regolazione), la differenza di regolazione viene condotta attraverso una banda morta (DEADBAND). Se DEADB_W = 0 la banda morta è disattivata.

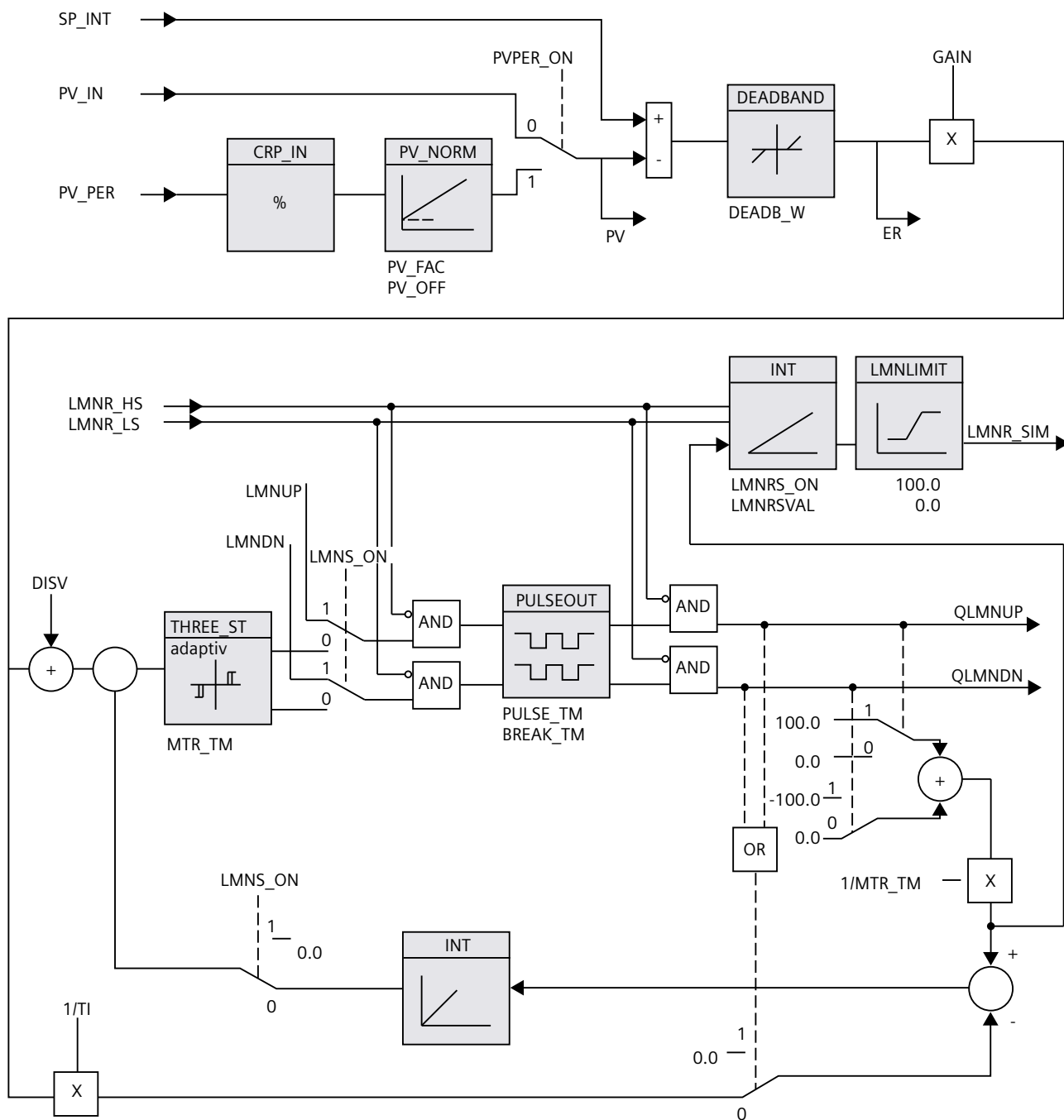
Algoritmo passo-passo PI

L'istruzione funziona senza risposta di posizione. Il componente I dell'algoritmo PI e la risposta di posizione prevista vengono calcolati in **un unico** integratore (INT) e confrontati con il componente P rimasto sotto forma di valore di ritorno. La differenza va a un regolatore a tre punti (THREE_ST) e un generatore di impulsi (PULSEOUT) che genera gli impulsi per la valvola di regolazione. La frequenza di commutazione del regolatore si riduce con un adattamento della soglia d'intervento dell'elemento a tre punti.

Inserzione di un segnale di disturbo

Nell'ingresso DISV è possibile inserire un segnale di disturbo.

10.4.2.3 Schema a blocchi di CONT_S



10.4.2.4 Parametri di ingresso di CONT_S

I nomi dei seguenti parametri si applicano sia al blocco dati sia all'accesso tramite l'API Openness.

Tabella 10-15

Parametri	Tipo di dati	Preassegnazione	Descrizione
COM_RST	BOOL	FALSE	L'istruzione dispone di una routine di inizializzazione che viene elaborata quando è impostato l'ingresso "Nuovo avviamento".
LMNR_HS	BOOL	FALSE	Il segnale "Valvola di regolazione sull'arresto superiore" viene interconnesso con l'ingresso "Segnale di arresto superiore della segnalazione di risposta di posizione". LMNR_HS=TRUE significa: la valvola di regolazione si trova sul riscontro superiore.
LMNR_LS	BOOL	FALSE	Il segnale "Valvola di regolazione sull'arresto inferiore" viene interconnesso con l'ingresso "Segnale di arresto inferiore della segnalazione di risposta di posizione". LMNR_LS=TRUE significa: la valvola di regolazione si trova sul riscontro inferiore.
LMNS_ON	BOOL	FALSE	Nell'ingresso "Inserire funzionamento manuale dei segnali dei valori regolanti" l'elaborazione del segnale del valore regolante viene commutata sul funzionamento manuale.
LMNUP	BOOL	FALSE	Con il funzionamento manuale dei segnali dei valori regolanti, nell'ingresso "Segnale del valore regolante, alto" viene comandato il segnale di uscita QLMNUP.
LMNDN	BOOL	FALSE	Con il funzionamento manuale dei segnali dei valori regolanti, nell'ingresso "Segnale del valore regolante, basso" viene comandato il segnale di uscita QLMNDN.
PVPER_ON	BOOL	FALSE	Per leggere il valore istantaneo dalla periferia è necessario interconnettere l'ingresso PV_PER con la periferia e impostare l'ingresso "Attivazione valore istantaneo dalla periferia".
CYCLE	TIME	T#1s	Il tempo che trascorre tra i richiami dei blocchi deve essere costante. L'ingresso "Tempo di campionamento" indica il tempo tra i richiami dei blocchi. CYCLE >= 1ms
SP_INT	REAL	0.0	L'ingresso "Setpoint interno" permette di predefinire un setpoint. Sono ammessi valori da -100 a 100 % o una grandezza fisica ¹⁾ .
PV_IN	REAL	0.0	Nell'ingresso "Ingresso valore istantaneo" è possibile parametrizzare un valore di messa in funzione o interconnettere un valore istantaneo esterno in virgola mobile. Sono ammessi valori da -100 a 100 % o una grandezza fisica ¹⁾ .
PV_PER	WORD	W#16#0-000	Il valore istantaneo in formato periferia viene interconnesso con il regolatore nell'ingresso "Valore istantaneo periferia".
GAIN	REAL	2.0	L'ingresso "Coefficiente proporzionale" indica il guadagno del regolatore.
TI	TIME	T#20s	L'ingresso "Tempo di integrazione" determina il comportamento temporale dell'integratore. TI >= CYCLE
DEADB_W	REAL	1.0	La differenza di regolazione viene condotta attraverso una banda morta. L'ingresso "Ampiezza banda morta" determina le dimensioni della banda morta. Sono ammessi valori da 0 a 100 % o una grandezza fisica ¹⁾ .
PV_FAC	REAL	1.0	L'ingresso "Fattore del valore istantaneo" viene moltiplicato per il valore istantaneo. Questo ingresso consente l'adeguamento del campo del valore istantaneo.
PV_OFF	REAL	0.0	L'ingresso "Offset del valore istantaneo" viene sommato al valore istantaneo. Questo ingresso consente l'adeguamento del campo del valore istantaneo.
PULSE_TM	TIME	T#3s	Nel parametro "Durata minima dell'impulso" è possibile parametrizzare la durata minima dell'impulso. PULSE_TM >= CYCLE

Parametri	Tipo di dati	Preassegnazione	Descrizione
BREAK_TM	TIME	T#3s	Nel parametro "Durata minima pausa" è possibile parametrizzare una lunghezza minima della pausa. BREAK_TM >= CYCLE
MTR_TM	TIME	T#30s	Nel parametro "Tempo di regolazione motore" viene registrato il tempo impiegato dalla valvola di regolazione tra un riscontro e l'altro. MTR_TM >= CYCLE
DISV	REAL	0.0	Per l'inserzione di un segnale di disturbo, il segnale viene interconnesso con l'ingresso "Segnale di disturbo". Sono ammessi valori da -100 a 100 % o una grandezza fisica ²⁾ .

¹⁾ Parametro nel ramo del setpoint/valore istantaneo con uguale unità

²⁾ Parametro nel ramo del valore regolante con uguale unità

10.4.2.5 Parametri di uscita di CONT_S

I nomi dei seguenti parametri si applicano sia al blocco dati sia all'accesso tramite l'API Openness.

Tabella 10-16

Parametri	Tipo di dati	Preassegnazione	Descrizione
QLMNUP	BOOL	FALSE	Se è impostata l'uscita "Segnale del valore regolante, alto", la valvola di regolazione deve essere aperta.
QLMNDN	BOOL	FALSE	Se è impostata l'uscita "Segnale del valore regolante, basso", la valvola di regolazione deve essere chiusa.
PV	REAL	0.0	Nell'uscita "Valore istantaneo" viene emesso il valore istantaneo effettivo.
ER	REAL	0.0	Nell'uscita "Differenza di regolazione" viene emessa la differenza di regolazione effettiva.

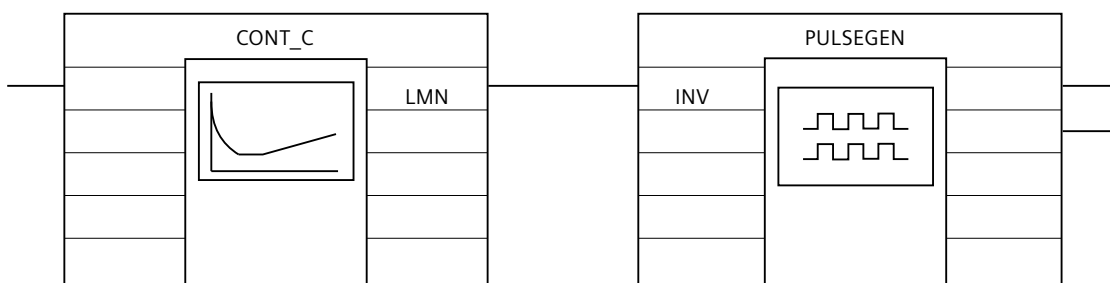
10.4.3 PULSEGEN

10.4.3.1 Descrizione di PULSEGEN

L'istruzione PULSEGEN consente di configurare un regolatore PID con uscita a impulsi per organi attuatori proporzionali. Tramite modulazione dell'ampiezza degli impulsi, PULSEGEN trasforma la grandezza di ingresso INV (= LMN del regolatore PID) in un treno di impulsi con periodo costante, che corrisponde al tempo di ciclo con il quale viene aggiornata la grandezza di ingresso.

Impiego

Con l'istruzione PULSEGEN è possibile creare regolatori PID a due o tre punti con modulazione di larghezza di impulso. Normalmente questa funzione viene utilizzata in combinazione con il regolatore continuo CONT_C.



Richiamo

L'istruzione PULSEGEN dispone di una routine di inizializzazione che viene eseguita se è impostato il parametro di ingresso COM_RST = TRUE. Tutte le uscite di segnale vengono impostate a zero. Dopo l'esecuzione della routine di inizializzazione, COM_RST = FALSE deve essere impostato.

Il calcolo dei valori nei blocchi di regolazione è corretto solo se il blocco viene richiamato a intervalli regolari. Perciò è opportuno richiamare i blocchi di regolazione in un OB di schedulazione orologio (OB 30-OB 38). Il tempo di campionamento va indicato nel parametro CYCLE.

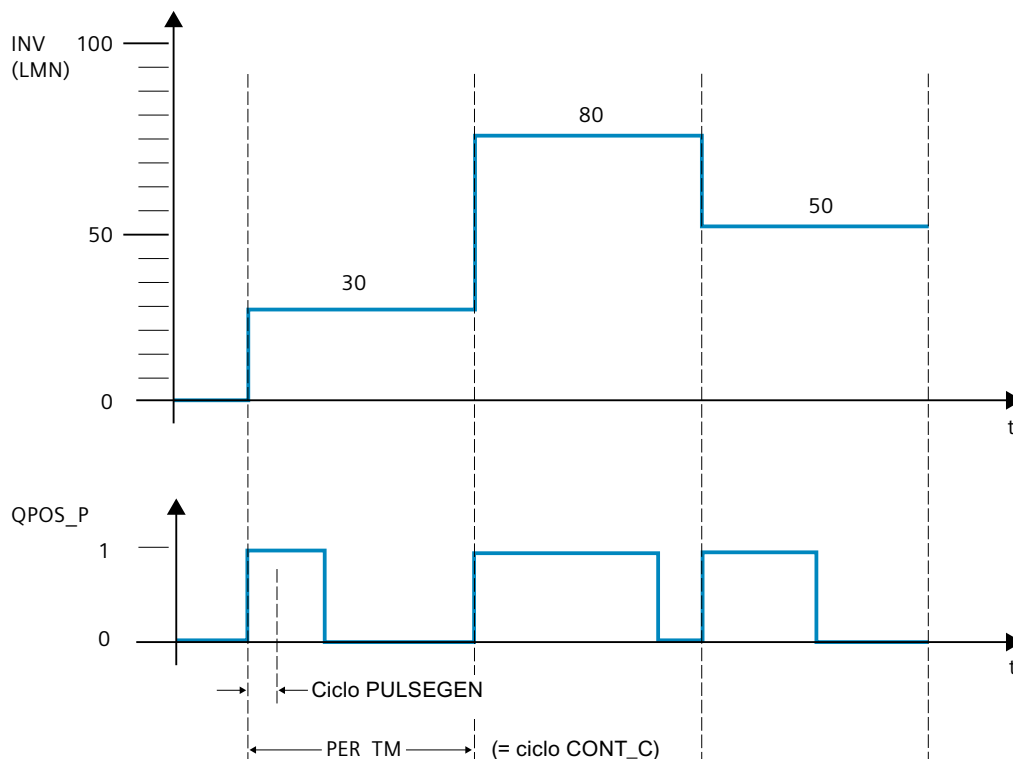
Comportamento in caso di errori

Il blocco non analizza la parola di segnalazione di errore RET_VAL.

10.4.3.2 Funzionamento di PULSEGEN

Modulazione di larghezza di impulso

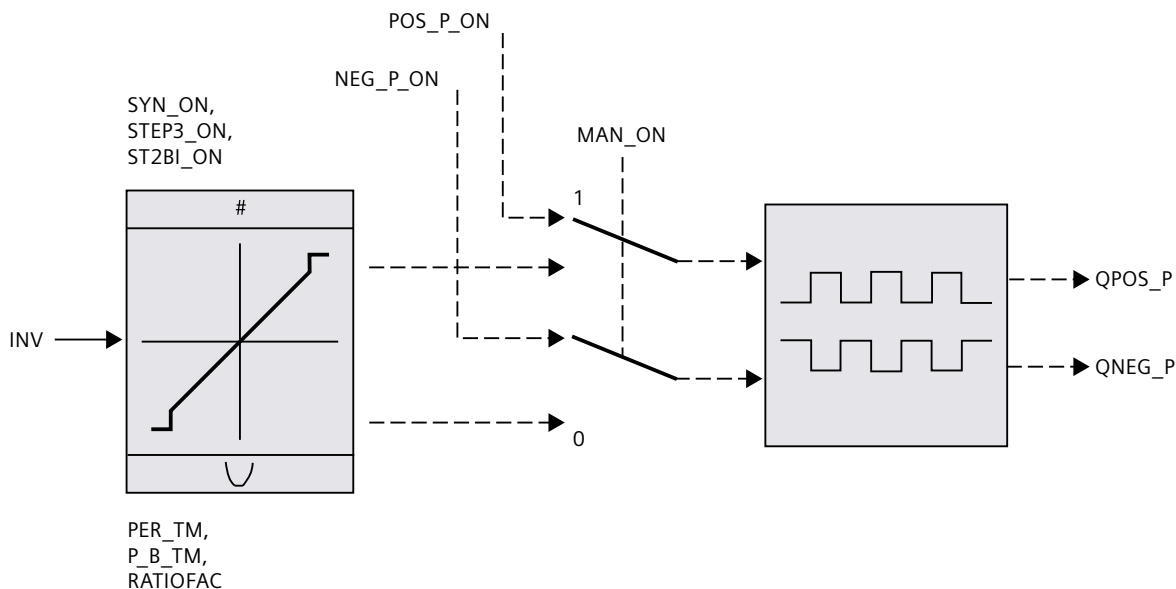
La durata di un impulso per periodo è proporzionale alla grandezza di ingresso. Il ciclo parametrizzato con PER_TM non è uguale al ciclo di elaborazione dell'istruzione PULSEGEN. Anzi, un ciclo PER_TM è formato da più cicli di elaborazione dell'istruzione PULSEGEN e il numero dei richiami di PULSEGEN per ogni ciclo PER_TM rappresenta un'unità di misura per il grado di precisione dell'ampiezza degli impulsi.



Una grandezza di ingresso del 30% e 10 richiami di PULSEGEN per ogni ciclo PER_TM significano pertanto:

- "uno" nell'uscita QPOS_P per i primi tre richiami di PULSEGEN (30% di 10 richiami)
- "zero" nell'uscita QPOS_P per sette ulteriori richiami di PULSEGEN (70% di 10 richiami)

Schema a blocchi



Grado di precisione del valore regolante

Considerato un "rapporto di campionamento" di 1:10 (richiami di CONT_C rispetto ai richiami di PULSEGEN), il grado di precisione del valore regolante di questo esempio è limitato al 10%, vale a dire che i valori di ingresso INV predefiniti possono essere rappresentati su una lunghezza dell'impulso nell'uscita QPOS_P solo in passi del 10%.

Di conseguenza, la precisione aumenta con l'aumentare del numero di richiami di PULSEGEN per ogni richiamo di CONT_C.

Richiamando PULSEGEN 100 volte più spesso di CONT_C, ad es., si ottiene una risoluzione dell'1% del campo del valore regolante.

NOTA

La scansione della frequenza di richiamo deve essere programmata individualmente.

Sincronizzazione automatica

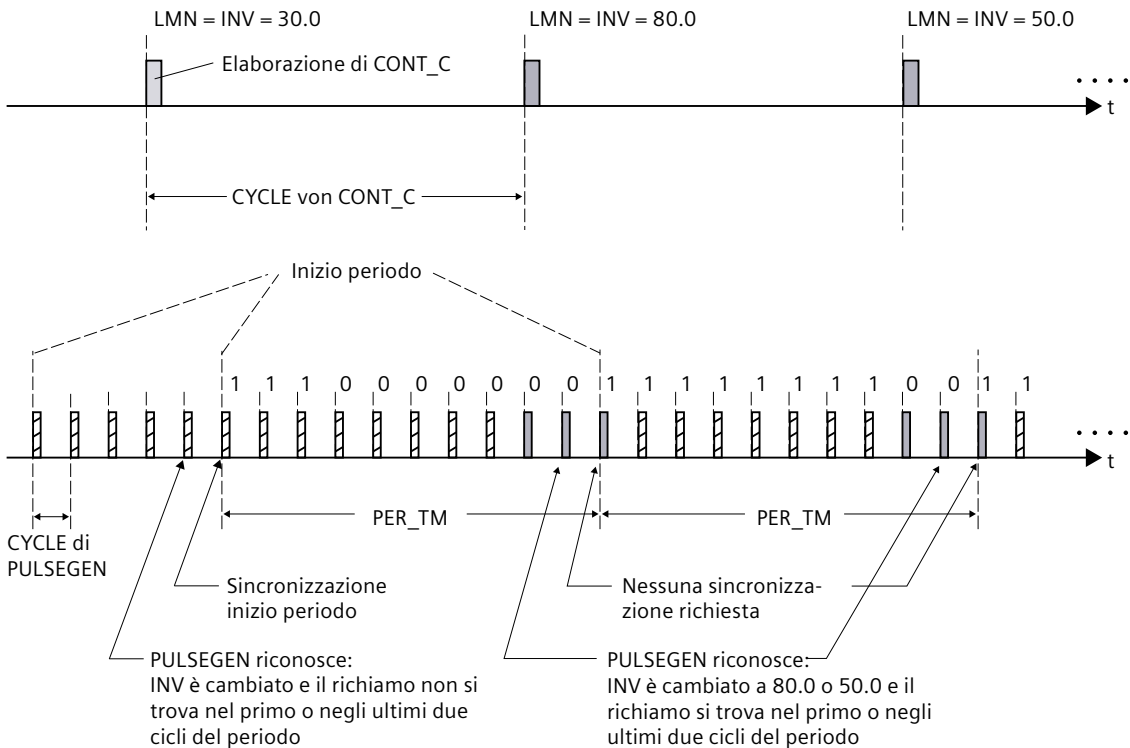
Esiste la possibilità di sincronizzare automaticamente l'emissione degli impulsi con l'istruzione che aggiorna la grandezza di ingresso INV (ad es. CONT_C). In questo modo si garantisce l'emissione tempestiva di una grandezza di ingresso variabile sotto forma di impulso.

Il generatore di impulsi analizza sempre nell'intervallo di tempo del periodo PER_TM la grandezza di ingresso INV e converte il valore in un segnale di impulso della lunghezza corrispondente.

Tuttavia, poiché INV viene generalmente calcolato in un livello di schedulazione orologio più lento, il generatore di impulsi dovrebbe iniziare la conversione del valore digitale in un segnale di impulso il più rapidamente possibile dopo l'aggiornamento di INV.

A questo scopo il blocco può sincronizzare automaticamente l'inizio del periodo secondo il seguente procedimento:

Se INV è stato modificato e il richiamo del blocco non si trova né nel primo né negli ultimi due cicli di richiamo di un periodo, la sincronizzazione viene eseguita. La durata dell'impulso viene ricalcolata e al ciclo successivo l'emissione comincia con un nuovo periodo.



Elaborazione di PULSEGEN

Elaborazione di PULSEGEN nel primo o nei due ultimi cicli del periodo

La sincronizzazione automatica è disattivata se SYN_ON = FALSE.

NOTA

Con l'inizio del nuovo periodo, il precedente valore di INV (cioè di LMN) viene riprodotto nel segnale di impulso in modo più o meno impreciso al termine della sincronizzazione.

10.4.3.3 Modi di funzionamento di PULSEGEN

Modi di funzionamento

A seconda della parametrizzazione del generatore di impulsi, i regolatori PID possono essere configurati a tre punti o con un'uscita a due punti bipolare o unipolare. La tabella seguente mostra l'impostazione delle combinazioni degli interruttori per i vari modi di funzionamento possibili.

Modo di funzionamento	MAN_ON	STEP3_ON	ST2BI_ON
Regolazione a tre punti	FALSE	TRUE	Qualsiasi
Regolazione a due punti con campo di regolazione bipolare (-100% ... 100%)	FALSE	FALSE	TRUE
Regolazione a due punti con campo di regolazione unipolare (0% ... 100%)	FALSE	FALSE	FALSE
Funzionamento manuale	TRUE	Qualsiasi	Qualsiasi

Funzionamento manuale per regolazione a due o tre punti

In funzionamento manuale (MAN_ON = TRUE) è possibile impostare le uscite binarie del regolatore a due o tre punti mediante i segnali POS_P_ON e NEG_P_ON indipendentemente da INV.

Regolazione	POS_P_ON	NEG_P_ON	QPOS_P	QNEG_P
Regolazione a tre punti	FALSE	FALSE	FALSE	FALSE
	TRUE	FALSE	TRUE	FALSE
	FALSE	TRUE	FALSE	TRUE
	TRUE	TRUE	FALSE	FALSE
Regolazione a due punti	FALSE	Qualsiasi	FALSE	TRUE
	TRUE	Qualsiasi	TRUE	FALSE

10.4.3.4 Regolazione a tre punti

Regolazione a tre punti

Con il modo di funzionamento "Regolazione a tre punti" si possono generare tre stati del segnale di regolazione. I valori di stato dei segnali di uscita binari QPOS_P e QNEG_P vengono assegnati ai rispettivi stati di funzionamento dell'organo attuatore. La tabella mostra un esempio di regolazione della temperatura.

Segnali di uscita	Riscaldamento	OFF	Raffreddamento
QPOS_P	TRUE	FALSE	FALSE
QNEG_P	FALSE	FALSE	TRUE

Dalla grandezza di ingresso viene calcolata la durata dell'impulso mediante una curva caratteristica. La forma di questa curva caratteristica viene definita dalla durata minima dell'impulso e della pausa e dal fattore proporzionale. Il valore normale per il fattore proporzionale è 1.

Le flessioni sulle curve caratteristiche sono causate dalla durata minima dell'impulso o della pausa.

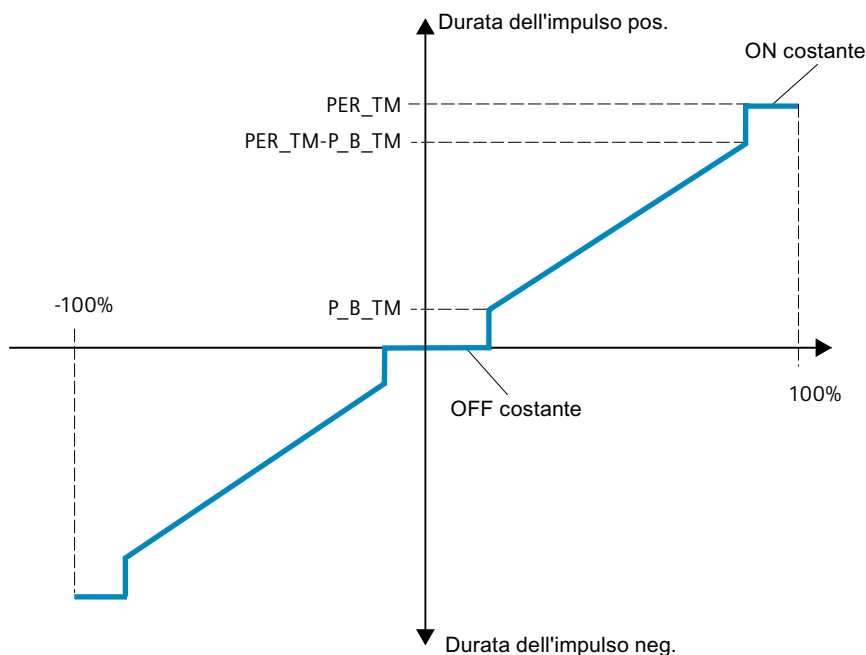
Durata minima degli impulsi o delle pause

Una durata minima degli impulsi o delle pause P_B_TM parametrizzata correttamente può evitare brevi tempi di inserzione o disinserione che compromettono la durata degli attuatori e dei dispositivi di regolazione. I valori assoluti di piccola entità della grandezza di ingresso LMN, che potrebbero generare impulsi di durata inferiore a P_B_TM , vengono soppressi. I valori di ingresso di grande entità che potrebbero generare una durata dell'impulso maggiore di $PER\_TM - P\_B\_TM$, invece, vengono impostati a 100% o -100%.

La durata degli impulsi positivi o negativi si calcola dalla grandezza di ingresso (in %) moltiplicata per il periodo:

$$\text{Durata dell'impulso} = \text{INV} / 100 * \text{PER\_TM}$$

La figura seguente mostra una curva caratteristica simmetrica del regolatore a tre punti (fattore proporzionale = 1)



Regolazione a tre punti asimmetrica

Il fattore proporzionale **RATIOFAC** consente di modificare il rapporto della durata degli impulsi positivi rispetto a quelli negativi. Nel caso di un processo termico, ad es., è così possibile considerare costanti di tempo del sistema regolato differenti per il riscaldamento e il raffreddamento.

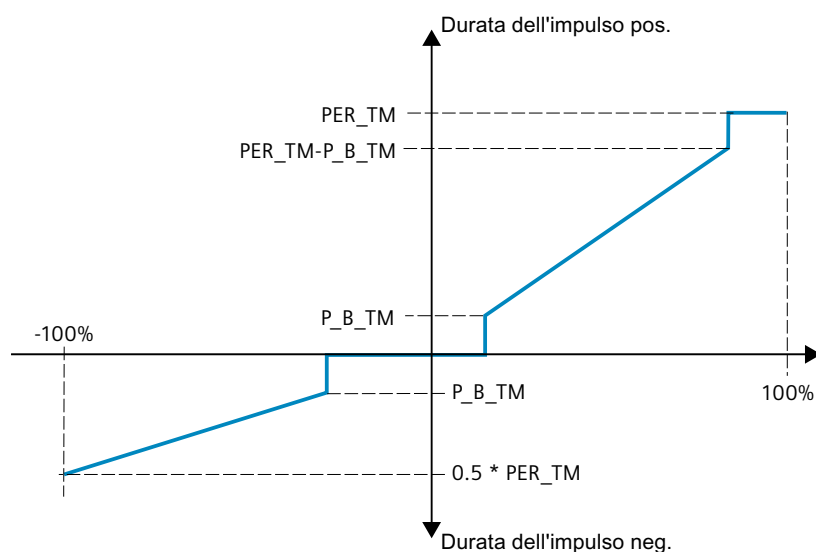
Fattore proporzionale < 1

La durata dell'impulso calcolata moltiplicando la grandezza di ingresso per la durata del periodo nell'uscita a impulsi negativi viene moltiplicata per il fattore proporzionale.

$$\text{Durata dell'impulso positivo} = \text{INV} / 100 * \text{PER\_TM}$$

$$\text{Durata dell'impulso negativo} = \text{INV} / 100 * \text{PER\_TM} * \text{RATIOFAC}$$

La figura seguente mostra la curva caratteristica asimmetrica del regolatore a tre punti (fattore proporzionale = 0.5)



Fattore proporzionale > 1

La durata dell'impulso calcolata moltiplicando la grandezza di ingresso per la durata del periodo nell'uscita a impulsi positivi viene divisa per il fattore proporzionale.

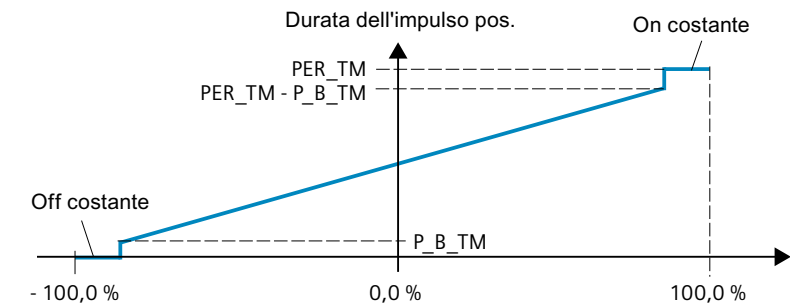
$$\text{Durata dell'impulso positivo} = \text{INV} / 100 * \text{PER\_TM} / \text{RATIOFAC}$$

$$\text{Durata dell'impulso negativo} = \text{INV} / 100 * \text{PER\_TM}$$

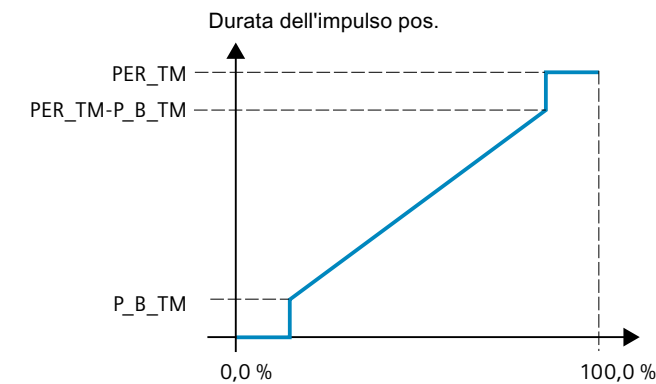
10.4.3.5 Regolazione a due punti

Con la regolazione a due punti si collega soltanto l'uscita degli impulsi positiva QPOS_P di PULSEGEN con il relativo organo attuatore di accensione/spegnimento. A seconda del campo del valore regolante utilizzato, il regolatore a due punti ha un campo del valore regolante bipolare o unipolare.

Regolazione a due punti con campo del valore regolante bipolare (-100% ... 100%)



Regolazione a due punti con campo del valore regolante unipolare (0% ... 100%)



In QNEG_P è disponibile il segnale di uscita negato se l'interconnessione del regolatore a due punti nel circuito di regolazione richiede un segnale binario logico inverso per gli impulsi di regolazione.

Impulso	Organo attuatore inserito	Organo attuatore disinserito
QPOS_P	TRUE	FALSE
QNEG_P	FALSE	TRUE

10.4.3.6 Parametri di ingresso di PULSEGEN

Nel blocco i valori dei parametri di ingresso non vengono limitati; i parametri non vengono controllati.

Tabella 10-17

Parametri	Tipo di dati	Preas-segnazione	Descrizione
INV	REAL	0.0	Nel parametro di ingresso "Variabile di ingresso" viene inserita una grandezza analogica del valore regolante. Sono ammessi valori da -100 a 100%.
PER_TM	TIME	T#1s	Nel parametro "Durata periodo" viene immesso un periodo costante della modulazione di larghezza di impulso. che corrisponde al tempo di campionamento del regolatore. Il rapporto tra il tempo di campionamento del generatore di impulsi e quello del regolatore determina la precisione della modulazione di larghezza di impulso. $PER\_TM \geq 20 * CYCLE$
P_B_TM	TIME	T#50ms	Nel parametro per la durata minima dell'impulso o della pausa è possibile parametrizzare una durata minima per l'impulso o per la pausa. $P\_B\_TM \geq CYCLE$
RATIOFAC	REAL	1.0	Con il parametro di ingresso "Fattore proporzionale" è possibile modificare il rapporto tra la durata degli impulsi negativi e quelli positivi. Nel caso di un processo termico è possibile compensare differenti costanti temporali per riscaldamento e raffreddamento (ad es. in un processo con riscaldamento elettrico e raffreddamento ad acqua). Sono ammessi valori da 0.1 a 10.0.
STEP3_ON	BOOL	TRUE	Nel parametro di ingresso "Inserisci regolazione a tre punti" si attiva il modo di funzionamento corrispondente. Con la regolazione a tre punti sono operativi entrambi i segnali di uscita.
ST2BI_ON	BOOL	FALSE	Il parametro di ingresso "Inserisci regolazione a due punti per area bipolare del valore regolante" consente di scegliere tra i modi di funzionamento "Regolazione a due punti per area bipolare del valore regolante" e "Regolazione a due punti per area unipolare del valore regolante". Qui si deve avere $STEP3\_ON = FALSE$.
MAN_ON	BOOL	FALSE	Con l'impostazione del parametro di ingresso "Inserire funzionamento manuale" è possibile impostare manualmente i segnali di uscita.
POS_P_ON	BOOL	FALSE	Con il funzionamento manuale di regolazione a tre punti è possibile comandare il segnale di uscita QPOS_P nel parametro di ingresso "Inserimento impulso positivo". Con il funzionamento manuale della regolazione a due punti, QNEG_P viene sempre impostato inversamente a QPOS_P.
NEG_P_ON	BOOL	FALSE	Con il funzionamento manuale della regolazione a tre punti è possibile comandare il segnale di uscita QNEG_P nel parametro di ingresso "Inserimento impulso negativo". Con il funzionamento manuale della regolazione a due punti, QNEG_P viene sempre impostato inversamente a QPOS_P.
SYN_ON	BOOL	TRUE	Impostando il parametro di ingresso "Inserisci sincronizzazione" è possibile sincronizzare automaticamente l'emissione degli impulsi con il blocco che aggiorna la grandezza di ingresso INV. In questo modo si garantisce che una grandezza di ingresso variabile venga emessa tempestivamente come impulso.
COM_RST	BOOL	FALSE	Il blocco dispone di una routine di inizializzazione che viene elaborata quando è impostato l'ingresso "Nuovo avviamento".
CYCLE	TIME	T#10ms	Il tempo che trascorre tra i richiami dei blocchi deve essere costante. L'ingresso "Tempo di campionamento" indica il tempo tra i richiami dei blocchi. $CYCLE \geq 1ms$

10.4.3.7 Parametri di uscita di PULSEGEN

Tabella 10-18

Parametri	Tipo di dati	Preassegnazione	Descrizione
QPOS_P	BOOL	FALSE	Il parametro di uscita "Segnale di uscita impulso positivo" viene impostato se deve essere emesso un impulso. Con la regolazione a tre punti l'impulso è positivo. Con la regolazione a due punti QNEG_P viene sempre impostato inversamente a QPOS_P.
QNEG_P	BOOL	FALSE	Il parametro di uscita "Segnale di uscita impulso negativo" viene impostato se deve essere emesso un impulso. Con la regolazione a tre punti l'impulso è negativo. Con la regolazione a due punti QNEG_P viene sempre impostato inversamente a QPOS_P.

10.4.4 TCONT_CP

10.4.4.1 Descrizione di TCONT_CP

L'istruzione TCONT_CP consente la regolazione di processi di temperatura con attivazione continua o a impulsi. Il funzionamento si basa sull'algoritmo di regolazione PID che dispone di funzioni aggiuntive per i processi di temperatura. Per migliorare il comportamento di regolazione nei circuiti di temperatura, il blocco dispone di una zona di regolazione e di una riduzione della componente P in caso di gradini del valore di riferimento.

L'istruzione è in grado di impostare automaticamente i parametri PI/PID ottimizzando la regolazione.

Impiego

Poiché alimenta un solo attuatore, un regolatore consente solo di riscaldare o di raffreddare. In caso di impiego in un processo di raffreddamento, GAIN deve essere parametrizzato con un valore negativo. L'inversione del regolatore così parametrizzata fa sì che, ad es., a un aumento della temperatura corrisponda un aumento della grandezza regolante LMN e quindi del potere di raffreddamento.

Richiamo

L'istruzione TCONT_CP deve essere richiamata in modo equidistante, utilizzando un livello di schedulazione orologio (ad es. l'OB35 per i sistemi S7-300).

L'istruzione TCONT_CP dispone di una routine di inizializzazione che viene eseguita se è impostato il parametro di ingresso COM_RST = TRUE. Durante l'inizializzazione l'integratore viene impostato sul valore di inizializzazione I_ITVAL. Tutte le altre uscite di segnale vengono impostate a zero. Una volta elaborata la routine di inizializzazione COM_RST il blocco viene reimpostato su FALSE. Per eseguire un'inizializzazione a ogni nuovo avviamento della CPU, richiamare il blocco nell'OB 100 con COM_RST = TRUE.

Vedere anche

[Funzionamento del generatore di impulsi \(Pagina 466\)](#)

[Schema a blocchi TCONT_CP \(Pagina 469\)](#)

10.4.4.2 Funzionamento di TCONT_CP**Ramo del setpoint**

Il setpoint può essere immesso nell'ingresso SP_INT in virgola mobile in unità fisiche o percentuali. Nel calcolo della differenza di regolazione, il setpoint e quello istantaneo devono essere espressi nella stessa unità.

Selezione del valore istantaneo (PVPER_ON)

A seconda del parametro PVPER_ON il valore istantaneo viene letto in formato periferia o in virgola mobile.

PVPER_ON	Introduzione del valore istantaneo
TRUE	Il valore istantaneo viene letto dalla periferia analogica (PEW xxx) nell'ingresso PV_PER.
FALSE	Il valore istantaneo viene letto in formato di virgola mobile nell'ingresso PV_IN.

Commutazione del formato del valore istantaneo CRP_IN (PER_MODE)

La funzione CRP_IN converte il valore di periferia PV_PER in un formato di virgola mobile in base all'interruttore PER_MODE secondo la seguente regola:

PER_MODE	Uscita di CRP_IN	Tipo di introduzione analogica	Unità
0	$PV\_PER * 0.1$	Termocoppie; PT100/Ni100; standard	°C;°F
1	$PV\_PER * 0.01$	PT100/Ni100; climatizzazione;	°C;°F
2	$PV\_PER * 100/27648$	Tensione/corrente	%

Normalizzazione del valore istantaneo PV_NORM (PV_FAC, PV_OFFS)

La funzione PV_NORM calcola l'uscita di CRP_IN secondo la regola seguente:

$$\text{"Uscita di PV\_NORM"} = \text{"uscita di CRP\_IN"} * \text{PV\_FAC} + \text{PV\_OFFS}$$

Questa funzione può essere utilizzata per i seguenti scopi:

- Adeguamento del valore istantaneo con PV_FAC come coefficiente del valore istantaneo e PV_OFFS come offset del valore istantaneo
- Normalizzazione della temperatura in percentuale

Per poter indicare il setpoint come percentuale è necessario convertire in percentuale il valore di temperatura misurato.

- Normalizzazione della percentuale secondo la temperatura

Per poter immettere il setpoint nella grandezza fisica della temperatura è necessario convertire in temperatura il valore di tensione/corrente misurato.

Calcolo dei parametri:

- $\text{PV\_FAC} = \text{campo di PV\_NORM} / \text{campo di CRP\_IN}$;
- $\text{PV\_OFFS} = \text{LIMINF(PV\_NORM)} - \text{PV\_FAC} * \text{LIMINF(CRP\_IN)}$;
con LIMINF: Limite inferiore

Con i valori di default (PV_FAC = 1.0 e PV_OFFS = 0.0) la normalizzazione è disattivata. Il valore di misura effettivamente attivo viene emesso nell'uscita PV.

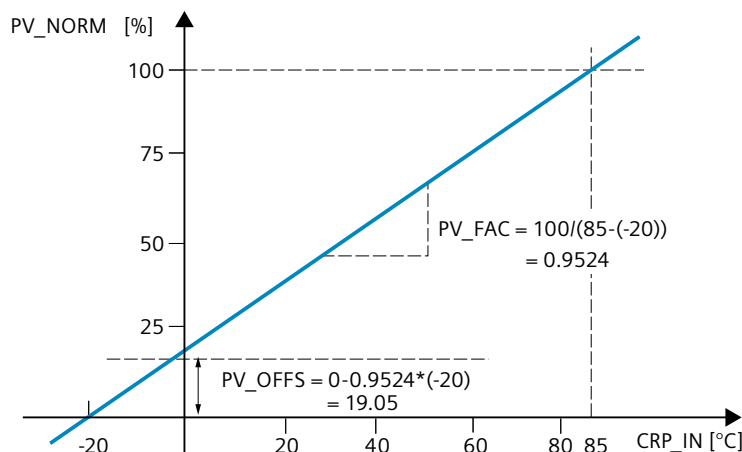
NOTA

In caso di regolazione a impulsi il valore istantaneo deve essere trasmesso al blocco nel richiamo veloce degli impulsi (motivo: filtraggio del valore medio). In caso contrario la qualità della regolazione potrebbe essere compromessa.

Esempio di normalizzazione del valore istantaneo

Se si desidera preimpostare il setpoint come percentuale e il campo della temperatura in CRP_IN è compreso tra -20 e 85 °C, è necessario convertire il campo della temperatura in percentuale.

La figura seguente mostra un esempio di adattamento del campo della temperatura da -20 a 85 °C al campo da 0 a 100 % interno:



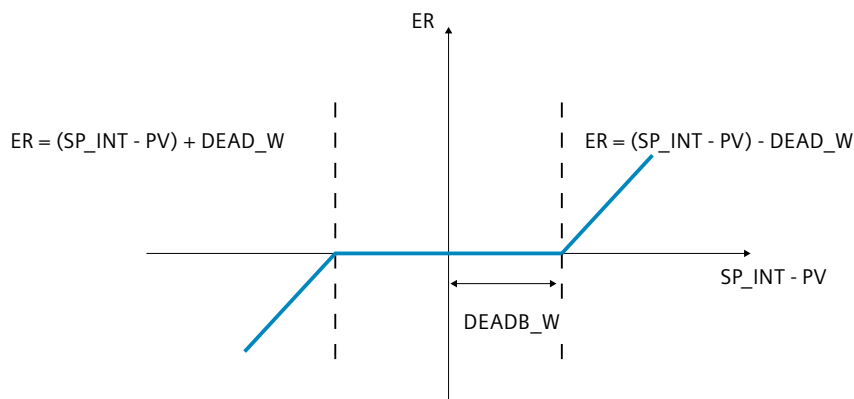
Calcolo della differenza di regolazione

Dalla differenza tra il setpoint e quello istantaneo risulta la differenza di regolazione prima della banda morta.

Il setpoint e il valore istantaneo devono avere la stessa unità.

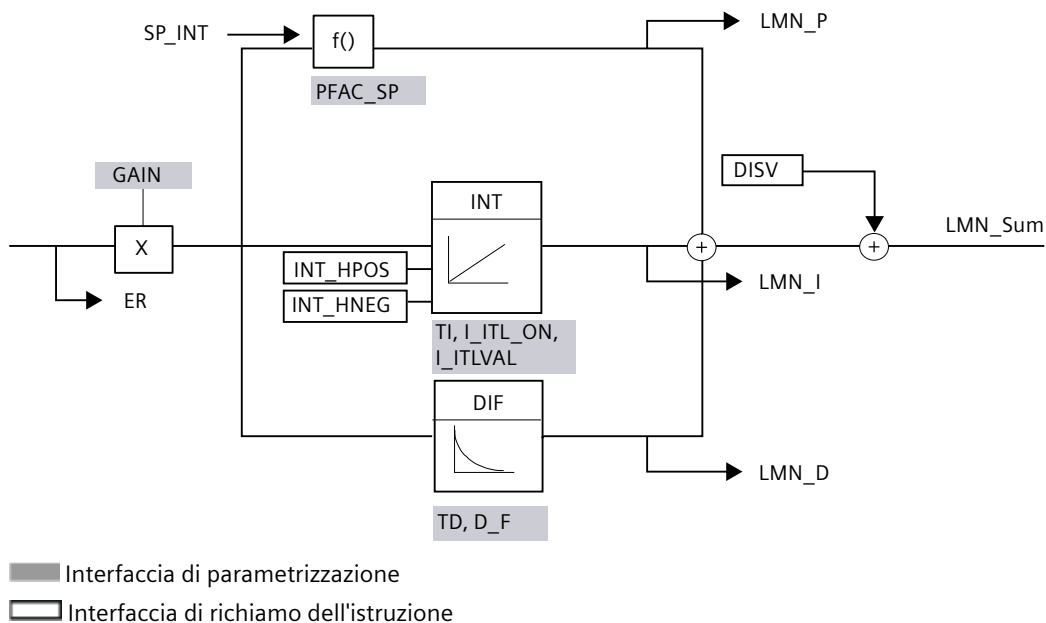
Banda morta (DEADB_W)

Per sopprimere una piccola oscillazione continua causata dalla quantizzazione delle grandezze regolanti (ad es. nel caso di una modulazione di larghezza di impulso con PULSEGEN), la differenza di regolazione viene condotta attraverso una banda morta (DEADBAND). Se DEADB_W = 0.0 la banda morta è disattivata. La differenza di regolazione effettiva è indicata nel parametro ER.



Algoritmo PID

La figura seguente mostra lo schema a blocchi dell'algoritmo PID:



Algoritmo PID (GAIN, TI, TD, D_F)

L'algoritmo PID utilizza un algoritmo di posizione. Le componenti proporzionale, integrale (INT) e differenziale (DIF) sono collegate in parallelo e possono essere inserite o disinserite singolarmente. In questo modo è possibile parametrizzare i regolatori P, PI, PD e PID.

L'ottimizzazione della regolazione supporta regolatori PI e PID. L'inversione del regolatore viene effettuata mediante un GAIN negativo (regolatore di raffreddamento).

Impostando TI e TD su 0.0 si ottiene un regolatore con sola funzione P sul punto di lavoro.

La risposta a gradino nell'area temporale è:

$$LMN\_Sum(t) = GAIN \cdot ER(0) \cdot \left(1 + \frac{1}{TI} \cdot t + D\_F \cdot e^{-\frac{t}{TD / D\_F}} \right)$$

Ciò significa:

LMN_Sum(t) la grandezza regolante in funzionamento automatico del regolatore

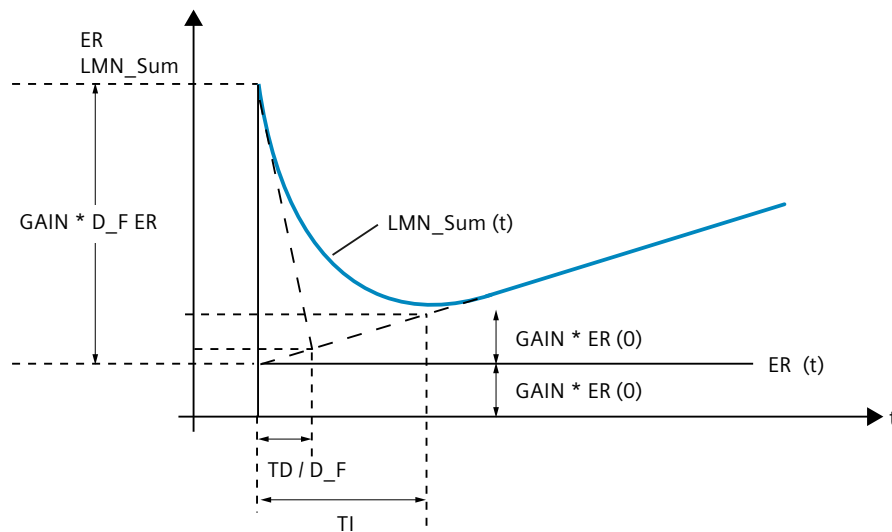
ER (0) l'altezza del gradino della differenza di regolazione normalizzata

GAIN il guadagno del regolatore

TI il tempo di integrazione

TD il tempo derivativo

D_F il fattore di derivazione



Integratore (TI, I_ITL_ON, I_ITLVAL)

Nel funzionamento manuale l'integratore viene comandato nel seguente modo: $LMN\_I = LMN - LMN\_P - DISV$.

In caso di limitazione del valore regolante la componente I del regolatore viene arrestata. La componente I viene riabilitata con una differenza di regolazione che la sposta in direzione del campo di regolazione interno.

Ulteriori variazioni della componente I si verificano nei seguenti casi:

- Disattivazione del componente I del regolatore impostando $TI = 0.0$
- Attenuazione del componente P al variare del setpoint
- Zona di regolazione
- I limiti del valore regolante si possono modificare online

Attenuazione della componente P in caso di variazioni del setpoint (PFAC_SP)

Per evitare sovraelongazioni è possibile ridurre il componente P con il parametro "Coefficiente proporzionale in caso di modifiche del setpoint" (PFAC_SP). PFAC_SP consente di definire in modo continuo tra 0.0 e 1.0 l'intensità con cui il componente P deve agire in caso di variazioni del setpoint:

- $PFAC\_SP = 1.0$: componente P pienamente effettivo in caso di variazione del setpoint
- $PFAC\_SP = 0.0$: componente P non attivo in caso di variazione del setpoint

La riduzione della componente P si ottiene tramite la compensazione della componente I.

Differenziatore (TD, D_F)

- Disattivazione del componente D del regolatore impostando $TD = 0.0$.
- Quando la componente D è collegata, attenersi alla seguente equazione:

$$TD = 0.5 * CYCLE * D\_F$$

Parametrizzazione di un regolatore P o PD con punto di lavoro

Disattivare il componente I nell'interfaccia di parametrizzazione ($TI = 0.0$) ed eventualmente il componente D ($TD = 0.0$). Effettuare inoltre la seguente parametrizzazione:

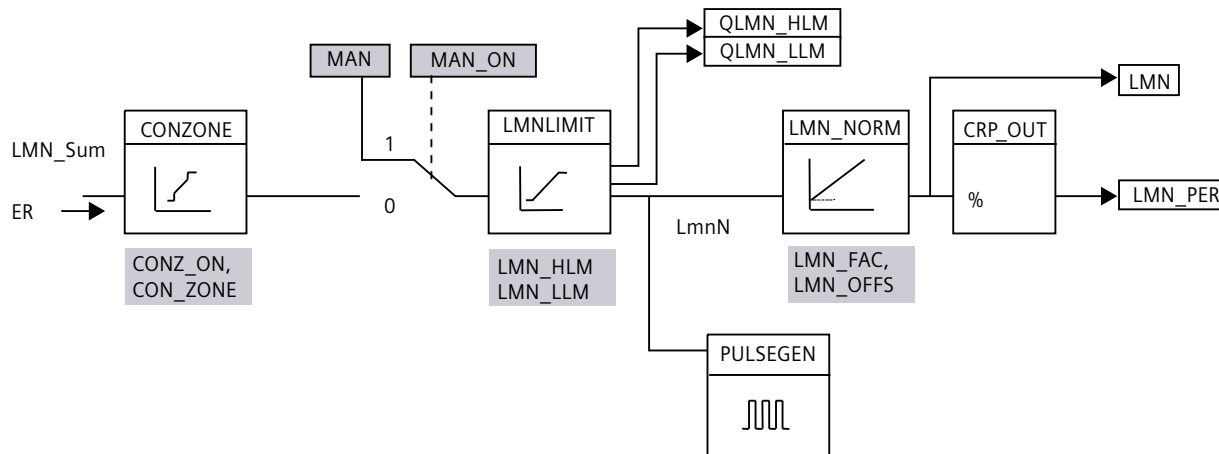
- $I\_ITL\_ON = TRUE$
- $I\_ITLVAL =$ punto di lavoro;

Inserzione di un segnale di disturbo (DISV)

Nell'ingresso DISV è possibile inserire anche un segnale di disturbo.

Calcolo del valore regolante

La seguente figura mostra lo schema a blocchi del calcolo del valore regolante:



- Interfaccia di parametrizzazione
- Interfaccia di richiamo dell'istruzione
- Interfaccia di parametrizzazione, interfaccia di richiamo dell'istruzione

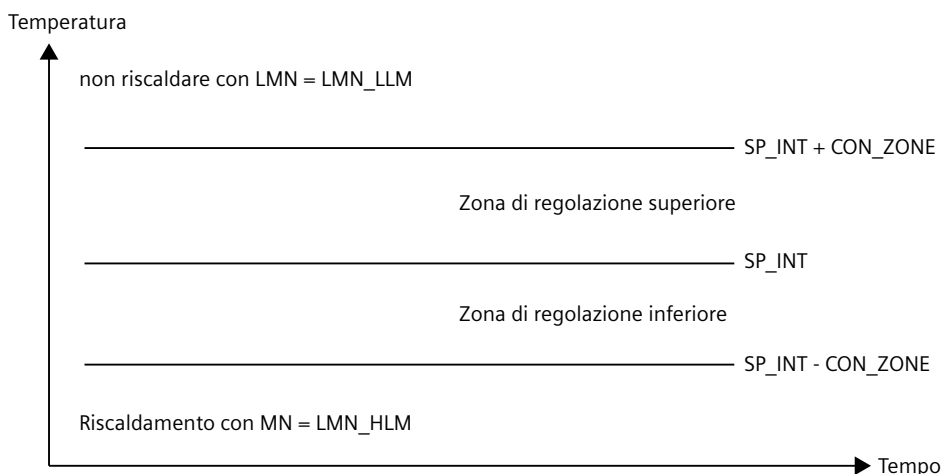
Zona di regolazione (CONZ_ON, CON_ZONE)

Se CONZ_ON = TRUE il regolatore funziona con una zona di regolazione. ovvero viene comandato con il seguente algoritmo:

- Se il valore istantaneo PV supera il setpoint SP_INT per un valore superiore a quello di CON_ZONE, come grandezza regolante viene emesso il valore LMN_LLM.
- Se il valore istantaneo PV è minore del setpoint SP_INT per un valore superiore a quello di CON_ZONE, viene emesso LMN_HLM.
- Se il valore istantaneo PV si muove all'interno della zona di regolazione (CON_ZONE), il valore regolante assume il valore dell'algoritmo PID LMN_Sum.

NOTA

La commutazione della grandezza regolante da LMN_LLM o LMN_HLM a LMN_Sum richiede il mantenimento di un'isteresi del 20% della zona di regolazione.



NOTA

Prima di inserire manualmente la zona di regolazione, assicurarsi che l'ampiezza impostata non sia troppo piccola. Se l'ampiezza della zona di regolazione impostata è insufficiente, si verificano oscillazioni nell'andamento della grandezza regolante e del valore istantaneo.

Vantaggio della zona di regolazione

All'ingresso nella zona di regolazione, la componente D collegata causa una rapida riduzione della grandezza regolante. Per questo motivo la zona di regolazione è opportuna soltanto se la componente D è attiva. Senza zona di regolazione, la grandezza regolante verrebbe sostanzialmente ridotta soltanto dal componente P che diminuisce. La zona di regolazione comporta un'assestamento più veloce senza sovra o sottooscillazioni, se la grandezza regolante minima e massima indicata si scosta notevolmente dalla grandezza regolante stazionaria necessaria per il nuovo punto di lavoro.

Elaborazione del valore manuale (MAN_ON, MAN)

È possibile commutare tra funzionamento manuale e automatico. Nel funzionamento manuale la grandezza di regolazione viene collegata a un valore manuale.

L'integratore (INT) viene impostato internamente su LMN - LMN_P - DISV mentre il derivatore (DIF) viene azzerato e sincronizzato internamente. La commutazione al funzionamento automatico avviene così in modo lineare.

NOTA

Durante l'ottimizzazione il parametro MAN_ON non è attivo.

Limitazione del valore regolante LMNLIMIT (LMN_HLM, LMN_LLM)

Il valore regolante viene limitato dalla funzione LMNLIMIT ai limiti LMN_HLM e LMN_LLM. Il raggiungimento dei limiti viene segnalato dai bit di segnalazione QLMN_HLM e QLMN_LLM.

In caso di limitazione del valore regolante la componente I del regolatore viene arrestata. La componente I viene riabilitata con una differenza di regolazione che la sposta in direzione del campo di regolazione interno.

Modifica dei limiti del valore regolante online

Se il campo del valore regolante viene ridotto e il nuovo valore regolante illimitato si trova oltre tali limiti, la componente I e quindi il valore regolante subiscono una variazione.

Il valore regolante viene ridotto pari alla variazione del limite del valore regolante. Se prima della variazione il valore regolante era illimitato, esso viene impostato esattamente sul nuovo limite (qui descritto per il limite superiore del valore regolante).

Normalizzazione del valore regolante LMN_NORM (LMN_FAC, LMN_OFFS)

La funzione LMN_NORM normalizza il valore regolante secondo la regola seguente:

$$LMN = LmnN * LMN\_FAC + LMN\_OFFS$$

Questa funzione può essere utilizzata per i seguenti scopi:

- Adeguamento del valore regolante con LMN_FAC come fattore del valore regolante e LMN_OFFS come offset del valore regolante

Il valore regolante è disponibile anche nel formato di periferia. La funzione CRP_OUT converte il valore in virgola mobile LMN in un valore di periferia secondo la regola seguente:

$$LMN\_PER = LMN * 27648/100$$

Con i valori di default (LMN_FAC = 1.0 e LMN_OFFS = 0.0) la normalizzazione è disattivata. Il valore regolante effettivo viene scritto nell'uscita LMN.

Memorizzazione dei parametri del regolatore SAVE_PAR

Se si ritiene che i parametri attuali del regolatore siano utilizzabili, prima di procedere a una modifica manuale è possibile memorizzarli in appositi parametri della struttura del DB di istanza dell'istruzione TCONT_CP. Con l'ottimizzazione della regolazione i parametri memorizzati vengono sovrascritti da quelli validi prima dell'ottimizzazione.

PFAC_SP, GAIN, TI, TD, D_F, CONZ_ON e CONZONE vengono scritti nella struttura PAR_SAVE.

Nuovo caricamento dei parametri del regolatore memorizzati UNDO_PAR

Tramite questa funzione i parametri del regolatore memorizzati per ultimi possono essere riattivati per il regolatore (solo in funzionamento manuale).

Commutazione tra i parametri PI e PID LOAD_PID (PID_ON)

Dopo l'ottimizzazione i parametri PI e PID vengono salvati nelle strutture PI_CON e PID_CON. Con LOAD_PID è possibile scrivere manualmente i parametri PI e PID sui parametri di regolazione effettivi in funzione di PID_ON.

Parametri PID PID_ON = TRUE	Parametri PI PID_ON = FALSE
• GAIN = PID_CON.GAIN • TI = PID_CON.TI • TD = PID_CON.TD	• GAIN = PI_CON.GAIN • TI = PI_CON.TI

NOTA

I parametri del regolatore vengono riscritti con UNDO_PAR o LOAD_PID solo se il guadagno del regolatore è diverso da zero:

Con LOAD_PID i parametri vengono copiati solo se $GAIN \neq 0$ (nel set di parametri PI o PID). In questo caso viene tenuto in considerazione il caso in cui non sia ancora stata eseguita un'ottimizzazione o che manchino parametri PID. Se prima $PID\_ON = TRUE$ e $PID.GAIN = FALSE$, PID_ON viene impostato a FALSE e i parametri PI vengono copiati.

- D_F, PFAC_SP vengono preimpostati dall'ottimizzazione. Successivamente essi possono essere modificati dall'utente. LOAD_PID non modifica questi parametri.
- La zona di regolazione viene sempre ricalcolata dall'inizio con LOAD_PID ($CON\_ZONE = 250/GAIN$) anche se $CONZ\_ON = FALSE$.

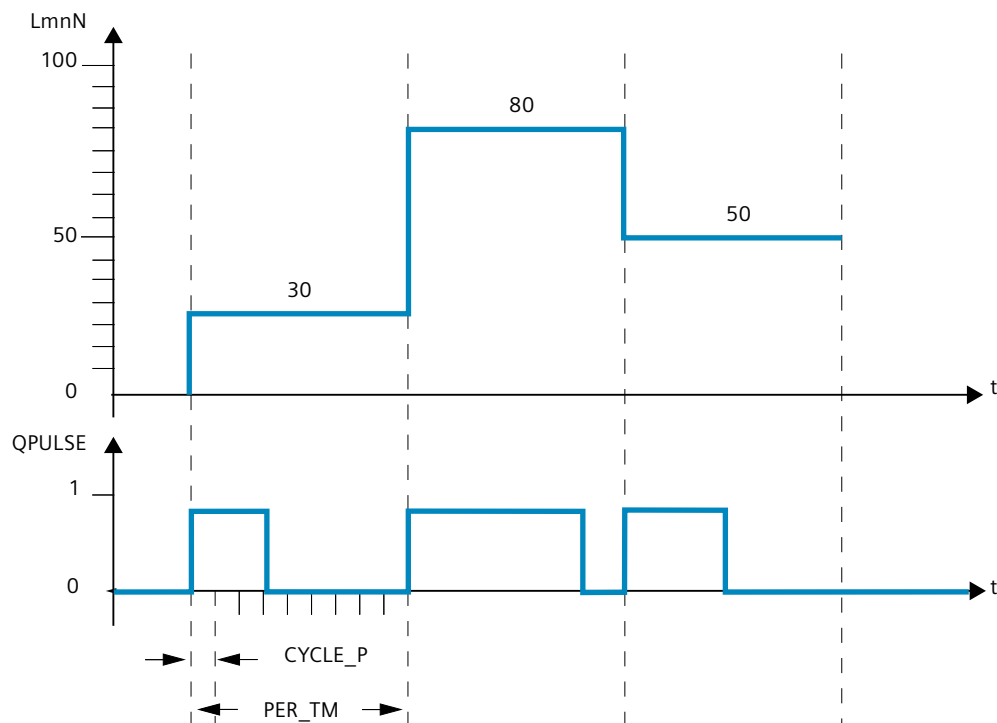
Vedere anche

[Funzionamento del generatore di impulsi \(Pagina 466\)](#)

[Schema a blocchi TCONT_CP \(Pagina 469\)](#)

10.4.4.3 Funzionamento del generatore di impulsi

La funzione PULSEGEN trasforma il valore regolante analogico LmnN in una sequenza di impulsi con periodo PER_TM tramite modulazione di larghezza di impulso. PULSEGEN si attiva quando PULSE_ON = TRUE e viene elaborata nel ciclo CYCLE_P.



Un valore regolante LmnN = 30 % e 10 richiami di PULSEGEN per PER_TM significano quindi:

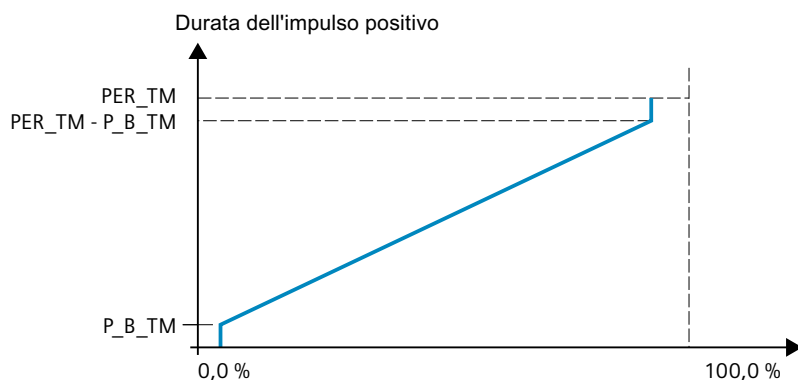
- TRUE nell'uscita QPULSE per i primi tre richiami di PULSEGEN (il 30 % di 10 richiami)
- FALSE nell'uscita QPULSE per sette ulteriori richiami di PULSEGEN (il 70 % di 10 richiami)

La durata di un impulso per periodo è proporzionale alla grandezza regolante e viene calcolata nel seguente modo:

$$\text{Durata dell'impulso} = \text{PER\_TM} * \text{LmnN} / 100$$

Se si sopprime la durata minima degli impulsi e delle pause, nell'area iniziale e finale della curva caratteristica del generatore compaiono delle flessioni.

La seguente figura illustra una regolazione a due punti con campo del valore regolante bipolare (da 0 % a 100 %):



Durata minima degli impulsi o delle pause (P_B_TM)

Tempi di attivazione e disattivazione brevi incidono negativamente sulla durata degli organi attuatori e dei dispositivi di regolazione. Questo inconveniente può essere evitato parametrizzando una durata minima degli impulsi e/o delle pause P_B_TM .

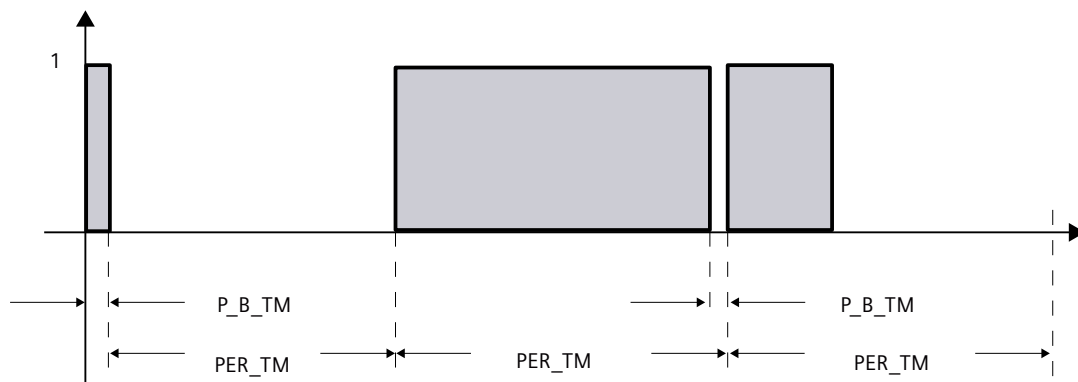
I valori assoluti di piccola entità della grandezza di ingresso $LmnN$, che potrebbero generare impulsi di durata inferiore a P_B_TM , vengono soppressi.

I valori di ingresso di grande entità che potrebbero generare impulsi di durata superiore a $PER\_TM - P\_B\_TM$ vengono invece impostati a 100 %. In questo modo si riduce la dinamica di generazione degli impulsi.

Per la durata minima degli impulsi e/o delle pause si consiglia di utilizzare valori di impostazione $P\_B\_TM \leq 0,1 * PER\_TM$.

Le flessioni delle curve caratteristiche rappresentate nella figura più sopra sono causate dalla durata minima degli impulsi e delle pause.

La figura seguente mostra il comportamento di commutazione dell'uscita degli impulsi:



Precisione della generazione degli impulsi

Minore è larghezza della griglia degli impulsi CYCLE_P rispetto alla durata periodo PER_TM, più precisa è la modulazione di larghezza di impulso. Per ottenere una regolazione sufficientemente precisa il rapporto dovrebbe essere il seguente:

$$\text{CYCLE\_P} \leq \text{PER\_TM}/50$$

In questo modo il valore regolante viene convertito in impulsi con una risoluzione di $\leq 2\%$.

NOTA

Se si richiama il regolatore nel ciclo del generatore di impulsi è importante considerare quanto segue:

Quando il regolatore viene richiamato nel ciclo del generatore di impulsi il valore istantaneo viene mediato. Di conseguenza nell'uscita PV possono essere presenti valori diversi per l'ingresso PV_IN o PV_PER. Se si vuole realizzare un inseguimento del setpoint si deve salvare il valore istantaneo del parametro di ingresso PV_IN durante i richiami dell'elaborazione complessiva del regolatore (QC_ACT = TRUE). Il valore istantaneo salvato deve essere quindi fornito ai parametri di ingresso PV_IN e SP_INT durante i richiami del generatore di impulsi.

Vedere anche

[Descrizione di TCONT_CP \(Pagina 464\)](#)

[Funzionamento di TCONT_CP \(Pagina 465\)](#)

[Schema a blocchi TCONT_CP \(Pagina 469\)](#)

[Parametri di ingresso di TCONT_CP \(Pagina 470\)](#)

[Parametri di uscita TCONT_CP \(Pagina 471\)](#)

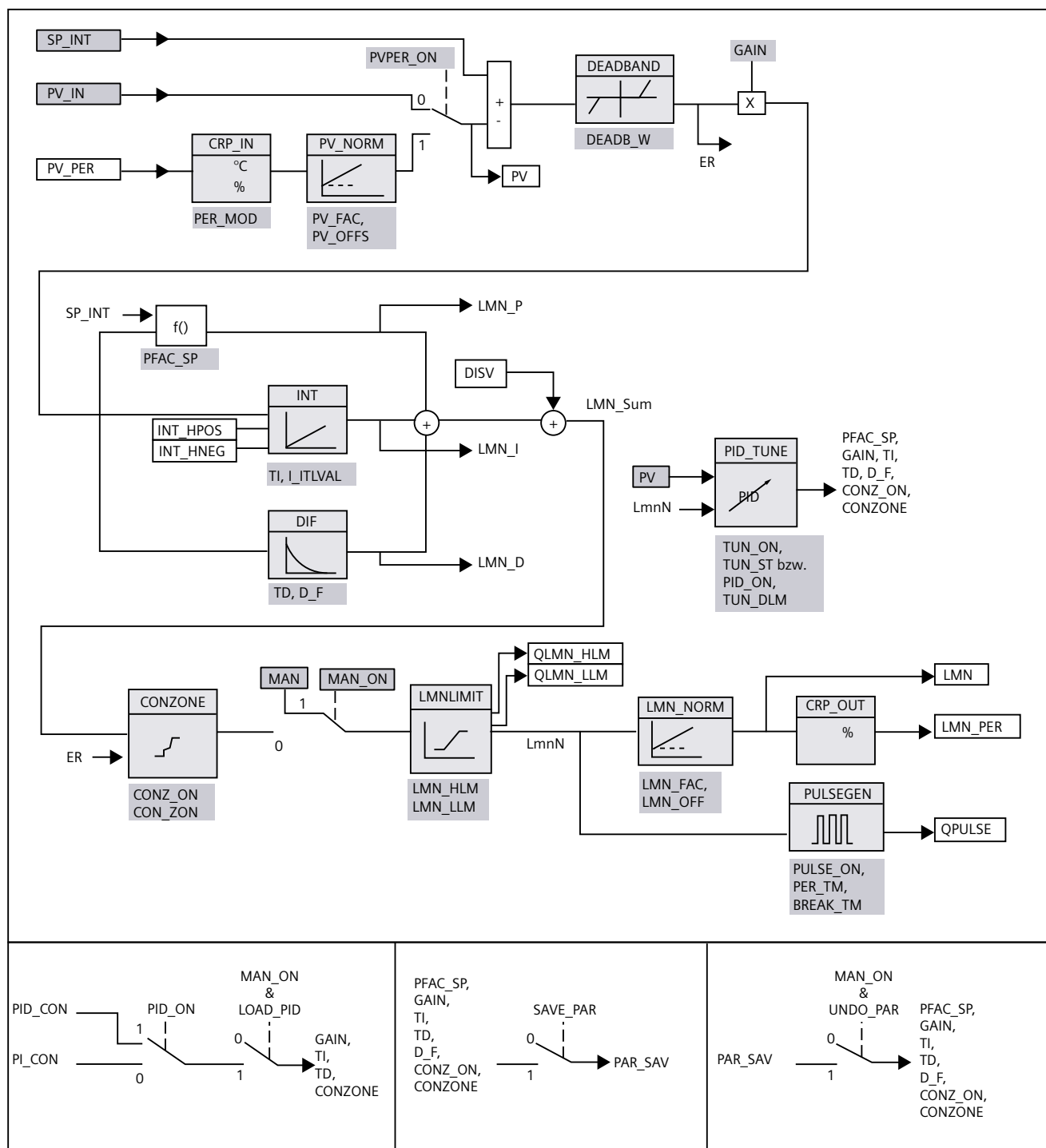
[Parametri di transito di TCONT_CP \(Pagina 472\)](#)

[Variabili statiche di TCONT_CP \(Pagina 472\)](#)

[Parametro STATUS_H \(Pagina 477\)](#)

[Parametro STATUS_D \(Pagina 478\)](#)

10.4.4.4 Schema a blocchi TCONT_CP



Vedere anche

[Descrizione di TCONT_CP \(Pagina 464\)](#)
[Funzionamento di TCONT_CP \(Pagina 465\)](#)
[Funzionamento del generatore di impulsi \(Pagina 474\)](#)
[Parametri di ingresso di TCONT_CP \(Pagina 470\)](#)
[Parametri di uscita TCONT_CP \(Pagina 471\)](#)
[Parametri di transito di TCONT_CP \(Pagina 472\)](#)
[Variabili statiche di TCONT_CP \(Pagina 472\)](#)
[Parametro STATUS_H \(Pagina 477\)](#)
[Parametro STATUS_D \(Pagina 478\)](#)

10.4.4.5 Parametri di ingresso di TCONT_CP

I nomi dei seguenti parametri si applicano sia al blocco dati sia all'accesso tramite l'API Openness.

Tabella 10-19

Parametri	Indirizzamento	Tipo di dati	Preassegnazione	Descrizione
PV_IN	0.0	REAL	0.0	Nell'ingresso "Ingresso valore istantaneo" è possibile parametrizzare un valore di messa in funzione o interconnettere un valore istantaneo esterno in virgola mobile. I valori validi variano a seconda dei sensori utilizzati.
PV_PER	4.0	INT	0	Il valore istantaneo in formato periferia viene interconnesso con il regolatore nell'ingresso "Valore istantaneo periferia".
DISV	6.0	REAL	0.0	Per l'inserzione di un segnale di disturbo, il segnale viene interconnesso con l'ingresso "Segnale di disturbo".
INT_HPOS	10.0	BOOL	FALSE	L'uscita dell'integratore può essere bloccata in direzione positiva. In questo caso l'ingresso INT_HPOS deve essere impostato a TRUE. Nel caso di una regolazione in cascata, INT_HPOS del regolatore pilota viene interconnesso con QLMN_HLM del regolatore asservito.
INT_HNEG	10.1	BOOL	FALSE	L'uscita dell'integratore può essere bloccata in direzione negativa. In questo caso l'ingresso INT_HNEG deve essere impostato a TRUE. Nel caso di una regolazione in cascata, INT_HNEG del regolatore pilota viene interconnesso con QLMN_LLM del regolatore asservito.
SELECT	12.0	INT	0	Se il generatore di impulsi è inserito, esistono diverse possibilità di richiamare l'algoritmo PID e il generatore di impulsi: • SELECT = 0: Il regolatore viene richiamato in un livello di schedulazione orologio rapida mentre vengono elaborati l'algoritmo PID e il generatore di impulsi. • SELECT = 1: Il regolatore viene richiamato nell'OB1 e viene elaborato solo l'algoritmo PID. • SELECT = 2: Il regolatore viene richiamato in un livello di schedulazione orologio rapida e viene elaborato solo il generatore di impulsi. • SELECT = 3: Il regolatore viene richiamato in un livello di schedulazione orologio lenta e viene elaborato solo l'algoritmo PID.

Vedere anche

[Funzionamento del generatore di impulsi \(Pagina 474\)](#)

[Schema a blocchi TCONT_CP \(Pagina 477\)](#)

10.4.4.6 Parametri di uscita TCONT_CP

I nomi dei seguenti parametri si applicano sia al blocco dati sia all'accesso tramite l'API Openness.

Tabella 10-20

Parametri	Indirizzamento	Tipo di dati	Preassegnazione	Descrizione
PV	14.0	REAL	0.0	Nell'uscita "Valore istantaneo" viene emesso il valore istantaneo effettivo. I valori validi variano a seconda dei sensori utilizzati.
LMN	18.0	REAL	0.0	Nell'uscita "Valore regolante" viene emesso il valore regolante effettivo in virgola mobile.
LMN_PER	22.0	INT	0	Il valore regolante in formato periferia viene interconnesso con il regolatore nell'uscita "Valore regolante periferia".
QPULSE	24.0	BOOL	FALSE	Il valore regolante viene emesso nell'uscita QPULSE con ampiezza di impulso modulata.
QLMN_HLM	24.1	BOOL	FALSE	Il valore regolante viene sempre limitato da un valore minimo e uno massimo. L'uscita QLMN_HLM segnala il raggiungimento del limite superiore.
QLMN_LLM	24.2	BOOL	FALSE	Il valore regolante viene sempre limitato da un valore minimo e uno massimo. L'uscita QLMN_LLM segnala il raggiungimento del limite inferiore.
QC_ACT	24.3	BOOL	TRUE	Questo parametro indica se, al successivo richiamo del blocco, verrà elaborato il componente del regolatore continuo (rilevante solo se SELECT ha il valore 0 o 1).

Vedere anche

[Funzionamento del generatore di impulsi \(Pagina 474\)](#)

[Schema a blocchi TCONT_CP \(Pagina 477\)](#)

[Parametro STATUS_H \(Pagina 477\)](#)

[Parametro STATUS_D \(Pagina 478\)](#)

10.4.4.7 Parametri di transito di TCONT_CP

I nomi dei seguenti parametri si applicano sia al blocco dati sia all'accesso tramite l'API Openness.

Tabella 10-21

Parametri	Indirizz-zo	Tipo di dati	Preas-segna-zione	Descrizione
CYCLE	26.0	REAL	0.1 s	Valore di default del tempo di campionamento per l'algoritmo PID. L'ottimizzatore calcola il tempo di campionamento nella fase 1 e lo registra in CYCLE. CYCLE > 0.001 s
CYCLE_P	30.0	REAL	0.02 s	In questo ingresso va immesso il tempo di campionamento per il componente del generatore di impulsi. L'istruzione TCONT_CP calcola il tempo di campionamento nella fase 1 e lo registra in CYCLE_P. CYCLE_P > 0.001 s
SP_INT	34.0	REAL	0.0	L'ingresso "Setpoint interno" permette di predefinire un setpoint. I valori validi variano a seconda dei sensori utilizzati.
MAN	38.0	REAL	0.0	L'ingresso "Valore manuale" consente di predefinire un valore manuale. In funzionamento automatico esso viene collegato con il valore regolante.
COM_RST	42.0	BOOL	FALSE	Il blocco ha una routine di inizializzazione che viene elaborata quando è impostato l'ingresso COM_RST.
MAN_ON	42.1	BOOL	TRUE	Se è impostato l'ingresso "Inserire funzionamento manuale" il circuito di regolazione è interrotto. Come valore regolante viene predefinito il valore manuale MAN.

Vedere anche

[Funzionamento del generatore di impulsi \(Pagina 474\)](#)

[Schema a blocchi TCONT_CP \(Pagina 477\)](#)

10.4.4.8 Variabili statiche di TCONT_CP

I nomi delle seguenti variabili si applicano sia al blocco dati sia all'accesso tramite l'API Openness.

Tabella 10-22

Parametri	Indirizzo	Tipo di dati	Preas-segna-zione	Descrizione
DEADB_W	44.0	REAL	0.0	Il differenziale di regolazione attraversa una banda morta. L'ingresso "Ampiezza banda morta" determina le dimensioni della banda morta. I valori validi variano a seconda dei sensori utilizzati.
I_ITLVAL	48.0	REAL	0.0	L'uscita dell'integratore può essere impostata nell'ingresso I_ITL_ON. Nell'ingresso "Valore di inizializzazione per componente I" si trova il valore di inizializzazione. Con un nuovo avviamento COM_RST = TRUE il componente I viene impostato al valore di inizializzazione. Sono ammessi valori da -100 a 100%.
LMN_HLM	52.0	REAL	100.0	Il valore regolante viene sempre limitato da un valore minimo e uno massimo. L'ingresso "Valore regolante, limite superiore" indica il limite superiore. LMN_HLM > LMN_LLM

Parametri	Indirizzo	Tipo di dati	Preas-segnazione	Descrizione
LMN_LLM	56.0	REAL	0.0	Il valore regolante viene sempre limitato da un valore minimo e uno massimo. L'ingresso "Valore regolante, limite inferiore" indica il limite inferiore. $LMN\_LLM < LMN\_HLM$
PV_FAC	60.0	REAL	1.0	L'ingresso "Fattore del valore istantaneo" viene moltiplicato per il "Valore istantaneo periferia". Questo ingresso consente l'adeguamento del campo del valore istantaneo.
PV_OFFS	64.0	REAL	0.0	L'ingresso "Offset del valore istantaneo" viene sommato al "Valore istantaneo periferia". Questo ingresso consente l'adeguamento del campo del valore istantaneo.
LMN_FAC	68.0	REAL	1.0	L'ingresso "Fattore del valore regolante" viene moltiplicato per il valore regolante. Questo ingresso consente l'adeguamento del campo del valore regolante.
LMN_OFFS	72.0	REAL	0.0	L'ingresso "Offset del valore regolante" viene sommato al valore regolante. Questo ingresso consente l'adeguamento del campo del valore regolante.
PER_TM	76.0	REAL	1.0 s	Nel parametro PER_TM viene immessa la durata periodo della modulazione di larghezza di impulso. Il rapporto tra durata di periodo e tempo di campionamento del generatore di impulsi determina la precisione della modulazione di larghezza di impulso. $PER\_TM \geq CYCLE$
P_B_TM	80.0	REAL	0.02 s	Nel parametro per la durata minima dell'impulso o della pausa è possibile parametrizzare una durata minima per l'impulso o per la pausa. P_B_TM viene limitato internamente a $> CYCLE\_P$.
TUN_DLMN	84.0	REAL	20.0	Il processo per l'ottimizzazione del regolatore viene attivato da una risposta a gradino del valore regolante di TUN_DLMN. Sono ammessi valori da -100 a 100%.
PER_MODE	88.0	INT	0	Questo interruttore consente di immettere il tipo di unità di ingresso e uscita. Il valore istantaneo nell'ingresso PV_PER viene così normalizzato nell'uscita PV. - PER_MODE = 0: Termocoppie; PT100/NI100; standard $PV\_PER * 0.1$ Unità: °C, °F - PER_MODE = 1: PT100/NI100; Clima $PV\_PER * 0.01$ Unità: °C, °F - PER_MODE = 2: Corrente/tensione $PV\_PER * 100/27648$ Unità: %
PVPER_ON	90.0	BOOL	FALSE	Per leggere il valore istantaneo dalla periferia è necessario interconnettere l'ingresso PV_PER con la periferia e impostare l'ingresso "Attivazione valore istantaneo dalla periferia".
I_ITL_ON	90.1	BOOL	FALSE	L'uscita dell'integratore può essere impostata sull'ingresso I_ITLVAL. Perciò deve essere impostato l'ingresso "Imposta componente I".
PULSE_ON	90.2	BOOL	FALSE	Con PULSE_ON = TRUE si attiva il generatore di impulsi.
TUN_KEEP	90.3	BOOL	FALSE	Il passaggio al funzionamento automatico viene effettuato soltanto se TUN_KEEP diventa FALSE.
ER	92.0	REAL	0.0	Nell'uscita "Differenza di regolazione" viene emessa la differenza di regolazione effettiva. I valori validi variano a seconda dei sensori utilizzati.
LMN_P	96.0	REAL	0.0	L'uscita "Componente P" contiene il componente proporzionale della grandezza regolante.

Parametri	Indirizzo	Tipo di dati	Preas-segnazione	Descrizione
LMN_I	100.0	REAL	0.0	L'uscita "Componente I" contiene il componente integrale della grandezza regolante.
LMN_D	104.0	REAL	0.0	L'uscita "Componente D" contiene il componente differenziale della grandezza regolante.
PHASE	108.0	INT	0	Nell'uscita PHASE viene visualizzata la fase attuale dell'ottimizzazione del regolatore. • PHASE = 0: Nessun funzionamento di ottimizzazione; funzionamento automatico o manuale • PHASE = 1: Ottimizzazione pronta; controllo dei parametri, attesa dell'attivazione, misura dei tempi di campionamento • PHASE = 2: Ottimizzazione effettiva: ricerca del punto di flesso con valore regolante costante. Immissione del tempo di campionamento nel DB di istanza. • PHASE = 3: Calcolo dei parametri di processo. Memorizzazione dei parametri del regolatore validi prima dell'ottimizzazione. • PHASE = 4: Progettazione del regolatore • PHASE = 5: Correzione del regolatore con una nuova grandezza regolante • PHASE = 7: Controllo del circuito di regolazione
STATUS_H	110.0	INT	0	STATUS_H indica un valore di diagnostica relativo alla ricerca del punto di flesso nel processo di riscaldamento.
STATUS_D	112.0	INT	0	STATUS_D indica un valore di diagnostica relativo al disegno del regolatore nel processo di riscaldamento.
QTUN_RUN	114.0	BOOL	0	L'ottimizzazione è iniziata con l'inserzione della grandezza regolante di ottimizzazione e si trova ancora nella fase 2 (ricerca del punto di flesso).
PI_CON	116.0	STRUCT		Parametro del regolatore PI
GAIN	+0.0	REAL	0.0	Guadagno del regolatore PI %/fisica Unità
TI	+4.0	REAL	0.0 s	Tempo di integrazione PI [s]
PID_CON	124.0	STRUCT		Parametro del regolatore PID
GAIN	+0.0	REAL	0.0	Guadagno del regolatore PID
TI	+4.0	REAL	0.0s	Tempo di integrazione PID [s]
TD	+8.0	REAL	0.0s	Tempo derivativo PID [s]
PAR_SAVE	136.0	STRUCT		In questa struttura vengono memorizzati i parametri PID.
PFAC_SP	+0.0	REAL	1.0	Fattore proporzionale in caso di modifica del setpoint Sono ammessi valori da 0.0 a 1.0.
GAIN	+4.0	REAL	0.0	Guadagno del regolatore %/fisica Unità
TI	+8.0	REAL	40.0 s	Tempo di integrazione [s]
TD	+12.0	REAL	10.0 s	Tempo derivativo [s]
D_F	+16.0	REAL	5.0	Fattore di derivazione Sono ammessi valori da 5.0 a 10.0.
CON_ZONE	+20.0	REAL	100.0	Ampiezza della zona di regolazione Se la differenza di regolazione è maggiore dell'ampiezza della zona di regolazione, come valore regolante viene emesso il limite superiore del valore regolante. Se la differenza di regolazione è minore dell'ampiezza della zona di regolazione negativa, come valore regolante viene emesso il limite inferiore del valore regolante. CON_ZONE ≥ 0.0

Parametri	Indirizzo	Tipo di dati	Preas-segnazione	Descrizione
CONZ_ON	+24.0	BOOL	FALSE	Attivazione della zona di regolazione
PFAC_SP	162.0	REAL	1.0	PFAC_SP indica il componente P effettivo in caso di modifiche del setpoint. Il parametro viene impostato tra 0 e 1. 1: componente P pienamente effettivo anche in caso di variazioni del setpoint 0: componente P non attivo in caso di variazioni del setpoint Sono ammessi valori da 0.0 a 1.0.
GAIN	166.0	REAL	2.0	L'ingresso "Coefficiente proporzionale" indica il guadagno del regolatore. L'inversione del senso di regolazione si ottiene con il segno negativo davanti a GAIN. %/fisica Unità
TI	170.0	REAL	40.0 s	L'ingresso "Tempo di integrazione" (tempo dell'azione integratrice) determina il comportamento temporale dell'integratore.
TD	174.0	REAL	10.0 s	L'ingresso "Tempo derivativo" (tempo dell'azione derivativa) determina il comportamento temporale del derivatore.
D_F	178.0	REAL	5.0	Il fattore di derivazione determina il ritardo del componente D. $D_F = \text{tempo derivativo} / \text{Ritardo del componente D}$ Sono ammessi valori da 5.0 a 10.0.
CON_ZONE	182.0	REAL	100.0	Se la differenza di regolazione è maggiore dell'ampiezza della zona di regolazione, come valore regolante viene emesso il limite superiore del valore regolante. Se la differenza di regolazione è minore dell'ampiezza della zona di regolazione negativa, come valore regolante viene emesso il limite inferiore del valore regolante. I valori validi variano a seconda dei sensori utilizzati.
CONZ_ON	186.0	BOOL	FALSE	Con CONZ_ON = TRUE è possibile attivare la zona di regolazione.
TUN_ON	186.1	BOOL	FALSE	Se TUN_ON = TRUE, viene effettuata la mediatura del valore regolante finché non viene inserita, in seguito a un gradino del setpoint o a TUN_ST = TRUE, l'attivazione del valore regolante TUN_DLMN.
TUN_ST	186.2	BOOL	FALSE	Se con l'ottimizzazione del regolatore nel punto di lavoro il setpoint deve restare costante, con TUN_ST = 1 viene inserito un gradino del valore regolante di TUN_DLMN.
UNDO_PAR	186.3	BOOL	FALSE	Carica i parametri del regolatore PFAC_SP, GAIN, TI, TD, D_F, CONZ_ON e CON_ZONE dalla struttura dei dati PAR_SAVE (solo in funzionamento manuale).
SAVE_PAR	186.4	BOOL	FALSE	Salva i parametri del regolatore PFAC_SP, GAIN, TI, TD, D_F, CONZ_ON e CON_ZONE nella struttura dei dati PAR_SAVE.
LOAD_PID	186.5	BOOL	FALSE	Carica i parametri del regolatore GAIN, TI, TD in funzione di PID_ON dalla struttura dei dati PI_CON o PID_CON (solo in funzionamento manuale).
PID_ON	186.6	BOOL	TRUE	Nell'ingresso PID_ON è possibile stabilire se utilizzare il regolatore ottimizzato come regolatore PI o PID. - Regolatore PID: PID_ON = TRUE - Regolatore PI: PID_ON = FALSE È tuttavia possibile che in alcuni tipi di circuito venga disegnato solo un regolatore PI nonostante PID_ON = TRUE.
GAIN_P	188.0	REAL	0.0	Guadagno del processo identificato. Con il tipo di circuito I, GAIN_P viene tendenzialmente stimato insufficiente.
TU	192.0	REAL	0.0	Tempo di ritardo identificato del processo. $TU \geq 3 * \text{CYCLE}$
TA	196.0	REAL	0.0	Tempo di compensazione identificato del processo. Con il tipo di circuito I, TA viene tendenzialmente stimato insufficiente.

Parametri	Indirizzo	Tipo di dati	Preas-segnazione	Descrizione
KIG	200.0	REAL	0.0	Aumento massimo del valore istantaneo con un'attivazione della grandezza regolante da 0 a 100% [1/s] $GAIN\_P = 0.01 * KIG * TA$
N_PTN	204.0	REAL	0.0	Questo parametro indica l'ordine del circuito. Sono possibili anche valori numerici non interi. Sono ammessi valori da 1.01 a 10.0.
TM_LAG_P	208.0	REAL	0.0	Costante temporale di un modello PTN (valori opportuni solo per $N\_PTN \geq 2$).
T_P_INF	212.0	REAL	0.0	Intervallo di tempo che trascorre dall'attivazione del processo al punto di flesso.
P_INF	216.0	REAL	0.0	Variazione del valore istantaneo dall'attivazione del processo al punto di flesso. I valori validi variano a seconda dei sensori utilizzati.
LMNO	220.0	REAL	0.0	Valore regolante all'inizio dell'ottimizzazione Viene determinato nella fase 1 (valore medio). Sono ammessi valori da 0 a 100%.
PV0	224.0	REAL	0.0	Valore istantaneo all'inizio dell'ottimizzazione
PVDT0	228.0	REAL	0.0	Incremento del valore istantaneo all'inizio dell'ottimizzazione [1/s] Segno adattato.
PVDT	232.0	REAL	0.0	Incremento momentaneo del valore istantaneo [1/s] Segno adattato.
PVDT_MAX	236.0	REAL	0.0	Variazione max. del valore istantaneo al secondo [1/s] Variazione massima del valore istantaneo nel punto di flesso (segno adattato, sempre > 0); viene utilizzata per calcolare TU e KIG.
NOI_PVDT	240.0	REAL	0.0	Rumorosità in PVDT_MAX in % Maggiore è la rumorosità, meno precisi sono i parametri del regolatore.
NOISE_PV	244.0	REAL	0.0	Rumorosità assoluta nel valore istantaneo Differenza tra valore istantaneo massimo e minimo nella fase 1.
FIL_CYC	248.0	INT	1	Numero dei cicli del filtro del valore medio Tramite cicli FIL_CYC viene calcolato il valore istantaneo medio. Se necessario, FIL_CYC viene aumentato automaticamente da 1 a max. 1024.
POI_CMAX	250.0	INT	2	Numero massimo di cicli dopo il punto di flesso Questo intervallo di tempo viene impiegato per cercare un ulteriore (cioè migliore) punto di flesso in caso di rumorosità di misura. Solo a quel punto si conclude l'ottimizzazione.
POI_CYCL	252.0	INT	0	Numero di cicli dopo il punto di flesso

Vedere anche

[Funzionamento del generatore di impulsi \(Pagina 474\)](#)

[Schema a blocchi TCONT_CP \(Pagina 477\)](#)

10.4.4.9 Parametro STATUS_H

STATUS_H	Descrizione	Rimedio
0	Default oppure non sono (ancora) presenti nuovi parametri del regolatore	
10000	Ottimizzazione conclusa + Sono stati trovati parametri del regolatore adatti	
2xxxx	Ottimizzazione conclusa + Parametri del regolatore incerti	
2xx2x	Punto di flesso non raggiunto (solo in caso di attivazione tramite gradino del setpoint)	Se il regolatore oscilla, immettere parametri del regolatore minori o ripetere il tentativo riducendo la differenza del valore regolante TUN_DLMN.
2x1xx	Errore di valutazione ($TU < 3 \cdot CYCLE$)	Ridurre CYCLE e ripetere il tentativo. Eccezione: circuito esclusivamente PT1: non ripetere il tentativo; eventualmente ridurre i valori dei parametri del regolatore
2x3xx	Errore di valutazione TU eccessivo	Ripetere il tentativo con condizioni migliori.
21xxx	Errore di valutazione $N\_PTN < 1$	Ripetere il tentativo con condizioni migliori.
22xxx	Errore di valutazione $N\_PTN > 10$	Ripetere il tentativo con condizioni migliori.
3xxxx	Ottimizzazione interrotta nella fase 1 causa parametrizzazione errata:	
30002	Differenza effettiva del valore regolante $< 5\%$	Correggere la differenza del valore regolante TUN_DLMN.
30005	I tempi di campionamento CYCLE e CYCLE_P si discostano dai valori misurati di oltre il 5%.	Confrontare CYCLE e CYCLE_P con il tempo di ciclo del livello di schedulazione orologio e osservare eventuali distributori dei richiami. Verificare la capacità della CPU. Un carico di lavoro eccessivo sulla CPU determina tempi di campionamento maggiori e inadeguati a CYCLE o CYCLE_P.

NOTA

Interrompendo l'ottimizzazione nelle fasi 1 o 2 si avrà STATUS_H = 0. STATUS_D, tuttavia, indica ancora lo stato dell'ultimo calcolo del regolatore.

Maggiore è il valore di STATUS_D, maggiore sarà l'ordine del sistema regolato, maggiore sarà il rapporto TU/TA e più imprecisi saranno i parametri del regolatore.

Vedere anche

[Funzionamento del generatore di impulsi \(Pagina 474\)](#)

[Schema a blocchi TCONT_CP \(Pagina 477\)](#)

10.4.4.10 Parametro STATUS_D

STATUS_D	Descrizione
0	Non sono stati calcolati parametri del regolatore
110	$N_{PTN} \leq 1.5$ tipo di circuito I veloce
121	$N_{PTN} > 1.5$ tipo di circuito I
200	$N_{PTN} > 1.9$ tipo di circuito II (area di transizione)
310	$N_{PTN} \geq 2.1$ tipo di circuito III veloce
320	$N_{PTN} > 2.6$ tipo di circuito III
111, 122, 201, 311, 321	I parametri sono stati corretti nella fase 7.

NOTA

Maggiore è il valore di STATUS_D, maggiore sarà l'ordine del sistema regolato, maggiore sarà il rapporto TU/TA e più imprecisi saranno i parametri del regolatore.

Vedere anche

[Funzionamento del generatore di impulsi \(Pagina 474\)](#)

[Schema a blocchi TCONT_CP \(Pagina 477\)](#)

10.4.5 TCONT_S

10.4.5.1 Descrizione di TCONT_S

L'istruzione TCONT_S consente la regolazione di processi tecnici di temperatura con segnali di uscita binari del valore regolante per organi attuatori integranti nei sistemi di automazione SIMATIC S7. La modalità di funzionamento si basa sull'algoritmo di regolazione PI del regolatore digitale. Il regolatore a passi funziona senza segnalazione di risposta della posizione.

Impiego

Il regolatore può essere utilizzato anche in una struttura a cascata come regolatore di posizione subordinato. La posizione dell'organo attuatore va predefinita nell'ingresso del setpoint SP_INT. In questo caso l'ingresso del valore istantaneo e il parametro TI (tempo di integrazione) devono essere impostati a zero. Un esempio di applicazione pratica è ad es. una regolazione della temperatura con regolazione del potere calorifico tramite un comando impulso-pausa e regolazione del potere di raffreddamento tramite una valvola a farfalla. Per chiudere completamente la valvola, la grandezza regolante ($ER \cdot GAIN$) dovrebbe diventare negativa.

Richiamo

L'istruzione TCONT_S deve essere richiamata in modo equidistante, Utilizzare un livello di schedulazione orologio (ad es. l'OB35 per i sistemi S7-300). Il tempo di campionamento viene predefinito nel parametro CYCLE.

Tempo di campionamento CYCLE

Il tempo di campionamento CYCLE deve coincidere con l'intervallo di tempo tra due richiami (tempo di ciclo dell'OB di schedulazione orologio considerate le scansioni).

Il tempo di campionamento del regolatore non deve superare di oltre il 10% il tempo di integrazione rilevato del regolatore (TI). Normalmente, per garantire la precisione necessaria del regolatore a passi, occorre impostare un tempo di campionamento molto inferiore.

Precisione necessaria G	MTR_TM	$CYCLE = MTR\_TM * G$	Commento
0,5 %	10 s	0,05 s	Il tempo di campionamento viene determinato dalla precisione necessaria per il regolatore a passi.

Avviamento

L'istruzione TCONT_S dispone di una routine di inizializzazione che viene eseguita se è impostato il parametro di ingresso COM_RST = TRUE. Una volta elaborata la routine di inizializzazione COM_RST il blocco viene reimpostato su FALSE. Tutte le altre uscite vengono impostate ai loro valori iniziali. Per eseguire un'inizializzazione a ogni nuovo avviamento della CPU, richiamare il blocco nell'OB 100 con COM_RST = TRUE.

Vedere anche

[Schema a blocchi di TCONT_S \(Pagina 484\)](#)

10.4.5.2 Funzionamento di TCONT_S

Ramo del setpoint

Il setpoint può essere immesso nell'ingresso SP_INT in virgola mobile in unità fisiche o percentuali. Nel calcolo della differenza di regolazione, il setpoint e quello istantaneo devono essere espressi nella stessa unità.

Selezione del valore istantaneo (PVPER_ON)

A seconda del parametro PVPER_ON il valore istantaneo viene letto in formato periferia o in virgola mobile.

PVPER_ON	Introduzione del valore istantaneo
TRUE	Il valore istantaneo viene letto dalla periferia analogica (PEW xxx) nell'ingresso PV_PER.
FALSE	Il valore istantaneo viene letto in formato di virgola mobile nell'ingresso PV_IN.

Commutazione del formato del valore istantaneo CRP_IN (PER_MODE)

La funzione CRP_IN converte il valore di periferia PV_PER in un formato di virgola mobile in base all'interruttore PER_MODE secondo la seguente regola:

PER_MODE	Uscita di CRP_IN	Tipo di introduzione analogica	Unità
0	$PV\_PER * 0.1$	Termocoppie; PT100/Ni100; standard	°C; °F
1	$PV\_PER * 0.01$	PT100/Ni100; Climat;	°C; °F
2	$PV\_PER * 100/27648$	Tensione/corrente	%

Normalizzazione del valore istantaneo PV_NORM (PV_FAC, PV_OFFS)

La funzione PV_NORM calcola l'uscita di CRP_IN secondo la regola seguente:

"Uscita di PV_NORM" = "uscita di CRP_IN" * PV_FAC + PV_OFFS

Questa funzione può essere utilizzata per i seguenti scopi:

- Adeguamento del valore istantaneo con PV_FAC come coefficiente del valore istantaneo e PV_OFFS come offset del valore istantaneo
- Normalizzazione della temperatura in percentuale
Per poter indicare il valore di riferimento come percentuale è necessario convertire in percentuale il setpoint misurato.
- Normalizzazione della percentuale secondo la temperatura
Per poter immettere il setpoint nella grandezza fisica della temperatura è necessario convertire in temperatura il valore di tensione/corrente misurato.

Calcolo dei parametri:

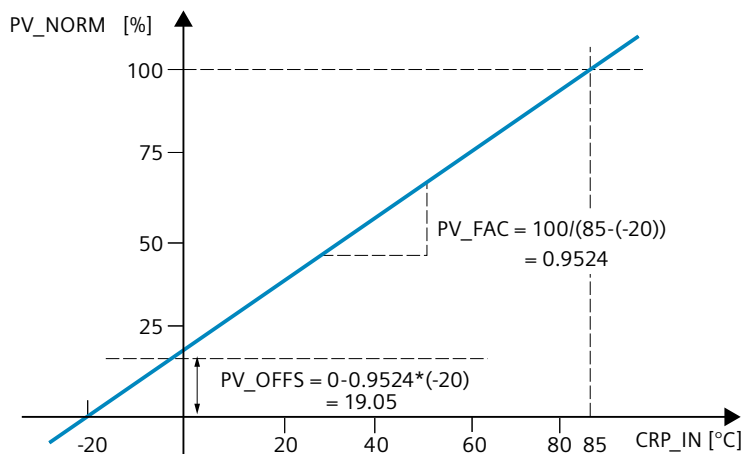
- $PV\_FAC = \text{campo di } PV\_NORM / \text{campo di } CRP\_IN$;
- $PV\_OFFS = LIMINF(PV\_NORM) - PV\_FAC * LIMINF(CRP\_IN)$;
con LIMINF: Limite inferiore

Con i valori di default ($PV\_FAC = 1.0$ e $PV\_OFFS = 0.0$) la normalizzazione è disattivata. Il valore di misura effettivamente attivo viene emesso nell'uscita PV.

Esempio di normalizzazione del valore istantaneo

Se si desidera preimpostare il setpoint come percentuale e il campo della temperatura in CRP_IN è compreso tra -20 e 85 °C, è necessario convertire il campo della temperatura in percentuale.

La figura seguente mostra un esempio di adeguamento del campo della temperatura da -20 a 85 °C al campo da 0 a 100% interno:



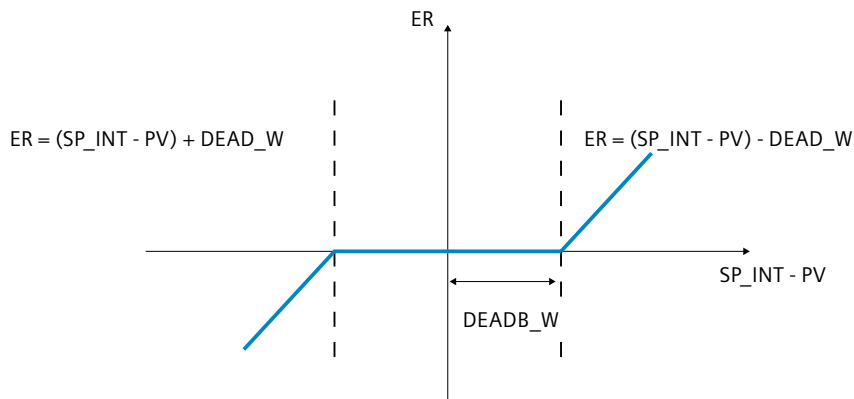
Calcolo della differenza di regolazione

Dalla differenza tra il setpoint e quello istantaneo risulta la differenza di regolazione prima della banda morta.

Il setpoint e il valore istantaneo devono avere la stessa unità.

Banda morta (DEADB_W)

Per sopprimere una piccola oscillazione continua causata dalla quantizzazione delle grandezze regolanti (ad es. nel caso di una modulazione di larghezza di impulso con PULSEGEN), la differenza di regolazione viene condotta attraverso una banda morta (DEADBAND). Se DEADB_W = 0.0 la banda morta è disattivata.



Algoritmo del regolatore a passi PI

L'istruzione TCONT_S funziona senza risposta di posizione. Il componente I dell'algoritmo PI e la risposta di posizione prevista vengono calcolati in un unico integratore (INT) e confrontati con il componente P rimasto sotto forma di valore di ritorno. La differenza va a un regolatore a tre punti (THREE_ST) e un generatore di impulsi (PULSEOUT) che genera gli impulsi per la valvola di regolazione. La frequenza di commutazione del regolatore viene ridotta tramite un adattamento della soglia d'intervento dell'elemento a tre punti.

Attenuazione del componente P al variare del setpoint

Per evitare sovraelongazioni è possibile ridurre il componente P con il parametro "Coefficiente proporzionale in caso di modifiche del setpoint" (PFAC_SP). PFAC_SP consente di definire in modo continuo tra 0.0 e 1.0 l'intensità con cui il componente P deve agire in caso di variazioni del setpoint:

- PFAC_SP = 1.0: componente P pienamente attivo in caso di variazione del setpoint
- PFAC_SP = 0.0: senza componente P in caso di variazione del setpoint

Come per il regolatore continuo, un valore di PFAC_SP < 1.0 è in grado di ridurre le sovraelongazioni se il tempo di funzionamento del motore MTR_TM è breve rispetto al tempo di compensazione TA e il rapporto è $TU/TA < 0.2$. Se MTR_TM raggiunge il 20% di TA, è sufficiente un piccolo miglioramento.

Inserzione di un segnale di disturbo

Nell'ingresso DISV è possibile inserire un segnale di disturbo.

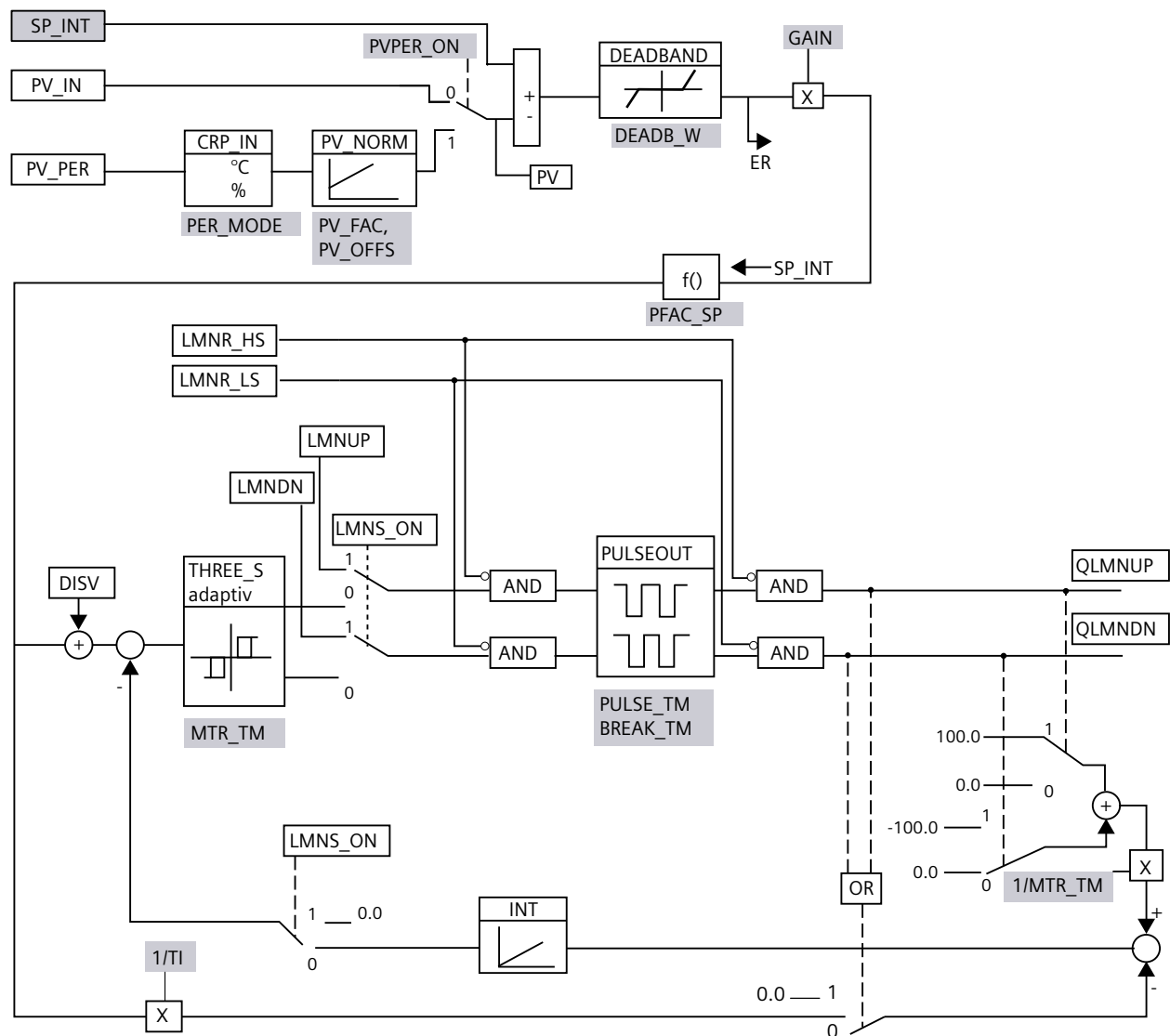
Elaborazione del valore manuale (LMNS_ON, LMNUP, LMNDN)

Con LMNS_ON è possibile commutare tra funzionamento manuale e automatico. In funzionamento manuale l'attuatore viene arrestato e l'integratore (INT) azzerato internamente. Con LMNUP e LMNDN l'attuatore può essere portato in posizione APERTO e CHIUSO. La commutazione sul funzionamento automatico avviene così con "sobbalzi". La differenza di regolazione presente porta a una variazione a gradino della grandezza regolante interna attraverso GAIN. Con l'attuatore integrale tuttavia si ottiene soltanto un comando del processo con andamento a rampa.

Vedere anche

[Schema a blocchi di TCONT_S \(Pagina 484\)](#)

10.4.5.3 Schema a blocchi di TCONT_S



■ Interfaccia di parametrizzazione

▬ Interfaccia di richiamo dell'istruzione

■ Interfaccia di parametrizzazione, interfaccia di richiamo dell'istruzione

Vedere anche

[Descrizione di TCONT_S \(Pagina 486\)](#)

[Funzionamento di TCONT_S \(Pagina 488\)](#)

[Parametri di ingresso di TCONT_S \(Pagina 485\)](#)

[Parametri di uscita di TCONT_S \(Pagina 486\)](#)

[Parametri di transito di TCONT_S \(Pagina 486\)](#)

[Variabili statiche di TCONT_S \(Pagina 487\)](#)

10.4.5.4 Parametri di ingresso di TCONT_S

I nomi dei seguenti parametri si applicano sia al blocco dati sia all'accesso tramite l'API Openness.

Tabella 10-23

Parametri	Indirizzo	Tipo di dati	Preassegnazione	Descrizione
CYCLE	0.0	REAL	0.1 s	In questo ingresso si immette il tempo di campionamento per il regolatore. $CYCLE \geq 0.001$
SP_INT	4.0	REAL	0.0	L'ingresso "Setpoint interno" permette di predefinire un setpoint. I valori validi variano a seconda dei sensori utilizzati.
PV_IN	8.0	REAL	0.0	Nell'ingresso "Ingresso valore istantaneo" è possibile parametrizzare un valore di messa in funzione o interconnettere un valore istantaneo esterno in virgola mobile. I valori validi variano a seconda dei sensori utilizzati.
PV_PER	12.0	INT	0	Il valore istantaneo in formato periferia viene interconnesso con il regolatore nell'ingresso "Valore istantaneo periferia".
DISV	14.0	REAL	0.0	Per l'inserzione di un segnale di disturbo, il segnale viene interconnesso con l'ingresso "Segnale di disturbo".
LMNR_HS	18.0	BOOL	FALSE	Il segnale "Valvola di regolazione sull'arresto superiore" viene interconnesso con l'ingresso "Segnale di arresto superiore della segnalazione di risposta di posizione". LMNR_HS=TRUE: la valvola di regolazione si trova sul riscontro superiore.
LMNR_LS	18.1	BOOL	FALSE	Il segnale "Valvola di regolazione sull'arresto inferiore" viene interconnesso con l'ingresso "Segnale di arresto inferiore della segnalazione di risposta di posizione". LMNR_LS=TRUE: La valvola di regolazione si trova sul riscontro inferiore.
LMNS_ON	18.2	BOOL	TRUE	Nell'ingresso "Inserire funzionamento manuale dei segnali dei valori regolanti" l'elaborazione del segnale del valore regolante viene commutata sul funzionamento manuale.
LMNUP	18.3	BOOL	FALSE	Con il funzionamento manuale dei segnali dei valori regolanti, nell'ingresso "Segnale del valore regolante, alto" viene comandato il segnale di uscita QLMNUP.
LMNDN	18.4	BOOL	FALSE	Con il funzionamento manuale dei segnali dei valori regolanti, nell'ingresso "Segnale del valore regolante, basso" viene comandato il segnale di uscita QLMNDN.

Vedere anche

[Schema a blocchi di TCONT_S \(Pagina 492\)](#)

10.4.5.5 Parametri di uscita di TCONT_S

I nomi dei seguenti parametri si applicano sia al blocco dati sia all'accesso tramite l'API Openness.

Tabella 10-24

Parametri	Indirizzo	Tipo di dati	Preassegnazione	Descrizione
QLMNUP	20.0	BOOL	FALSE	Se è impostata l'uscita "Segnale del valore regolante, alto", la valvola di regolazione deve essere aperta.
QLMNDN	20.1	BOOL	FALSE	Se è impostata l'uscita "Segnale del valore regolante, basso", la valvola di regolazione deve essere chiusa.
PV	22.0	REAL	0.0	Nell'uscita "Valore istantaneo" viene emesso il valore istantaneo effettivo.
ER	26.0	REAL	0.0	Nell'uscita "Differenza di regolazione" viene emessa la differenza di regolazione effettiva.

Vedere anche

[Schema a blocchi di TCONT_S \(Pagina 492\)](#)

10.4.5.6 Parametri di transito di TCONT_S

I nomi dei seguenti parametri si applicano sia al blocco dati sia all'accesso tramite l'API Openness.

Tabella 10-25

Parametri	Indirizzo	Tipo di dati	Preassegnazione	Descrizione
COM_RST	30.0	BOOL	FALSE	Il blocco ha una routine di inizializzazione che viene elaborata quando è impostato l'ingresso COM_RST.

Vedere anche

[Schema a blocchi di TCONT_S \(Pagina 492\)](#)

10.4.5.7 Variabili statiche di TCONT_S

I nomi delle seguenti variabili si applicano sia al blocco dati sia all'accesso tramite l'API Openness.

Tabella 10-26

Parametri	Indiriz- zo	Tipo di dati	Preas- segna- zione	Descrizione
PV_FAC	32.0	REAL	1.0	L'ingresso "Fattore del valore istantaneo" viene moltiplicato per il valore istantaneo. Questo ingresso consente l'adeguamento del campo del valore istantaneo.
PV_OFFS	36.0	REAL	0.0	L'ingresso "Offset del valore istantaneo" viene sommato al valore istantaneo. Questo ingresso consente l'adeguamento del campo del valore istantaneo. I valori validi variano a seconda dei sensori utilizzati.
DEADB_W	40.0	REAL	0.0	Il differenziale di regolazione attraversa una banda morta. L'ingresso "Ampiezza banda morta" determina le dimensioni della banda morta. $DEADB\_W \geq 0.0$
PFAC_SP	44.4	REAL	1.0	PFAC_SP indica il componente P effettivo in caso di modifiche del setpoint. 1: componente P pienamente effettivo in caso di variazione del setpoint 0: componente P non attivo in caso di variazione del setpoint Sono ammessi valori da 0.0 a 1.0.
GAIN	48.0	REAL	2.0	L'ingresso "Coefficiente proporzionale" indica il guadagno del regolatore. L'inversione del senso di regolazione si ottiene con il segno negativo davanti a GAIN. %/fisica Unità
TI	52.0	REAL	40.0 s	L'ingresso "Tempo di integrazione" (tempo dell'azione integratrice) determina il comportamento temporale dell'integratore.
MTR_TM	56.0	REAL	30 s	Nel parametro "Tempo di regolazione motore" viene registrato il tempo impiegato dalla valvola di regolazione tra un riscontro e l'altro. $MTR\_TM \geq CYCLE$
PULSE_TM	60.0	REAL	0.0 s	Nel parametro "Durata minima dell'impulso" è possibile parametrizzare la durata minima dell'impulso.
BREAK_TM	64.0	REAL	0.0 s	Nel parametro "Durata minima pausa" è possibile parametrizzare una lunghezza minima della pausa.
PER_MODE	68.0	INT	0	Questo interruttore consente di immettere il tipo di unità di ingresso e uscita. Il valore istantaneo nell'ingresso PV_PER viene così normalizzato nell'uscita PV. - PER_MODE = 0: Termocoppie; PT100/NI100; standard $PV\_PER * 0.1$ Unità: °C, °F - PER_MODE = 1: PT100/NI100; Clima $PV\_PER * 0.01$ Unità: °C, °F - PER_MODE = 2: Corrente/tensione $PV\_PER * 100/27648$ Unità: %
PVPER_ON	70.0	BOOL	FALSE	Per leggere il valore istantaneo dalla periferia è necessario interconnettere l'ingresso PV_PER con la periferia e impostare l'ingresso "Attivazione valore istantaneo dalla periferia".

Vedere anche[Schema a blocchi di TCONT_S \(Pagina 492\)](#)**10.4.6 Funzioni di sistema integrate****10.4.6.1 CONT_C_SF****CONT_C_SF**

L'istruzione CONT_C_SF è integrata nelle CPU Compact S7-300. L'istruzione non deve essere trasmessa durante la fase di caricamento nella CPU S7-300. L'estensione della funzione corrisponde all'istruzione CONT_C.

Vedere anche[Descrizione di CONT_C \(Pagina 444\)](#)[Funzionamento di CONT_C \(Pagina 445\)](#)[Schema a blocchi di CONT_C \(Pagina 447\)](#)[Parametri di ingresso di CONT_C \(Pagina 448\)](#)[Parametri di uscita di CONT_C \(Pagina 449\)](#)**10.4.6.2 CONT_S_SF****CONT_S_SF**

L'istruzione CONT_S_SF è integrata nelle CPU Compact S7-300. L'istruzione non deve essere trasmessa durante la fase di caricamento nella CPU S7-300. L'estensione della funzione corrisponde all'istruzione CONT_S.

Vedere anche[Descrizione di CONT_S \(Pagina 450\)](#)[Funzionamento di CONT_S \(Pagina 451\)](#)[Schema a blocchi di CONT_S \(Pagina 452\)](#)[Parametri di ingresso di CONT_S \(Pagina 453\)](#)[Parametri di uscita di CONT_S \(Pagina 454\)](#)

10.4.6.3 PULSEGEN_SF

PULSEGEN_SF

L'istruzione PULSEGEN_SF è integrata nelle CPU Compact S7-300. L'istruzione non deve essere trasmessa durante la fase di caricamento nella CPU S7-300. L'estensione della funzione corrisponde all'istruzione PULSEGEN.

Vedere anche

[Descrizione di PULSEGEN \(Pagina 455\)](#)

[Funzionamento di PULSEGEN \(Pagina 456\)](#)

[Modi di funzionamento di PULSEGEN \(Pagina 459\)](#)

[Regolazione a tre punti \(Pagina 459\)](#)

[Regolazione a due punti \(Pagina 462\)](#)

[Parametri di ingresso di PULSEGEN \(Pagina 463\)](#)

[Parametri di uscita di PULSEGEN \(Pagina 464\)](#)

10.5 Polyline

10.5.1 Compatibilità con CPU e FW

La tabella seguente mostra quali versioni di Polyline sono compatibili con le varie CPU .:

CPU	FW	Polyline
S7-1200	da V4.2	V1.0
S7-1200 G2	da V1.0	V1.0
CPU basate su S7-1500	da V2.0	V1.0

10.5.2 Descrizione Polyline

Descrizione

L'istruzione Polyline forma il valore di ingresso Input tramite la caratteristica sul valore di uscita Output. La caratteristica viene definita come tratto poligonale con max. 50 punti di interpolazione. Tra questi punti viene eseguita l'interpolazione lineare. Dal numero e dalla configurazione dei punti di interpolazione è possibile adeguare il tratto poligonale alla caratteristica desiderata.

L'istruzione Polyline può essere impiegata ad es. per effettuare la linearizzazione del comportamento non lineare di sensori e attuatori.

Calcolo interpolazione

Polyline calcola il valore di uscita sul parametro Output per il valore di ingresso sul parametro Input, che si colloca tra i valori di punto di interpolazione x_i e x_{i+1} con interpolazione lineare. Il calcolo dell'interpolazione lineare avviene con la seguente formula:

$$\text{Output} = \frac{(\text{Input} - x_i)}{(x_{i+1} - x_i)} (y_{i+1} - y_i) + y_i$$

Il parametro Reset = TRUE consente anche di definire un valore di uscita alternativo tramite il parametro SubstituteOutput.

Dati del tratto poligonale

Le coppie di valori per il tratto poligonale sono contenuti nell'area Static dell'istruzione.

NOTA

- Il numero minimo delle coppie di valori da configurare ammonta a 2.
 - Il numero massimo delle coppie di valori da configurare ammonta a 50.
 - Per una configurazione valida i valori x devono essere predefiniti in ordine crescente.
-

Per consentire modifiche ai dati del tratto poligonale senza che queste ultime siano immediatamente attive, le coppie di valori di questo tratto sono duplicate e contenute nelle seguenti strutture:

- UserData

I dati del tratto poligonale in questa struttura sono editabili.

Utilizzare questa struttura per definire o modificare i dati del tratto poligonale. Eventuali modifiche apportate in questa struttura si ripercuotono sul calcolo dell'interpolazione soltanto dopo che il controllo e la copia dei dati nella struttura WorkingData sono stati avviati. Ciò avviene impostando Validate = TRUE oppure automaticamente durante la prima elaborazione di Polyline dopo la commutazione dello stato di funzionamento della CPU da STOP a RUN.

La predefinizione dei valori in questa struttura non costituisce una configurazione valida. Per l'impiego di questi valori nel calcolo dell'interpolazione, provvedere all'impostazione delle variabili su valori validi.

- WorkingData

I dati del tratto poligonale in questa struttura non sono editabili. Questi dati vengono utilizzati per il calcolo dell'interpolazione. Non modificare manualmente i dati in questa struttura.

Entrambe le strutture presentano lo stesso tipo di dati e pertanto lo stesso contenuto:

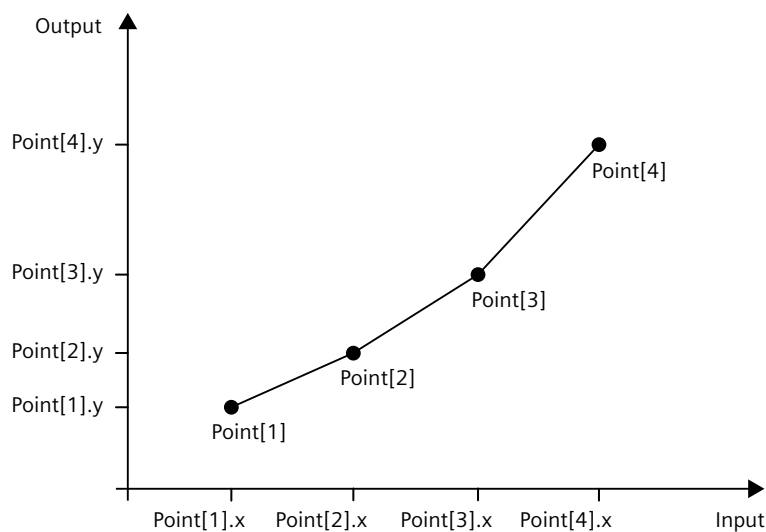
- NumberOfUsedPoints

Numero di punti utilizzati per il calcolo dell'interpolazione

- Point

L'array con 50 elementi contiene le coppie di valori dei punti di interpolazione Point[i].x e Point[i].y con l'indice "i" da 1 a 50.

La figura seguente illustra un tratto poligonale con quattro punti di interpolazione:



Richiamo

In un OB Polyline viene richiamato come DB di istanza singola; in un FC Polyline può essere richiamato come DB di istanza singola e come DB di istanza di un parametro; in un FB Polyline può essere richiamato sia come DB di istanza singola, sia come DB di multiistanza o come DB di istanza di un parametro.

Al richiamo dell'istruzione non viene creato nessun oggetto tecnologico. Nella finestra di ispezione dell'editor di programma è possibile configurare le polilinee è disponibile per le singole istanze.

Avviamento

Le variabili nelle strutture UserData e WorkingData non sono a ritenzione. Queste variabili vengono inizializzate con i valori di avvio dopo ogni commutazione dello stato di funzionamento della CPU da STOP a RUN.

Se si modificano online i valori attuali nella struttura UserData e si desidera che questi valori vengano conservati dopo la commutazione dello stato di funzionamento della CPU da STOP a RUN, salvare i valori nei valori di avvio del blocco dati.

Al primo richiamo dell'istruzione Polyline, dopo la commutazione dello stato di funzionamento della CPU da STOP a RUN, la validità dei dati nella struttura UserData viene controllata automaticamente. Se il controllo si conclude con esito positivo, i dati vengono inoltrati alla struttura WorkingData.

Comportamento in caso di errori

L'istruzione Polyline individua svariati errori che possono verificarsi durante il calcolo dell'interpolazione. Il risultato del calcolo dell'interpolazione può essere emesso sull'uscita nonostante la presenza di un errore. Se un errore impedisce il calcolo corretto del risultato dell'interpolazione, sull'uscita viene emesso un valore di uscita sostitutivo.

Qualora un errore impedisse il calcolo corretto del risultato dell'interpolazione, nella variabile ErrorMode definire il valore di uscita sostitutivo come indicato nel seguito:

ErrorMode	Output
0	Valore del parametro Input
1	Valore del parametro SubstituteOutput
2	L'ultimo risultato valido del calcolo dell'interpolazione 0.0, in assenza di risultati validi

Inoltre, per tutti i valori della variabile ErrorMode vale quanto segue:

- Se il valore di uscita sostitutivo non è un valore REAL valido, come valore di uscita viene emesso 0.0.
- Il valore di uscita sostitutivo viene limitato al campo valori del tipo di dati REAL. Solo in quel momento viene emesso il valore di uscita sostitutivo nel parametro Output.
- La variabile ErrorMode è effettiva solo se è impostato il parametro Reset = FALSE. Se è impostato il parametro Reset = TRUE, il valore del parametro SubstituteOutput viene emesso nel parametro Output.

Il parametro Error indica se attualmente è presente un errore. Se l'errore non è più presente, Error viene impostato su FALSE. Il parametro ErrorBits indica quali errori si sono verificati. ErrorBits è a ritenzione e viene resettato soltanto da un fronte di salita nel parametro Reset o ErrorAck.

10.5.3 Funzionamento Polyline

Dati del tratto poligonale

Per la modifica dei dati del tratto poligonale, editare i valori nella struttura UserData. Dopo il controllo della relativa validità, i valori vengono trasferiti nella struttura WorkingData. Soltanto dopo l'inoltro nella struttura WorkingData, i valori vengono utilizzati per il calcolo dell'interpolazione

I valori vengono controllati e trasferiti se

- Il parametro Validate viene impostato su TRUE, mentre il parametro Reset è impostato su FALSE.
- Polyline dopo la commutazione dello stato di funzionamento di CPU da STOP a RUN viene richiamato per la prima volta, mentre il parametro Reset è impostato su FALSE.

Se Polyline è già stato ad es. richiamato in OB100, un nuovo controllo automatico dei valori nei richiami successivi non ha luogo.

Se i dati del tratto poligonale nella struttura UserData non sono validi, i dati precedenti nella struttura WorkingData rimangono invariati e viene emesso un messaggio di errore. Se il controllo viene eseguito per la prima volta, nella struttura WorkingData non sono disponibili valori validi e viene emesso un messaggio di errore corrispondente. Il parametro Output viene in questo caso definito con il valore di uscita sostitutivo che viene configurato nella variabile ErrorMode.

Il controllo e l'inoltro dei valori della struttura UserData richiede un tempo di elaborazione maggiore da parte della CPU rispetto al calcolo dell'interpolazione. In applicazioni con criticità temporale, la prima esecuzione di Polyline può essere effettuata nell'OB di avvio 100. In questo modo le dispendiose operazioni, una tantum, di controllo e trasferimento dei dati del tratto poligonale, si concludono già prima della suddivisione ciclica del programma utente.

Validità dei dati del tratto poligonale

Per consentire la disponibilità di un tratto poligonale valido per il calcolo dell'interpolazione, durante il controllo nella struttura UserData, i valori devono soddisfare le seguenti condizioni:

- $2 \leq \text{UserData.NumberOfUsedPoints} \leq 50$
- $\text{UserData.Point}[j].x < \text{UserData.Point}[j+1].x$ con l'indice $j = 1..(\text{UserData.NumberOfUsedPoints} - 1)$
- $-3.402823\text{e}+38 \leq \text{UserData.Point}[i].x \leq 3.402823\text{e}+38$ con l'indice $i = 1..\text{UserData.NumberOfUsedPoints}$
- $-3.402823\text{e}+38 \leq \text{UserData.Point}[i].y \leq 3.402823\text{e}+38$ con l'indice $i = 1..\text{UserData.NumberOfUsedPoints}$
- $\text{UserData.Point}[i].x$ and $\text{UserData.Point}[i].y$ sono valori REAL validi ($\neq \text{NaN}$) con l'indice $i = 1..\text{UserData.NumberOfUsedPoints}$

Qualora durante il controllo una o più condizioni non siano soddisfatte, i valori nella struttura UserData non vengono acquisiti dalla struttura WorkingData. Il messaggio di errore corrispondente viene emesso nel parametro ErrorBits ([Pagina 498](#)).

La predefinitone dei valori nella struttura UserData non costituisce una configurazione valida. Per l'impiego delle variabili nel calcolo dell'interpolazione, provvedere all'impostazione delle stesse su valori validi.

NOTA

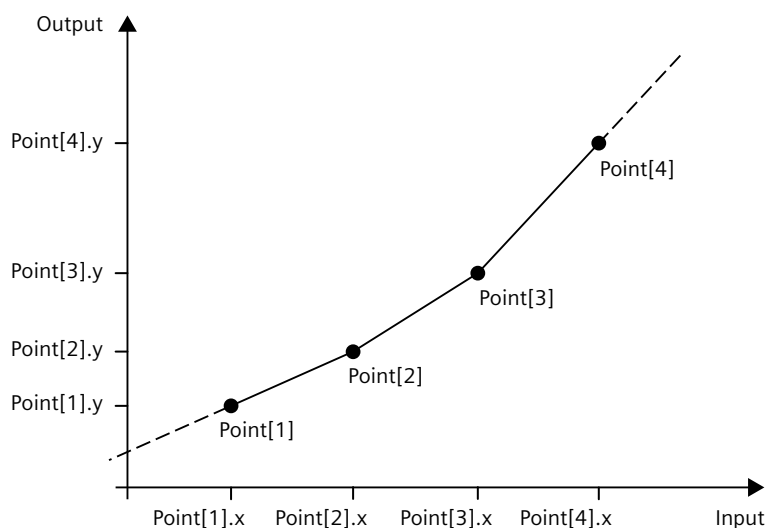
Qualora l'applicazione richieda una quantità superiore al numero massimo di 50 punti di interpolazione, impiegare una o più istanze di Polyline.

Calcolo del valore di uscita

Se il valore di ingresso nel parametro Input si trova al di sotto del primo o al di sopra dell'ultimo valore x dei punti di interpolazione utilizzati, configurare il parametro Output nella variabile OutOfRangeMode con le seguenti impostazioni:

- OutOfRangeMode = 0

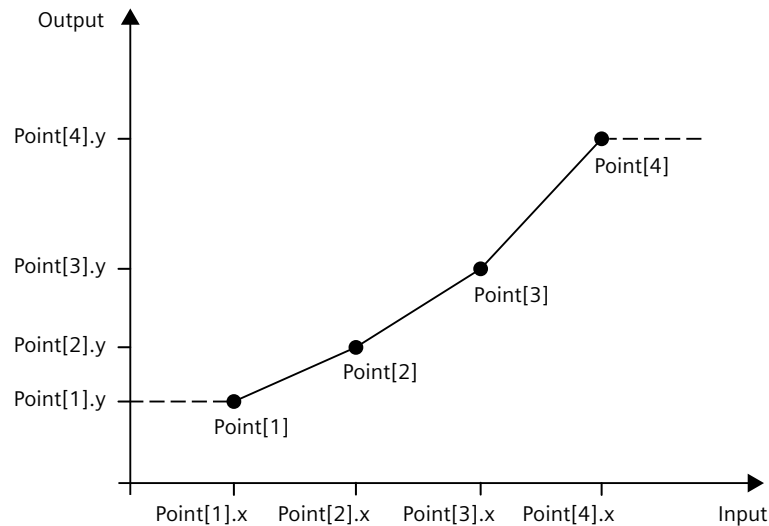
Il valore di uscita viene estrapolato con la pendenza dei primi o degli ultimi due punti di interpolazione.



Se la variabile OutOfRangeMode si trova al di fuori del campo valori consentito da 0 a 1, acquisisce validità l'impostazione di default 0.

- OutOfRangeMode = 1

Il valore di uscita viene limitato sul valore y del primo o dell'ultimo punto di interpolazione.



Il parametro Output ha un campo valori valido appartenente al tipo di dati REAL esteso tra $-3.402823e+38$ e $3.402823e+38$. La validità del valore di uscita nel parametro Output viene controllata ogni volta che viene eseguita l'istruzione Polyline. Se dal calcolo dell'interpolazione risulta un valore REAL non valido, il valore di uscita viene sostituito con le impostazioni nella variabile ErrorMode.

Comportamento di abilitazione EN/ENO

Se una delle seguenti condizioni è soddisfatta, l'uscita di abilitazione ENO viene impostata su FALSE:

- L'ingresso di abilitazione EN è impostato su TRUE e il parametro Output è definito tramite un valore uscita sostitutivo in presenza dei messaggi di errore $\text{ErrorBits} \geq 16\#0001\_0000$.
- Ingresso di abilitazione EN impostato su FALSE.

Diversamente l'uscita di abilitazione ENO viene impostata su TRUE:

Punti di interpolazione attuali

La variabile NextXIndex emette l'indice del prossimo valore x più elevato per il valore di uscita attuale. Ciò consente di rilevare i punti di interpolazione utilizzati nel calcolo attuale dell'interpolazione.

$\text{WorkingData.Point}[\text{NextXIndex}-1].x < \text{Input} \leq \text{WorkingData.Point}[\text{NextXIndex}].x$

Esempio:

- Se il valore del parametro Input si colloca tra $\text{WorkingData.Point}[3].x$ e $\text{WorkingData.Point}[4].x$ la variabile NextXIndex ha il valore 4.
- Se il valore del parametro Input è minore di $\text{WorkingData.Point}[1].x$, la variabile NextXIndex ha il valore 1.
- Se il valore del parametro Input è maggiore di $\text{WorkingData.Point}[\text{WorkingData.NumberOfUsedPoints}].x$ e supera pertanto l'ultimo valore x del tratto poligonale, la variabile NextXIndex ha il valore della variabile $\text{WorkingData.NumberOfUsedPoints} + 1$. Pertanto il valore massimo consentito della variabile NextXIndex è 51.

10.5.4 Parametro di ingresso Polyline

Parametri	Tipo di dati	Preassegnazione	Descrizione
Input	REAL	0.0	Valore di ingresso
SubstituteOutput	REAL	0.0	SubstituteOutput viene impiegato come valore di uscita sostitutivo se • Reset = TRUE oppure • se un errore con messaggio di errore $\text{ErrorBits} \geq 16\#0001\_0000$, impedisce il calcolo corretto del risultato dell'interpolazione, e ErrorMode è configurato sul valore 1.
Validate	BOOL	FALSE	Se Validate è impostato su TRUE, viene effettuato il controllo validità dei dati del tratto poligonale in UserData che vengono poi acquisiti in WorkingData.
ErrorAck	BOOL	FALSE	Cancella i messaggi di errore • Fronte FALSE -> TRUE ErrorBits viene resettato
Reset	BOOL	FALSE	Esegue il riavvio dell'istruzione • Fronte FALSE -> TRUE • ErrorBits viene resettato. • Fintantoché Reset è impostato su TRUE, il valore di uscita sostitutivo SubstituteOutput viene emesso sull'uscita. • Fintantoché Reset è impostato su FALSE, il calcolo dell'interpolazione viene eseguito.

10.5.5 Parametro di uscita Polyline

Parametro	Tipo di dati	Preassegnazione	Descrizione
Output	REAL	0.0	Valore di uscita
Error	BOOL	FALSE	Se Error è impostato su TRUE si è attualmente in presenza di almeno un errore.
ErrorBits	DWORD	DW#16#0	Il parametro ErrorBits (Pagina 498) visualizza i messaggi di errore presenti. ErrorBits è a ritenzione e viene resettato in presenza di un fronte di salita in Reset o ErrorAck.

10.5.6 Variabili statiche Polyline

Variabile	Tipo di dati	Preassegnazione	Descrizione
UserData	AuxFct_PointTable	-	Area di ingresso del tratto poligonale I dati del tratto poligonale nella struttura UserData sono editabili. Eventuali modifiche apportate in questa struttura si ripercuotono sul calcolo dell'interpolazione soltanto dopo che il controllo e la copia dei dati nella struttura WorkingData sono stati avviati.
UserData.NumberOfUsedPoints	INT	0	Numero di punti utilizzati per il calcolo dell'interpolazione Campo di valori ammesso: 2 ... 50
UserData.Point	Array[1..50] of AuxFct_Point	-	Punti per il calcolo dell'interpolazione L'array con 50 elementi del tipo di dati AuxFct_Point contiene le coppie di valori dei punti di interpolazione.
UserData.Point[i]	AuxFct_Point	-	Punto per il calcolo dell'interpolazione Un elemento con l'indice "i" dell'array "Point".
UserData.Point[i].x	REAL	0.0	Valore x del punto di interpolazione Campo di valori ammesso: Point[i].x < Point[i+1].x
UserData.Point[i].y	REAL	0.0	Valore y del punto di interpolazione
WorkingData	AuxFct_PointTable	-	Area di visualizzazione dei dati del tratto poligonale attualmente validi I dati del tratto poligonale nella struttura WorkingData non sono editabili. Questi dati vengono utilizzati per il calcolo dell'interpolazione.
WorkingData.NumberOfUsedPoints	INT	0	Numero di punti utilizzati per il calcolo dell'interpolazione Campo di valori ammesso: 2 ... 50
WorkingData.Point	Array[1..50] of AuxFct_Point	-	Punti per il calcolo dell'interpolazione L'array con 50 elementi di tipo AuxFct_Point contiene le coppie di valori dei punti di interpolazione.
WorkingData.Point[i]	AuxFct_Point	-	Punto per il calcolo dell'interpolazione Un elemento con l'indice "i" dell'array "Point".
WorkingData.Point[i].x	REAL	0.0	Valore x del punto di interpolazione Campo di valori ammesso: Point[i].x < Point[i+1].x

Variabile	Tipo di dati	Preassegnazione	Descrizione
WorkingData.Point[i].y	REAL	0.0	Valore y del punto di interpolazione
ErrorMode	INT	0	Selezione del valore di uscita sostitutivo in caso di errori 0 = Input 1 = SubstituteOutput 2 = ultimo valore di uscita valido Campo di valori ammesso: 0 ... 2
OutOfRangeMode	INT	0	Selezione del valore di uscita se il valore di ingresso si trova al di fuori dei valori x stabiliti. 0 = mantieni pendenza 1 = valore y del primo / ultimo punto di interpolazione Campo di valori ammesso: 0 - 1
NextXIndex	INT	2	Indice del prossimo valore x Consente il monitoraggio dell'indice dei punti di interpolazione utilizzati nel calcolo attuale dell'interpolazione. Ha validità la seguente condizione: $\text{WorkingData.Point}[\text{NextXIndex}-1].x < \text{Input} \leq \text{WorkingData.Point}[\text{NextXIndex}].x$ Non modificare manualmente questo valore.

10.5.7 Parametro ErrorBits

Se si verificano diversi errori contemporaneamente, i valori di ErrorBits vengono visualizzati in un'addizione binaria. Se viene visualizzato ad es. ErrorBits = 16#0000_0003 sono presenti contemporaneamente gli errori 16#0000_0001 e 16#0000_0002.

In caso di Polyline gli errori emessi nel parametro ErrorBits sono suddivisi in due categorie.

- Errore con messaggio di errore ErrorBits < 16#0001_0000
- Errore con messaggio di errore ErrorBits ≥ 16#0001_0000

Errore con messaggio di errore ErrorBits < 16#0001_0000

In presenza di uno o più errori con i messaggi di errore ErrorBits < 16#0001_0000, Polyline ha la seguente reazione:

- Nonostante questo errore, il valore di uscita viene rilevato come indicato nel seguito:
 - Se Reset = FALSE, calcolo interpolazione
 - Se Reset = TRUE, emissione di SubstituteOutput
- Il parametro di uscita Error viene impostato.
- L'uscita di abilitazione ENO non viene modificata.

Il parametro di uscita Error viene eliminato non appena tutti gli errori sono stati eliminati.

ErrorBits (DW#16#...)	Descrizione
0000_0000	Nessun errore
0000_0001	Causa degli errore e reazione agli errori: Il parametro Output è stato limitato a -3.402823e+38 oppure +3.402823e+38. Rimedio: Se sull'uscita viene emesso il valore di interpolazione (Reset = FALSE e ErrorBits < 16#0001_0000), controllare le seguenti variabili utilizzate nel calcolo dell'interpolazione: • Input • WorkingData.Point[i].x • WorkingData.Point[i].y Se ErrorBits ≥ 16#0001_0000 e Reset = FALSE, il valore di uscita sostitutivo viene limitato al momento della relativa emissione. Successivamente, a seconda del valore impostato nella variabile ErrorMode, controllare i seguenti parametri: • Input • SubstituteOutput Se Reset = TRUE, controllare il parametro SubstituteOutput.
0000_0002	Causa dell'errore: Una o più variabili nella struttura UserData presenta valori non validi durante il controllo dei dati del tratto poligonale (Validate = TRUE e Reset = FALSE). Reazione all'errore: I dati del tratto poligonale nella struttura UserData non vengono acquisiti nella struttura WorkingData, pertanto le modifiche apportate nella struttura UserData non diventano attive. FB Polyline prosegue il calcolo dell'interpolazione con i dati del tratto poligonale validi e invariati nella struttura WorkingData. Rimedio: Se il parametro Validate è impostato su TRUE, accertarsi che siano soddisfatte le seguenti condizioni: • $2 \leq \text{UserData.NumberOfUsedPoints} \leq 50$ • $\text{UserData.Point}[j].x < \text{UserData.Point}[j+1].x$ con l'indice $j = 1..(\text{UserData.NumberOfUsedPoints} - 1)$ • $-3.402823e+38 \leq \text{UserData.Point}[i].x \leq 3.402823e+38$ con l'indice $i = 1.. \text{UserData.NumberOfUsedPoints}$ • $-3.402823e+38 \leq \text{UserData.Point}[i].y \leq 3.402823e+38$ con l'indice $i = 1.. \text{UserData.NumberOfUsedPoints}$ • $\text{UserData.Point}[i].x$ e $\text{UserData.Point}[i].y$ sono valori REAL validi ($\neq \text{NaN}$) con l'indice $i = 1.. \text{UserData.NumberOfUsedPoints}$

Errore con messaggio di errore ErrorBits ≥ 16#0001_0000

In presenza di uno o più errori con i messaggi di errore ErrorBits ≥ 16#0001_0000, Polyline ha la seguente reazione:

- Il valore di uscita non può essere rilevato come previsto. Viene invece emesso il valore di uscita sostitutivo.
- Il parametro di uscita Error viene impostato.
- L'uscita di abilitazione ENO viene impostata su FALSE.

Non appena tutti gli errori segnalati con i messaggi ErrorBits ≥ 16#0001_0000 sono stati eliminati, Polyline ha le seguenti reazioni:

- Il valore di uscita viene rilevato come indicato nel seguito:
 - Se Reset = FALSE, calcolo interpolazione
 - Se Reset = TRUE, emissione di SubstituteOutput
- L'uscita di abilitazione ENO viene impostata su TRUE.

Il parametro di uscita Error viene eliminato non appena tutti gli errori sono stati eliminati.

ErrorBits (DW#16#...)	Descrizione		
0001_0000	Causa dell'errore: Il parametro SubstituteOutput o Input, utilizzato come valore di uscita, non ha un valore REAL valido. Reazione all'errore: L'uscita viene impostata su 0.0. Rimedio: Accertarsi che il parametro utilizzato come valore di uscita sia un valore REAL valido (≠ NaN, ad es. 16#7FFF_FFFF). Il parametro utilizzato come valore di uscita è in funzione di Reset e ErrorMode:		
	Reset	ErrorMode	Valore di uscita
	FALSE	0	Input
	FALSE	1	SubstituteOutput
	TRUE	-	SubstituteOutput
0002_0000	Causa dell'errore: Il parametro Input non ha un valore REAL valido, durante il calcolo dell'interpolazione (Reset = FALSE). Reazione all'errore: Nel parametro Output viene emesso il valore di uscita sostitutivo configurato nella variabile ErrorMode. Con ErrorMode = 0, 0.0 viene impiegato come valore di uscita. La variabile NextXIndex non viene aggiornata fintantoché il parametro Input ha un valore REAL valido. Rimedio: Accertarsi che il parametro Input sia un valore REAL valido (≠ NaN, ad es. 16#7FFF_FFFF).		
0004_0000	Causa dell'errore: Il calcolo dell'interpolazione fornisce un valore REAL non valido per il parametro Output. Reazione all'errore: Nel parametro Output viene emesso il valore di uscita sostitutivo configurato nella variabile ErrorMode. Fintantoché è presente quest'errore, la variabile NextXIndex può avere un valore non valido. Rimedio: Controllare la validità dei valori REAL nella struttura WorkingData. Informazione supplementare: Per la modifica dei dati del tratto poligonale, editare innanzitutto la struttura UserData, quindi impostare il parametro Validate = TRUE. Non modificare manualmente i dati nella struttura WorkingData.		

ErrorBits (DW#16#...)	Descrizione
0008_0000	Causa dell'errore: Durante il controllo dei dati del tratto poligonale, una o più variabili nella struttura UserData presenta valori non validi. Reazione all'errore: I dati del del tratto poligonale nella struttura UserData non vengono acquisiti nella struttura WorkingData, pertanto i valori nella struttura UserData non diventano attivi. FB Polyline non emette il valore di interpolazione nel parametro Output perché la struttura WorkingData non contiene dati del tratto poligonale validi. Nel parametro Output viene emesso il valore di uscita sostitutivo configurato nella variabile ErrorMode. Rimedio: Durante il controllo dei dati del tratto poligonale, accertarsi che siano soddisfatte le seguenti condizioni: • $2 \leq \text{UserData.NumberOfUsedPoints} \leq 50$ • $\text{UserData.Point}[j].x < \text{UserData.Point}[j+1].x$ con l'indice $j = 1..(\text{UserData.NumberOfUsedPoints} - 1)$ • $-3.402823e+38 \leq \text{UserData.Point}[i].x \leq 3.402823e+38$ con l'indice $i = 1..\text{UserData.NumberOfUsedPoints}$ • $-3.402823e+38 \leq \text{UserData.Point}[i].y \leq 3.402823e+38$ con l'indice $i = 1..\text{UserData.NumberOfUsedPoints}$ • $\text{UserData.Point}[i].x$ e $\text{UserData.Point}[i].y$ sono valori REAL validi ($\neq \text{NaN}$) con l'indice $i = 1..\text{UserData.NumberOfUsedPoints}$ Informazione supplementare: I dati del tratto poligonale nella struttura UserData vengono controllati se • il parametro Validate è impostato su TRUE, mentre il parametro Reset è impostato su FALSE. oppure • Polyline dopo la commutazione dello stato di funzionamento della CPU viene richiamato per la prima volta da STOP a RUN con il parametro Reset = FALSE. Tenere presente che tutte le variabili nelle strutture UserData e WorkingData non sono a ritenzione. Queste variabili vengono inizializzate con i valori di avvio dopo ogni commutazione dello stato di funzionamento della CPU da STOP a RUN.

10.6 SplitRange

10.6.1 Compatibilità con CPU e FW

La tabella seguente mostra quali versioni di SplitRange sono compatibili con le varie CPU :

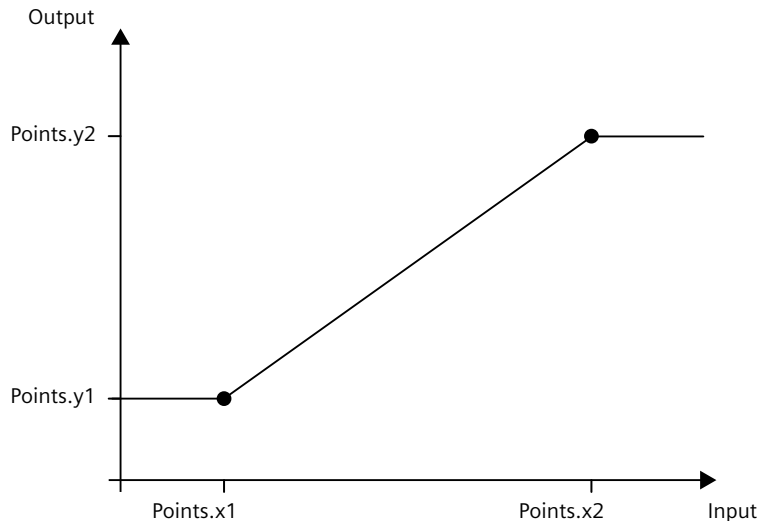
CPU	FW	SplitRange
S7-1200	da V4.2	V1.0
S7-1200 G2	da V1.0	V1.0
CPU basate su S7-1500	da V2.0	V1.0

10.6.2 Descrizione di SplitRange

Descrizione

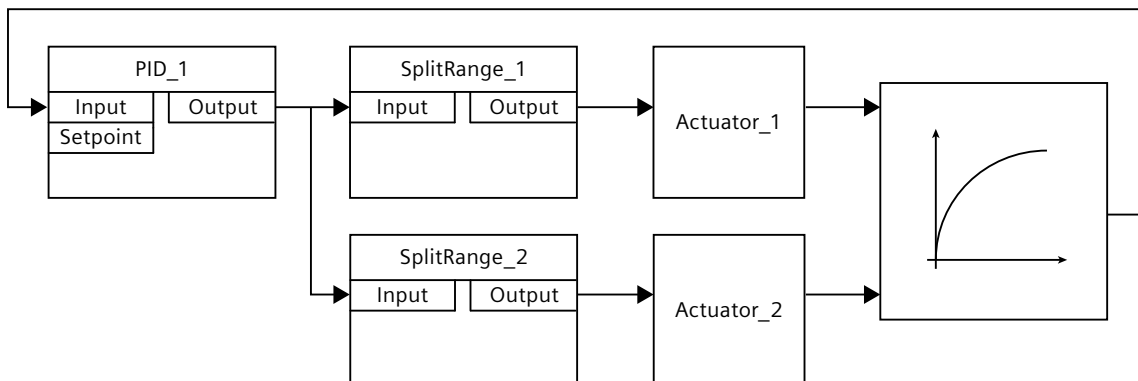
L'istruzione SplitRange converte il valore di ingresso in un valore di uscita. Il valore di ingresso è compreso nel campo di valori limitato da Points.x1 e Points.x2. Il valore di uscita è compreso nel campo di valori limitato da Points.y1 e Points.y2.

La figura seguente mostra la caratteristica di una configurazione di esempio dell'istruzione SplitRange.



Utilizzare SplitRange se è necessaria la regolazione di un processo che viene influenzato da diversi attuatori. SplitRange suddivide il campo dei valori di uscita del regolatore PID in diversi sottocampi. Assegnare un sottocampo a ogni attuatore. Il programma utente richiama il blocco una volta per ogni sottocampo. Il valore di ingresso di ogni istanza SplitRange è collegato al valore di uscita del regolatore PID.

La figura seguente mostra ad es. un circuito di regolazione con due istanze SplitRange e due attuatori.



Validità dei dati di SplitRange

Le coppie di valori nella struttura Points definiscono il campo dei valori di ingresso e di uscita di SplitRange. Entrambe le coppie di valori sono comprese nel campo statico del blocco SplitRange.

SplitRange verifica a ogni richiamo se sono soddisfatte le condizioni seguenti, in modo che per il calcolo del valore di uscita siano disponibili valori validi:

- $\text{Points.x1} < \text{Points.x2}$
- Points.x1 , Points.y1 , Points.x2 e Points.y2 sono compresi nel campo di valori consentito $-3.402823\text{e}+38 \dots 3.402823\text{e}+38$
- Points.x1 , Points.y1 , Points.x2 e Points.y2 sono valori REAL validi ($\neq \text{NaN}$ ad es. $16\#7\text{FFF\_FFFF}$)

Se una o più condizioni non sono soddisfatte non è possibile calcolare correttamente il valore di uscita. Un messaggio di errore corrispondente viene emesso nel parametro ErrorBits.

La preimpostazione dei valori x e y con 0.0 non è una configurazione valida. Per utilizzare le variabili per il calcolo del valore di uscita, impostare le variabili con valori validi.

Comportamento di abilitazione EN/ENO

Se una delle seguenti condizioni è soddisfatta, l'uscita di abilitazione ENO viene impostata su FALSE:

- L'ingresso di abilitazione EN è impostato su TRUE e il parametro Output è definito da un valore di uscita sostitutivo in presenza dei messaggi di errore $\text{ErrorBits} \geq 16\#0001\_0000$.
- Ingresso di abilitazione EN impostato su FALSE.

Diversamente l'uscita di abilitazione ENO viene impostata su TRUE:

Richiamo

In un OB o FC SplitRange viene richiamato come DB di un'istanza singola. In un FB SplitRange può essere richiamata sia come DB di istanza singola, sia come DB di multiistanza e come DB di istanza di parametro.

Al richiamo dell'istruzione non viene creato nessun oggetto tecnologico. Non è disponibile un'interfaccia utente per la parametrizzazione e la messa in servizio. La parametrizzazione di SplitRange avviene direttamente nel DB di istanza e la messa in servizio di SplitRange viene eseguita in una tabella di controllo del programma utente nella CPU oppure in HMI.

Avviamento

Le variabili nel campo statico di SplitRange non sono a ritenzione. Queste variabili vengono inizializzate con i valori di avvio dopo ogni commutazione dello stato di funzionamento della CPU da STOP a RUN.

Se si modificano online i valori attuali nella struttura Points e si vuole che questi valori vengano conservati dopo la commutazione dello stato di funzionamento della CPU da STOP a RUN, salvare questi valori nei valori di avvio del blocco dati.

Comportamento in caso di errori

L'istruzione SplitRange individua svariati errori che possono verificarsi durante il calcolo del valore di uscita. Il risultato del calcolo può essere emesso sull'uscita anche se è presente un errore. Se un errore impedisce il calcolo corretto del valore di uscita, sull'uscita viene emesso un valore di uscita sostitutivo.

Se un errore impedisce di calcolare correttamente il valore di uscita, definire nella variabile ErrorMode il valore di uscita sostitutivo nel modo seguente.

ErrorMode	Output
0	Valore del parametro Input
1	Valore del parametro SubstituteOutput
2	L'ultimo risultato valido del calcolo del valore di uscita 0.0, in assenza di risultati validi

Inoltre, per tutti i valori della variabile ErrorMode vale quanto segue:

- Se il valore di uscita sostitutivo non è un valore REAL valido, come valore di uscita viene emesso 0.0.
- Il valore di uscita sostitutivo viene limitato al campo valori del tipo di dati REAL. Solo in quel momento viene emesso il valore di uscita sostitutivo nel parametro Output.
- La variabile ErrorMode è effettiva solo se è impostato il parametro Reset = FALSE. Se è impostato il parametro Reset = TRUE, il valore del parametro SubstituteOutput viene emesso nel parametro Output.

Il parametro Error indica se attualmente è presente un errore. Se l'errore non è più presente, Error viene impostato su FALSE. Il parametro ErrorBits indica quali errori si sono verificati. ErrorBits è a ritenzione e viene resettato soltanto da un fronte di salita nel parametro Reset o ErrorAck.

10.6.3 Parametro di ingresso SplitRange

Parametri	Tipo di dati	Preassegnazione	Descrizione
Input	REAL	0.0	Valore di ingresso
SubstituteOutput	REAL	0.0	SubstituteOutput viene impiegato come valore di uscita sostitutivo se • Reset = TRUE oppure • se un errore con messaggio di errore ErrorBits $\geq$ 16#0001_0000 impedisce il calcolo corretto del valore di uscita e ErrorMode è configurato sul valore 1.
ErrorAck	BOOL	FALSE	Cancella i messaggi di errore • Fronte FALSE -> TRUE ErrorBits viene resettato
Reset	BOOL	FALSE	Esegue il riavvio dell'istruzione • Fronte FALSE -> TRUE ErrorBits viene resettato. • Finché Reset è impostato su TRUE, il valore di uscita sostitutivo SubstituteOutput viene emesso sull'uscita. • Finché Reset è impostato su FALSE, il calcolo del valore di uscita viene eseguito.

10.6.4 Parametro di uscita SplitRange

Parametro	Tipo di dati	Preassegnazione	Descrizione
Output	REAL	0.0	Valore di uscita
ErrorBits	DWORD	DW#16#0	Il parametro ErrorBits (Pagina 506) visualizza i messaggi di errore presenti. ErrorBits è a ritenzione e viene resettato in presenza di un fronte di salita in Reset o ErrorAck.
Error	BOOL	FALSE	Se Error è impostato su TRUE si è attualmente in presenza di almeno un errore.

10.6.5 Variabili statiche SplitRange

Variabile	Tipo di dati	Preassegnazione	Descrizione
Points	AuxFct_SplitRange_Points	-	Dati sui punti di intersezione
Points.x1	REAL	0.0	Valore x del punto di interpolazione 1 Campo di valori ammesso: Points.x1 < Points.x2
Points.y1	REAL	0.0	Valore y del punto di interpolazione 1
Points.x2	REAL	0.0	Valore x del punto di interpolazione 2 Campo di valori ammesso: Points.x1 < Points.x2
Points.y2	REAL	0.0	Valore y del punto di interpolazione 2
ErrorMode	INT	0	Selezione del valore di uscita sostitutivo in caso di errori 0 = Input 1 = SubstituteOutput 2 = ultimo valore di uscita valido Campo di valori ammesso: 0 ... 2

10.6.6 Parametro ErrorBits

Se si verificano diversi errori contemporaneamente, i valori di ErrorBits vengono visualizzati in un'addizione binaria. Se viene visualizzato ad es. ErrorBits = 16#0000_0003 sono presenti contemporaneamente gli errori 16#0000_0001 e 16#0000_0002.

Nel caso di SplitRange gli errori emessi nel parametro ErrorBits sono suddivisi in due categorie.

- Errore con messaggio di errore ErrorBits < 16#0001_0000
- Errore con messaggio di errore ErrorBits ≥ 16#0001_0000

Errore con messaggio di errore ErrorBits < 16#0001_0000

In presenza di uno o più errori con i messaggi di errore ErrorBits < 16#0001_0000, SplitRange ha la seguente reazione:

- Nonostante questo errore, il valore di uscita viene rilevato come indicato nel seguito:
 - Se Reset = FALSE, calcolo del valore di uscita
 - Se Reset = TRUE, emissione di SubstituteOutput
- Il parametro di uscita Error viene impostato.
- L'uscita di abilitazione ENO non viene modificata.

Il parametro di uscita Error viene eliminato non appena tutti gli errori sono stati eliminati.

ErrorBits (DW#16#...)	Descrizione
0000_0000	Nessun errore
0000_0001	Causa degli errore e reazione agli errori: Il parametro Output è stato limitato a -3.402823e+38 oppure +3.402823e+38. Rimedio: Se ErrorBits ≥ 16#0001_0000 e Reset = FALSE, il valore di uscita sostitutivo viene limitato al momento della relativa emissione. Successivamente, a seconda del valore impostato nella variabile ErrorMode, controllare i seguenti parametri: • Input • SubstituteOutput Se Reset = TRUE, controllare il parametro SubstituteOutput.

Errore con messaggio di errore ErrorBits $\geq$ 16#0001_0000

In presenza di uno o più errori con i messaggi di errore ErrorBits $\geq$ 16#0001_0000, SplitRange ha la seguente reazione:

- Il valore di uscita non può essere rilevato come previsto. Viene invece emesso il valore di uscita sostitutivo.
- Il parametro di uscita Error viene impostato.
- L'uscita di abilitazione ENO viene impostata su FALSE.

Non appena tutti gli errori segnalati con i messaggi ErrorBits $\geq$ 16#0001_0000 sono stati eliminati, SplitRange ha le seguenti reazioni:

- Il valore di uscita viene rilevato come indicato nel seguito:
 - Se Reset = FALSE, calcolo del valore di uscita
 - Se Reset = TRUE, emissione di SubstituteOutput
- L'uscita di abilitazione ENO viene impostata su TRUE.

Il parametro di uscita Error viene eliminato non appena tutti gli errori sono stati eliminati.

ErrorBits (DW#16#...)	Descrizione		
0001_0000	Causa dell'errore: Il parametro SubstituteOutput o Input, utilizzato come valore di uscita, non ha un valore REAL valido. Reazione all'errore: L'uscita viene impostata su 0.0. Rimedio: Accertarsi che il parametro utilizzato come valore di uscita sia un valore REAL valido ($\neq$ NaN, ad es. 16#7FFF_FFFF). Il parametro utilizzato come valore di uscita è in funzione di Reset e ErrorMode:		
	Reset	ErrorMode	Valore di uscita
	FALSE	0	Input
	FALSE	1	SubstituteOutput
	TRUE	-	SubstituteOutput
0002_0000	Causa dell'errore: Il parametro Input non ha un valore REAL valido mentre viene eseguito il calcolo del valore di uscita (Reset = FALSE). Reazione all'errore: Nel parametro Output viene emesso il valore di uscita sostitutivo configurato nella variabile ErrorMode. Con ErrorMode = 0, 0.0 viene impiegato come valore di uscita. Rimedio: Accertarsi che il parametro Input sia un valore REAL valido ($\neq$ NaN, ad es. 16#7FFF_FFFF).		
0004_0000	Possibile causa di errore: • Una o più variabili nella struttura Points presenta valori non validi. • Il calcolo del valore di uscita fornisce un valore REAL non valido per il parametro Output. Reazione all'errore: Nel parametro Output viene emesso il valore di uscita sostitutivo configurato nella variabile ErrorMode. Rimedio: Accertarsi che siano soddisfatte le seguenti condizioni: 1. Points.x1 < Points.x2 2. Points.x1, Points.y1, Points.x2 e Points.y2 sono compresi nel campo di valori consentito -3.402823e+38 ... 3.402823e+38 3. Points.x1, Points.y1, Points.x2 e Points.y2 sono valori REAL validi ($\neq$ NaN ad es. 16#7FFF_FFFF) Informazione supplementare:		

ErrorBits (DW#16#...)	Descrizione
	Tenere presente che le variabili nella struttura Points non sono a ritenzione. Queste variabili vengono inizializzate con i valori di avvio dopo ogni commutazione dello stato di funzionamento della CPU da STOP a RUN.

10.7 RampFunction

10.7.1 Compatibilità con CPU e FW

La tabella seguente mostra quali versioni di RampFunction sono compatibili con le varie CPU :

CPU	FW	RampFunction
S7-1200	da V4.2	V1.0
S7-1200 G2	da V1.0	V1.0
CPU basate su S7-1500	da V2.0	V1.0

10.7.2 Descrizione di RampFunction

Descrizione

L'istruzione RampFunction limita la velocità di variazione di un segnale. RampFunction emette un gradino del segnale nell'ingresso come funzione di rampa del valore di uscita.

Utilizzare RampFunction per evitare gradini del segnale ad es. nei casi seguenti:

- Tra sorgente del setpoint e ingresso del setpoint del regolatore, per ottenere un comportamento pilota più graduale senza influenzare la risposta ai disturbi.
- Tra l'uscita del regolatore e l'ingresso dell'attuatore per proteggere l'attuatore, ad es. un motore con riduttore, o il processo.

Per la velocità di variazione si possono impostare i seguenti valori limite:

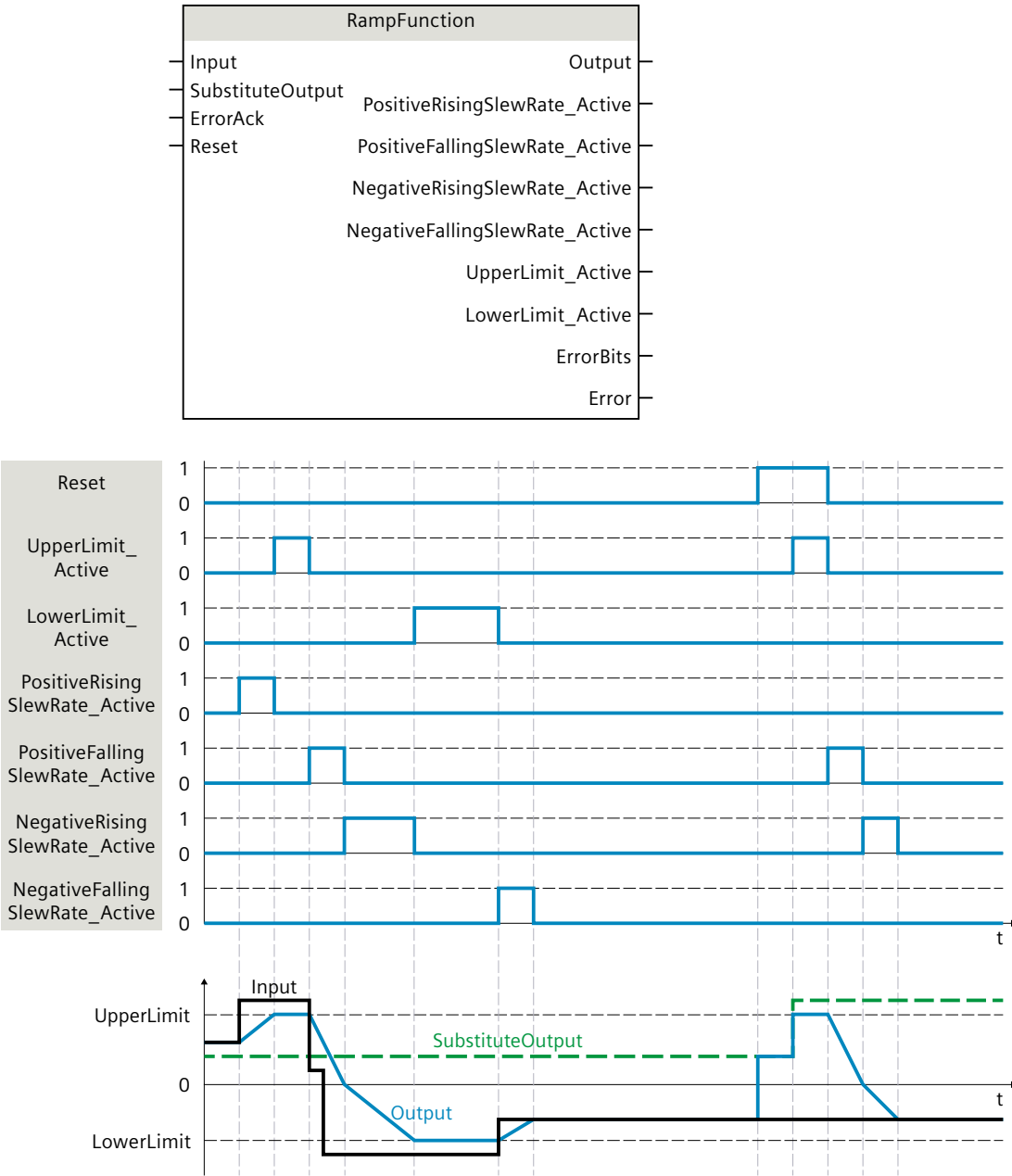
- Velocità di variazione crescente nel campo di valori positivo
- Velocità di variazione decrescente nel campo di valori positivo
- Velocità di variazione crescente nel campo di valori negativo
- Velocità di variazione decrescente nel campo di valori negativo

Inoltre l'istruzione RampFunction limita il valore di uscita al valore limite inferiore e superiore.

Se si raggiunge il limite della velocità di variazione o il limite inferiore o superiore, RampFunction imposta il bit dell'uscita corrispondente a TRUE.

Diagramma funzionale

La figura seguente mostra l'istruzione RampFunction e un esempio di diagramma funzionale.



Richiamo

In un OB o FC RampFunction viene richiamato come DB di un'istanza singola. In un FB RampFunction può essere richiamata sia come DB di istanza singola, sia come DB di multiistanza e come DB di istanza di parametro.

Al richiamo dell'istruzione non viene creato nessun oggetto tecnologico. Non è disponibile un'interfaccia utente per la parametrizzazione e la messa in servizio. La parametrizzazione di RampFunction avviene direttamente nel DB di istanza e la messa in servizio di RampFunction viene eseguita in una tabella di controllo del programma utente nella CPU oppure in HMI.

Avviamento

Le variabili nel campo statico di RampFunction non sono a ritenzione. Queste variabili vengono inizializzate con i valori di avvio dopo ogni commutazione dello stato di funzionamento della CPU da STOP a RUN.

Se si modificano i valori attuali dei valori limite online e si vuole che questi valori vengano conservati dopo la commutazione dello stato di funzionamento della CPU, salvare questi valori nei valori di avvio del blocco dati.

Nella variabile StartMode si definisce il valore di inizializzazione per il parametro Output.

Il valore di inizializzazione viene emesso al primo richiamo di RampFunction dopo la

- Commutazione dello stato di funzionamento della CPU

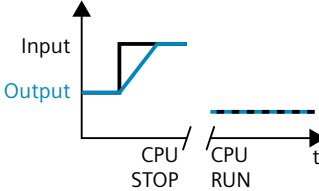
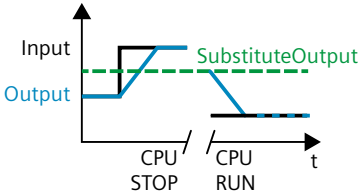
oppure

- Dopo l'esecuzione della funzione "Carica valori di avvio come valori attuali" (solo con l'opzione "Tutti i valori", non con l'opzione "Solo valori di impostazione")

nel parametro Output.

Ai richiami successivi RampFunction calcola il valore di uscita partendo da questo valore di inizializzazione e basandosi sul valore di ingresso e sui valori limite per la velocità di variazione.

La tabella seguente mostra il rapporto di dipendenza tra la variabile StartMode e il parametro Output. I valori nella colonna Output vengono emessi dopo la commutazione dello stato di funzionamento della CPU nel parametro Output.

StartMode	Output	Esempio
0	Valore del parametro Input	
1	Valore del parametro SubstituteOutput	

StartMode	Output	Esempio
2	Rimane invariato. Il parametro Output è a ritenzione.	
3	0.0	
4	Valore della variabile LowerLimit	
5	Valore della variabile UpperLimit	

Inoltre, per tutti i valori della variabile StartMode vale quanto segue:

- Se i valori delle variabili UpperLimit e LowerLimit sono validi, il valore di inizializzazione viene limitato al campo di valori di queste variabili. Solo in quel momento viene emesso il valore di inizializzazione nel parametro Output.
- Se il valore di inizializzazione non è un valore REAL valido, viene emesso il valore di uscita sostitutivo nel parametro Output. Il valore di uscita sostitutivo viene configurato con la variabile ErrorMode. Il valore di uscita sostitutivo è sempre limitato al campo di valori delle variabili UpperLimit e LowerLimit. Se il valore di uscita sostitutivo è a sua volta un valore REAL non valido, 0.0 viene emesso nel parametro Output. Ai richiami successivi l'istruzione calcola il valore di uscita basandosi su questo valore di uscita sostitutivo.
- Soltanto se al primo richiamo dell'istruzione è impostato il parametro Reset = FALSE e al contempo non è presente alcun errore con la segnalazione ErrorBits $\geq$ 16#0002_0000, la variabile StartMode è attiva. Se è impostato il parametro Reset = TRUE, il valore del parametro SubstituteOutput viene emesso nel parametro Output. In presenza di un errore con segnalazione ErrorBits $\geq$ 16#0002_0000, nel parametro Output viene emesso il valore di uscita sostitutivo configurato nella variabile ErrorMode.

Comportamento in caso di errori

L'istruzione RampFunction individua svariati errori che possono verificarsi durante il calcolo del valore di uscita. Il risultato di questo calcolo può essere emesso sull'uscita anche se è presente un errore. Se un errore impedisce il calcolo corretto del valore di uscita, sull'uscita viene emesso un valore di uscita sostitutivo.

Se un errore impedisce di calcolare correttamente il valore di uscita, definire nella variabile ErrorMode il valore di uscita sostitutivo come indicato nel seguito.

La tabella seguente mostra il rapporto di dipendenza tra la variabile ErrorMode e il valore di uscita sostitutivo che RampFunction emette nel parametro Output:

ErrorMode	Output
0	Valore del parametro Input
1	Valore del parametro SubstituteOutput
2	Ultimo valore di uscita valido nel parametro Output
3	0.0
4	Valore della variabile LowerLimit
5	Valore della variabile UpperLimit

Inoltre, per tutti i valori della variabile ErrorMode vale quanto segue:

- Se il valore di uscita sostitutivo non è un valore REAL valido, come valore di uscita viene emesso 0.0.
- Se i valori delle variabili UpperLimit e LowerLimit sono validi, il valore di uscita sostitutivo viene limitato al campo di valori di queste variabili. Solo in quel momento viene emesso il valore di uscita sostitutivo nel parametro Output.
- La variabile ErrorMode è effettiva solo se è impostato il parametro Reset = FALSE. Se è impostato il parametro Reset = TRUE, il valore del parametro SubstituteOutput viene emesso nel parametro Output.
- Se è presente un errore che impedisce di calcolare correttamente il valore di uscita, RampFunction commuta il parametro Output dal valore di uscita calcolato al valore di uscita sostitutivo. A seconda del valore della variabile ErrorMode può verificarsi un gradino del valore di uscita.

Il parametro Error indica se attualmente è presente un errore. Se l'errore non è più presente, Error viene impostato su FALSE. Il parametro ErrorBits indica quali errori si sono verificati. ErrorBits è a ritenzione e viene resettato soltanto da un fronte di salita nel parametro Reset o ErrorAck.

10.7.3 Modo di funzionamento di RampFunction

Limitazione della velocità di variazione

Per la velocità di variazione del segnale di ingresso si possono configurare quattro valori limite. Il valore limite attivo di volta in volta dipende dai fattori seguenti:

- Segno del valore di uscita nel parametro Output
- Direzione di variazione del valore assoluto del valore di uscita nel parametro Output

La tabella seguente mostra le variabili effettive per il valore limite della velocità di variazione in funzione del parametro Output.

Output	Variabile effettiva
Output ≥ 0 e Output di salita	PositiveRisingSlewRate
Output ≥ 0 e Output di discesa	PositiveFallingSlewRate
Output < 0 e Output di salita	NegativeRisingSlewRate
Output < 0 e Output di discesa	NegativeFallingSlewRate

Il valore assoluto dei valori limite della velocità di variazione definisce la variazione max. del valore di uscita al secondo.

Esempio:

Per l'esempio vale la situazione seguente:

- PositiveRisingSlewRate = 10.0
- Tempo di richiamo di RampFunction = 0.1 s
- Input > Output ≥ 0.0

Risultato:

Il valore di uscita Output aumenta di 1.0 a ogni richiamo (10.0 al secondo) fino a raggiungere il valore nel parametro Input.

Per disattivare la limitazione della velocità di variazione di uno o più campi impostare la rispettiva variabile sul valore 3.402823e+38.

Se il valore di uscita Output è attualmente limitato da un valore limite della velocità di variazione, RampFunction imposta il corrispondente bit di uscita a TRUE:

- PositiveRisingSlewRate_Active
- PositiveFallingSlewRate_Active
- NegativeRisingSlewRate_Active
- NegativeFallingSlewRate_Active

Se il parametro Reset è TRUE, i valori limite della velocità di variazione non sono attivi. Perciò i gradini nel parametro SubstituteOutput causano gradini nel parametro Output.

RampFunction verifica a ogni richiamo se sono soddisfatte le condizioni seguenti per le variabili PositiveRisingSlewRate, PositiveFallingSlewRate, NegativeRisingSlewRate e NegativeFallingSlewRate:

- I valori sono compresi nel campo di valori consentito maggiore di 0.0 ... 3.402823e+38
- I valori sono valori REAL validi ($\neq$ NaN ad es. 16#7FFF_FFFF)

Se una o più condizioni non sono soddisfatte viene emesso il valore di uscita sostitutivo nel parametro Output. Il messaggio di errore corrispondente viene emesso nel parametro ErrorBits.

Limitazione del valore di uscita

Il valore di uscita Output è sempre limitato al campo di valori delle variabili UpperLimit e LowerLimit finché queste variabili hanno valori validi.

Se il valore di uscita Output è attualmente limitato a questo campo di valori, RampFunction imposta il corrispondente bit di uscita a TRUE:

- UpperLimit_Active
- LowerLimit_Active

La limitazione del valore di uscita ha una priorità maggiore rispetto alla limitazione della velocità di variazione. Perciò le modifiche delle variabili UpperLimit e LowerLimit causano gradini del valore di uscita Output se ciò è necessario per rispettare i valori limite delle variabili UpperLimit e LowerLimit. La limitazione della velocità di variazione in questo caso non viene considerata.

Esempio:

Se si riduce il valore limite UpperLimit da 100.0 a 80.0 mentre il valore dei parametri Input e Output è 90.0, il valore di uscita Output salta a 80.0. Il valore di uscita Output salta a 80.0 anche se ciò dovesse violare il valore limite configurato per la velocità di variazione.

RampFunction verifica a ogni richiamo se sono soddisfatte le condizioni seguenti:

- LowerLimit < UpperLimit
- LowerLimit e UpperLimit sono compresi nel campo di valori consentito 3.402823e+38 ... 3.402823e+38
- LowerLimit e UpperLimit sono valori REAL validi ($\neq$ NaN ad es. 16#7FFF_FFFF)

Se una o più condizioni non sono soddisfatte viene emesso il valore di uscita sostitutivo nel parametro Output. Il messaggio di errore corrispondente viene emesso nel parametro ErrorBits.

Inoltre RampFunction verifica a ogni richiamo che il valore di uscita Output abbia il campo di valori consentito per un tipo di dati REAL -3.402823e+38 ... 3.402823e+38. Se dal calcolo del valore di uscita risulta un valore REAL non valido, nel parametro Output viene emesso il valore di uscita sostitutivo. Il valore di uscita sostitutivo si configura nella variabile ErrorMode.

Comportamento di abilitazione EN/ENO

Se una delle seguenti condizioni è soddisfatta, l'uscita di abilitazione ENO viene impostata su FALSE:

- L'ingresso di abilitazione EN è impostato su TRUE e il parametro Output è definito da un valore di uscita sostitutivo in presenza dei messaggi di errore ErrorBits $\geq$ 16#0001_0000.
- Ingresso di abilitazione EN impostato su FALSE.

Diversamente l'uscita di abilitazione ENO viene impostata su TRUE:

Misura automatica del tempo di ciclo

Per calcolare la velocità di variazione del valore di uscita, RampFunction impiega il tempo trascorso dall'ultimo richiamo di RampFunction.

Per default il tempo di ciclo viene misurato automaticamente e a partire dal secondo richiamo viene emesso nella variabile CycleTime.Value. RampFunction misura il tempo di ciclo a ogni richiamo dell'istruzione e perciò non è utilizzabile in cicli di richiamo equidistanti, ad es. nell'OB1.

Osservare che i richiami condizionati dell'istruzione, i punti di arresto o il caricamento di istantanee come valori attuali, protraggono il tempo di ciclo nella misura automatica.

Se la misura del tempo di ciclo non fornisce un risultato valido, RampFunction calcola il valore di uscita attuale con l'ultimo tempo di ciclo valido. Inoltre RampFunction emette un messaggio di errore nel parametro ErrorBits.

Se si disattiva la misura automatica del tempo di ciclo impostando la variabile CycleTime.EnableMeasurement = FALSE, si deve predefinire il tempo di ciclo manualmente nella variabile CycleTime.Value. RampFunction verifica la validità della variabile CycleTime.Value a ogni richiamo.

Misura automatica del tempo di ciclo con punti di arresto

Se tra due richiami di RampFunction sono attivi dei punti di arresto, la misura automatica del tempo di ciclo fornisce il tempo effettivamente trascorso tra due richiami. Se è attivo un punto di arresto, la CPU è in stato di funzionamento ALT.

NOTA

I punti di arresto attivi prolungano l'intervallo tra due richiami di RampFunction.

Con un intervallo di tempo più lungo tra due richiami aumenta anche la variazione max. consentita del valore di uscita nel parametro Output.

Esempio:

Per l'esempio vale la situazione seguente:

- PositiveRisingSlewRate = 10.0
- Tempo di richiamo di RampFunction = 0.1 s
- Input > Output ≥ 0.0

Risultato senza punti di arresto:

Il valore di uscita Output aumenta di 1.0 a ogni richiamo fino a raggiungere il valore nel parametro Input.

Risultato con un punto di arresto attivo per dieci secondi:

Al richiamo successivo il valore di uscita Output aumenta di 100.0.

Se non si necessita del calcolo del valore di uscita basato sul tempo effettivo con punti di arresto, eseguire i passi seguenti:

- Disattivare la misura automatica del tempo di ciclo impostando la variabile CycleTime.EnableMeasurement = FALSE.
- Immettere il tempo di ciclo manualmente nella variabile CycleTime.Value.

10.7.4 Parametro di ingresso RampFunction

Parametri	Tipo di dati	Preassegnazione	Descrizione
Input	REAL	0.0	Valore di ingresso
SubstituteOutput	REAL	0.0	SubstituteOutput viene impiegato come valore di uscita sostitutivo se - Reset = TRUE oppure - se un errore con messaggio di errore ErrorBits $\geq 16\#0001\_0000$ impedisce il calcolo corretto del valore di uscita e ErrorMode è configurato sul valore 1.
ErrorAck	BOOL	FALSE	Cancella i messaggi di errore Fronte FALSE -> TRUE ErrorBits viene resettato
Reset	BOOL	FALSE	Esegue il riavvio dell'istruzione - Fronte FALSE -> TRUE ErrorBits viene resettato. - Fintantoché Reset è impostato su TRUE, il valore di uscita sostitutivo SubstituteOutput viene emesso sull'uscita. - Finché Reset è impostato su FALSE, il calcolo del valore di uscita viene eseguito.

10.7.5 Parametro di uscita RampFunction

Parametro	Tipo di dati	Preassegnazione	Descrizione
Output	REAL	0.0	Valore di uscita
PositiveRisingSlewRate_Active	BOOL	FALSE	Se PositiveRisingSlewRate_Active = TRUE, il valore di uscita è attualmente limitato da PositiveRisingSlewRate.
PositiveFallingSlewRate_Active	BOOL	FALSE	Se PositiveFallingSlewRate_Active = TRUE, il valore di uscita è attualmente limitato da PositiveFallingSlewRate.
NegativeRisingSlewRate_Active	BOOL	FALSE	Se NegativeRisingSlewRate_Active = TRUE, il valore di uscita è attualmente limitato da NegativeRisingSlewRate.
NegativeFallingSlewRate_Active	BOOL	FALSE	Se NegativeFallingSlewRate_Active = TRUE, il valore di uscita è attualmente limitato da NegativeFallingSlewRate.
UpperLimit_Active	BOOL	FALSE	Se UpperLimit_Active = TRUE, il valore di uscita è attualmente limitato da UpperLimit.
LowerLimit_Active	BOOL	FALSE	Se LowerLimit_Active = TRUE, il valore di uscita è attualmente limitato da LowerLimit.
ErrorBits	DWORD	DW#16#0	Il parametro ErrorBits (Pagina 518) visualizza i messaggi di errore presenti. ErrorBits è a ritenzione e viene resettato in presenza di un fronte di salita in Reset o ErrorAck.
Error	BOOL	FALSE	Se Error = TRUE è attualmente presente almeno un errore.

10.7.6 Variabili statiche RampFunction

Variabile	Tipo di dati	Preassegnazione	Descrizione
PositiveRisingSlewRate	REAL	10.0	Valore limite per la velocità di variazione del valore di uscita al secondo nel campo positivo in caso di valore assoluto crescente Con PositiveRisingSlewRate = 3.402823e+38 questa limitazione della velocità di variazione è disattivata. Campo di valori ammesso: > 0.0
PositiveFallingSlewRate	REAL	10.0	Valore limite per la velocità di variazione del valore di uscita al secondo nel campo positivo in caso di valore assoluto decrescente Con PositiveFallingSlewRate = 3.402823e+38 questa limitazione della velocità di variazione è disattivata. Campo di valori ammesso: > 0.0
NegativeRisingSlewRate	REAL	10.0	Valore limite per la velocità di variazione del valore di uscita al secondo nel campo negativo in caso di valore assoluto crescente Con NegativeRisingSlewRate = 3.402823e+38 questa limitazione della velocità di variazione è disattivata Campo di valori ammesso: > 0.0
NegativeFallingSlewRate	REAL	10.0	Valore limite per la velocità di variazione del valore di uscita al secondo nel campo negativo in caso di valore assoluto decrescente Con NegativeFallingSlewRate = 3.402823e+38 questa limitazione della velocità di variazione è disattivata. Campo di valori ammesso: > 0.0
UpperLimit	REAL	100.0	Limite superiore del valore di uscita Campo di valori ammesso: > LowerLimit
LowerLimit	REAL	0.0	Limite inferiore del valore di uscita Campo di valori ammesso: < UpperLimit
ErrorMode	INT	2	Selezione del valore di uscita sostitutivo in caso di errori • 0 = Input • 1 = SubstituteOutput • 2 = ultimo valore di uscita valido • 3 = 0.0 • 4 = LowerLimit • 5 = UpperLimit Campo di valori ammesso: 0 ... 5
StartMode	INT	2	Selezione del valore di uscita per il primo richiamo dell'istruzione • 0 = Input • 1 = SubstituteOutput • 2 = ultimo valore di uscita • 3 = 0.0 • 4 = LowerLimit • 5 = UpperLimit Campo di valori ammesso: 0 ... 5

Variabile	Tipo di dati	Preassegnazione	Descrizione
CycleTime	AuxFct_CycleTime	-	Dati del tempo di ciclo
CycleTime.Value	REAL	0.1	Intervallo tra due richiami dell'istruzione in secondi Campo di valori ammesso: > 0.0
CycleTime.EnableMeasurement	BOOL	TRUE	Misura automatica del tempo di ciclo - FALSE = disattivato - TRUE = attivato

10.7.7 Parametro ErrorBits

Se si verificano diversi errori contemporaneamente, i valori di ErrorBits vengono visualizzati in un'addizione binaria. Se viene visualizzato ad es. ErrorBits = 16#0000_0003 sono presenti contemporaneamente gli errori 16#0000_0001 e 16#0000_0002.

Nel caso di RampFunction gli errori emessi nel parametro ErrorBits sono suddivisi in due categorie:

- Errore con messaggio di errore ErrorBits < 16#0001_0000
- Errore con messaggio di errore ErrorBits ≥ 16#0001_0000

Errore con messaggio di errore ErrorBits < 16#0001_0000

In presenza di uno o più errori con i messaggi di errore ErrorBits < 16#0001_0000, RampFunction ha la seguente reazione:

- Nonostante questo errore, il valore di uscita viene rilevato come indicato nel seguito:
 - Se Reset = FALSE, calcolo del valore di uscita
 - Se Reset = TRUE, emissione di SubstituteOutput
- Il parametro di uscita Error viene impostato.
- L'uscita di abilitazione ENO non viene modificata.

Il parametro di uscita Error viene eliminato non appena tutti gli errori sono stati eliminati.

ErrorBits (DW#16#...)	Descrizione
0000_0000	Nessun errore
0000_0002	Causa dell'errore: La misura del tempo di ciclo fornisce un valore non valido mentre viene eseguito il calcolo del valore di uscita (Reset = FALSE). Reazione all'errore: Se è già stato misurato un valore valido per il tempo di ciclo, RampFunction calcola il valore di uscita a partire da quest'ultimo valore della variabile CycleTime.Value. Se per il tempo di ciclo non è ancora stato misurato un valore valido, RampFunction continua ad emettere il valore di uscita configurato con la variabile StartMode nel parametro Output.

Errore con messaggio di errore ErrorBits $\geq$ 16#0001_0000

In presenza di uno o più errori con i messaggi di errore ErrorBits $\geq$ 16#0001_0000, RampFunction ha la seguente reazione:

- Il valore di uscita non può essere rilevato come previsto. Viene invece emesso il valore di uscita sostitutivo.
- Il parametro di uscita Error viene impostato.
- L'uscita di abilitazione ENO viene impostata su FALSE.
- La limitazione del valore di uscita rimane effettiva finché le variabili LowerLimit e UpperLimit hanno valori validi.
- La limitazione della velocità di variazione non è più effettiva. In una delle seguenti situazioni si possono verificare gradini nel valore di uscita:
 - Quando viene rilevato l'errore RampFunction commuta dal valore di uscita calcolato al valore di uscita sostitutivo. Il verificarsi o meno di un salto dipende dal valore della variabile ErrorMode.
 - Il valore di uscita sostitutivo viene modificato mentre è effettivo.

Non appena tutti gli errori segnalati con i messaggi ErrorBits $\geq$ 16#0001_0000 sono stati eliminati, RampFunction ha le seguenti reazioni:

- Il valore di uscita viene rilevato come indicato nel seguito:
 - Se Reset = FALSE, calcolo del valore di uscita
 - Se Reset = TRUE, emissione di SubstituteOutput
- L'uscita di abilitazione ENO viene impostata su TRUE.

Il parametro di uscita Error viene eliminato non appena tutti gli errori sono stati eliminati.

ErrorBits (DW#16#...)	Descrizione		
0001_0000	Causa dell'errore: Il parametro SubstituteOutput o un'altra variabile utilizzata come valore di uscita non ha un valore REAL valido.		
	Reazione all'errore: L'uscita viene impostata su 0.0 e limitata dalle variabili LowerLimit e UpperLimit.		
	Rimedio: Accertarsi che la variabile utilizzata come valore di uscita sia un valore REAL valido ($\neq$ NaN ad es. 16#7FFF_FFFF). La variabile utilizzata come valore di uscita è in funzione di Reset e ErrorMode:		
	Reset	ErrorMode	Valore di uscita
	FALSE	0	Input
	FALSE	1	SubstituteOutput
	FALSE	4	LowerLimit
FALSE	FALSE	5	UpperLimit
	TRUE	-	SubstituteOutput

ErrorBits (DW#16#...)	Descrizione
0002_0000	Causa dell'errore: Il parametro Input non ha un valore REAL valido mentre viene eseguito il calcolo del valore di uscita (Reset = FALSE). Reazione all'errore: Nel parametro Output viene emesso il valore di uscita sostitutivo configurato nella variabile ErrorMode e limitato dalle variabili UpperLimit e LowerLimit. Con ErrorMode = 0, 0.0 viene impiegato come valore di uscita. Rimedio: Accertarsi che il parametro Input sia un valore REAL valido ($\neq$ NaN, ad es. 16#7FFF_FFFF).
0004_0000	Causa dell'errore: Il calcolo del valore di uscita fornisce un valore REAL non valido per il parametro Output. Reazione all'errore: Nel parametro Output viene emesso il valore di uscita sostitutivo configurato nella variabile ErrorMode e limitato dalle variabili UpperLimit e LowerLimit. Rimedio: Controllare tutte le variabili comprese nel calcolo del valore di uscita: • Input • PositiveRisingSlewRate • PositiveFallingSlewRate • NegativeRisingSlewRate • NegativeFallingSlewRate • CycleTime.Value Queste variabili hanno i seguenti valori validi. Con questa combinazione delle variabili, il calcolo del valore di uscita non riesce.
0008_0000	Causa dell'errore: La variabile LowerLimit o UpperLimit non ha un valore valido. Reazione all'errore: Nel parametro Output viene emesso il valore seguente, a seconda del parametro Reset: • Reset = FALSE Nel parametro Output viene emesso il valore di uscita sostitutivo configurato nella variabile ErrorMode. • Reset = TRUE Nel parametro Output viene emesso il valore del parametro SubstituteOutput. In entrambi i casi il parametro Output viene limitato al campo di valori $-3.402823e+38 \dots 3.402823e+38$ del tipo di dati REAL. Rimedio: Accertarsi che siano soddisfatte le seguenti condizioni: 1. LowerLimit < UpperLimit 2. LowerLimit e UpperLimit sono compresi nel campo di valori consentito $-3.402823e+38 \dots 3.402823e+38$ 3. LowerLimit e UpperLimit sono valori REAL validi ($\neq$ NaN ad es. 16#7FFF_FFFF)
0010_0000	Causa dell'errore: Almeno una delle seguenti variabili presenta valori non validi mentre viene eseguito il calcolo del valore di uscita (Reset = FALSE): 1. PositiveRisingSlewRate 2. PositiveFallingSlewRate 3. NegativeRisingSlewRate 4. NegativeFallingSlewRate Reazione all'errore: Nel parametro Output viene emesso il valore di uscita sostitutivo configurato nella variabile ErrorMode e limitato dalle variabili UpperLimit e LowerLimit. Rimedio:

ErrorBits (DW#16#...)	Descrizione
	Accertarsi che siano soddisfatte le seguenti condizioni per tutte e quattro le variabili sopra citate: • I valori sono compresi nel campo di valori consentito maggiore 0.0 ... 3.402823e+38 • I valori sono valori REAL validi ($\neq$ NaN ad es. 16#7FFF_FFFF)
0020_0000	Causa dell'errore: La variabile (configurata con StartMode) per l'inizializzazione del parametro Output al primo richiamo dell'istruzione non ha un valore REAL valido. Reazione all'errore: Al primo richiamo dell'istruzione nel parametro Output viene emesso il valore di uscita sostitutivo configurato nella variabile ErrorMode e limitato dalle variabili LowerLimit e UpperLimit. Ai richiami successivi RampFunction calcola il valore di uscita basandosi su questo valore di uscita sostitutivo. Rimedio: Accertarsi che la variabile utilizzata per l'inizializzazione del parametro Output sia un valore REAL valido ($\neq$ NaN, ad es. 16#7FFF_FFFF). Se è impostato Reset = FALSE, l'inizializzazione diventa effettiva al primo richiamo dell'istruzione dopo la commutazione dello stato di funzionamento della CPU da STOP a RUN. La variabile utilizzata per l'inizializzazione del parametro Output è in funzione di StartMode: • StartMode = 1: Substitute Output • StartMode = 2: Output
0040_0000	Causa dell'errore: La variabile CycleTime.Value non ha un valore valido mentre viene eseguito il calcolo del valore di uscita (Reset = FALSE). Reazione all'errore: Nel parametro Output viene emesso il valore di uscita sostitutivo configurato nella variabile ErrorMode e limitato dalle variabili UpperLimit e LowerLimit. Rimedio: Accertarsi che siano soddisfatte le seguenti condizioni: • $0.0 < \text{CycleTime.Value} \leq 3.402823e+38$ • CycleTime.Value è un valore REAL valido ($\neq$ NaN ad es. 16#7FFF_FFFF) Informazione supplementare: Per il calcolo automatico del valore della variabile CycleTime.Value impostare la variabile CycleTime.EnableMeasurement su TRUE.

10.8 RampSoak

10.8.1 Compatibilità con CPU e FW

La tabella seguente indica le CPU compatibili con le varie versioni di RampSoak :

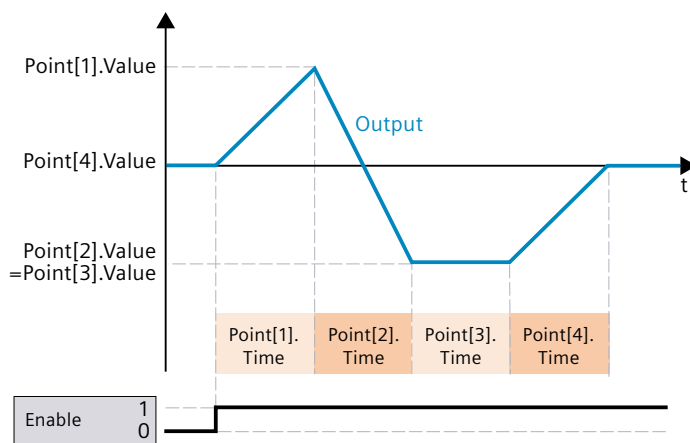
CPU	FW	RampSoak
S7-1200	da V4.2	V1.0
S7-1200 G2	da V1.0	V1.0
CPU basate su S7-1500	da V2.0	V1.0

10.8.2 Descrizione di RampSoak

Descrizione

Con l'istruzione RampSoak si genera un valore di uscita che segue cronologicamente un profilo configurabile. Ogni punto del profilo è provvisto di un valore di destinazione e di un valore temporale. Durante l'esecuzione del profilo, il valore di destinazione del punto attuale viene raggiunto prima che sia trascorso il valore temporale.

La figura seguente illustra un profilo con 4 punti:



L'istruzione RampSoak può essere impiegata ad es. per rendere disponibile un profilo del setpoint per la regolazione di un processo termico.

Calcolo del valore di uscita

Il valore di uscita viene calcolato con un'interpolazione tra il punto attuale e il punto precedente utilizzando la formula seguente:

$$\text{Output}(t) = \text{Point}[i].\text{Value} - \frac{\text{RemainingTime\_Point}}{\text{Point}[i].\text{Time}} (\text{Point}[i].\text{Value} - \text{Point}[i-1].\text{Value})$$

i rappresenta il valore del parametro CurrentPoint.

Sulla base del tempo trascorso viene determinato quali punti del profilo devono essere attualmente inclusi nel calcolo del valore di uscita.

Richiamo

In un OB o FC RampSoak viene richiamato come DB di un'istanza singola. In un FB RampSoak può essere richiamato sia come DB di istanza singola, sia come DB di multiistanza o come DB di istanza di un parametro.

Al richiamo dell'istruzione non viene creato nessun oggetto tecnologico. Non è disponibile un'interfaccia utente per la parametrizzazione e la messa in servizio. La parametrizzazione di RampSoak avviene direttamente nel DB di istanza e la messa in servizio di RampSoak viene eseguita in una tabella di controllo del programma utente nella CPU oppure in HMI.

Avviamento

Le variabili nel campo statico di RampSoak e quindi anche i dati del profilo nelle strutture UserData e WorkingData non sono a ritenzione. Queste variabili vengono inizializzate con i valori di avvio dopo ogni commutazione dello stato di funzionamento della CPU da STOP a RUN.

Se si modificano online i valori attuali nella struttura UserData e si desidera che questi valori vengano conservati dopo la commutazione dello stato di funzionamento della CPU da STOP a RUN, salvare i valori nei valori di avvio del blocco dati.

Al primo richiamo dell'istruzione RampSoak, dopo la commutazione dello stato di funzionamento della CPU da STOP a RUN, la validità dei dati nella struttura UserData viene controllata automaticamente. Se il controllo si conclude con esito positivo, i dati vengono inoltrati alla struttura WorkingData.

La variabile StartMode (Pagina 533) consente di determinare il comportamento all'avvio dell'istruzione RampSoak al primo richiamo dopo la commutazione dello stato di funzionamento della CPU da STOP a RUN.

Comportamento in caso di errori

Se il valore di uscita non può essere calcolato correttamente, l'istruzione RampSoak emette un valore di uscita sostitutivo e genera un errore con il messaggio di errore ErrorBits >= 16#0002_0000. Il valore di uscita sostitutivo può essere determinato con la variabile ErrorMode (Pagina 542) come segue:

ErrorMode	Output
0	Valore della variabile WorkingData.StartValue Assicurarsi che i dati del profilo in UserData siano stati controllati con il parametro Validate e acquisiti in WorkingData. Se i dati del profilo non sono mai stati controllati e acquisiti, si applica il valore predefinito 0.0 di WorkingData.StartValue.
1	Valore del parametro SubstituteOutput
2	Ultimo valore di uscita valido dell'esecuzione del profilo 0.0 in assenza di un valore di uscita valido dell'esecuzione del profilo
3	0.0

Inoltre, per tutti i valori della variabile ErrorMode vale quanto segue:

- Se il valore di uscita sostitutivo non è un valore REAL valido, come valore di uscita viene emesso 0.0.
- Il valore di uscita sostitutivo viene limitato al campo valori $-3.402823e+38$.. $+3.402823e+38$ del tipo di dati REAL. Solo in quel momento viene emesso il valore di uscita sostitutivo nel parametro Output.
- La variabile ErrorMode è effettiva solo se è impostato il parametro Reset = FALSE. Se è impostato il parametro Reset = TRUE, nel parametro Output viene emesso il valore del parametro SubstituteOutput oppure 0.0.

Il parametro Error indica se attualmente è presente un errore. Se l'errore non è più presente, Error viene impostato su FALSE. Il parametro ErrorBits indica quali errori si sono verificati. ErrorBits è a ritenzione e viene resettato soltanto da un fronte di salita nel parametro Reset o ErrorAck.

10.8.3 Modo di funzionamento RampSoak

10.8.3.1 Configurazione e controllo dei dati del profilo

Configurazione dei dati del profilo

Il profilo viene configurato nella struttura statica UserData.

Quest'ultima contiene i seguenti elementi:

- NumberOfUsedPoints

Numero di punti di interpolazione utilizzati per il profilo.

- StartValue

Valore di uscita opzionale durante l'avvio ([Pagina 533](#)), l'arresto ([Pagina 536](#)) o la conclusione dell'esecuzione del profilo.

- Point

L'array con 50 elementi contiene le coppie di valori dei punti di interpolazione Point[i].Value e Point[i].Time:

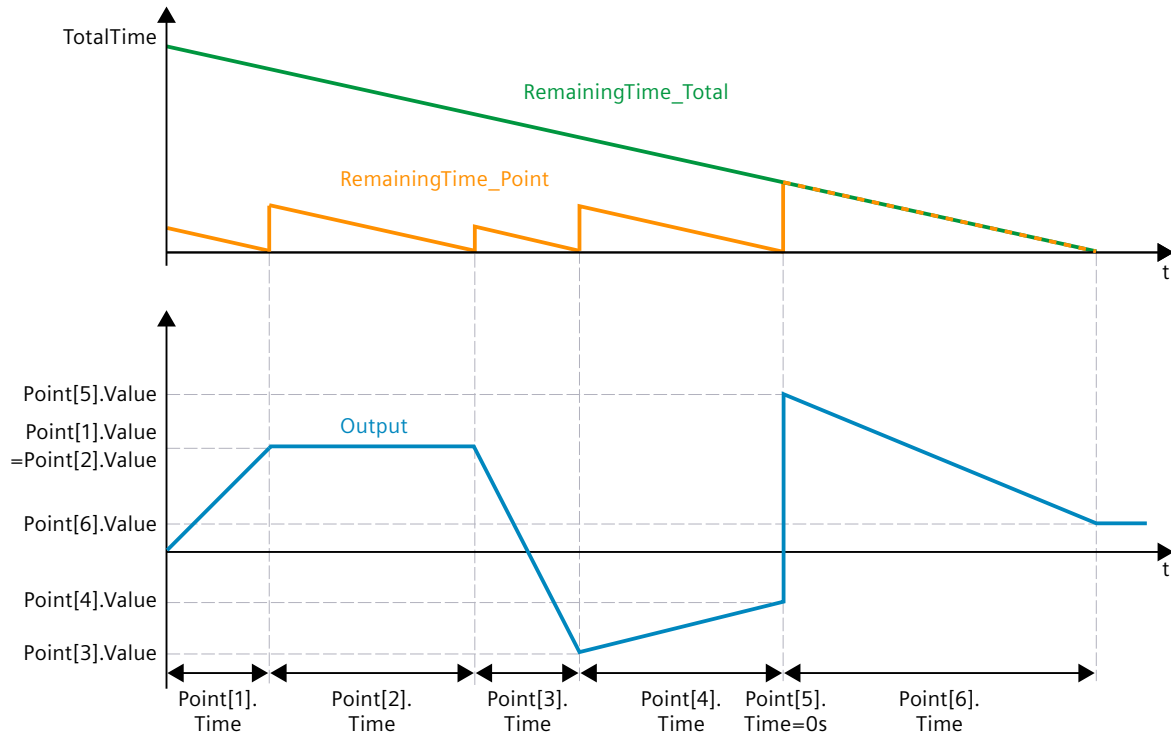
- Point[i].Value

Finché il punto è attivo, il valore di uscita viene modificato gradualmente fino a raggiungere questo valore prima che sia trascorso l'intervallo Point[i].Time.

- Point[i].Time

Questo valore definisce la durata del punto in secondi.

La figura seguente illustra un profilo con 6 punti. Il punto numero 5 ha una durata di 0 secondi, di conseguenza il valore di uscita viene saltato.



Controllo dei dati del profilo

Prima di utilizzare i dati del profilo per l'esecuzione del profilo, questi devono essere controllati nella struttura UserData.

Il controllo viene avviato nei seguenti casi:

- Quando il parametro Validate viene impostato su TRUE.
- Se il parametro Enable, dopo la commutazione dello stato di funzionamento della CPU da STOP a RUN, viene impostato per la prima volta su TRUE senza un controllo preventivo dei dati del profilo.

Nella struttura UserData vengono controllate le seguenti condizioni dei dati del profilo:

- $1 \leq \text{UserData.NumberOfUsedPoints} \leq 50$
- $\text{NextPoint} \leq \text{UserData.NumberOfUsedPoints}$
- Quando è attiva l'esecuzione del profilo (Enable = TRUE):
 - $\text{CurrentPoint} \leq \text{UserData.NumberOfUsedPoints}$
- $-3.402823\text{e}+38 \leq \text{UserData.StartValue} \leq 3.402823\text{e}+38$
- $-3.402823\text{e}+38 \leq \text{UserData.Point}[i].\text{Value} \leq 3.402823\text{e}+38$ con l'indice $i = 1..\text{UserData.NumberOfUsedPoints}$

- $0.0 \leq \text{UserData.Point}[i].\text{Time} \leq 3.402823\text{e}+38$ con l'indice $i = 1..\text{UserData.NumberOfUsedPoints}$
- $0.0 < \text{UserData.Point}[1].\text{Time} + \text{UserData.Point}[2].\text{Time} + \dots + \text{UserData.Point}[\text{UserData.NumberOfUsedPoints}].\text{Time} \leq 3.402823\text{e}+38$

Effetti del controllo dei dati del profilo e possibili messaggi di errore

Se le condizioni vengono soddisfatte, i nuovi dati del profilo vengono trasferiti dalla struttura UserData nella struttura WorkingData e utilizzati per l'esecuzione del profilo. Il parametro TotalTime viene aggiornato. Il parametro RemainingTime_Total viene aggiornato durante l'esecuzione del profilo.

NOTA

Il valore e la durata del punto attuale vengono salvati negli appunti e restano invariati anche dopo la conclusione del controllo dei nuovi dati del profilo fino al termine dell'elaborazione del punto attuale. I nuovi dati del profilo vengono utilizzati a partire dall'inizio del punto successivo.

Quando una delle condizioni non viene soddisfatta, i dati del profilo già presenti nella struttura WorkingData restano invariati. È presente un errore con il messaggio di errore ErrorBits = 16#0000_0004 ([Pagina 542](#)).

Quando il parametro Enable o il parametro Next è impostato su TRUE ma nella struttura WorkingData non sono presenti dati del profilo validi, viene emesso un errore con il messaggio di errore ErrorBits = 16#0008_0000 ([Pagina 542](#)).

NOTA

Se i valori predefiniti nella struttura UserData non vengono modificati prima del controllo, il controllo dei dati del profilo non riesce.

NOTA

I valori offline della struttura WorkingData non può essere modificata. Per modificare i dati del profilo, editare innanzitutto la struttura UserData, quindi impostare il parametro Validate = TRUE.

10.8.3.2 Esecuzione del profilo

Per eseguire un profilo e calcolare il valore di uscita sono necessari dei dati del profilo verificati e un fronte di salita nel parametro Enable.

Avviare e arrestare l'esecuzione del profilo - parametro Enable

L'esecuzione del profilo e il calcolo del valore di uscita vengono avviati con un fronte di salita nel parametro Enable. L'esecuzione del profilo ha inizio nel punto indicato nel parametro di transito NextPoint.

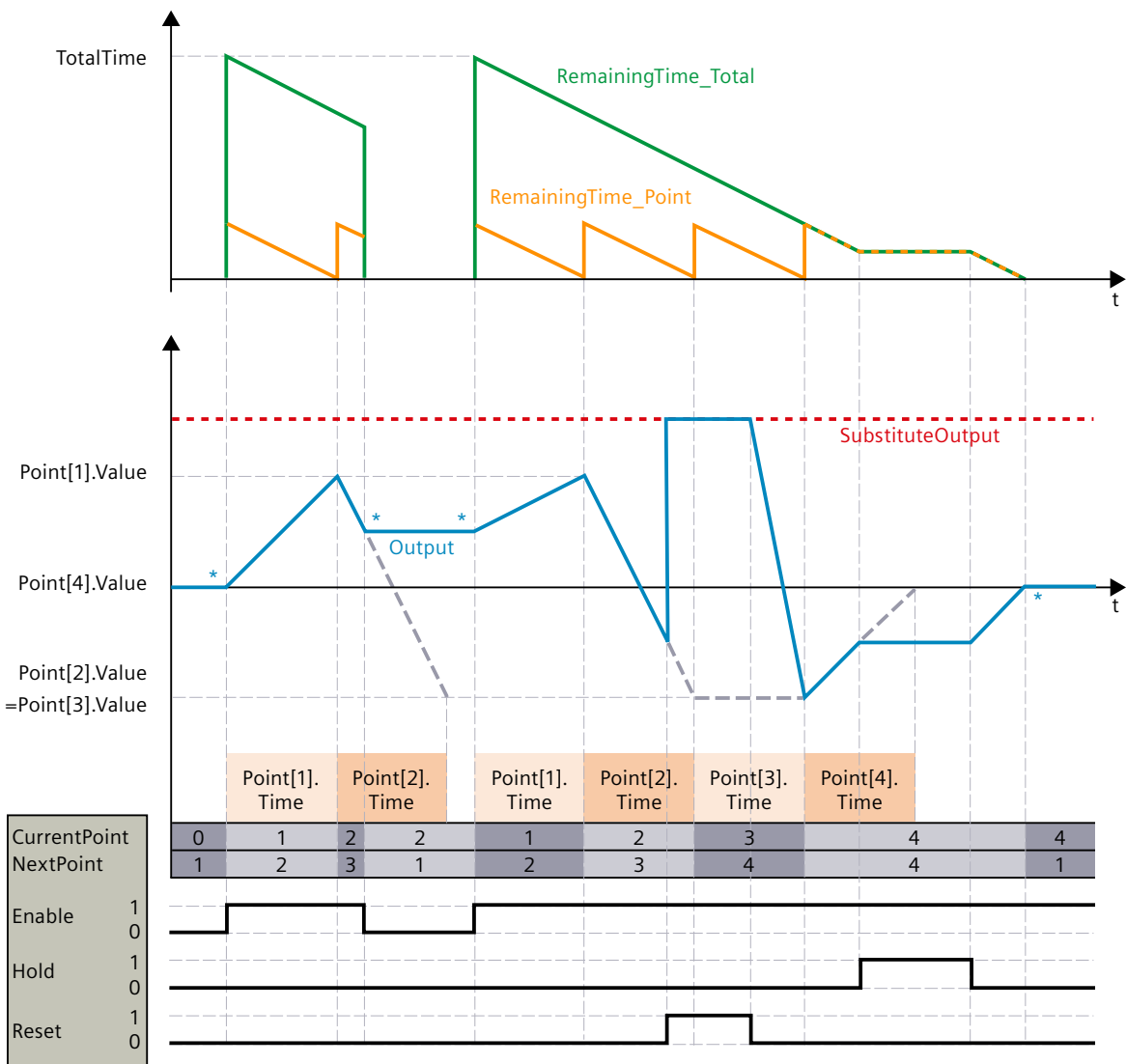
L'esecuzione del profilo si arresta con un fronte di discesa nel parametro Enable.

Quando il profilo viene arrestato o è stato completato il parametro di transito NextPoint viene impostato automaticamente su 1. Pertanto, l'esecuzione del profilo successiva inizia nel primo punto del profilo se il parametro di transito NextPoint non viene modificato.

Il comportamento all'avvio, all'arresto e al termine dell'esecuzione del profilo può essere configurato con le variabili statiche StartMode ([Pagina 533](#)) e StopMode ([Pagina 536](#)).

L'esecuzione del profilo determina solo il valore di uscita nel parametro Output, se Reset = FALSE. Se è impostato RESET = TRUE, l'esecuzione del profilo attiva prosegue sullo sfondo.

La figura seguente mostra in che modi i parametri di ingresso Enable, Reset e Hold determinano l'esecuzione del profilo:



Questo esempio illustra il comportamento all'avvio e all'arresto dell'esecuzione del profilo (contrassegnato con *) con la preassegnazione StartMode = StopMode = 2 (avvio dall'ultimo valore di uscita / mantieni ultimo valore di uscita).

Arrestare l'esecuzione del profilo - parametro Hold

L'esecuzione del profilo attiva si arresta con il parametro Hold = TRUE.

Il parametro Hold = TRUE ha le seguenti ripercussioni:

- I parametri Output, CurrentPoint, NextPoint, RemainingTime_Total e RemainingTime_Point restano invariati.
- Un fronte nei parametri Enable o Next viene ritardato e attivato solo quando Hold = FALSE.
- Se il parametro Reset passa da TRUE a FALSE, il parametro Hold influisce sulle modifiche del parametro Output. Per maggiori informazioni consultare la descrizione del parametro Reset.

Il parametro Hold non influenza l'elaborazione dei parametri Validate e ErrorAck.

Sovrascrivere il valore di uscita - parametro Reset

Il parametro Reset = TRUE consente di resettare l'istruzione e sovrascrivere il valore di uscita nel parametro Output con il valore del parametro SubstituteOutput.

L'esecuzione del profilo attiva prosegue sullo sfondo. È possibile continuare ad utilizzare i parametri Enable, Hold e Next.

Se l'esecuzione del profilo è ancora attiva (Enable=TRUE e il profilo non è ancora stato completato), e il parametro Reset subisce una variazione da TRUE a FALSE, questo comportamento dipende dal parametro Hold:

- Hold = FALSE

Il parametro Output viene modificato gradualmente dal valore SubstituteOutput al valore del punto attuale che raggiunge entro il tempo rimanente del punto attuale (RemainingTime_Point).

- Hold = TRUE

Il parametro Output salta da SubstituteOutput al valore dell'esecuzione del profilo arrestata.

Quando l'esecuzione del profilo non è più attiva (Enable=FALSE o il profilo è stato completato) e il parametro Reset commuta da TRUE a FALSE, il parametro Output salta sull'ultimo valore dell'esecuzione del profilo.

Continuare l'esecuzione del profilo con un punto specifico - parametro Next

L'esecuzione del profilo continua in presenza di un fronte di salita nel parametro Next con Point[NextPoint].

Quando l'esecuzione del profilo è disattivata, il parametro Next = TRUE ha il seguente effetto:

- Il parametro Output acquisisce il valore di Point[NextPoint].Value.
- Il parametro CurrentPoint acquisisce il valore di NextPoint.
- Il parametro NextPoint viene aumentato. Se NextPoint è l'ultimo punto del profilo, il valore di NextPoint viene impostato su 1.

Quando l'esecuzione del profilo è attivata, il parametro Next = TRUE ha il seguente effetto:

- Il valore attuale del parametro Output viene modificato gradualmente fino a raggiungere il valore di Point[NextPoint].Value durante Point[NextPoint].Time.
- Il parametro RemainingTime_Point acquisisce il valore di Point[NextPoint].Time.
- Il parametro RemainingTime_Total viene aggiornato di conseguenza.
- Il parametro CurrentPoint acquisisce il valore di NextPoint.
- Il parametro NextPoint viene aumentato. Se NextPoint è l'ultimo punto del profilo, il nuovo valore dipende da StopMode. Per maggiori informazioni consultare la descrizione del parametro NextPoint.

NOTA

Al primo richiamo dell'istruzione RampSoak, dopo la commutazione dello stato di funzionamento della CPU da STOP a RUN, il comportamento è determinato da StartMode. Il parametro Next è attivo dai richiami successivi.

Un profilo di esempio con il parametro Next si trova alla sezione "Continuare l'esecuzione del profilo con un punto specifico - parametro NextPoint".

Continuare l'esecuzione del profilo con un punto specifico - parametro NextPoint

Il parametro NextPoint determina con quale punto deve continuare l'esecuzione del profilo. Se l'elaborazione del punto attuale è conclusa, l'istruzione RampSoak aggiorna automaticamente il valore del parametro NextPoint al punto successivo del profilo. Modificando il valore del parametro NextPoint è possibile definire se l'esecuzione del profilo deve continuare in un punto diverso.

Il parametro NextPoint può influenzare il comportamento dell'esecuzione del profilo quando è disattivata come descritto di seguito:

- Con un fronte di salita nel parametro Next, il parametro Output acquisisce il valore di Point[NextPoint].Value.
- Quando viene avviata l'esecuzione del profilo, NextPoint rappresenta il primo CurrentPoint. Il valore del parametro Output, definito con StartMode, viene modificato gradualmente fino a raggiungere il valore di Point[NextPoint].Value durante Point[NextPoint].Time.

Il parametro NextPoint può influenzare il comportamento dell'esecuzione del profilo quando è attivata come descritto di seguito:

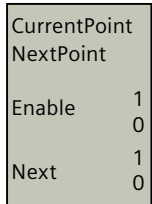
- Senza fronte di salita nel parametro Next:
Dopo la conclusione del punto attuale l'elaborazione prosegue con NextPoint.
- Con fronte di salita nel parametro Next:
L'elaborazione prosegue immediatamente con NextPoint.

Mentre è attivo l'ultimo punto dell'esecuzione del profilo ($\text{CurrentPoint} = \text{WorkingData.NumberOfUsedPoints}$), StopMode determina l'aggiornamento automatico del valore NextPoint come segue:

- StopMode $\neq$ 4
Mentre l'ultimo punto è attivo: $\text{NextPoint} = \text{CurrentPoint} = \text{WorkingData.NumberOfUsedPoints}$.
L'esecuzione del profilo termina dopo l'ultimo punto.
- StopMode = 4
Mentre l'ultimo punto è attivo: $\text{NextPoint} = 1$.
L'esecuzione del profilo viene riavviata dopo l'ultimo punto.

Quando il profilo viene arrestato o è stato completato il parametro di transito NextPoint viene impostato automaticamente su 1. Pertanto, l'esecuzione del profilo successiva inizia nel primo punto del profilo se il parametro di transito NextPoint non viene modificato.

da un'immissione dell'utente, questo comportamento viene contrassegnato con →:



4.

10.8.3.3 Configurazione del comportamento all'avvio - variabile statica StartMode

La variabile StartMode consente di determinare il comportamento dell'istruzione RampSoak nei seguenti casi:

- Alla prima esecuzione dell'istruzione RampSoak dopo la commutazione dello stato di funzionamento della CPU da STOP a RUN (riavvio).
- Viene avviata l'esecuzione del profilo.
- Dopo l'esecuzione della funzione "Carica valori di avvio come valori attuali" (solo con l'opzione "Tutti i valori", non con l'opzione "Solo valori di impostazione")

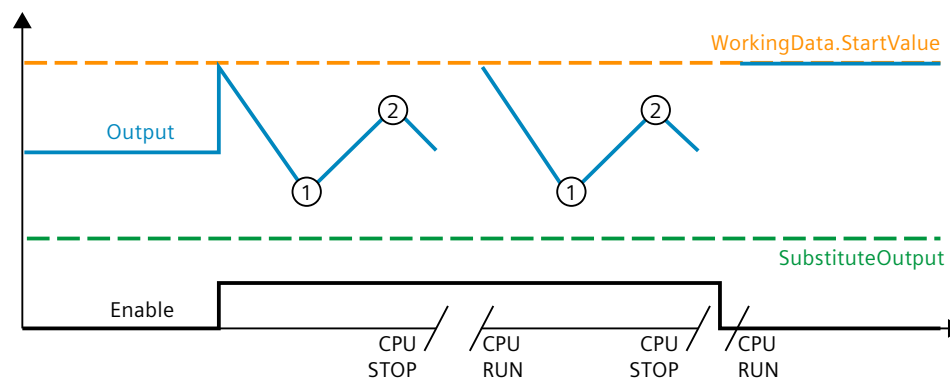
Sono possibili le seguenti impostazioni della variabile StartMode:

- StartMode = 0

Il parametro Output acquisisce il valore di WorkingData.StartValue.

L'esecuzione del profilo inizia con questo valore se dopo un riavvio della CPU è impostato Enable = TRUE o se Enable passa da FALSE a TRUE.

La figura seguente mostra l'avvio dell'esecuzione del profilo e il riavvio della CPU con StartMode = 0:



NOTA

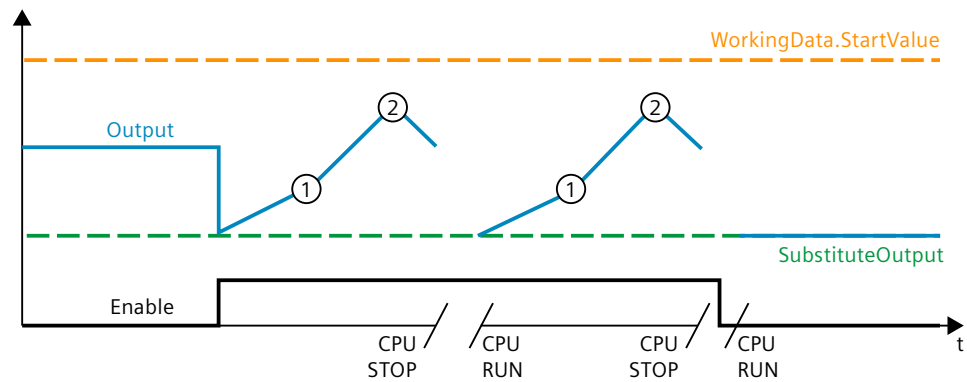
Con questa impostazione, al primo richiamo dell'istruzione dopo il riavvio della CPU è necessario effettuare un controllo dei dati del profilo e acquisirli secondo WorkingData. In caso contrario viene utilizzato il valore predefinito 0.0 di WorkingData.StartValue.

- StartMode = 1

Il parametro Output acquisisce il valore di SubstituteOutput.

L'esecuzione del profilo inizia con questo valore se dopo un riavvio della CPU è impostato Enable = TRUE o se Enable passa da FALSE a TRUE.

La figura seguente mostra l'avvio dell'esecuzione del profilo e il riavvio della CPU con StartMode = 1:

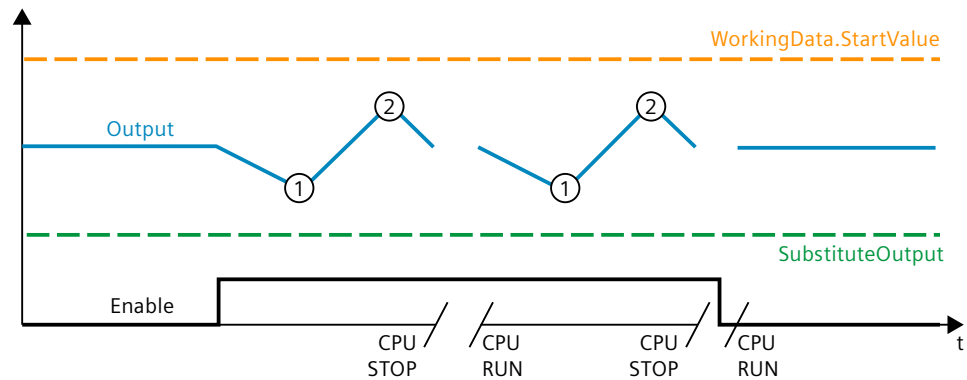


- StartMode = 2

Il parametro Output resta invariato.

L'esecuzione del profilo inizia con il valore non modificato del parametro Output se dopo un riavvio della CPU è impostato Enable = TRUE o se Enable passa da FALSE a TRUE.

La figura seguente mostra l'avvio dell'esecuzione del profilo e il riavvio della CPU con StartMode = 2:

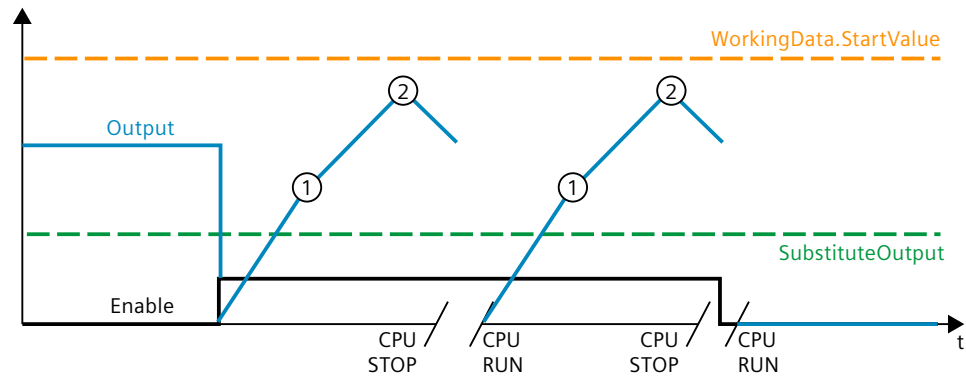


- StartMode = 3

Il parametro Output acquisisce il valore 0.0.

L'esecuzione del profilo inizia con questo valore se dopo un riavvio della CPU è impostato Enable = TRUE o se Enable passa da FALSE a TRUE.

La figura seguente mostra l'avvio dell'esecuzione del profilo e il riavvio della CPU con StartMode = 3:



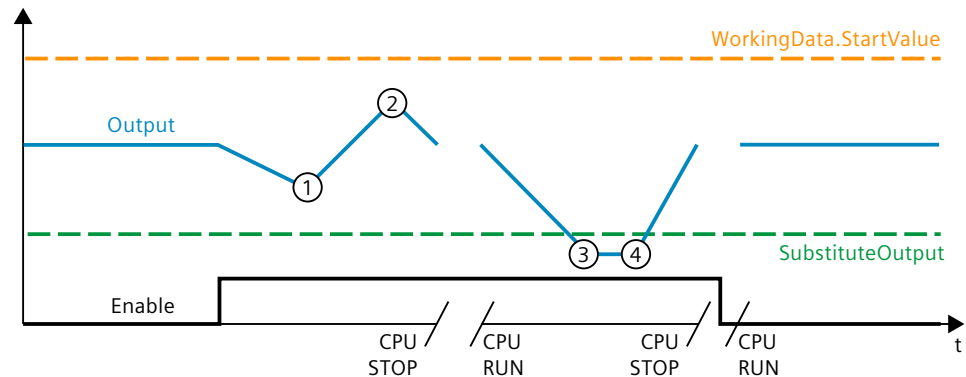
- StartMode = 4

Il parametro Output resta invariato.

L'esecuzione del profilo prosegue se dopo il riavvio della CPU rimane impostato Enable = TRUE e se l'esecuzione del profilo era attiva prima del riavvio.

L'esecuzione del profilo inizia con il valore non modificato del parametro Output se dopo un riavvio della CPU è impostato Enable = TRUE e l'esecuzione del profilo era disattivata prima del riavvio o se Enable passa da FALSE a TRUE.

La figura seguente mostra l'avvio dell'esecuzione del profilo e il riavvio della CPU con StartMode = 4:



Inoltre, per tutti i valori della variabile StartMode vale quanto segue:

- Il parametro Enable, la variabile StartMode e i dati del profilo nelle strutture UserData e WorkingData non sono a ritenzione. Queste variabili vengono inizializzate con i valori di avvio dopo ogni commutazione dello stato di funzionamento della CPU da STOP a RUN. Assicurarsi che al primo richiamo dell'istruzione RampSoak, dopo la commutazione dello stato di funzionamento della CPU da STOP a RUN, i valori di queste variabili siano idonei per ottenere il comportamento desiderato.
- Il valore selezionato tramite StartMode viene limitato al campo valori del tipo di dati REAL. Solo successivamente il valore viene emesso nel parametro Output.
- Se il valore selezionato tramite StartMode non è un valore REAL valido, viene emesso il valore di uscita sostitutivo nel parametro Output. Il valore di uscita sostitutivo viene configurato con la variabile ErrorMode e limitato al campo valori del tipo di dati REAL. Se attiva, l'esecuzione del profilo inizia con questo valore di uscita sostitutivo.

- Soltanto se è impostato il parametro `Reset = FALSE` e al contempo non è presente alcun errore con il messaggio `ErrorBits ≥ 16#0002_0000` la variabile `StartMode` ha effetto sul parametro `Output`. Se è impostato il parametro `Reset = TRUE`, il valore del parametro `SubstituteOutput` viene emesso nel parametro `Output`. In presenza di un errore con il messaggio `ErrorBits ≥ 16#0002_0000`, nel parametro `Output` viene emesso il valore di uscita sostitutivo configurato nella variabile `ErrorMode`.

10.8.3.4 Configurazione del comportamento all'arresto - variabile statica `StopMode`

La variabile `StopMode` consente di determinare il comportamento dell'istruzione `RampSoak` nei seguenti casi:

- L'esecuzione del profilo si conclude quando viene raggiunto l'ultimo punto.
- L'esecuzione del profilo viene arrestata resettando `Enable`.

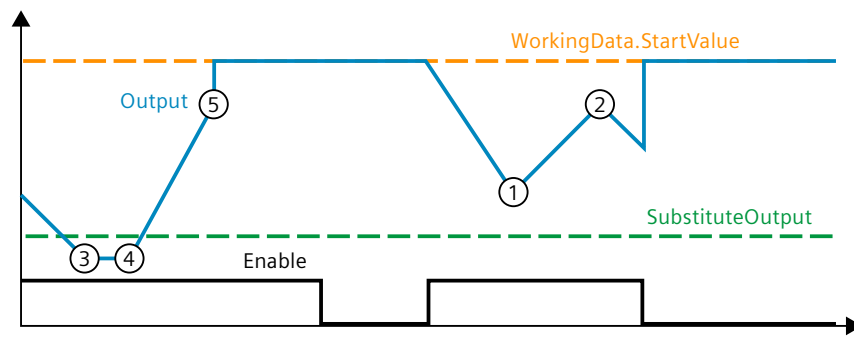
Il valore impostato con `StopMode` viene utilizzato come valore di uscita fino a quando non viene attivata una nuova azione, ad es. l'avvio dell'esecuzione del profilo.

Sono possibili le seguenti impostazioni della variabile `StopMode`:

- `StopMode = 0`

Il parametro `Output` acquisisce il valore di `WorkingData.StartValue`.

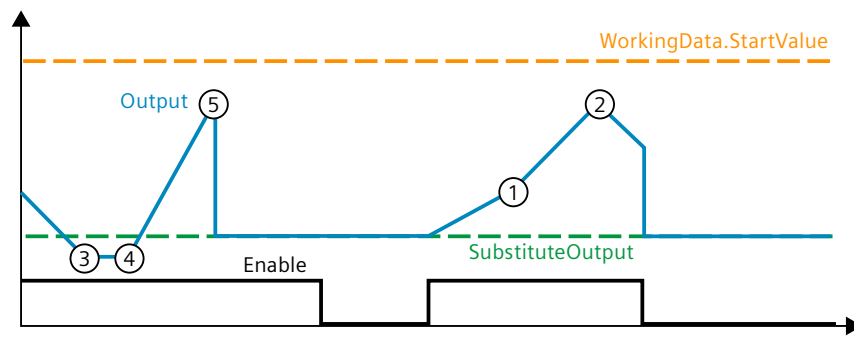
La figura seguente mostra come terminare e arrestare l'esecuzione e di un profilo con 5 punti con `StopMode = 0`:



- `StopMode = 1`

Il parametro `Output` acquisisce il valore di `SubstituteOutput`.

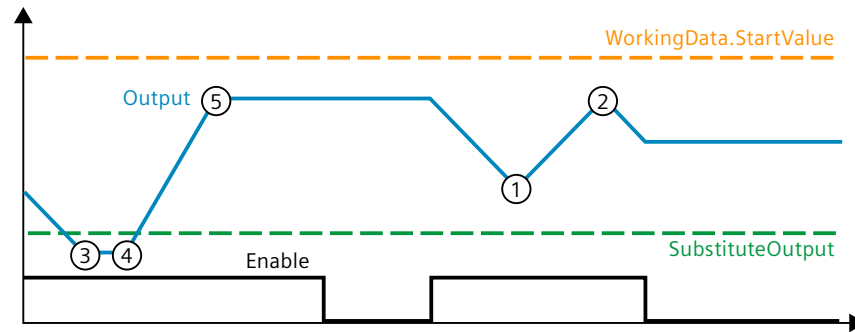
La figura seguente mostra come terminare e arrestare l'esecuzione e di un profilo con 5 punti con `StopMode = 1`:



- StopMode = 2

Il parametro Output acquisisce l'ultimo valore valido dell'esecuzione del profilo.

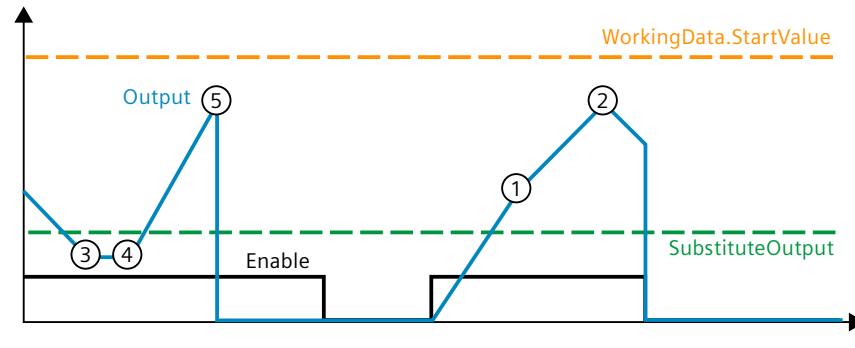
La figura seguente mostra come terminare e arrestare l'esecuzione e di un profilo con 5 punti con StopMode = 2:



- StopMode = 3

Il parametro Output acquisisce il valore 0.0.

La figura seguente mostra come terminare e arrestare l'esecuzione e di un profilo con 5 punti con StopMode = 3:

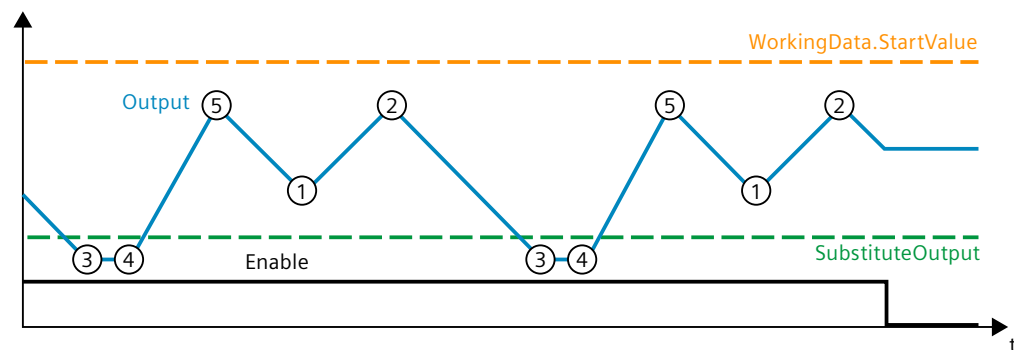


- StopMode = 4

Dopo la conclusione dell'esecuzione del profilo al raggiungimento dell'ultimo punto, essa viene riavviata automaticamente e prosegue con WorkingData.Point[1]. L'esecuzione del profilo viene ripetuta finché Enable non viene resettato.

Quando l'esecuzione del profilo viene arrestata con Enable = FALSE, il parametro Output acquisisce l'ultimo valore valido dell'esecuzione del profilo.

La figura seguente mostra come terminare e arrestare l'esecuzione e di un profilo con 5 punti con StopMode = 4:



Inoltre, per tutti i valori della variabile StopMode vale quanto segue:

- Il valore selezionato tramite StopMode viene limitato al campo valori del tipo di dati REAL. Solo successivamente il valore viene emesso nel parametro Output.
- Se il valore selezionato tramite StopMode non è un valore REAL valido, viene emesso e mantenuto il valore di uscita sostitutivo nel parametro Output. Il valore di uscita sostitutivo viene configurato con la variabile ErrorMode e limitato al campo valori del tipo di dati REAL.
- Soltanto se è impostato il parametro Reset = FALSE e al contempo non è presente alcun errore con il messaggio ErrorBits $\geq 16\#0002\_0000$ la variabile StopMode ha effetto sul parametro Output. Se è impostato il parametro Reset = TRUE, il valore del parametro SubstituteOutput viene emesso nel parametro Output. In presenza di un errore con il messaggio ErrorBits $\geq 16\#0002\_0000$, nel parametro Output viene emesso il valore di uscita sostitutivo configurato nella variabile ErrorMode.

10.8.3.5 Misura del tempo di ciclo

Misura automatica del tempo di ciclo

Per il calcolo del valore di uscita, RampSoak impiega il tempo trascorso dall'ultimo richiamo di RampSoak.

Per default il tempo di ciclo viene misurato automaticamente e a partire dal secondo richiamo viene emesso nella variabile CycleTime.Value. RampSoak misura il tempo di ciclo a ogni richiamo dell'istruzione e perciò non è utilizzabile in cicli di richiamo equidistanti, ad es. nell' OB1.

Osservare che i richiami condizionati dell'istruzione, i punti di arresto o il caricamento di istantanee come valori attuali, protraggono il tempo di ciclo nella misura automatica.

Se la misura del tempo di ciclo non fornisce un risultato valido, RampSoak calcola il valore di uscita attuale con l'ultimo tempo di ciclo valido. Inoltre RampSoak emette un messaggio di errore nel parametro ErrorBits.

Se si disattiva la misura automatica del tempo di ciclo impostando la variabile CycleTime.EnableMeasurement = FALSE, il tempo di ciclo deve essere calcolato manualmente nella variabile CycleTime.Value. RampSoak verifica la validità della variabile CycleTime.Value a ogni richiamo.

Misura automatica del tempo di ciclo con punti di arresto

Se tra due richiami di RampSoak sono attivi dei punti di arresto, la misura automatica del tempo di ciclo fornisce il tempo effettivamente trascorso tra due richiami. Se è attivo un punto di arresto, la CPU è nello stato di funzionamento ALT.

NOTA

I punti di arresto attivi prolungano l'intervallo tra due richiami di RampSoak.

Con un intervallo di tempo più lungo tra due richiami aumenta anche la modifica del valore di uscita nel parametro Output. A seconda del tempo trascorso e dei dati del profilo configurati, alcuni punti possono essere saltati.

Se non si necessita del calcolo del valore di uscita basato sul tempo effettivo con punti di arresto attivi, eseguire i passi seguenti:

- Disattivare la misura automatica del tempo di ciclo impostando la variabile `CycleTime.EnableMeasurement = FALSE`.
- Inserire il tempo di ciclo manualmente nella variabile `CycleTime.Value`.

10.8.3.6 Comportamento di abilitazione EN/ENO

Se una delle seguenti condizioni è soddisfatta, l'uscita di abilitazione ENO viene impostata su FALSE:

- L'ingresso di abilitazione EN è impostato su TRUE ed è presente un errore con messaggio di errore `ErrorBits ≥ 16#0001_0000`.
- L'ingresso di abilitazione EN impostato su FALSE.

In tutti gli altri casi l'uscita di abilitazione ENO viene impostata su TRUE.

10.8.4 Parametro di ingresso RampSoak

Parametri	Tipo di dati	Preassegnazione	Descrizione
Enable	BOOL	FALSE	Con un fronte di salita nel parametro Enable l'esecuzione del profilo e il calcolo del valore di uscita vengono avviati. Con un fronte di discesa nel parametro Enable l'esecuzione del profilo si arresta.
Hold	BOOL	FALSE	Se Hold è impostato su TRUE, l'esecuzione del profilo viene arrestata. Il valore di uscita rimane costante.
Next	BOOL	FALSE	In presenza di un fronte di salita nel parametro Next l'esecuzione del profilo continua con <code>Point[NextPoint]</code> .
SubstituteOutput	REAL	0.0	SubstituteOutput viene impiegato come valore di uscita sostitutivo se <code>Reset = TRUE</code> o se è attualmente attiva una delle seguenti modalità: • <code>ErrorMode = 1</code> • <code>StartMode = 1</code> • <code>StopMode = 1</code>

Parametri	Tipo di dati	Preassegnazione	Descrizione
Validate	BOOL	FALSE	Se Validate è impostato su TRUE, viene effettuato il controllo della validità dei dati del profilo in UserData che vengono poi acquisiti in WorkingData.
ErrorAck	BOOL	FALSE	Cancella i messaggi di errore. Il fronte FALSE -> TRUE ErrorBits viene resettato.
Reset	BOOL	FALSE	Resetta l'istruzione. - Il fronte FALSE -> TRUE ErrorBits viene resettato. - Finché Reset è impostato su TRUE, il valore di uscita sostitutivo SubstituteOutput viene emesso sull'uscita. Se contemporaneamente viene eseguito il profilo, l'esecuzione prosegue sullo sfondo. - Finché Reset è impostato su FALSE, il valore di uscita viene determinato dall'esecuzione del profilo.

10.8.5 Parametro di uscita RampSoak

Parametro	Tipo di dati	Preassegnazione	Descrizione
Output	REAL	0.0	Valore di uscita Il valore di uscita è a ritenzione.
CurrentPoint	INT	0	Numero del punto attualmente utilizzato per l'esecuzione del profilo e il calcolo del valore di uscita.
TotalTime	REAL	0.0	Tempo complessivo del profilo (somma dei tempi di tutti i punti utilizzati) in secondi
RemainingTime_Total	REAL	0.0	Tempo rimanente del profilo in secondi
RemainingTime_Point	REAL	0.0	Tempo rimanente del punto attuale in secondi
ErrorBits	DWORD	16#0	Il parametro ErrorBits (Pagina 542) mostra quali messaggi di errore sono presenti. ErrorBits è a ritenzione e viene resettato in presenza di un fronte di salita in Reset o ErrorAck.
Error	BOOL	FALSE	Se Error è impostato su TRUE si è attualmente in presenza di almeno un errore.

10.8.6 Parametro di transito RampSoak

Parametri	Tipo di dati	Preassegnazione	Descrizione
NextPoint	Int	1	Numero del punto successivo da utilizzare. Campo di valori ammesso: 1 ... WorkingData.NumberOfUsedPoints

10.8.7 Variabili statiche RampSoak

Variabile	Tipo di dati	Preassegnazione	Descrizione
UserData	AuxFct_Ramp- Soak_Profile	-	Nella struttura UserData vengono inseriti i dati del profilo. I dati del profilo nella struttura UserData sono editabili. Eventuali modifiche apportate in questa struttura si ripercuotono sull'esecuzione del profilo soltanto dopo che il controllo e la copia dei dati nella struttura WorkingData sono stati avviati.
UserData.NumberOfUsedPoints	INT	0	Numero di punti di interpolazione del profilo utilizzati Campo di valori ammesso: 1 ... 50
UserData.StartValue	REAL	0.0	StartValue viene impiegato come valore di uscita opzionale se è attualmente attiva una delle seguenti modalità: - ErrorMode = 0 - StartMode = 0 - StopMode = 0
UserData.Point	Array[1..50] of AuxFct_Ramp- Soak_Point	-	Punti di interpolazione del profilo
UserData.Point[i].Value	REAL	0.0	Valore di uscita di questo punto
UserData.Point[i].Time	REAL	0.0	Durata del punto in secondi Campo di valori consentito: Point[i].Time ≥ 0.0
WorkingData	AuxFct_Ramp- Soak_Profile	-	Nella struttura WorkingData vengono visualizzati i dati del profilo attualmente attivi. I dati del profilo nella struttura WorkingData non sono editabili.
WorkingData.NumberOfUsedPoints	INT	0	Numero di punti di interpolazione del profilo utilizzati Campo di valori ammesso: 1 ... 50
WorkingData.StartValue	REAL	0.0	WorkingData.StartValue viene impiegato come valore di uscita opzionale se è attualmente attiva una delle seguenti modalità: - ErrorMode = 0 - StartMode = 0 - StopMode = 0
WorkingData.Point	Array[1..50] of AuxFct_Ramp- Soak_Point	-	Punti di interpolazione del profilo
WorkingData.Point[i].Value	REAL	0.0	Valore di uscita di questo punto
WorkingData.Point[i].Time	REAL	0.0	Durata del punto in secondi Campo di valori consentito: Point[i].Time ≥ 0.0
ErrorMode	INT	2	Selezione del valore di uscita sostitutivo in caso di errori 0 = WorkingData.StartValue 1 = SubstituteOutput 2 = ultimo valore di uscita valido dell'esecuzione del profilo 3 = 0.0 Campo di valori ammesso: 0 - 3 Se il valore di ErrorMode non corrisponde al campo di valori consentito, ErrorMode = 2.

Variabile	Tipo di dati	Preassegnazione	Descrizione
StartMode	INT	2	Selezione del comportamento all'avvio 0 = WorkingData.StartValue 1 = SubstituteOutput 2 = avvio dall'ultimo valore di uscita 3 = 0.0 4 = continuazione dall'ultimo valore di uscita Campo di valori ammesso: 0 ... 4 Se il valore di StartMode non corrisponde al campo di valori consentito, StartMode = 2.
StopMode		2	Selezione del comportamento all'arresto 0 = WorkingData.StartValue 1 = SubstituteOutput 2 = ultimo valore di uscita valido dell'esecuzione del profilo 3 = 0.0 4 = funzionamento ciclo Campo di valori ammesso: 0 ... 4 Se il valore di StopMode non corrisponde al campo di valori consentito, StopMode = 2.
CycleTime	AuxFct_CycleTime	-	Dati del tempo di ciclo
CycleTime.Value	REAL	0.1	Tempo di ciclo in secondi (intervallo tra due richiami) Campo di valori consentito: CycleTime.Value > 0.0
CycleTime.EnableMeasurement	BOOL	TRUE	Misura automatica del tempo di ciclo - FALSE = disattivato - TRUE = attivato

10.8.8 Parametro ErrorBits

Se si verificano diversi errori contemporaneamente, i valori di ErrorBits vengono visualizzati in un'addizione binaria. Se viene visualizzato ad es. ErrorBits = 16#0000_0003 sono presenti contemporaneamente gli errori 16#0000_0001 e 16#0000_0002.

Nel caso di RampSoak gli errori emessi nel parametro ErrorBits sono suddivisi in due categorie:

- Errore con messaggio di errore ErrorBits < 16#0001_0000
Il valore di uscita può essere calcolato nonostante l'errore.
- Errore con messaggio di errore ErrorBits ≥ 16#0001_0000
L'errore impedisce il calcolo del valore di uscita. Viene emesso un valore di uscita sostitutivo.

Errore con messaggio di errore ErrorBits < 16#0001_0000

In presenza di uno o più errori con i messaggi di errore ErrorBits < 16#0001_0000, RampSoak ha la seguente reazione:

- Nonostante questo errore, il valore di uscita viene rilevato come indicato nel seguito:
 - Se Reset = FALSE, calcolo del valore di uscita mediante esecuzione del profilo
 - Se Reset = TRUE, emissione di SubstituteOutput
- Il parametro di uscita Error viene impostato.
- L'uscita di abilitazione ENO non viene modificata.

Il parametro di uscita Error viene eliminato non appena tutti gli errori sono stati eliminati.

ErrorBits (DW#16#...)	Descrizione
0000_0000	Nessun errore
0000_0001	Causa degli errore e reazione agli errori: Il parametro Output è stato limitato a -3.402823e+38 oppure +3.402823e+38. Rimedio: Se ErrorBits < 16#0001_0000 e Reset = FALSE, il valore di uscita viene limitato mentre viene determinato da StartMode o StopMode. Successivamente, a seconda del valore impostato nelle variabili StartMode o StopMode, controllare i seguenti parametri: • WorkingData.StartValue • SubstituteOutput Se ErrorBits ≥ 16#0001_0000 e Reset = FALSE, il valore di uscita sostitutivo viene limitato al momento della relativa emissione. Successivamente, a seconda del valore impostato nella variabile ErrorMode, controllare i seguenti parametri • WorkingData.StartValue • SubstituteOutput Se Reset = TRUE, controllare il parametro SubstituteOutput. Informazione supplementare: Per modificare WorkingData.StartValue, editare innanzitutto UserData.StartValue, quindi impostare il parametro Validate = TRUE. Non modificare manualmente i dati nella struttura WorkingData.
0000_0002	Causa dell'errore: La misura del tempo di ciclo fornisce un valore non valido mentre è attiva l'esecuzione del profilo (Enable = TRUE). Reazione all'errore: Se è già stato misurato un valore valido per il tempo di ciclo, RampSoak esegue il profilo a partire da quest'ultimo valore della variabile CycleTime.Value. Se per il tempo di ciclo non è ancora stato misurato un valore valido, RampSoak continua ad emettere il valore di uscita configurato con la variabile StartMode nel parametro Output.
0000_0004	Causa dell'errore: Durante il controllo dei dati del profilo, una o più variabili nella struttura UserData presentano valori non validi. Reazione all'errore: I dati del profilo nella struttura UserData non vengono acquisiti nella struttura WorkingData, pertanto le modifiche apportate nella struttura UserData non diventano attive. Rimedio: Durante il controllo dei dati del profilo, accertarsi che siano soddisfatte le seguenti condizioni: • $1 \leq \text{UserData.NumberOfUsedPoints} \leq 50$ • $-3.402823e+38 \leq \text{UserData.Point}[i].\text{Value} \leq 3.402823e+38$ con l'indice $i = 1..\text{UserData.NumberOfUsedPoints}$ • $0.0 \leq \text{UserData.Point}[i].\text{Time} \leq 3.402823e+38$ con l'indice $i = 1..\text{UserData.NumberOfUsedPoints}$ • $-3.402823e+38 \leq \text{UserData.StartValue} \leq 3.402823e+38$ • $\text{NextPoint} \leq \text{UserData.NumberOfUsedPoints}$

ErrorBits (DW#16#...)	Descrizione
	$0.0 < \text{UserData.Point}[1].\text{Time} + \text{UserData.Point}[2].\text{Time} + \dots + \text{UserData.Point}[\text{UserData.NumberOfUsedPoints}].\text{Time} \leq 3.402823\text{e}+38$ - Se è attiva l'esecuzione del profilo: $\text{CurrentPoint} \leq \text{UserData.NumberOfUsedPoints}$ Informazione supplementare: I dati del profilo nella struttura UserData vengono controllati nei seguenti casi: - Quando il parametro Validate è impostato su TRUE. - Oppure quando RampSoak viene richiamato per la prima volta dopo la commutazione dello stato di funzionamento della CPU da STOP a RUN con il parametro Enable = TRUE senza un controllo preventivo dei dati del profilo. Tenere presente che tutte le variabili nelle strutture UserData e WorkingData non sono a ritenzione. Queste variabili vengono inizializzate con i valori di avvio dopo ogni commutazione dello stato di funzionamento della CPU da STOP a RUN.
0000_0008	Causa dell'errore: Il parametro NextPoint contiene un valore non valido. Reazione all'errore: NextPoint viene resettato all'ultimo valore valido. Rimedio: Accertarsi che venga soddisfatta la seguente condizione: $1 \leq \text{NextPoint} \leq \text{WorkingData.NumberOfUsedPoints}$

Errore con messaggio di errore ErrorBits $\geq 16\#0001\_0000$

In presenza di uno o più errori con i messaggi di errore ErrorBits $\geq 16\#0001\_0000$, RampSoak ha la seguente reazione:

- Il valore di uscita non può essere rilevato come previsto. La seguente tabella descrive la reazione del parametro Output e l'esecuzione del profilo.
- Il parametro di uscita Error viene impostato.
- L'uscita di abilitazione ENO viene impostata su FALSE.

Non appena tutti gli errori segnalati con i messaggi ErrorBits $\geq 16\#0001\_0000$ sono stati eliminati, RampSoak ha le seguenti reazioni:

- Il valore di uscita viene rilevato come indicato nel seguito:
 - Se Reset = FALSE, valore di uscita mediante esecuzione del profilo
 - Se Reset = TRUE, emissione di SubstituteOutput
- L'uscita di abilitazione ENO viene impostata su TRUE.

Il parametro di uscita Error viene eliminato non appena tutti gli errori sono stati eliminati.

ErrorBits (DW#16#...)	Descrizione
0001_0000	Causa dell'errore: Il parametro SubstituteOutput o la variabile WorkingData.StartValue è attualmente utilizzato/a per determinare il valore di uscita, ma non contiene un valore REAL valido. Reazione all'errore: Se Reset = TRUE e SubstituteOutput è un valore REAL valido, il parametro Output continua ad emettere SubstituteOutput. In tutti gli altri casi il parametro Output viene impostato su 0.0. Rimedio:

ErrorBits (DW#16#...)	Descrizione			
0001_0000	Accertarsi che il parametro SubstituteOutput e la variabile WorkingData.StartValue siano valori REAL validi (≠NaN ad es. 16#7FFF_FFFF). La variabile utilizzata dipende dal Reset, dagli errori presenti e da ErrorMode:			
	Reset	ErrorBits	ErrorMode	Variabile utilizzata
	-	>= 16#0002_0000	0	WorkingData.StartValue
	-	>= 16#0002_0000	1	SubstituteOutput
	TRUE	-	-	SubstituteOutput
Informazione supplementare: Per modificare WorkingData.StartValue, editare innanzitutto UserData.StartValue, quindi impostare il parametro Validate = TRUE. Non modificare manualmente i dati nella struttura WorkingData.				
0004_0000	Causa dell'errore: Il calcolo durante l'esecuzione del profilo fornisce un valore REAL non valido. Reazione all'errore: L'esecuzione del profilo viene interrotta. Se Reset = FALSE, nel parametro Output viene emesso e mantenuto il valore di uscita sostitutivo configurato nella variabile ErrorMode. Se Reset = TRUE, il parametro Output continua ad emettere SubstituteOutput. Rimedio: Controllare la validità dei valori REAL nella struttura WorkingData e riavviare l'esecuzione del profilo se necessario. Informazione supplementare: Per modificare i dati del profilo, editare innanzitutto la struttura UserData, quindi impostare il parametro Validate = TRUE. Non modificare manualmente i dati nella Struktur WorkingData.			
0008_0000	Causa dell'errore: Il parametro Enable o il parametro Next è impostato su TRUE ma nella struttura WorkingData non sono presenti dati del profilo validi. Reazione all'errore: Il parametro Enable e il parametro Next non sono attivi. Se Reset = FALSE, nel parametro Output viene emesso il valore di uscita sostitutivo configurato nella variabile ErrorMode. Se Reset = TRUE, il parametro Output continua ad emettere SubstituteOutput. Rimedio: Inserire dei dati del profilo validi nella struttura UserData e successivamente impostare il parametro Validate = TRUE. In questo modo i dati del profilo dopo il controllo vengono trasferiti nella struttura WorkingData. Informazione supplementare: Se il parametro Enable o il parametro Next sono ancora impostati su TRUE, essi vengono attivati non appena nella struttura WorkingData sono presenti dei dati del profilo validi. Un nuovo fronte positivo non è necessario. Tenere presente che tutte le variabili nelle strutture UserData e WorkingData non sono a ritenzione. Queste variabili vengono inizializzate con i valori di avvio dopo ogni commutazione dello stato di funzionamento della CPU da STOP a RUN.			
0010_0000	Causa dell'errore: La variabile (configurata con StopMode) che determina il valore di uscita non contiene un valore REAL valido alla conclusione o all'arresto dell'esecuzione del profilo. Reazione all'errore: Se Reset = FALSE, nel parametro Output viene emesso e mantenuto il valore di uscita sostitutivo configurato nella variabile ErrorMode. Se Reset = TRUE, il parametro Output continua ad emettere SubstituteOutput. Rimedio: Accertarsi che la variabile sia un valore REAL valido (≠ NaN ad es. 16#7FFF_FFFF). La variabile utilizzata dipende da StopMode: - StopMode = 0: WorkingData.StartValue - StopMode = 1: SubstituteOutput			

ErrorBits (DW#16#...)	Descrizione
	Informazione supplementare: Per modificare WorkingData.StartValue, editare innanzitutto UserData.StartValue, quindi impostare il parametro Validate = TRUE. Non modificare manualmente i dati nella struttura WorkingData.
0020_0000	Causa dell'errore: La variabile (configurata con StartMode) che determina il valore di uscita non contiene un valore REAL valido al primo richiamo dell'istruzione o all'avvio dell'esecuzione del profilo. Reazione all'errore: Se Reset = FALSE, nel parametro Output viene emesso e mantenuto il valore di uscita sostitutivo configurato nella variabile ErrorMode. Quando viene attivata (Enable = TRUE), l'esecuzione del profilo inizia a partire da questo valore di uscita sostitutivo. Se Reset = TRUE, il parametro Output continua ad emettere SubstituteOutput. Rimedio: Accertarsi che la variabile sia un valore REAL valido ($\neq$ NaN ad es. 16#7FFF_FFFF). La variabile utilizzata dipende da StartMode: - StartMode = 0: WorkingData.StartValue - StartMode = 1: SubstituteOutput - StartMode = 2: Output Informazione supplementare: Per modificare WorkingData.StartValue, editare innanzitutto UserData.StartValue, quindi impostare il parametro Validate = TRUE. Non modificare manualmente i dati nella struttura WorkingData.
0040_0000	Causa dell'errore: La variabile CycleTime.Value contiene un valore non valido mentre è attiva l'esecuzione del profilo (Enable = TRUE). Reazione all'errore: L'esecuzione del profilo viene arrestata. Se Reset = FALSE, nel parametro Output viene emesso il valore di uscita sostitutivo configurato nella variabile ErrorMode. Se Reset = TRUE, il parametro Output continua ad emettere SubstituteOutput. L'esecuzione del profilo prosegue quando l'errore non è più presente. Se l'esecuzione del profilo viene arrestata anticipatamente, in seguito il valore di uscita sostitutivo viene mantenuto. Rimedio: Accertarsi che siano soddisfatte le seguenti condizioni: $0.0 < \text{CycleTime.Value} \leq 3.402823\text{e}+38$ - CycleTime.Value è un valore REAL valido ($\neq$ NaN ad es. 16#7FFF_FFFF) Informazione supplementare: Per il calcolo automatico del valore della variabile CycleTime.Value, impostare la variabile CycleTime.EnableMeasurement su TRUE.

10.9 Filter_PT1

10.9.1 Compatibilità con CPU e FW

La tabella seguente mostra quali versioni di Filter_PT1 sono compatibili con le varie CPU :

CPU	FW	Filter_PT1
S7-1200	da V4.2	V1.0
S7-1200 G2	da V1.0	V1.0
CPU basate su S7-1500	da V2.0	V1.0

10.9.2 Descrizione del filtro_PT1

Descrizione

L'istruzione Filter_PT1 è un elemento di trasmissione proporzionale con un ritardo di prim'ordine, denominato anche regolatore PT1.

Filter_PT1 può essere impiegato per i seguenti scopi:

- Filtro passabasso per attenuare le percentuali ad alta frequenza di un segnale, quali ad es. la rumorosità.
- Componente di ritardo per il livellamento di gradini del segnali, ad es. del setpoint o del valore di uscita di un regolatore.
- Blocco di simulazione del processo, per la realizzazione di un circuito di regolazione chiuso all'interno della CPU. Ciò consente ad esempio di testare il regolatore prima della messa in servizio.

Possono essere preimpostati i seguenti parametri del filtro:

- Guadagno proporzionale (Gain)
- Costante tempo di ritardo (Lag)

NOTA

Differenze tra un regolatore PT1 a tempo continuo e un filtro_PT1

Poiché Filter_PT1 viene eseguito in un programma PLC, Filter_PT1 è la realizzazione a tempo discreto di un regolatore PT1. I sistemi a tempo discreto non possono presentare le stesse proprietà del modello corrispondente a tempo continuo. Tuttavia, in funzione del tempo di ciclo, questi sistemi sono in grado di emulare con una buona precisione un sistema a tempo continuo: Più breve e costante è il tempo di ciclo, minore sarà la divergenze tra le proprietà del Filter_PT1 e quelle di un regolatore PT1 a tempo continuo. Le proprietà di un regolatore PT1-a tempo continuo, sono la funzione di trasmissione, il comportamento temporale e il comportamento della frequenza illustrati nel seguito.

Per una buona riproduzione del comportamento della frequenza si raccomanda un tempo ciclo massimo pari ad un decimo della durata più breve dei periodi del segnale di ingresso. Ad esempio un segnale con periodi di frequenza che raggiungono 50 Hz, ha una durata periodo più breve, pari a 20 ms. Per realizzare una buona riproduzione del comportamento della frequenza si raccomanda, per il presente esempio, un tempo di ciclo massimo di 2 ms.

Funzione di trasmissione di un regolatore PT1

La seguente formula mostra la funzione di trasmissione di un regolatore PT1, dove s è uguale a Laplace operator:

$$G(s) = \frac{\text{Output}(s)}{\text{Input}(s)} = \frac{\text{Gain}}{1 + \text{Lag} \cdot s}$$

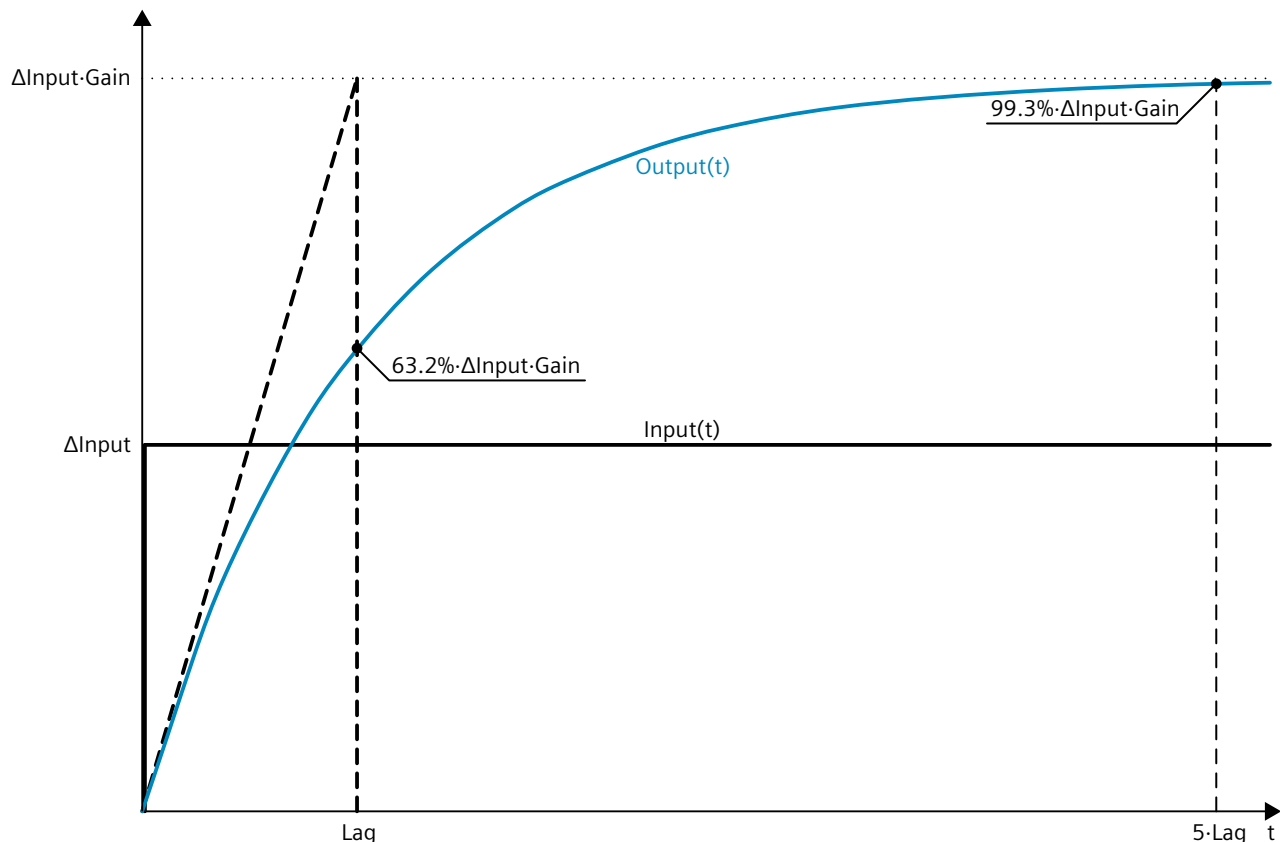
Funzione di Comportamento temporale di un regolatore PT1

La risposta a gradino è la reazione del valore di uscita ad un gradino del valore di ingresso.

La risposta a gradino per un gradino del valore di ingresso da 0 a ΔInput può essere calcolata con la seguente formula:

$$\text{Output}(t) = \Delta\text{Input} \cdot \text{Gain} \cdot (1 - e^{-\frac{t}{\text{Lag}}})$$

La figura seguente mostra la risposta a gradino di un regolatore PT1:



Comportamento della frequenza di un regolatore PT1

Il comportamento della frequenza di un elemento di trasmissione viene descritto dalla caratteristica di ampiezza e dalla caratteristica di fase.

La caratteristica di ampiezza descrive il guadagno di un segnale tramite l'elemento di trasmissione in funzione della frequenza di circuito del segnale.

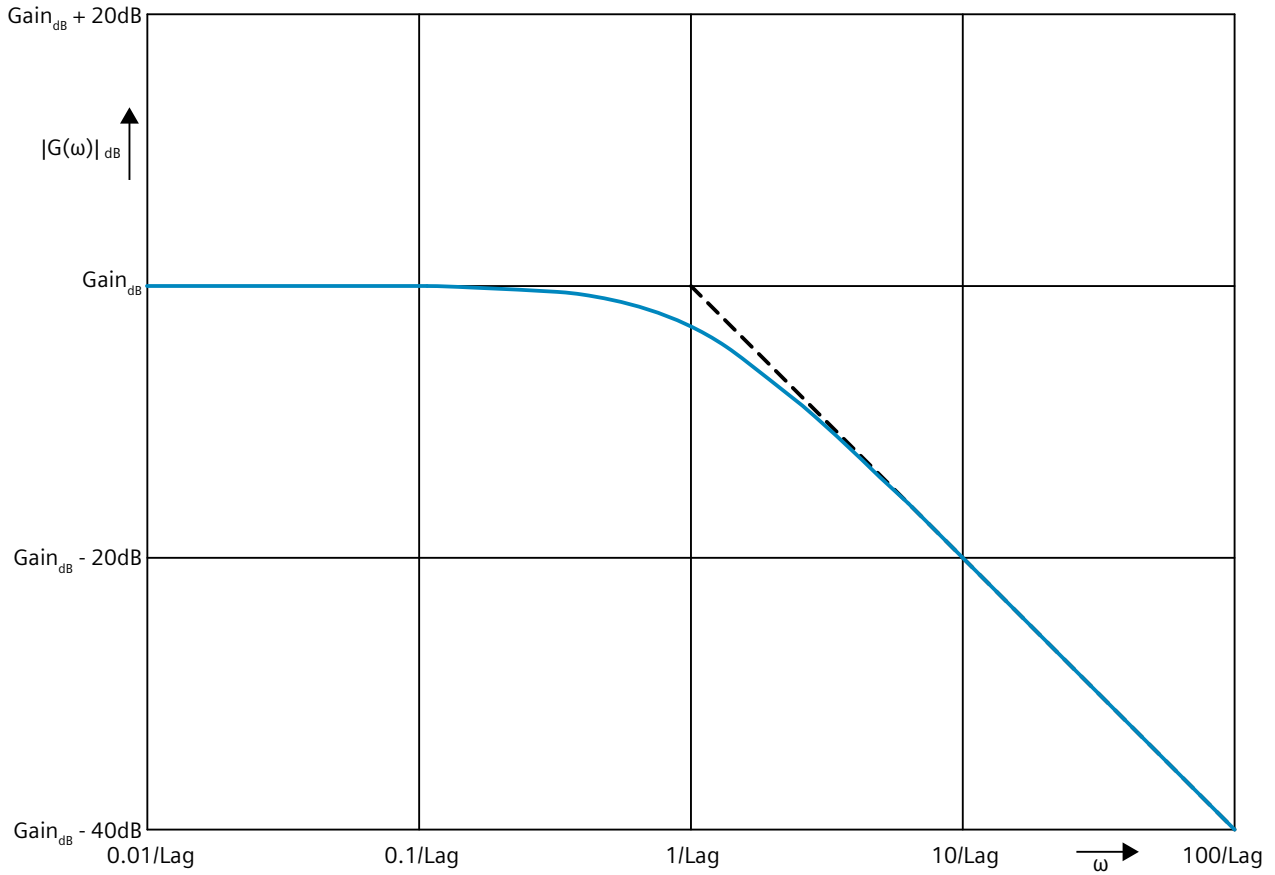
La seguente equazione descrive la caratteristica di ampiezza di un regolatore PT1:

$$|G(\omega)| = \frac{\text{Gain}}{\sqrt{1 + (\omega \cdot \text{Lag})^2}}$$

$|G(\omega)|$ Guadagno di un segnale sicurezza in funzione della frequenza di circuito

ω Frequenza di circuito

La seguente figura illustra la caratteristica di ampiezza di un regolatore PT1:



La caratteristica di fase descrive l'offset di fase di un segnale tramite l'elemento di trasmissione in funzione della frequenza di circuito del segnale.

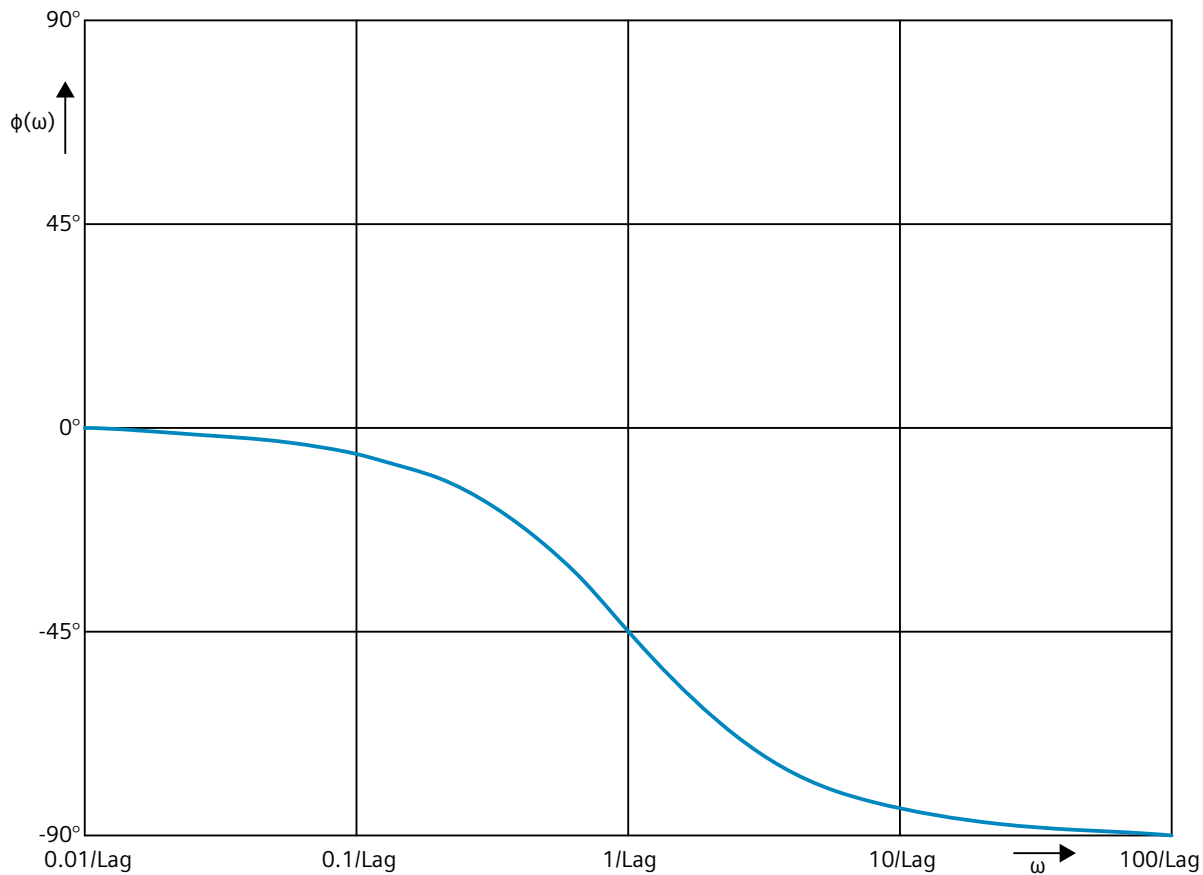
La seguente equazione descrive la caratteristica di fase di un regolatore PT1:

$$\varphi(\omega) = -\tan^{-1}(\omega \cdot \text{Lag})$$

$\varphi(\omega)$ Caratteristica di fase in funzione della frequenza di circuito

ω Frequenza di circuito

La seguente figura illustra la caratteristica di fase di un regolatore PT1:



Richiamo

In un OB o FC Filter_PT1 viene richiamato come DB di un'istanza singola. In un FB Filter_PT1 può essere richiamato sia come DB di istanza singola, sia come DB di multiistanza e come DB di istanza di un parametro.

Al richiamo dell'istruzione non viene creato nessun oggetto tecnologico. Non è disponibile un'interfaccia utente per la parametrizzazione e la messa in servizio. La parametrizzazione di Filter_PT1 avviene direttamente nel DB di istanza e la messa in servizio di Filter_PT1 viene eseguita in una tabella di controllo del programma utente nella CPU oppure in HMI.

Avvio

Le variabili nel campo statico di Filter_PT1 non sono a ritenzione. Queste variabili vengono inizializzate con i valori di avvio dopo ogni commutazione dello stato di funzionamento della CPU da STOP a RUN. Per mantenere i valori attuali eventualmente modificati nel funzionamento online anche dopo la commutazione dello stato di funzionamento della CPU, salvare questi valori nei valori di avvio del blocco dati.

Nella variabile StartMode si definisce il valore di inizializzazione per il parametro Output.

Il valore di inizializzazione viene emesso al primo richiamo di Filter_PT1 dopo la

- commutazione dello stato di funzionamento della CPU

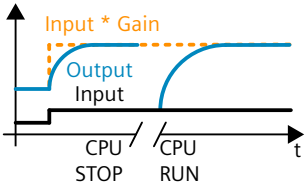
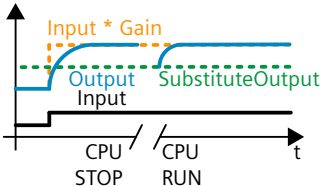
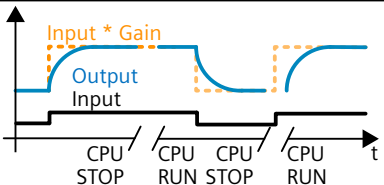
oppure

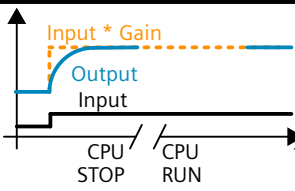
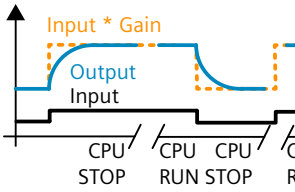
- dopo l'esecuzione della funzione "Carica valori di avvio come valori attuali" (solo con l'opzione "Tutti i valori", non con l'opzione "Solo valori di impostazione")

nel parametro Output.

Ai richiami successivi Filter_PT1 calcola il valore di uscita partendo da questo valore di inizializzazione e basandosi sul valore di ingresso e sulla configurazione del filtro.

La tabella seguente mostra l'interdipendenza tra la variabile StartMode e il parametro Output. I valori nella colonna Output vengono emessi, dopo la commutazione dello stato di funzionamento della CPU, nel parametro Output.

StartMode	Output	Esempio
0	Valore del parametro Input	
1	Valore del parametro SubstituteOutput	
2	Rimane invariato Il parametro Output è a ritenzione Impostazione standard Viene utilizzato da StartMode non nel campo 0...4	

StartMode	Output	Esempio
3	0.0	
4	Valore del prodotto Input * Gain	

Inoltre, per tutti i valori della variabile StartMode vale quanto segue:

- Il valore di inizializzazione viene limitato al campo valori del tipo di dati REAL. Solo a questo punto viene emesso il valore di inizializzazione nel parametro Output.
- Se il valore di inizializzazione non è un valore REAL valido, viene emesso il valore di uscita sostitutivo nel parametro Output. Il valore di uscita sostitutivo viene configurato con la variabile ErrorMode. Il valore di uscita sostitutivo viene limitato al campo valori del tipo di dati REAL prima di essere emesso nel parametro Output. Se il valore di uscita sostitutivo è a sua volta un valore REAL non valido, 0.0 viene emesso nel parametro Output. Basandosi su questo valore di uscita sostitutivo, l'istruzione calcola il valore di uscita ai richiami successivi.
- Soltanto se al primo richiamo dell'istruzione è impostato il parametro Reset = FALSE e al contempo non è presente alcun errore con la segnalazione ErrorBits $\geq$ 16#0002_0000, la variabile StartMode è attiva. Se è impostato il parametro Reset = TRUE, il valore del parametro SubstituteOutput viene emesso nel parametro Output. In presenza di un errore con segnalazione ErrorBits $\geq$ 16#0002_0000, nel parametro Output viene emesso il valore di uscita sostitutivo configurato nella variabile ErrorMode.

Comportamento in caso di errori

L'istruzione Filter_PT1 individua svariati errori che possono verificarsi durante il calcolo del valore di uscita. Il risultato di questo calcolo può essere emesso sull'uscita anche se è presente un errore. Se un errore impedisce il calcolo corretto del valore di uscita, sull'uscita viene emesso un valore di uscita sostitutivo.

Se un errore impedisce di calcolare correttamente il valore di uscita, definire nella variabile ErrorMode il valore di uscita sostitutivo come indicato nel seguito.

La tabella seguente mostra l'interdipendenza tra il valore della variabile ErrorMode e il valore di uscita sostitutivo che Filter_PT1 emette nel parametro Output:

ErrorMode	Output
0	Valore del parametro Input
1	Valore del parametro SubstituteOutput

ErrorMode	Output
2	Ultimo valore di uscita valido del filtro 0.0, in assenza di un valore di uscita valido del filtro Impostazione standard Viene utilizzato se ErrorMode non si trova nel campo 0...4
3	0.0
4	Valore del prodotto Input * Gain

Inoltre, per tutti i valori della variabile ErrorMode vale quanto segue:

- Se il valore di uscita sostitutivo non è un valore REAL valido, come valore di uscita viene emesso 0.0.
- Il valore di uscita sostitutivo viene limitato al campo valori del tipo di dati REAL. Solo in quel momento viene emesso il valore di uscita sostitutivo nel parametro Output.
- La variabile ErrorMode è effettiva solo se è impostato il parametro Reset = FALSE. Se è impostato il parametro Reset = TRUE, il valore del parametro SubstituteOutput viene emesso nel parametro Output.

Il parametro Error indica se attualmente è presente un errore. Se l'errore non è più presente, Error viene impostato su FALSE. Il parametro ErrorBits indica quali errori si sono verificati. ErrorBits è a ritenzione e viene resettato soltanto da un fronte di salita nel parametro Reset o ErrorAck.

10.9.3 Modo di funzionamento Filter_PT1

Comportamento in caso di reset

Filter_PT1 si comporta, in funzione del parametro Reset, nel seguente modo:

- Se è impostato il parametro Reset = TRUE, il valore del parametro SubstituteOutput viene emesso nel parametro Output.
- Se è impostato il parametro Reset = FALSE, il valore emesso nel parametro Output viene calcolato con l'algoritmo del filtro.
- Se il parametro Reset di FALSE è impostato su TRUE, il valore del parametro Output commuta direttamente sul valore del parametro SubstituteOutput. Questa commutazione può essere repentina. Inoltre viene resettato il parametro ErrorBits.
- Se il parametro Reset di TRUE viene impostato su FALSE, l'algoritmo del filtro viene impostato in modo da consentire che la commutazione avvenga bumpless.

Comportamento di abilitazione EN/ENO

Se una delle seguenti condizioni è soddisfatta, l'uscita di abilitazione ENO viene impostata su FALSE:

- L'ingresso di abilitazione EN è impostato su TRUE e il parametro Output è definito da un valore di uscita sostitutivo in presenza dei messaggi di errore ErrorBits $\geq 16\#0001\_0000$.
- Ingresso di abilitazione EN impostato su FALSE.

Diversamente l'uscita di abilitazione ENO viene impostata su TRUE:

Misura automatica del tempo di ciclo

Per il calcolo del valore di uscita, Filter_PT1 impiega il tempo trascorso dall'ultimo richiamo di Filter_PT1.

Per default il tempo di ciclo viene misurato automaticamente e a partire dal secondo richiamo viene emesso nella variabile CycleTime.Value. Filter_PT1 misura il tempo di ciclo a ogni richiamo dell'istruzione e perciò non è utilizzabile in cicli di richiamo equidistanti, ad es. nell'OB1.

Osservare che i richiami condizionati dell'istruzione, i punti di arresto o il caricamento di istantanee come valori attuali, protraggono il tempo di ciclo nella misura automatica. Un tempo di ciclo troppo lungo viene individuato come errore con la segnalazione di errore ErrorBits = 16#0008_0000.

Se la misura del tempo di ciclo non fornisce un risultato valido, Filter_PT1 calcola il valore di uscita attuale con l'ultimo tempo di ciclo valido. Inoltre Filter_PT1 emette un messaggio di errore nel parametro ErrorBits.

Se si disattiva la misura automatica del tempo di ciclo impostando la variabile CycleTime.EnableMeasurement = FALSE, il tempo di ciclo deve essere calcolato manualmente nella variabile CycleTime.Value. Filter_PT1 verifica la validità della variabile CycleTime.Value a ogni richiamo.

Misura automatica del tempo di ciclo con punti di arresto

Se tra due richiami di Filter_PT1 sono attivi dei punti di arresto, la misura automatica del tempo di ciclo fornisce il tempo effettivamente trascorso tra due richiami. Se è attivo un punto di arresto, la CPU è nello stato di funzionamento ALT.

NOTA

I punti di arresto attivi prolungano l'intervallo tra due richiami di Filter_PT1.

Con un intervallo di tempo più lungo tra due richiami aumenta anche la variazione del valore di uscita nel parametro Output.

Può inoltre accadere che, a causa di intervalli più lunghi, la condizione $\text{Lag} \geq \text{CycleTime.Value}/2$ venga violata e che venga pertanto rilevato un errore con la segnalazione di errore ErrorBits = 16#0008_0000.

Se non si necessita del calcolo del valore di uscita basato sul tempo effettivo con punti di arresto, eseguire i passi seguenti:

- Disattivare la misura automatica del tempo di ciclo impostando la variabile CycleTime.EnableMeasurement = FALSE.
- Inserire manualmente il tempo di ciclo nella variabile CycleTime.Value.

10.9.4 Parametro di ingresso Filter_PT1

Parametri	Tipo di dati	Preassegnazione	Descrizione
Input	REAL	0.0	Valore di ingresso
SubstituteOutput	REAL	0.0	SubstituteOutput viene impiegato come valore di uscita sostitutivo se - Reset = TRUE oppure - se un errore con messaggio di errore ErrorBits $\geq 16\#0001\_0000$ impedisce il calcolo corretto del valore di uscita e ErrorMode è configurato sul valore 1.
ErrorAck	BOOL	FALSE	Cancella i messaggi di errore Fronte FALSE -> TRUE ErrorBits viene resettato.
Reset	BOOL	FALSE	Esegue il riavvio dell'istruzione - Fronte FALSE -> TRUE ErrorBits viene resettato. - Finché Reset è impostato su TRUE, il valore di uscita sostitutivo SubstituteOutput viene emesso sull'uscita. - Finché Reset è impostato su FALSE, il calcolo del valore di uscita viene eseguito.

10.9.5 Parametro di uscita Filter_PT1

Parametro	Tipo di dati	Preassegnazione	Descrizione
Output	REAL	0.0	Valore di uscita Il valore di uscita è a ritenzione.
ErrorBits	DWORD	DW#16#0	Il parametro ErrorBits (Pagina 556) mostra quali messaggi di errore sono presenti. ErrorBits è a ritenzione e viene resettato in presenza di un fronte di salita in Reset o ErrorAck.
Error	BOOL	FALSE	Se Error è impostato su TRUE si è attualmente in presenza di almeno un errore.

10.9.6 Variabili Filter_PT1 statiche

Variabile	Tipo di dati	Preassegnazione	Descrizione
Gain	REAL	1.0	Guadagno proporzionale
Lag	REAL	25.0	Costante tempo di ritardo in secondi Campo di valori consentito: $Lag \geq CycleTime.Value/2$
ErrorMode	INT	2	Selezione del valore di uscita sostitutivo in caso di errori 0 = Input 1 = SubstituteOutput 2 = ultimo valore di uscita valido del filtro 3 = 0.0 4 = Input * Gain Campo di valori ammesso: 0 ... 4

Variabile	Tipo di dati	Preassegnazione	Descrizione
StartMode	INT	2	Selezione del valore di uscita per il primo richiamo dell'istruzione 0 = Input 1 = SubstituteOutput 2 = ultimo valore di uscita 3 = 0.0 4 = Input * Gain Campo di valori ammesso: 0 ... 4
CycleTime	AuxFct_CycleTime	-	Dati del tempo di ciclo
CycleTime.Value	REAL	0.1	Tempo di ciclo in secondi (intervallo tra due richiami) Campo di valori consentito: CycleTime.Value > 0.0
CycleTime.EnableMeasurement	BOOL	TRUE	Misura automatica del tempo di ciclo - FALSE = disattivato - TRUE = attivato

10.9.7 Parametro ErrorBits

Se si verificano diversi errori contemporaneamente, i valori di ErrorBits vengono visualizzati in un'addizione binaria. Se viene visualizzato ad es. ErrorBits = 16#0000_0003 sono presenti contemporaneamente gli errori 16#0000_0001 e 16#0000_0002.

Nel caso di Filter_PT1 gli errori emessi nel parametro ErrorBits sono suddivisi in due categorie:

- Errore con messaggio di errore ErrorBits < 16#0001_0000
- Errore con messaggio di errore ErrorBits ≥ 16#0001_0000

Errore con messaggio di errore ErrorBits < 16#0001_0000

In presenza di uno o più errori con i messaggi di errore ErrorBits < 16#0001_0000, Filter_PT1 ha la seguente reazione:

- Nonostante questo errore, il valore di uscita viene rilevato come indicato nel seguito:
 - Se Reset = FALSE, calcolo del valore di uscita con l'algoritmo del filtro
 - Se Reset = TRUE, emissione di SubstituteOutput
- Il parametro di uscita Error viene impostato.
- L'uscita di abilitazione ENO non viene modificata.

Il parametro di uscita Error viene eliminato non appena tutti gli errori sono stati eliminati.

ErrorBits (DW#16#...)	Descrizione
0000_0000	Nessun errore
0000_0001	Causa degli errore e reazione agli errori: Il parametro Output è stato limitato a -3.402823e+38 oppure +3.402823e+38. Rimedio: Se sull'uscita viene emesso il valore stabilito dalla funzione del filtro (Reset = FALSE e ErrorBits < 16#0001_0000), controllare le seguenti variabili utilizzate nel calcolo del filtro: • Input • Gain • Lag • CycleTime.Value

ErrorBits (DW#16#...)	Descrizione
	Se ErrorBits $\geq$ 16#0001_0000 e Reset = FALSE, il valore di uscita sostitutivo viene limitato al momento della relativa emissione. Successivamente, a seconda del valore impostato nella variabile ErrorMode:, controllare i seguenti parametri • Input • SubstituteOutput • Il prodotto di Input e Gain Se Reset = TRUE, controllare il parametro SubstituteOutput.
0000_0002	Causa dell'errore: La misura del tempo di ciclo fornisce un valore non valido mentre viene eseguito il calcolo del valore di uscita (Reset = FALSE). Reazione all'errore: Se è già stato misurato un valore valido per il tempo di ciclo, Filter_PT1 calcola il valore di uscita a partire da quest'ultimo valore della variabile CycleTime.Value. Se per il tempo di ciclo non è ancora stato misurato un valore valido, Filter_PT1 continua ad emettere il valore di uscita configurato con la variabile StartMode nel parametro Output.

Errore con messaggio di errore ErrorBits $\geq$ 16#0001_0000

In presenza di uno o più errori con i messaggi di errore ErrorBits $\geq$ 16#0001_0000, Filter_PT1 ha la seguente reazione:

- Il valore di uscita non può essere rilevato come previsto. Viene invece emesso il valore di uscita sostitutivo.
- Il parametro di uscita Error viene impostato.
- L'uscita di abilitazione ENO viene impostata su FALSE.

Non appena tutti gli errori segnalati con i messaggi ErrorBits $\geq$ 16#0001_0000 sono stati eliminati, Filter_PT1 ha le seguenti reazioni:

- Il valore di uscita viene rilevato come indicato nel seguito:
 - Se Reset = FALSE, calcolo del valore di uscita con l'algoritmo del filtro
 - Se Reset = TRUE, emissione di SubstituteOutput
- L'uscita di abilitazione ENO viene impostata su TRUE.

Il parametro di uscita Error viene eliminato non appena tutti gli errori sono stati eliminati.

ErrorBits (DW#16#...)	Descrizione		
0001_0000	Causa dell'errore: Il parametro SubstituteOutput o un'altra variabile, utilizzata come valore di uscita non ha un valore REAL valido. Reazione all'errore: L'uscita viene impostata su 0.0. Rimedio: Accertarsi che la variabile utilizzata come valore di uscita sia un valore REAL ($\neq$ NaN ad es. 16#7FFF_FFFF) valido. La variabile utilizzata come valore di uscita è in funzione di Reset e ErrorMode:		
	Reset	ErrorMode	Valore di uscita
	FALSE	0	Input
	FALSE	1	SubstituteOutput

ErrorBits (DW#16#...)	Descrizione		
0001_0000	FALSE	4	Il prodotto di Input e Gain
	TRUE	-	SubstituteOutput
0002_0000	Causa dell'errore: Il parametro Input non ha un valore REAL valido mentre viene eseguito il calcolo del valore di uscita (Reset = FALSE). Reazione all'errore: Nel parametro Output viene emesso il valore di uscita sostitutivo configurato nella variabile ErrorMode. Con ErrorMode = 0, 0.0 viene impiegato come valore di uscita. Rimedio: Accertarsi che il parametro Input sia un valore REAL valido ($\neq$ NaN, ad es. 16#7FFF_FFFF).		
0004_0000	Causa dell'errore: Il calcolo del valore di uscita fornisce un valore REAL non valido per il parametro Output. Reazione all'errore: Nel parametro Output viene emesso il valore di uscita sostitutivo configurato nella variabile ErrorMode. Rimedio: Controllare tutte le variabili comprese nel calcolo del valore di uscita: • Input • Gain • Lag • CycleTime.Value Queste variabili hanno i seguenti valori validi. Con questa combinazione delle variabili, il calcolo del valore di uscita non riesce.		
0008_0000	Causa dell'errore: La variabile Gain, o Lag non ha un valore valido mentre viene eseguito il calcolo del valore di uscita (Reset = FALSE). Reazione all'errore: Nel parametro Output viene emesso il valore di uscita sostitutivo configurato nella variabile ErrorMode. Rimedio: Accertarsi che per i valori delle variabili Gain, e Lag siano soddisfatte le seguenti condizioni: • $-3.402823e+38 \leq \text{Gain} \leq 3.402823e+38$ • $\text{CycleTime.Value}/2 \leq \text{Lag} \leq 3.402823e+38$ • I valori sono valori REAL validi ($\neq$ NaN ad es. 16#7FFF_FFFF) Informazione supplementare: Osservare che nei seguenti casi può accadere che la condizione $\text{CycleTime.Value}/2 \leq \text{Lag}$ venga violata: • L'intervallo tra due richiami di Filter_PT1 supera $2 * \text{Lag}$, ad es. a causa di richiami condizionati durante l'esecuzione del programma o di punti di arresto attivi. • Un'istantanea del DB di istanza Filter_PT1 viene caricata come valore attuale nella CPU e la generazione di quest'istanza presenta una datazione superiore a $2 * \text{Lag}$. In questi casi viene, durante la misura automatica, individuato un errore con la segnalazione di errore ErrorBits = 16#0008_0000.		
0020_0000	Causa dell'errore: La variabile (configurata con StartMode) per l'inizializzazione del parametro Output al primo richiamo dell'istruzione non ha un valore REAL valido. Reazione all'errore: Al primo richiamo dell'istruzione nel parametro Output viene emesso il valore di uscita sostitutivo configurato nella variabile ErrorMode. Ai richiami successivi Filter_PT1 calcola il valore di uscita basandosi su questo valore di uscita sostitutivo. Rimedio:		

ErrorBits (DW#16#...)	Descrizione
	Accertarsi che la variabile utilizzata per l'inizializzazione del parametro Output sia un valore REAL valido ($\neq$ NaN ad es. 16#7FFF_FFFF). Se è impostato Reset = FALSE, l'inizializzazione diventa effettiva al primo richiamo dell'istruzione dopo la commutazione dello stato di funzionamento della CPU da STOP a RUN. La variabile utilizzata per l'inizializzazione del parametro Output è in funzione di StartMode: - StartMode = 1: SubstituteOutput - StartMode = 2: Output - StartMode = 4: Il prodotto di Input e Gain
0040_0000	Causa dell'errore: La variabile CycleTime.Value durante l'esecuzione del calcolo del valore di uscita (Reset = FALSE)) non ha un valore valido. Reazione all'errore: Nel parametro Output viene emesso il valore di uscita sostitutivo configurato nella variabile ErrorMode. Rimedio: Accertarsi che siano soddisfatte le seguenti condizioni: $0.0 < \text{CycleTime.Value} \leq 3.402823\text{e}+38$ - CycleTime.Value è un valore REAL valido ($\neq$ NaN ad es. 16#7FFF_FFFF) Informazione supplementare: Per il calcolo automatico del valore della variabile CycleTime.Value, impostare la variabile CycleTime.EnableMeasurement su TRUE.

10.10 Filter_PT2

10.10.1 Compatibilità con CPU e FW

La tabella seguente mostra quali versioni di Filter_PT2 sono compatibili con le varie CPU:

CPU	FW	Filter_PT2
S7-1200	da V4.2	V1.0
S7-1200 G2	da V1.0	V1.0
CPU basate su S7-1500	da V2.0	V1.0

10.10.2 Descrizione del filtro_PT2

Descrizione

L'istruzione Filter_PT2 è un elemento di trasmissione proporzionale con un ritardo di secondo ordine, denominato anche regolatore PT2.

Filter_PT2 può essere impiegato per i seguenti scopi:

- Filtro passabasso per attenuare le percentuali ad alta frequenza di un segnale, quali ad es. la rumorosità.
- Elemento di ritardo per il livellamento di gradini dei segnali, ad es. del setpoint o del valore di uscita di un regolatore.
- Blocco di simulazione del processo, per la realizzazione di un circuito di regolazione chiuso all'interno della CPU. Ciò consente ad esempio di testare il regolatore prima della messa in servizio.

Possono essere preimpostati i seguenti parametri del filtro:

- Guadagno proporzionale (Gain)
- Costante di tempo (TimeConstant)
- Attenuazione (Damping)

NOTA

Differenze tra un regolatore PT2 a tempo continuo e un Filter_PT2

Poiché Filter_PT2 viene eseguito in un programma PLC, Filter_PT2 è una realizzazione a tempo discreto di un regolatore PT2. I sistemi a tempo discreto non possono presentare le stesse proprietà del modello corrispondente a tempo continuo. Tuttavia, in funzione del tempo di ciclo, sistemi a tempo discreto sono in grado di emulare con una buona precisione un sistema a tempo continuo: Più breve e costante è il tempo di ciclo, minore sarà la divergenza tra le proprietà del Filter_PT2 e quelle di un regolatore PT2 a tempo continuo. Le proprietà di un regolatore PT2-a tempo continuo, sono la funzione di trasmissione, il comportamento temporale e il comportamento della frequenza illustrati nel seguito.

Per una buona riproduzione del comportamento della frequenza si raccomanda un tempo ciclo massimo pari ad un decimo della durata più breve dei periodi del segnale di ingresso. Ad esempio un segnale con periodi di frequenza che raggiungono 50 Hz, ha una durata periodo più breve, pari a 20 ms. Per realizzare una buona riproduzione del comportamento della frequenza si raccomanda, per il presente esempio, un tempo di ciclo massimo di 2 ms.

Funzione di trasmissione di un regolatore PT2

La seguente formula mostra la funzione di trasmissione di un regolatore PT2, dove s è uguale a Laplace operator:

$$G(s) = \frac{\text{Output}(s)}{\text{Input}(s)} = \frac{\text{Gain}}{1 + 2 \cdot \text{Damping} \cdot \text{TimeConstant} \cdot s + \text{TimeConstant}^2 \cdot s^2}$$

Se $\text{Damping} \geq 1$, il regolatore PT2 può essere descritto come due regolatori PT1 collegati in serie:

$$G(s) = \frac{\text{Output}(s)}{\text{Input}(s)} = \frac{\text{Gain}}{(1 + \text{Lag}_1 \cdot s) \cdot (1 + \text{Lag}_2 \cdot s)}$$

Le costanti del tempo di ritardo dei due regolatori PT1 collegati in serie, vengono calcolate come segue:

$$\text{Lag}_{1/2} = \text{TimeConstant} \cdot \left(\text{Damping} \pm \sqrt{\text{Damping}^2 - 1} \right)$$

Comportamento temporale di un regolatore PT2

La risposta a gradino è la reazione del valore di uscita ad un gradino del valore di ingresso.

La risposta a gradino per un gradino del valore di ingresso da 0 a ΔInput può essere calcolata con le seguenti formule:

Per $\text{Damping} < 1$ vale la seguente formula:

$$\text{Output}(t) = \Delta\text{Input} \cdot \text{Gain} \cdot \left(1 - e^{-\frac{\text{Damping}}{\text{TimeConstant}} \cdot t} \cdot \left[\cos\left(\frac{\sqrt{1 - \text{Damping}^2}}{\text{TimeConstant}} \cdot t\right) + \frac{\text{Damping}}{\sqrt{1 - \text{Damping}^2}} \cdot \sin\left(\frac{\sqrt{1 - \text{Damping}^2}}{\text{TimeConstant}} \cdot t\right) \right] \right)$$

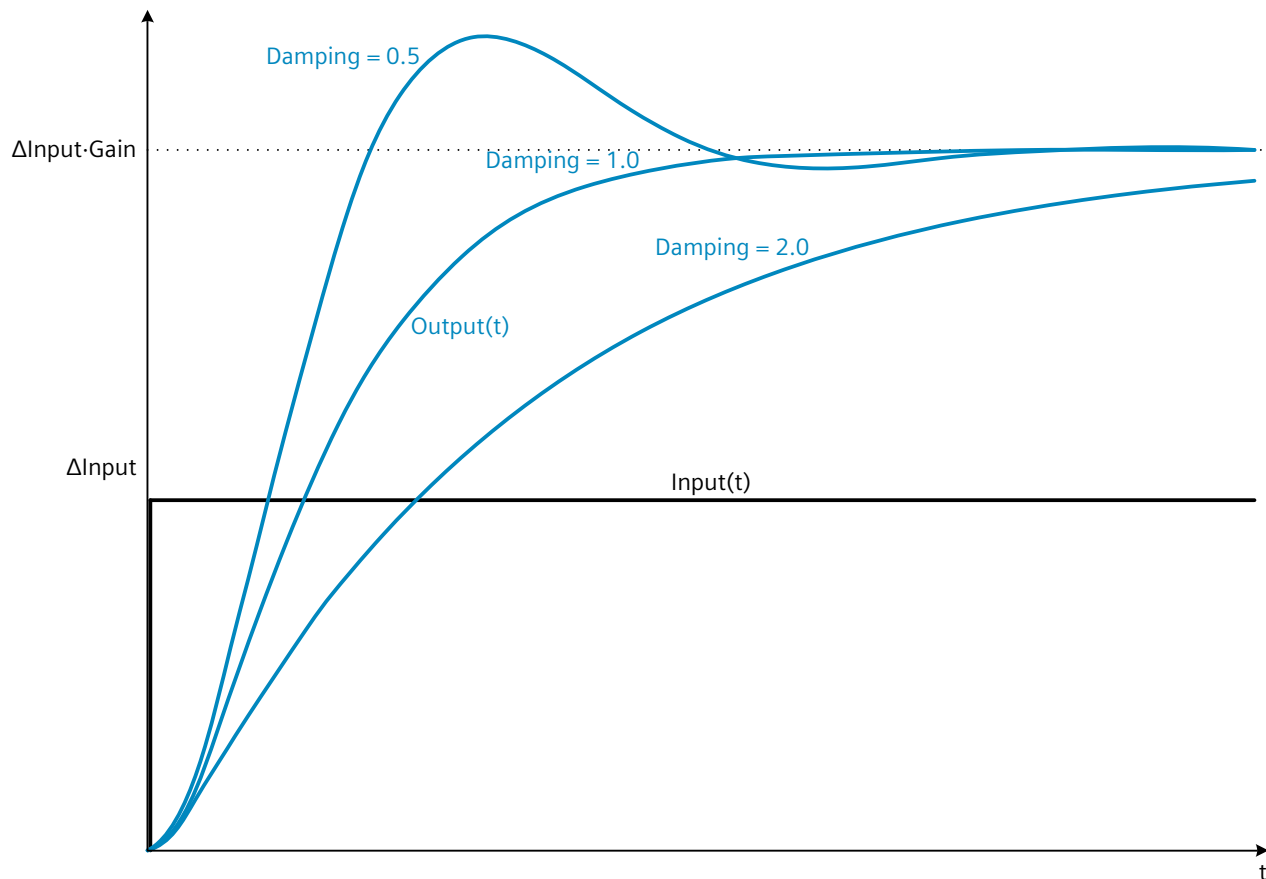
Per $\text{Damping} = 1$ vale la seguente formula:

$$\text{Output}(t) = \Delta\text{Input} \cdot \text{Gain} \cdot \left(1 - e^{-\frac{t}{\text{TimeConstant}}} - \frac{t}{\text{TimeConstant}} \cdot e^{-\frac{t}{\text{TimeConstant}}} \right)$$

Per $\text{Damping} > 1$ vale la seguente formula con Lag_1 e Lag_2 calcolati in precedenza:

$$\text{Output}(t) = \Delta\text{Input} \cdot \text{Gain} \cdot \left(1 - \frac{\text{Lag}_1}{\text{Lag}_1 - \text{Lag}_2} \cdot e^{-\frac{t}{\text{Lag}_1}} + \frac{\text{Lag}_2}{\text{Lag}_1 - \text{Lag}_2} \cdot e^{-\frac{t}{\text{Lag}_2}} \right)$$

La figura seguente mostra la risposta a gradino di un regolatore PT2 con diversi valori per l'attenuazione:



Comportamento della frequenza di un regolatore PT2

Il comportamento della frequenza di un elemento di trasmissione viene descritto dalla caratteristica di ampiezza e dalla caratteristica di fase.

La caratteristica di ampiezza descrive il guadagno di un segnale tramite l'elemento di trasmissione in funzione della frequenza di circuito del segnale.

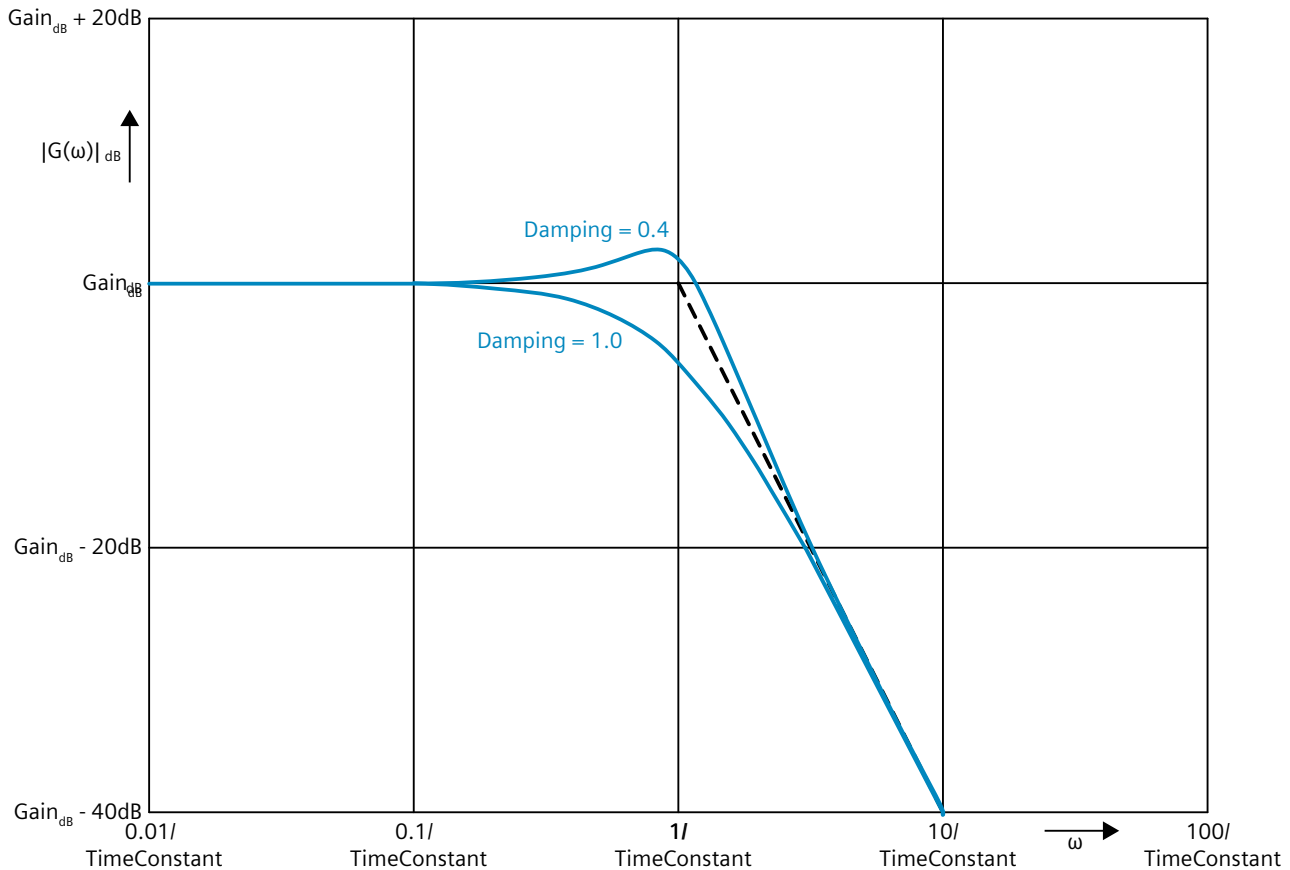
La seguente equazione descrive la caratteristica di ampiezza di un regolatore PT2:

$$|G(\omega)| = \frac{\text{Gain}}{\sqrt{(1 - [\omega \cdot \text{TimeConstant}]^2)^2 + (\omega \cdot 2 \cdot \text{Damping} \cdot \text{TimeConstant})^2}}$$

$|G(\omega)|$ Guadagno di un segnale in funzione della frequenza di circuito

ω Frequenza di circuito

La seguente figura illustra la caratteristica di ampiezza di un regolatore PT2 con un'attenuazione di 0,4 e 1,0:



NOTA

Se $\text{Damping} < 1/\sqrt{2}$, nella caratteristica di ampiezza appare un'esaltazione di risonanza.

La caratteristica di fase descrive l'offset di fase di un segnale tramite l'elemento di trasmissione in funzione della frequenza di circuito del segnale.

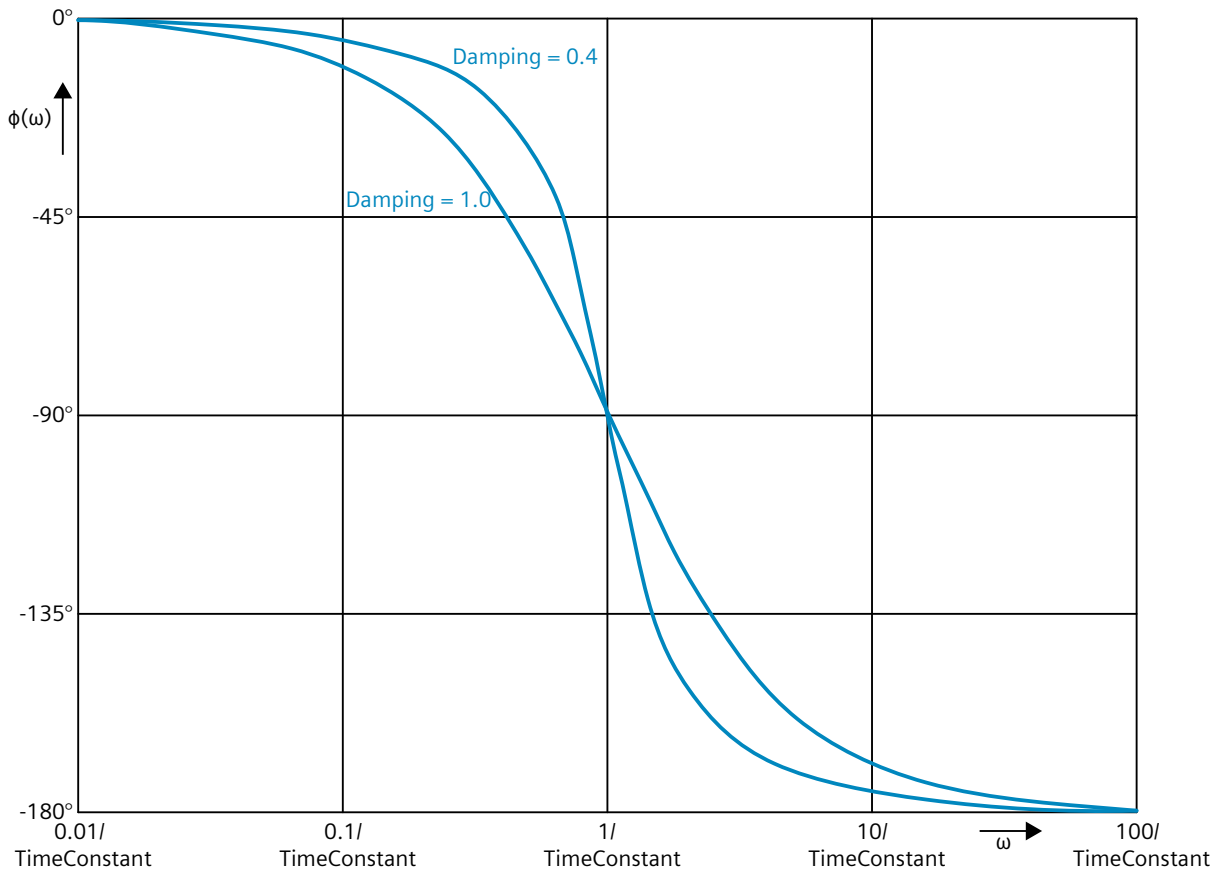
La seguente equazione descrive la caratteristica di fase di un regolatore PT2:

$$\phi(\omega) = -\tan^{-1} \left(\frac{\omega \cdot 2 \cdot \text{Damping} \cdot \text{TimeConstant}}{1 - (\omega \cdot \text{TimeConstant})^2} \right)$$

$\phi(\omega)$ Caratteristica di fase in funzione della frequenza di circuito

ω Frequenza di circuito

La seguente figura illustra la caratteristica di fase di un regolatore PT2:



Richiamo

In un OB o FC Filter_PT2 viene richiamato come DB di un'istanza singola. In un FB Filter_PT2 può essere richiamato sia come DB di istanza singola, sia come DB di multiistanza e come DB di istanza di un parametro.

Al richiamo dell'istruzione non viene creato nessun oggetto tecnologico. Non è disponibile un'interfaccia utente per la parametrizzazione e la messa in servizio. La parametrizzazione di Filter_PT2 avviene direttamente nel DB di istanza e la messa in servizio di Filter_PT2 viene eseguita in una tabella di controllo del programma utente nella CPU oppure in HMI.

Avvio

Le variabili nel campo statico di Filter_PT2 non sono a ritenzione. Queste variabili vengono inizializzate con i valori di avvio dopo ogni commutazione dello stato di funzionamento della CPU da STOP a RUN. Per mantenere i valori attuali eventualmente modificati nel funzionamento online anche dopo la commutazione dello stato di funzionamento della CPU, salvare questi valori nei valori di avvio del blocco dati.

Nella variabile StartMode si definisce il valore di inizializzazione per il parametro Output.

Il valore di inizializzazione viene emesso al primo richiamo di Filter_PT2 dopo la

- commutazione dello stato di funzionamento della CPU

oppure

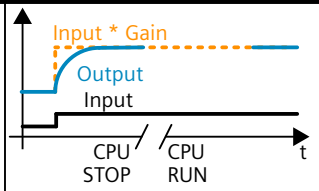
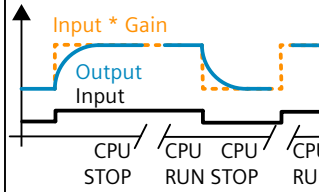
- Dopo l'esecuzione della funzione "Carica valori di avvio come valori attuali" (solo con l'opzione "Tutti i valori", non con l'opzione "Solo valori di impostazione")

nel parametro Output.

Ai richiami successivi Filter_PT2 calcola il valore di uscita partendo da questo valore di inizializzazione e basandosi sul valore di ingresso e sulla configurazione del filtro.

La tabella seguente mostra l'interdipendenza tra la variabile StartMode e il parametro Output. I valori nella colonna Output vengono emessi, dopo la commutazione dello stato di funzionamento della CPU, nel parametro Output.

StartMode	Output	Esempio
0	Valore del parametro Input	
1	Valore del parametro SubstituteOutput	
2	Rimane invariato Il parametro Output è a ritenzione. Impostazione standard Viene utilizzato da StartMode non nel campo 0...4	

StartMode	Output	Esempio
3	0.0	
4	Valore del prodotto Input * Gain	

Inoltre, per tutti i valori della variabile StartMode vale quanto segue:

- Il valore di inizializzazione viene limitato al campo valori del tipo di dati REAL. Solo a questo punto viene emesso il valore di inizializzazione nel parametro Output.
- Se il valore di inizializzazione non è un valore REAL valido, viene emesso il valore di uscita sostitutivo nel parametro Output. Il valore di uscita sostitutivo viene configurato con la variabile ErrorMode. Il valore di uscita sostitutivo viene limitato al campo valori del tipo di dati REAL prima di essere emesso nel parametro Output. Se il valore di uscita sostitutivo è a sua volta un valore REAL non valido, 0.0 viene emesso nel parametro Output. Basandosi su questo valore di uscita sostitutivo, l'istruzione calcola il valore di uscita ai richiami successivi.
- Soltanto se al primo richiamo dell'istruzione è impostato il parametro Reset = FALSE e al contempo non è presente alcun errore con la segnalazione ErrorBits $\geq$ 16#0002_0000, la variabile StartMode è attiva. Se è impostato il parametro Reset = TRUE, il valore del parametro SubstituteOutput viene emesso nel parametro Output. In presenza di un errore con segnalazione ErrorBits $\geq$ 16#0002_0000, nel parametro Output viene emesso il valore di uscita sostitutivo configurato nella variabile ErrorMode.

Comportamento in caso di errori

L'istruzione Filter_PT2 individua svariati errori che possono verificarsi durante il calcolo del valore di uscita. Il risultato di questo calcolo può essere emesso sull'uscita anche se è presente un errore. Se un errore impedisce il calcolo corretto del valore di uscita, sull'uscita viene emesso un valore di uscita sostitutivo.

Se un errore impedisce di calcolare correttamente il valore di uscita, definire nella variabile ErrorMode il valore di uscita sostitutivo come indicato nel seguito.

La tabella seguente mostra l'interdipendenza tra il valore della variabile ErrorMode e il valore di uscita sostitutivo che Filter_PT2 emette nel parametro Output:

ErrorMode	Output
0	Valore del parametro Input
1	Valore del parametro SubstituteOutput

ErrorMode	Output
2	Ultimo valore di uscita valido del filtro 0.0, in assenza di un valore di uscita valido del filtro Impostazione standard Viene utilizzato se ErrorMode non si trova nel campo 0...4
3	0.0
4	Valore del prodotto Input * Gain

Inoltre, per tutti i valori della variabile ErrorMode vale quanto segue:

- Se il valore di uscita sostitutivo non è un valore REAL valido, come valore di uscita viene emesso 0.0.
- Il valore di uscita sostitutivo viene limitato al campo valori del tipo di dati REAL. Solo in quel momento viene emesso il valore di uscita sostitutivo nel parametro Output.
- La variabile ErrorMode è effettiva solo se è impostato il parametro Reset = FALSE. Se è impostato il parametro Reset = TRUE, il valore del parametro SubstituteOutput viene emesso nel parametro Output.

Il parametro Error indica se attualmente è presente un errore. Se l'errore non è più presente, Error viene impostato su FALSE. Il parametro ErrorBits indica quali errori si sono verificati. ErrorBits è a ritenzione e viene resettato soltanto da un fronte di salita nel parametro Reset o ErrorAck.

10.10.3 Modo di funzionamento Filter_PT2

Comportamento in caso di reset

Filter_PT2 si comporta, in funzione del parametro Reset, nel seguente modo:

- Se è impostato il parametro Reset = TRUE, il valore del parametro SubstituteOutput viene emesso nel parametro Output.
- Se è impostato il parametro Reset = FALSE, il valore emesso nel parametro Output viene calcolato con l'algoritmo del filtro.
- Se il parametro Reset di FALSE è impostato su TRUE, il valore del parametro Output commuta direttamente sul valore del parametro SubstituteOutput. Questa commutazione può essere repentina. Inoltre viene resettato il parametro ErrorBits.
- Se il parametro Reset di TRUE viene impostato su FALSE, l'algoritmo del filtro viene impostato in modo da consentire che la commutazione avvenga bumpless.

Comportamento di abilitazione EN/ENO

Se una delle seguenti condizioni è soddisfatta, l'uscita di abilitazione ENO viene impostata su FALSE:

- L'ingresso di abilitazione EN è impostato su TRUE e il parametro Output è definito da un valore di uscita sostitutivo in presenza dei messaggi di errore ErrorBits $\geq 16\#0001\_0000$.
- Ingresso di abilitazione EN impostato su FALSE.

Diversamente l'uscita di abilitazione ENO viene impostata su TRUE:

Misura automatica del tempo di ciclo

Per calcolare il valore di uscita, Filter_PT2 impiega il tempo trascorso dall'ultimo richiamo di Filter_PT2.

Per default il tempo di ciclo viene misurato automaticamente e a partire dal secondo richiamo viene emesso nella variabile CycleTime.Value. Filter_PT2 misura il tempo di ciclo a ogni richiamo dell'istruzione e perciò non è utilizzabile in cicli di richiamo equidistanti, ad es. nell'OB1.

Osservare che i richiami condizionati dell'istruzione, i punti di arresto o il caricamento di istantanee come valori attuali, protraggono il tempo di ciclo nella misura automatica. Un tempo di ciclo troppo lungo viene individuato come errore con la segnalazione di errore ErrorBits = 16#0008_0000.

Se la misura del tempo di ciclo non fornisce un risultato valido, Filter_PT2 calcola il valore di uscita attuale con l'ultimo tempo di ciclo valido. Inoltre Filter_PT2 emette un messaggio di errore nel parametro ErrorBits.

Se si disattiva la misura automatica del tempo di ciclo impostando la variabile CycleTime.EnableMeasurement = FALSE, il tempo di ciclo deve essere calcolato manualmente nella variabile CycleTime.Value. Filter_PT2 verifica la validità della variabile CycleTime.Value a ogni richiamo.

Misura automatica del tempo di ciclo con punti di arresto

Se tra due richiami di Filter_PT2 sono attivi dei punti di arresto, la misura automatica del tempo di ciclo fornisce il tempo effettivamente trascorso tra due richiami. Se è attivo un punto di arresto, la CPU è nello stato di funzionamento ALT.

NOTA

I punti di arresto attivi prolungano l'intervallo tra due richiami di Filter_PT2.

Con un intervallo di tempo più lungo tra due richiami aumenta anche la variazione del valore di uscita nel parametro Output.

Può inoltre accadere che, a causa di intervalli più lunghi, la condizione $\text{TimeConstant} \geq \text{CycleTime.Value}/2$ venga violata e che venga pertanto rilevato un errore con la segnalazione di errore ErrorBits = 16#0008_0000.

Se non si necessita del calcolo del valore di uscita basato sul tempo effettivo con punti di arresto, eseguire i passi seguenti:

- Disattivare la misura automatica del tempo di ciclo impostando la variabile CycleTime.EnableMeasurement = FALSE.
- Inserire il tempo di ciclo manualmente nella variabile CycleTime.Value.

10.10.4 Parametro di ingresso Filter_PT2

Parametri	Tipo di dati	Preassegnazione	Descrizione
Input	REAL	0.0	Valore di ingresso
SubstituteOutput	REAL	0.0	SubstituteOutput viene impiegato come valore di uscita sostitutivo se - Reset = TRUE oppure - se un errore con messaggio di errore ErrorBits $\geq 16\#0001\_0000$ impedisce il calcolo corretto del valore di uscita e ErrorMode è configurato sul valore 1.
ErrorAck	BOOL	FALSE	Cancella i messaggi di errore Fronte FALSE -> TRUE ErrorBits viene resettato.
Reset	BOOL	FALSE	Esegue il riavvio dell'istruzione - Fronte FALSE -> TRUE ErrorBits viene resettato. - Fintantoché Reset è impostato su TRUE, il valore di uscita sostitutivo SubstituteOutput viene emesso sull'uscita. - Finché Reset è impostato su FALSE, il calcolo del valore di uscita viene eseguito.

10.10.5 Parametro di uscita Filter_PT2

Parametro	Tipo di dati	Preassegnazione	Descrizione
Output	REAL	0.0	Valore di uscita Il valore di uscita è a ritenzione.
ErrorBits	DWORD	DW#16#0	Il parametro ErrorBits (Pagina 569) mostra quali messaggi di errore sono presenti. ErrorBits è a ritenzione e viene resettato in presenza di un fronte di salita in Reset o ErrorAck.
Error	BOOL	FALSE	Se Error è impostato su TRUE si è attualmente in presenza di almeno un errore.

10.10.6 Variabili statiche Filter_PT2

Variabile	Tipo di dati	Preassegnazione	Descrizione
Gain	REAL	1.0	Guadagno proporzionale
TimeConstant	REAL	25.0	Costante di tempo in secondi Campo di valori consentito: TimeConstant $\geq$ CycleTime.Value/2
Damping	REAL	1.0	Attenuazione Campo di valori consentito: Damping > 0.0
ErrorMode	INT	2	Selezione del valore di uscita sostitutivo in caso di errori 0 = Input 1 = SubstituteOutput 2 = ultimo valore di uscita valido del filtro 3 = 0.0 4 = Input * Gain Campo di valori ammesso: 0 ... 4

Variabile	Tipo di dati	Preassegnazione	Descrizione
StartMode	INT	2	Selezione del valore di uscita per il primo richiamo dell'istruzione 0 = Input 1 = SubstituteOutput 2 = ultimo valore di uscita 3 = 0.0 4 = Input * Gain Campo di valori ammesso: 0 ... 4
CycleTime	AuxFct_CycleTime	-	Dati del tempo di ciclo
CycleTime.Value	REAL	0.1	Tempo di ciclo in secondi (intervallo tra due richiami) Campo di valori consentito: CycleTime.Value > 0.0
CycleTime.EnableMeasurement	BOOL	TRUE	Misura automatica del tempo di ciclo - FALSE = disattivato - TRUE = attivato

10.10.7 Parametro ErrorBits

Se si verificano diversi errori contemporaneamente, i valori di ErrorBits vengono visualizzati in un'addizione binaria. Se viene visualizzato ad es. ErrorBits = 16#0000_0003 sono presenti contemporaneamente gli errori 16#0000_0001 e 16#0000_0002.

Nel caso di Filter_PT2 gli errori emessi nel parametro ErrorBits sono suddivisi in due categorie:

- Errore con messaggio di errore ErrorBits < 16#0001_0000
- Errore con messaggio di errore ErrorBits ≥ 16#0001_0000

Errore con messaggio di errore ErrorBits < 16#0001_0000

In presenza di uno o più errori con i messaggi di errore ErrorBits < 16#0001_0000, Filter_PT2 ha la seguente reazione:

- Nonostante questo errore, il valore di uscita viene rilevato come indicato nel seguito:
 - Se Reset = FALSE, , calcolo del valore di uscita con l'algoritmo del filtro
 - Se Reset = TRUE, emissione di SubstituteOutput
- Il parametro di uscita Error viene impostato.
- L'uscita di abilitazione ENO non viene modificata.

Il parametro di uscita Error viene eliminato non appena tutti gli errori sono stati eliminati.

ErrorBits (DW#16#...)	Descrizione
0000_0000	Nessun errore
0000_0001	Causa degli errore e reazione agli errori: Il parametro Output è stato limitato a -3.402823e+38 oppure +3.402823e+38. Rimedio:

ErrorBits (DW#16#...)	Descrizione
	Se sull'uscita viene emesso il valore stabilito dalla funzione del filtro (Reset = FALSE e ErrorBits < 16#0001_0000), controllare le seguenti variabili utilizzate nel calcolo del filtro: • Input • Gain • TimeConstant • Damping • CycleTime.Value Se ErrorBits ≥ 16#0001_0000 e Reset = FALSE, il valore di uscita sostitutivo viene limitato al momento della relativa emissione. Successivamente, a seconda del valore impostato nella variabile ErrorMode, controllare i seguenti parametri: • Input • SubstituteOutput • Il prodotto di Input e Gain Se Reset = TRUE, controllare il parametro SubstituteOutput.
0000_0002	Causa dell'errore: La misura del tempo di ciclo fornisce un valore non valido mentre viene eseguito il calcolo del valore di uscita (Reset = FALSE). Reazione all'errore: Se è già stato misurato un valore valido per il tempo di ciclo, Filter_PT2 calcola il valore di uscita a partire da quest'ultimo valore della variabile CycleTime.Value. Se non è ancora stato misurato un valore valido per il tempo di ciclo, Filter_PT2 continua ad emettere, nel parametro Output, il valore di uscita configurato con la variabile StartMode.

Errore con messaggio di errore ErrorBits ≥ 16#0001_0000

In presenza di uno o più errori con i messaggi di errore ErrorBits ≥ 16#0001_0000, Filter_PT2 ha la seguente reazione:

- Il valore di uscita non può essere rilevato come previsto. Viene invece emesso il valore di uscita sostitutivo.
- Il parametro di uscita Error viene impostato.
- L'uscita di abilitazione ENO viene impostata su FALSE.

Non appena tutti gli errori segnalati con i messaggi ErrorBits ≥ 16#0001_0000 sono stati eliminati, Filter_PT2 ha le seguenti reazioni:

- Il valore di uscita viene rilevato come indicato nel seguito:
 - Se Reset = FALSE, calcolo del valore di uscita con l'algoritmo del filtro
 - Se Reset = TRUE, emissione di SubstituteOutput
- L'uscita di abilitazione ENO viene impostata su TRUE.

Il parametro di uscita Error viene eliminato non appena tutti gli errori sono stati eliminati.

ErrorBits (DW#16#...)	Descrizione		
0001_0000	Causa dell'errore: Il parametro SubstituteOutput o un'altra variabile, utilizzata come valore di uscita non ha un valore REAL valido.		
	Reazione all'errore: L'uscita viene impostata su 0.0.		
	Rimedio: Accertarsi che la variabile, utilizzata come valore di uscita sia un valore REALvalido (≠NaN ad es. 16#7FFF_FFFF). La variabile utilizzata come valore di uscita è in funzione di Reset e ErrorMode:		
	Reset	ErrorMode	Valore di uscita
	FALSE	0	Input
	FALSE	1	SubstituteOutput
FALSE	4	Il prodotto di Input e Gain	
	TRUE	-	SubstituteOutput
	0002_0000		
Causa dell'errore: Il parametro Input non ha un valore REAL valido mentre viene eseguito il calcolo del valore di uscita (Reset = FALSE).			
Reazione all'errore: Nel parametro Output viene emesso il valore di uscita sostitutivo configurato nella variabile ErrorMode. Con ErrorMode = 0, 0.0 viene impiegato come valore di uscita.			
Rimedio: Accertarsi che il parametro Input sia un valore REAL valido (≠NaN, ad es. 16#7FFF_FFFF).			
0004_0000	Causa dell'errore: Il calcolo del valore di uscita fornisce un valore REAL non valido per il parametro Output.		
	Reazione all'errore: Nel parametro Output viene emesso il valore di uscita sostitutivo configurato nella variabile ErrorMode.		
0008_0000	Rimedio: Controllare tutte le variabili comprese nel calcolo del valore di uscita:		
	• Input • Gain • TimeConstant • Damping • CycleTime.Value Queste variabili hanno i seguenti valori validi. Con questa combinazione delle variabili, il calcolo del valore di uscita non riesce.		
0008_0000	Causa dell'errore: La variabile Gain, TimeConstant o Damping non ha un valore valido mentre viene eseguito il calcolo del valore di uscita (Reset = FALSE).		
	Reazione all'errore: Nel parametro Output viene emesso il valore di uscita sostitutivo configurato nella variabile ErrorMode.		
0008_0000	Rimedio: Accertarsi che per i valori delle variabili Gain, TimeConstant e Damping siano soddisfatte le seguenti condizioni:		
	• -3.402823e+38 ≤ Gain ≤ 3.402823e+38 • CycleTime.Value/2 ≤ TimeConstant ≤ 3.402823e+38 • 0.0 < Damping ≤ 3.402823e+38 • I valori sono valori REAL validi (≠ NaN ad es. 16#7FFF_FFFF) Informazione supplementare:		

ErrorBits (DW#16#...)	Descrizione
	Osservare che nei seguenti casi può accadere che la condizione $\text{CycleTime.Value}/2 \leq \text{TimeConstant}$ venga violata: • L'intervallo tra due richiami di Filter_PT2 supera $2 * \text{TimeConstant}$, ad es. a causa di richiami condizionati durante l'esecuzione del programma o di punti di arresto attivi. • Un'istanza del DB di istanza Filter_PT2 viene caricata come valore attuale nella CPU e la generazione di quest'istanza presenta una datazione superiore a $2 * \text{TimeConstant}$. In questi casi viene, durante la misura automatica, individuato un errore con la segnalazione di errore ErrorBits = 16#0008_0000.
0020_0000	Causa dell'errore: La variabile (configurata con StartMode) per l'inizializzazione del parametro Output al primo richiamo dell'istruzione non ha un valore REAL valido. Reazione all'errore: Al primo richiamo dell'istruzione nel parametro Output viene emesso il valore di uscita sostitutivo configurato nella variabile ErrorMode. Ai richiami successivi Filter_PT2 calcola il valore di uscita basandosi su questo valore di uscita sostitutivo. Rimedio: Accertarsi che la variabile utilizzata per l'inizializzazione del parametro Output sia un valore REAL valido ($\neq \text{NaN}$ ad es. 16#7FFF_FFFF). Se è impostato Reset = FALSE, l'inizializzazione diventa effettiva al primo richiamo dell'istruzione dopo la commutazione dello stato di funzionamento della CPU da STOP a RUN. La variabile utilizzata per l'inizializzazione del parametro Output è in funzione di StartMode: • StartMode = 1: SubstituteOutput • StartMode = 2: Output • StartMode = 4: il prodotto di Input e Gain
0040_0000	Causa dell'errore: La variabile CycleTime.Value durante l'esecuzione del calcolo del valore di uscita (Reset = FALSE) non ha un valore valido. Reazione all'errore: Nel parametro Output viene emesso il valore di uscita sostitutivo configurato nella variabile ErrorMode. Rimedio: Accertarsi che siano soddisfatte le seguenti condizioni: • $0.0 < \text{CycleTime.Value} \leq 3.402823\text{e}+38$ • CycleTime.Value è un valore REAL valido ($\neq \text{NaN}$ ad es. 16#7FFF_FFFF) Informazione supplementare: Per il calcolo automatico del valore della variabile CycleTime.Value, impostare la variabile CycleTime.EnableMeasurement su TRUE.

10.11 Filter_DT1

10.11.1 Compatibilità con CPU e FW

La tabella seguente indica le CPU compatibili con le varie versioni di Filter_DT1.

CPU	FW	Filter_DT1
S7-1200	da V4.2	V1.0
S7-1200 G2	da V1.0	V1.0
CPU basate su S7-1500	da V2.0	V1.0

10.11.2 Descrizione del filtro_PT1

Descrizione

L'istruzione Filter_DT1 è un regolatore differenziale con un ritardo di prim'ordine, denominato anche regolatore DT1.

Filter_DT1 può essere impiegato per i seguenti scopi:

- Filtro passa-alto per attenuare le percentuali ad bassa frequenza di un segnale.
- Regolatore differenziale per il calcolo della derivazione di un segnale, quale ad esempio la velocità basandosi su valori di posizione.
- Inserzione di un segnale di disturbo per attenuare gli effetti di disturbi misurabili nel processo.

Possono essere preimpostati i seguenti parametri del filtro:

- Tempo derivativo (Td)
- Costante tempo di ritardo (Lag)

NOTA

Differenze tra un regolatore DT1a tempo continuo e un Filter_DT1

Poiché Filter_DT1 viene eseguito in un programma PLC, Filter_DT1 è una realizzazione a tempo discreto di un regolatore DT1. I sistemi a tempo discreto non possono presentare le stesse proprietà del modello corrispondente a tempo continuo. Tuttavia, in funzione del tempo di ciclo, sistemi a tempo discreto sono in grado di emulare con una buona precisione un sistema a tempo continuo: Più breve e costante è il tempo di ciclo, minore sarà la divergenze tra le proprietà del Filter_DT1 e quelle di un regolatore DT1a tempo continuo. Le proprietà di un regolatore DT1-a tempo continuo, sono la funzione di trasmissione, il comportamento temporale e il comportamento della frequenza illustrati nel seguito.

Per una buona riproduzione del comportamento della frequenza si raccomanda un tempo ciclo massimo pari ad un decimo della durata più breve dei periodi del segnale di ingresso. Ad esempio un segnale con periodi di frequenza che raggiungono 50 Hz, ha una durata periodo più breve, pari a 20 ms. Per realizzare una buona riproduzione del comportamento della frequenza si raccomanda, per il presente esempio, un tempo di ciclo massimo di 2 ms.

Funzione di trasmissione di un regolatore DT1

La seguente formula mostra la funzione di trasmissione di un regolatore DT1, dove s è uguale a Laplace operator:

$$G(s) = \frac{\text{Output}(s)}{\text{Input}(s)} = \frac{Td \cdot s}{1 + \text{Lag} \cdot s}$$

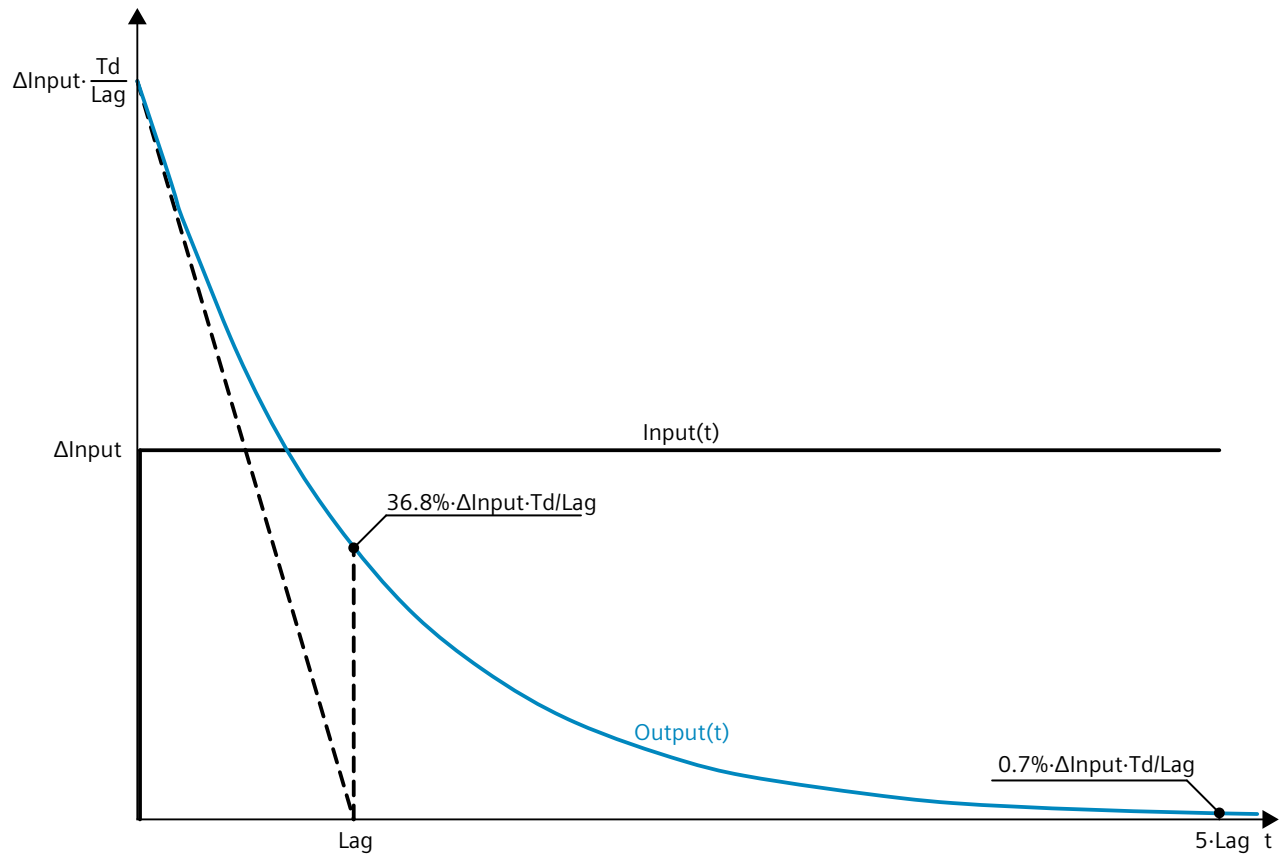
Comportamento temporale di un regolatore DT1

La risposta a gradino è la reazione del valore di uscita ad un gradino del valore di ingresso.

La risposta a gradino per un gradino del valore di ingresso da 0 a ΔInput può essere calcolata con la seguente formula:

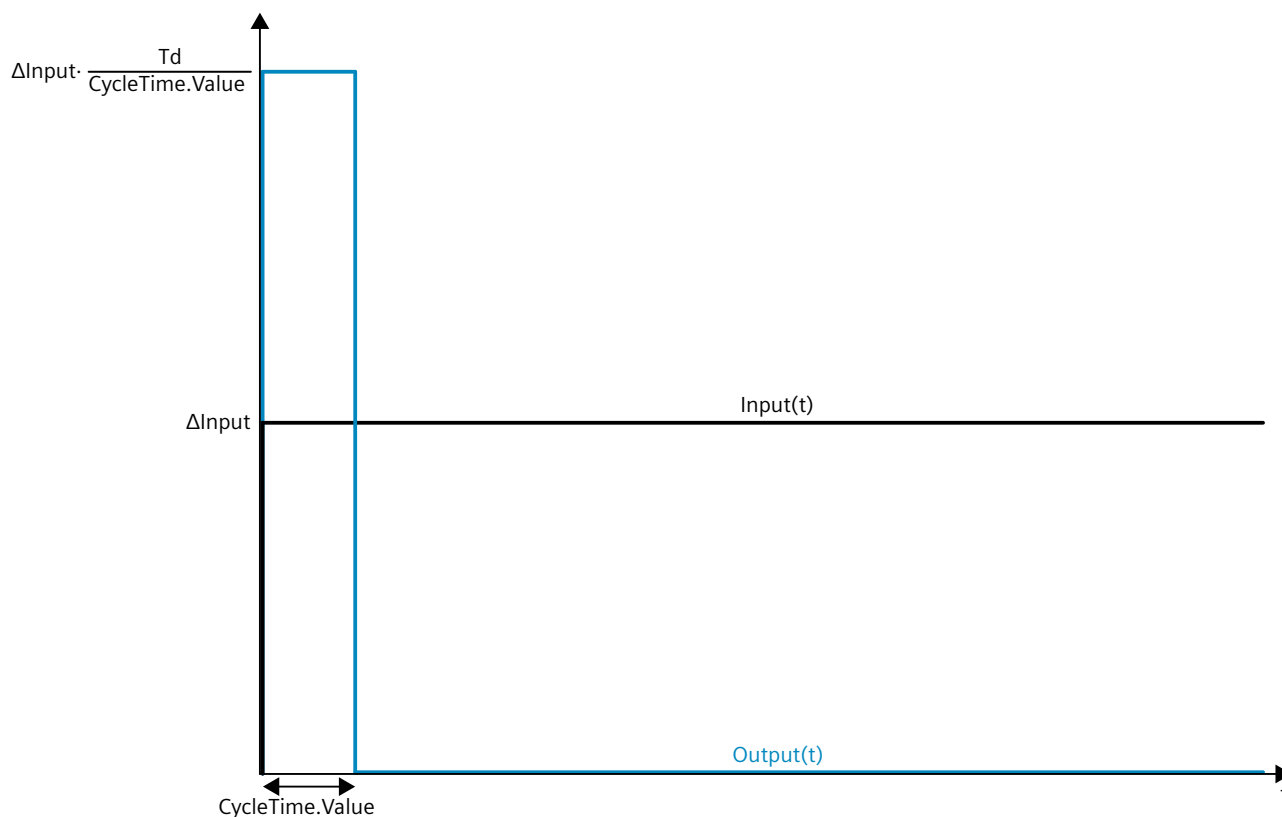
$$\text{Output}(t) = \Delta\text{Input} \cdot \frac{T_d}{\text{Lag}} \cdot e^{-\frac{t}{\text{Lag}}}$$

La figura seguente mostra la risposta a gradino di un regolatore DT1:



Per disattivare il ritardo, impostare il parametro Lag sul valore minimo $\text{CycleTime.Value} / 2$. Le modifiche al valore di ingresso vengono in questo caso moltiplicate per $T_d / \text{CycleTime.Value}$ ed emesse nel parametro Output. Dopo un tempo di ciclo, il valore di uscita è 0.0.

La figura seguente mostra la risposta a gradino nel caso in cui $Lag = CycleTime.Value / 2$:



Comportamento della frequenza di un regolatore DT1

Il comportamento della frequenza di un elemento di trasmissione viene descritto dalla caratteristica di ampiezza e dalla caratteristica di fase.

La caratteristica di ampiezza descrive il guadagno di un segnale tramite l'elemento di trasmissione in funzione della frequenza di circuito del segnale.

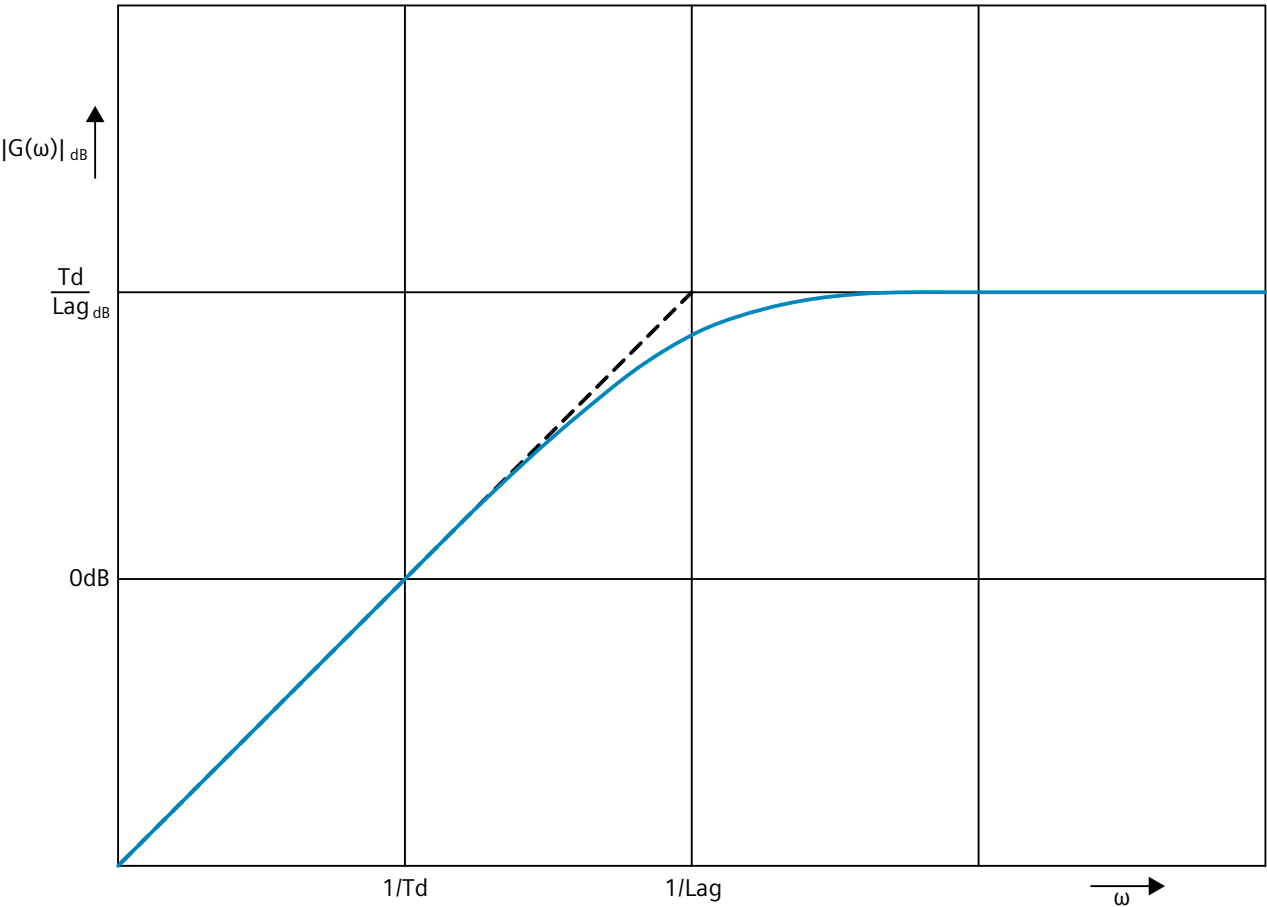
La seguente equazione descrive la caratteristica di ampiezza di un regolatore DT1:

$$|G(\omega)| = \frac{\omega \cdot T_d}{\sqrt{1 + (\omega \cdot Lag)^2}}$$

$|G(\omega)|$ Guadagno di un segnale in funzione della frequenza di circuito

ω Frequenza di circuito

La seguente figura illustra la caratteristica di ampiezza di un regolatore DT1:



La caratteristica di fase descrive l'offset di fase di un segnale tramite l'elemento di trasmissione in funzione della frequenza di circuito del segnale.

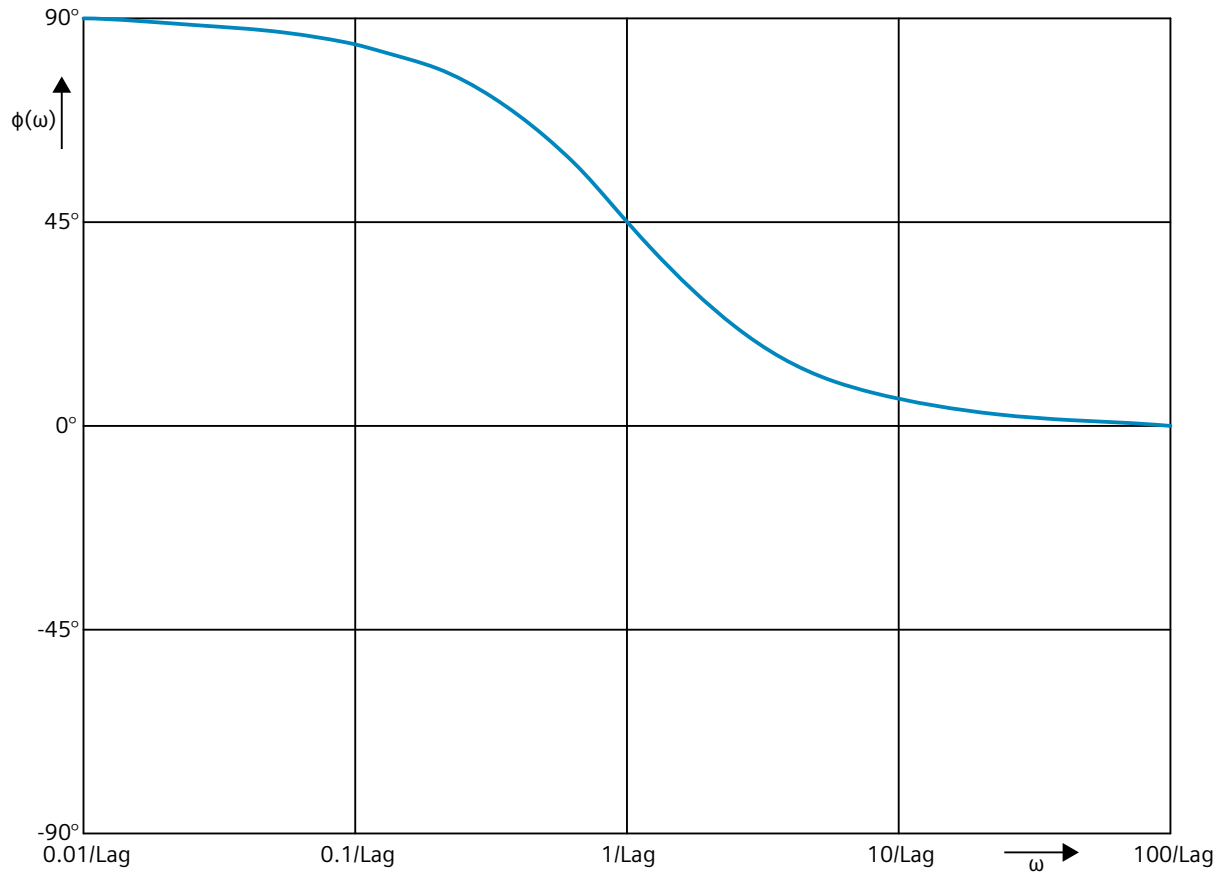
La seguente equazione descrive la caratteristica di fase di un regolatore DT1:

$$\varphi(\omega) = \tan^{-1}(\omega \cdot \text{Lag})$$

$\varphi(\omega)$ Caratteristica di fase in funzione della frequenza di circuito

ω Frequenza di circuito

La seguente figura illustra la caratteristica di fase di un regolatore DT1:



Richiamo

In un OB o FC Filter_DT1 viene richiamato come DB di un'istanza singola. In un FB Filter_DT1 può essere richiamato sia come DB di istanza singola, sia come DB di multiistanza e come DB di istanza di un parametro.

Al richiamo dell'istruzione non viene creato nessun oggetto tecnologico. Non è disponibile un'interfaccia utente per la parametrizzazione e la messa in servizio. La parametrizzazione di Filter_DT1 avviene direttamente nel DB di istanza e la messa in servizio di Filter_DT1 viene eseguita in una tabella di controllo del programma utente nella CPU oppure in HMI.

Avvio

Le variabili nel campo statico di Filter_DT1 non sono a ritenzione. Queste variabili vengono inizializzate con i valori di avvio dopo ogni commutazione dello stato di funzionamento della CPU da STOP a RUN. Per mantenere i valori attuali eventualmente modificati nel funzionamento online anche dopo la commutazione dello stato di funzionamento della CPU, salvare questi valori nei valori di avvio del blocco dati.

Nella variabile StartMode si definisce il valore di inizializzazione per il parametro Output.

Il valore di inizializzazione viene emesso al primo richiamo di Filter_DT1 dopo la

- Commutazione dello stato di funzionamento della CPU

oppure

- Dopo l'esecuzione della funzione "Carica valori di avvio come valori attuali" (solo con l'opzione "Tutti i valori", non con l'opzione "Solo valori di impostazione")

nel parametro Output.

Ai richiami successivi Filter_DT1 calcola il valore di uscita partendo da questo valore di inizializzazione e basandosi sul valore di ingresso e sulla configurazione del filtro.

La tabella seguente mostra l'interdipendenza tra la variabile StartMode e il parametro Output. I valori nella colonna Output vengono emessi, dopo la commutazione dello stato di funzionamento della CPU, nel parametro Output.

StartMode	Output	Esempio
0	Valore del parametro Input	
1	Valore del parametro SubstituteOutput	
2	Rimane invariato Il parametro Output è a ritenzione. Impostazione standard Viene utilizzato da StartMode non nel campo 0...3	
3	0.0	

Inoltre, per tutti i valori della variabile StartMode vale quanto segue:

- Il valore di inizializzazione viene limitato al campo valori del tipo di dati REAL. Solo a questo punto viene emesso il valore di inizializzazione nel parametro Output.
- Se il valore di inizializzazione non è un valore REAL valido, viene emesso il valore di uscita sostitutivo nel parametro Output. Il valore di uscita sostitutivo viene configurato con la variabile ErrorMode. Il valore di uscita sostitutivo viene limitato al campo valori del tipo di dati REAL prima di essere emesso nel parametro Output. Se il valore di uscita sostitutivo è a sua volta un valore REAL non valido, 0.0 viene emesso nel parametro Output. Basandosi su questo valore di uscita sostitutivo, l'istruzione calcola il valore di uscita ai richiami successivi.
- Soltanto se al primo richiamo dell'istruzione è impostato il parametro Reset = FALSE e al contempo non è presente alcun errore con la segnalazione ErrorBits $\geq 16\#0002\_0000$, la variabile StartMode è attiva. Se è impostato il parametro Reset = TRUE, il valore del parametro SubstituteOutput viene emesso nel parametro Output. In presenza di un errore con segnalazione ErrorBits $\geq 16\#0002\_0000$, nel parametro Output viene emesso il valore di uscita sostitutivo configurato nella variabile ErrorMode.

Comportamento in caso di errori

L'istruzione Filter_DT1 individua svariati errori che possono verificarsi durante il calcolo del valore di uscita. Il risultato di questo calcolo può essere emesso sull'uscita anche se è presente un errore. Se un errore impedisce il calcolo corretto del valore di uscita, sull'uscita viene emesso un valore di uscita sostitutivo.

Se un errore impedisce di calcolare correttamente il valore di uscita, definire nella variabile ErrorMode il valore di uscita sostitutivo come indicato nel seguito.

La tabella seguente mostra l'interdipendenza tra il valore della variabile ErrorMode e il valore di uscita sostitutivo che Filter_DT1 emette nel parametro Output:

ErrorMode	Output
0	Valore del parametro Input
1	Valore del parametro SubstituteOutput
2	Ultimo valore di uscita valido del filtro 0.0, in assenza di un valore di uscita valido del filtro Impostazione standard Viene utilizzato se ErrorMode non si trova nel campo 0...3
3	0.0

Inoltre, per tutti i valori della variabile ErrorMode vale quanto segue:

- Se il valore di uscita sostitutivo non è un valore REAL valido, come valore di uscita viene emesso 0.0.
- Il valore di uscita sostitutivo viene limitato al campo valori del tipo di dati REAL. Solo in quel momento viene emesso il valore di uscita sostitutivo nel parametro Output.
- La variabile ErrorMode è effettiva solo se è impostato il parametro Reset = FALSE. Se è impostato il parametro Reset = TRUE, il valore del parametro SubstituteOutput viene emesso nel parametro Output.

Il parametro Error indica se attualmente è presente un errore. Se l'errore non è più presente, Error viene impostato su FALSE. Il parametro ErrorBits indica quali errori si sono verificati. ErrorBits è a ritenzione e viene resettato soltanto da un fronte di salita nel parametro Reset o ErrorAck.

10.11.3 Modo di funzionamento Filter_DT1

Comportamento in caso di reset

Filter_DT1 si comporta, in funzione del parametro Reset, nel seguente modo:

- Se è impostato il parametro Reset = TRUE, il valore del parametro SubstituteOutput viene emesso nel parametro Output.
- Se è impostato il parametro Reset = FALSE, il valore emesso nel parametro Output viene calcolato con l'algoritmo del filtro.
- Se il parametro Reset di FALSE è impostato su TRUE, il valore del parametro Output commuta direttamente sul valore del parametro SubstituteOutput. Questa commutazione può essere repentina. Inoltre viene resettato il parametro ErrorBits.
- Se il parametro Reset di TRUE viene impostato su FALSE, l'algoritmo del filtro viene impostato in modo da consentire che la commutazione avvenga bumpless.

Comportamento di abilitazione EN/ENO

Se una delle seguenti condizioni è soddisfatta, l'uscita di abilitazione ENO viene impostata su FALSE:

- L'ingresso di abilitazione EN è impostato su TRUE e il parametro Output è definito tramite un valore di uscita sostitutivo in presenza dei messaggi di errore ErrorBits $\geq$ 16#0001_0000
- Ingresso di abilitazione EN impostato su FALSE.

Diversamente l'uscita di abilitazione ENO viene impostata su TRUE:

Misura automatica del tempo di ciclo

Per calcolare il valore di uscita, Filter_DT1 impiega il tempo trascorso dall'ultimo richiamo di Filter_DT1.

Per default il tempo di ciclo viene misurato automaticamente e a partire dal secondo richiamo viene emesso nella variabile CycleTime.Value. Filter_DT1 misura il tempo di ciclo a ogni richiamo dell'istruzione e perciò non è utilizzabile in cicli di richiamo equidistanti, ad es. nell'OB1.

Osservare che i richiami condizionati dell'istruzione, i punti di arresto o il caricamento di istantanee come valori attuali, protraggono il tempo di ciclo nella misura automatica. Un tempo di ciclo troppo lungo viene individuato come errore con la segnalazione di errore ErrorBits = 16#0008_0000.

Se la misura del tempo di ciclo non fornisce un risultato valido, Filter_DT1 calcola il valore di uscita attuale con l'ultimo tempo di ciclo valido. Inoltre Filter_DT1 emette un messaggio di errore nel parametro ErrorBits.

Se si disattiva la misura automatica del tempo di ciclo impostando la variabile `CycleTime.EnableMeasurement = FALSE`, il tempo di ciclo deve essere calcolato manualmente nella variabile `CycleTime.Value`. Filter_DT1 verifica la validità della variabile `CycleTime.Value` a ogni richiamo.

Misura automatica del tempo di ciclo con punti di arresto

Se tra due richiami di Filter_DT1 sono attivi dei punti di arresto, la misura automatica del tempo di ciclo fornisce il tempo effettivamente trascorso tra due richiami. Se è attivo un punto di arresto, la CPU è nello stato di funzionamento ALT.

NOTA

I punti di arresto attivi prolungano l'intervallo tra due richiami di Filter_DT1.

Con un intervallo di tempo più lungo tra due richiami aumenta anche la variazione del valore di uscita nel parametro Output.

Può inoltre accadere che, a causa di intervalli più lunghi, le condizioni

$Lag \geq CycleTime.Value/2$ o $Td \geq CycleTime.Value$ vengano violate. Viene quindi individuato un errore con la segnalazione di errore `ErrorBits = 16#0008_0000`.

Se non si necessita del calcolo del valore di uscita basato sul tempo effettivo con punti di arresto, eseguire i passi seguenti:

- Disattivare la misura automatica del tempo di ciclo impostando la variabile `CycleTime.EnableMeasurement = FALSE`.
- Inserire il tempo di ciclo manualmente nella variabile `CycleTime.Value`.

10.11.4 Parametro di ingresso Filter_DT1

Parametri	Tipo di dati	Preassegnazione	Descrizione
Input	REAL	0.0	Valore di ingresso
SubstituteOutput	REAL	0.0	SubstituteOutput viene impiegato come valore di uscita sostitutivo se • Reset = TRUE oppure • se un errore con messaggio di errore <code>ErrorBits ≥ 16#0001_0000</code> impedisce il calcolo corretto del valore di uscita e <code>ErrorMode</code> è configurato sul valore 1.
ErrorAck	BOOL	FALSE	Cancella i messaggi di errore • Fronte FALSE -> TRUE ErrorBits viene resettato.
Reset	BOOL	FALSE	Esegue il riavvio dell'istruzione • Fronte FALSE -> TRUE ErrorBits viene resettato. • Finché Reset è impostato su TRUE, il valore di uscita sostitutivo SubstituteOutput viene emesso sull'uscita. • Finché Reset è impostato su FALSE, il calcolo del valore di uscita viene eseguito.

10.11.5 Parametro di uscita Filter_DT1

Parametro	Tipo di dati	Preassegnazione	Descrizione
Output	REAL	0.0	Valore di uscita Il valore di uscita è a ritenzione.
ErrorBits	DWORD	DW#16#0	Il parametro ErrorBits (Pagina 583) mostra quali messaggi di errore sono presenti. ErrorBits è a ritenzione e viene resettato in presenza di un fronte di salita in Reset o ErrorAck.
Error	BOOL	FALSE	Se Error è impostato su TRUE si è attualmente in presenza di almeno un errore.

10.11.6 Variabili Filter_DT1 statiche

Variabile	Tipo di dati	Preassegnazione	Descrizione
Td	REAL	25.0	Tempo derivativo in secondi Campo di valori consentito: $Td \geq \text{CycleTime.Value}$
Lag	REAL	5.0	Costante tempo di ritardo in secondi Campo di valori consentito: $Lag \geq \text{CycleTime.Value}/2$
ErrorMode	INT	2	Selezione del valore di uscita sostitutivo in caso di errori 0 = Input 1 = SubstituteOutput 2 = ultimo valore di uscita valido del filtro 3 = 0.0 Campo di valori ammesso: 0 - 3
StartMode	INT	2	Selezione del valore di uscita per il primo richiamo dell'istruzione 0 = Input 1 = SubstituteOutput 2 = ultimo valore di uscita 3 = 0.0 Campo di valori ammesso: 0 - 3
CycleTime	AuxFct_CycleTime	-	Dati del tempo di ciclo
CycleTime.Value	REAL	0.1	Tempo di ciclo in secondi (intervallo tra due richiami) Campo di valori consentito: $\text{CycleTime.Value} > 0.0$
CycleTime.EnableMeasurement	BOOL	TRUE	Misura automatica del tempo di ciclo - FALSE = disattivato - TRUE = attivato

10.11.7 Parametro ErrorBits

Se si verificano diversi errori contemporaneamente, i valori di ErrorBits vengono visualizzati in un'addizione binaria. Se viene visualizzato ad es. ErrorBits = 16#0000_0003 sono presenti contemporaneamente gli errori 16#0000_0001 e 16#0000_0002.

Nel caso di Filter_DT1 gli errori emessi nel parametro ErrorBits sono suddivisi in due categorie:

- Errore con messaggio di errore ErrorBits < 16#0001_0000
- Errore con messaggio di errore ErrorBits ≥ 16#0001_0000

Errore con messaggio di errore ErrorBits < 16#0001_0000

In presenza di uno o più errori con i messaggi di errore ErrorBits < 16#0001_0000, Filter_DT1 ha la seguente reazione:

- Nonostante questo errore, il valore di uscita viene rilevato come indicato nel seguito:
 - Se Reset = FALSE, calcolo del valore di uscita con l'algoritmo del filtro
 - Se Reset = TRUE, emissione di SubstituteOutput
- Il parametro di uscita Error viene impostato.
- L'uscita di abilitazione ENO non viene modificata.

Il parametro di uscita Error viene eliminato non appena tutti gli errori sono stati eliminati.

ErrorBits (DW#16#...)	Descrizione
0000_0000	Nessun errore
0000_0001	Causa degli errore e reazione agli errori: Il parametro Output è stato limitato a -3.402823e+38 oppure +3.402823e+38. Rimedio: Se sull'uscita viene emesso il valore stabilito dalla funzione del filtro (Reset = FALSE e ErrorBits < 16#0001_0000), controllare le seguenti variabili utilizzate nel calcolo del filtro: • Input • Td • Lag • CycleTime.Value Se ErrorBits ≥ 16#0001_0000 e Reset = FALSE, il valore di uscita sostitutivo viene limitato al momento della relativa emissione. Successivamente, a seconda del valore impostato nella variabile ErrorMode:, controllare i seguenti parametri • Input • SubstituteOutput Se Reset = TRUE, controllare il parametro SubstituteOutput.
0000_0002	Causa dell'errore: La misura del tempo di ciclo fornisce un valore non valido mentre viene eseguito il calcolo del valore di uscita (Reset = FALSE). Reazione all'errore: Se è già stato misurato un valore valido per il tempo di ciclo, Filter_DT1 calcola il valore di uscita a partire da quest'ultimo valore della variabile CycleTime.Value. Se non è ancora stato misurato un valore valido per il tempo di ciclo, Filter_DT1 continua ad emettere, nel parametro Output, il valore di uscita configurato con la variabile StartMode.

Errore con messaggio di errore ErrorBits $\geq$ 16#0001_0000

In presenza di uno o più errori con i messaggi di errore ErrorBits $\geq$ 16#0001_0000, Filter_DT1 ha la seguente reazione:

- Il valore di uscita non può essere rilevato come previsto. Viene invece emesso il valore di uscita sostitutivo.
- Il parametro di uscita Error viene impostato.
- L'uscita di abilitazione ENO viene impostata su FALSE.

Non appena tutti gli errori segnalati con i messaggi ErrorBits $\geq$ 16#0001_0000 sono stati eliminati, Filter_DT1 ha le seguenti reazioni:

- Il valore di uscita viene rilevato come indicato nel seguito:
 - Se Reset = FALSE, calcolo del valore di uscita con l'algoritmo del filtro
 - Se Reset = TRUE, emissione di SubstituteOutput
- L'uscita di abilitazione ENO viene impostata su TRUE.

Il parametro di uscita Error viene eliminato non appena tutti gli errori sono stati eliminati.

ErrorBits (DW#16#...)	Descrizione		
0001_0000	Causa dell'errore: Il parametro SubstituteOutput o un'altra variabile utilizzata come valore di uscita non ha un valore REAL valido. Reazione all'errore: L'uscita viene impostata su 0.0. Rimedio: Accertarsi che la variabile, utilizzata come valore di uscita sia un valore REAL valido ($\neq$ NaN ad es. 16#7FFF_FFFF). La variabile utilizzata come valore di uscita è in funzione di Reset e ErrorMode:		
	Reset	ErrorMode	Valore di uscita
	FALSE	0	Input
	FALSE	1	SubstituteOutput
	TRUE	-	SubstituteOutput
0002_0000	Causa dell'errore: Il parametro Input non ha un valore REAL valido mentre viene eseguito il calcolo del valore di uscita (Reset = FALSE). Reazione all'errore: Nel parametro Output viene emesso il valore di uscita sostitutivo configurato nella variabile ErrorMode. Con ErrorMode = 0, 0.0 viene impiegato come valore di uscita. Rimedio: Accertarsi che il parametro Input sia un valore REAL valido ($\neq$ NaN, ad es. 16#7FFF_FFFF).		
0004_0000	Causa dell'errore: Il calcolo del valore di uscita fornisce un valore REAL non valido per il parametro Output. Reazione all'errore: Nel parametro Output viene emesso il valore di uscita sostitutivo configurato nella variabile ErrorMode. Rimedio: Controllare tutte le variabili comprese nel calcolo del valore di uscita: • Input • Td • Lag • CycleTime.Value Queste variabili hanno i seguenti valori validi. Con questa combinazione delle variabili, il calcolo del valore di uscita non riesce.		

ErrorBits (DW#16#...)	Descrizione
0008_0000	Causa dell'errore: La variabile Td, o Lag non ha un valore valido mentre viene eseguito il calcolo del valore di uscita (Reset = FALSE). Reazione all'errore: Nel parametro Output viene emesso il valore di uscita sostitutivo configurato nella variabile ErrorMode. Rimedio: Accertarsi che per i valori delle variabili Td, e Lag siano soddisfatte le seguenti condizioni: • $\text{CycleTime.Value} \leq \text{Td} \leq 3.402823\text{e}+38$ • $\text{CycleTime.Value}/2 \leq \text{Lag} \leq 3.402823\text{e}+38$ • I valori sono valori REAL validi ($\neq$ NaN ad es. 16#7FFF_FFFF) Informazione supplementare: Si noti che nei seguenti casi può accadere che le condizioni $\text{CycleTime.Value} \leq \text{Td}$ e $\text{CycleTime.Value}/2 \leq \text{Lag}$ vengano violate: • L'intervallo tra due richiami di Filter_DT1 supera $2 * \text{Lag}$ o Td, ad es. a causa di richiami condizionati durante l'esecuzione del programma o di punti di arresto attivi. • Un'istanza del DB di istanza Filter_DT1 viene caricata come valore attuale nella CPU e la generazione di quest'istanza presenta una datazione superiore a $2 * \text{Lag}$ o Td. In questi casi, durante la misura automatica viene rilevato un errore con la segnalazione di errore ErrorBits = 16#0008_0000.
0020_0000	Causa dell'errore: La variabile (configurata con StartMode) per l'inizializzazione del parametro Output al primo richiamo dell'istruzione non ha un valore REAL valido. Reazione all'errore: Al primo richiamo dell'istruzione nel parametro Output viene emesso il valore di uscita sostitutivo configurato nella variabile ErrorMode. Ai richiami successivi Filter_DT1 calcola il valore di uscita basandosi su questo valore di uscita sostitutivo. Rimedio: Accertarsi che la variabile utilizzata per l'inizializzazione del parametro Output sia un valore REAL valido ($\neq$ NaN ad es. 16#7FFF_FFFF). Se è impostato Reset = FALSE, l'inizializzazione diventa effettiva al primo richiamo dell'istruzione dopo la commutazione dello stato di funzionamento della CPU da STOP a RUN. La variabile utilizzata per l'inizializzazione del parametro Output è in funzione di StartMode: • StartMode = 1: SubstituteOutput • StartMode = 2: Output
0040_0000	Causa dell'errore: La variabile CycleTime.Value durante l'esecuzione del calcolo del valore di uscita (Reset = FALSE) non ha un valore valido. Reazione all'errore: Nel parametro Output viene emesso il valore di uscita sostitutivo configurato nella variabile ErrorMode. Rimedio: Accertarsi che siano soddisfatte le seguenti condizioni: • $0.0 < \text{CycleTime.Value} \leq 3.402823\text{e}+38$ • CycleTime.Value è un valore REAL valido ($\neq$ NaN ad es. 16#7FFF_FFFF) Informazione supplementare: Per il calcolo automatico del valore della variabile CycleTime.Value, impostare la variabile CycleTime.EnableMeasurement su TRUE.

10.12 Filter_Universal

10.12.1 Compatibilità con CPU e FW

La tabella seguente indica le CPU compatibili con le varie versioni di Filter_Universal.

CPU	FW	Filter_Universal
S7-1200 G2	da V1.0	V1.0
CPU basate su S7-1500	da V2.0	V1.0

10.12.2 Descrizione di Filter_Universal

Descrizione

L'istruzione Filter_Universal è un filtro configurabile dell'ordine da 1 a 10.

Viene impiegato per elaborare un segnale in modo tale da far passare o attenuare le componenti di frequenza specifiche di tale segnale.

Filter_Universal può essere impiegato per i seguenti scopi:

- Filtro passa alto
- Filtro passa basso
- Filtro passa banda
- Filtro elimina banda

Per ottenere il comportamento desiderato per il filtro è possibile impostare i seguenti parametri del filtro con le variabili corrispondenti:

- Tipo (variabile Type)
- Frequenza (variabile Frequency)
- Larghezza di banda (variabile Bandwidth)
- Ordine (variabile Order)
- Caratteristica (variabile Characteristic)

Una descrizione più approfondita dei parametri del filtro e delle relative variabili è disponibile nel capitolo Parametri del filtro ([Pagina 588](#))

Richiamo

Filter_Universal richiede un tempo di ciclo costante e quindi deve essere richiamato in un OB di schedulazione orologio.

In un OB o FC Filter_Universal viene richiamato come DB di un'istanza singola. In un FB Filter_Universal può essere richiamato sia come DB di istanza singola, sia come DB di multiistanza o come DB di istanza di un parametro.

Al richiamo dell'istruzione non viene creato nessun oggetto tecnologico. Nella finestra di ispezione dell'editor di programma è disponibile un'interfaccia di parametrizzazione per le singole istanze. Questa rappresenta la risposta in frequenza determinata dai parametri configurati per il filtro.

Avviamento

Le variabili nel campo statico di Filter_Universal non sono a ritenzione. Queste variabili vengono inizializzate con i valori di avvio dopo ogni commutazione dello stato di funzionamento della CPU da STOP a RUN. Se si modificano online i valori attuali e si desidera che questi valori vengano conservati dopo la commutazione dello stato di funzionamento della CPU da STOP a RUN, salvarli nei valori di avvio del blocco dati.

La variabile StartMode ([Pagina 594](#)) consente di determinare il comportamento all'avvio dell'istruzione Filter_Universal al primo richiamo dopo la commutazione dello stato di funzionamento della CPU da STOP a RUN.

Comportamento in caso di errori

Se il valore di uscita non può essere calcolato correttamente, l'istruzione Filter_Universal emette un valore di uscita sostitutivo e genera un errore con il messaggio di errore ErrorBits >= 16#0002_0000. Il valore di uscita sostitutivo può essere determinato con la variabile ErrorMode ([Pagina 601](#)) come segue:

ErrorMode	Output
0	Valore del parametro Input
1	Valore del parametro SubstituteOutput
2	Ultimo valore di uscita valido del filtro 0.0, in assenza di un valore di uscita valido del filtro.
3	0.0

Inoltre, per tutti i valori della variabile ErrorMode vale quanto segue:

- Se il valore di uscita sostitutivo non è un valore REAL valido, come valore di uscita viene emesso 0.0.
- Il valore di uscita sostitutivo viene limitato al campo valori -3.402823e+38 .. +3.402823e+38 del tipo di dati REAL. Solo in quel momento viene emesso il valore di uscita sostitutivo nel parametro Output.
- La variabile ErrorMode è effettiva solo se è impostato il parametro Reset = FALSE. Se è impostato il parametro Reset = TRUE, nel parametro Output viene emesso il valore del parametro SubstituteOutput oppure 0.0.

Il parametro Error indica se attualmente è presente un errore. Se l'errore non è più presente, Error viene impostato su FALSE. Il parametro ErrorBits indica quali errori si sono verificati. ErrorBits è a ritenzione e viene resettato soltanto da un fronte di salita nel parametro Reset o ErrorAck.

Quando non sono più presenti errori con segnalazione di errore $\text{ErrorBits} \geq 16\ 0002\_0000$, Filter_Universal torna al calcolo del valore di uscita con l'algoritmo del filtro. La commutazione dipende dal tipo di filtro:

- Per i filtri passa alto e passa banda (Type = 1 o 2) l'algoritmo del filtro viene impostato in modo tale per cui è come se il filtro fosse in uno stato stazionario con $\text{Output} = 0.0$. Se il parametro Input rimane costante, il valore di uscita salta a $\text{Output} = 0.0$. Se il parametro Input cambia, il valore di uscita salta ad un valore corrispondente.
- Per i filtri passa basso ed elimina banda (Type = 0 o 3) l'algoritmo del filtro viene impostato in modo tale per cui è come se il filtro fosse in uno stato stazionario con $\text{Output} = \text{SubstituteOutput}$. La commutazione avviene bumpless.

10.12.3 **Modo di funzionamento Filter_Universal**

10.12.3.1 **Parametri del filtro**

Per ottenere il comportamento del filtro desiderato è possibile impostare i parametri di tipo, frequenza, larghezza di banda, caratteristica e ordine del filtro con le variabili corrispondenti.

Tipo di filtro

Il tipo di filtro determina il comportamento generale di trasmissione per le diverse componenti di frequenza del segnale di ingresso. Viene definito con la variabile Type.

La tabella seguente mostra i diversi tipi di filtro:

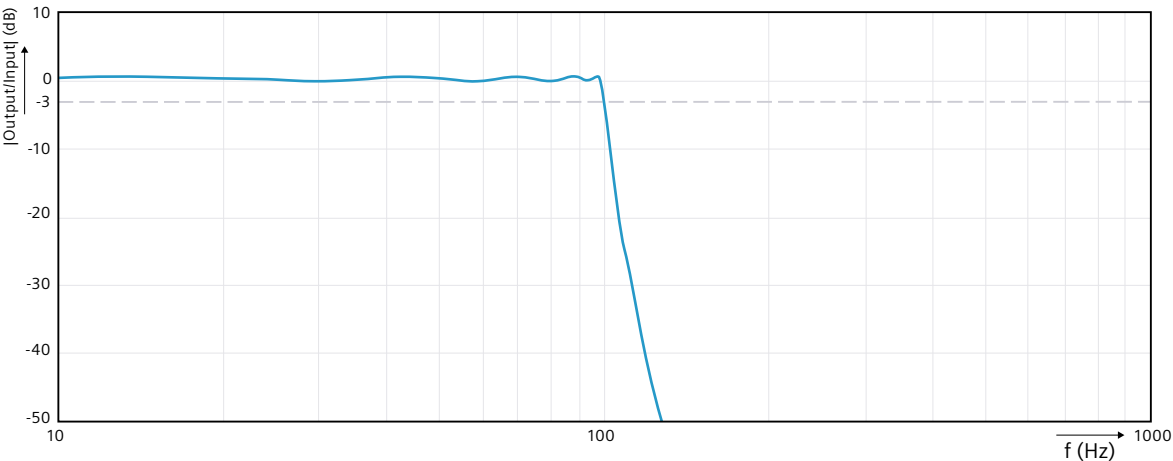
Type	Descrizione	Esempio applicativo
0	Filtro passa basso: Il filtro fa passare le componenti di frequenza inferiori alla frequenza limite e attenua quelle superiori.	Riduzione del rumore con un dato valore di ingresso misurato per ottenere un andamento del segnale più livellato
1	Filtro passa alto: Il filtro fa passare le componenti di frequenza superiori alla frequenza limite e attenua quelle inferiori.	Soppressione di componenti di frequenza continua o bassa, ad es. componente continua nel segnale
2	Filtro passa banda: Il filtro fa passare le componenti di frequenza che rientrano in un determinato campo contiguo alla frequenza media e attenua quelle esterne a tale campo.	Calcolo del segnale utile con un determinato campo di frequenza da un segnale che contiene ulteriori componenti di frequenza
3	Filtro elimina banda: Il filtro attenua le componenti di frequenza che rientrano in un determinato campo contiguo alla frequenza media e fa passare quelle esterne a tale campo.	Attenuazione di interferenze in un determinato campo di frequenza, ad es. interferenze dovute alla frequenza di rete

Frequenza e larghezza di banda

Per i filtri passa basso e passa alto la frequenza limite viene definita con la variabile Frequency. La frequenza limite è la frequenza per la quale il guadagno viene ridotto a $1/\sqrt{2} \approx 0.707 \approx -3\text{dB}$. Un segnale di ingresso sinusoidale con ampiezza 1.0 e una frequenza pari alla frequenza limite comporterebbe un segnale di uscita sinusoidale con ampiezza 0.707.

Il comportamento del valore di uscita rispetto al valore di ingresso (guadagno) dipendente dalla frequenza può essere rappresentato nella caratteristica di ampiezza. Il valore 0 dB corrisponde a un guadagno = 1.0.

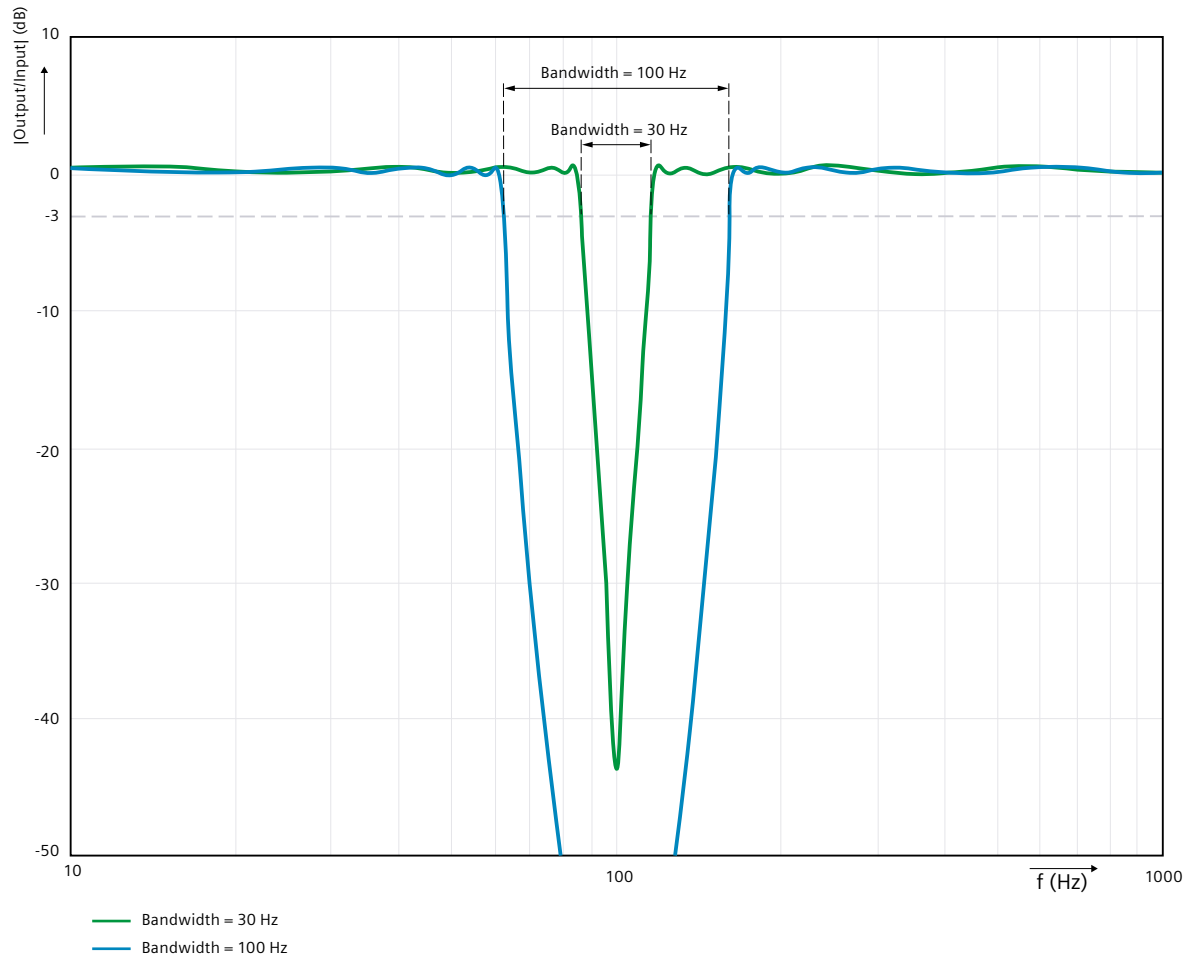
La seguente figura illustra la caratteristica di ampiezza di un filtro passa basso con Frequency = 100 Hz (Order = 10, Characteristic = 2):



I filtri passa banda ed elimina banda hanno una frequenza limite superiore e inferiore la cui posizione viene determinata dalla frequenza media e dalla larghezza di banda. La tabella seguente mostra le variabili configurabili per tali filtri:

Variabile	Descrizione
Frequency	Determina la frequenza media, che è la media geometrica della frequenza limite inferiore e superiore. Nella scala logaritmica della frequenza, la frequenza media si trova a metà tra la frequenza limite inferiore e superiore.
Bandwidth	La larghezza di banda determina la differenza tra la frequenza limite inferiore e superiore. Essa definisce l'ampiezza del campo di frequenza che deve essere attenuato o lasciato passare.

La seguente figura illustra la caratteristica di ampiezza di un filtro elimina banda con Frequency = 100 Hz e valori diversi per la variabile Bandwidth (Order = 10, Characteristic = 2):



La massima frequenza limite o media utilizzabile dipende dal tempo di ciclo. Il campo di valori ammesso è $Frequency < 0.5 / CycleTime.Value$.

La larghezza di banda massima utilizzabile dipende dal tempo di ciclo e dalla frequenza media. Il campo di valori ammesso è $Bandwidth < 0.5 / CycleTime.Value - Frequency$.

NOTA

Per evitare distorsioni, la velocità di campionamento in Filter_Universal ($= 1 / \text{tempo di ciclo}$) deve essere almeno il doppio della frequenza massima dei segnali o delle componenti del segnale da elaborare. Si consiglia di impostare un tempo di ciclo inferiore a tale valore limite.

Esempio: Un segnale con componenti di frequenza fino a 100 Hz richiede una velocità di campionamento superiore a 200 Hz. Ciò corrisponde a un tempo di ciclo massimo di 5 ms; si consiglia tuttavia di impostare un valore inferiore.

Ordine

L'ordine del filtro determina la velocità con cui lo smorzamento aumenta oltre la frequenza limite. Ciò corrisponde alla pendenza della caratteristica di ampiezza oltre la frequenza limite. L'ordine del filtro viene definito con la variabile Order.

Con un filtro di ordine superiore:

- Una stessa frequenza oltre la frequenza limite viene attenuata maggiormente. La caratteristica di ampiezza ha una pendenza maggiore.
- Aumenta il tempo di esecuzione di Filter_Universal.
- La sovraelongazione della risposta a gradino aumenta dopo un gradino di ingresso per la caratteristica del filtro Butterworth e Chebyshev (variabile Characteristic = 1 o 2).

Per i filtri passa banda ed elimina banda si consiglia di utilizzare solo filtri di ordine superiore, in caso contrario può accadere che, nel campo di frequenza contiguo alla frequenza media, non si ottenga l'azione di filtraggio desiderata.

Per l'ordine del filtro si possono configurare valori da 0 a 10 nella variabile Order.

Con l'impostazione Order = 0 il filtro è inefficace e Output = Input.

Caratteristica

La caratteristica del filtro viene definita con la variabile Characteristic.

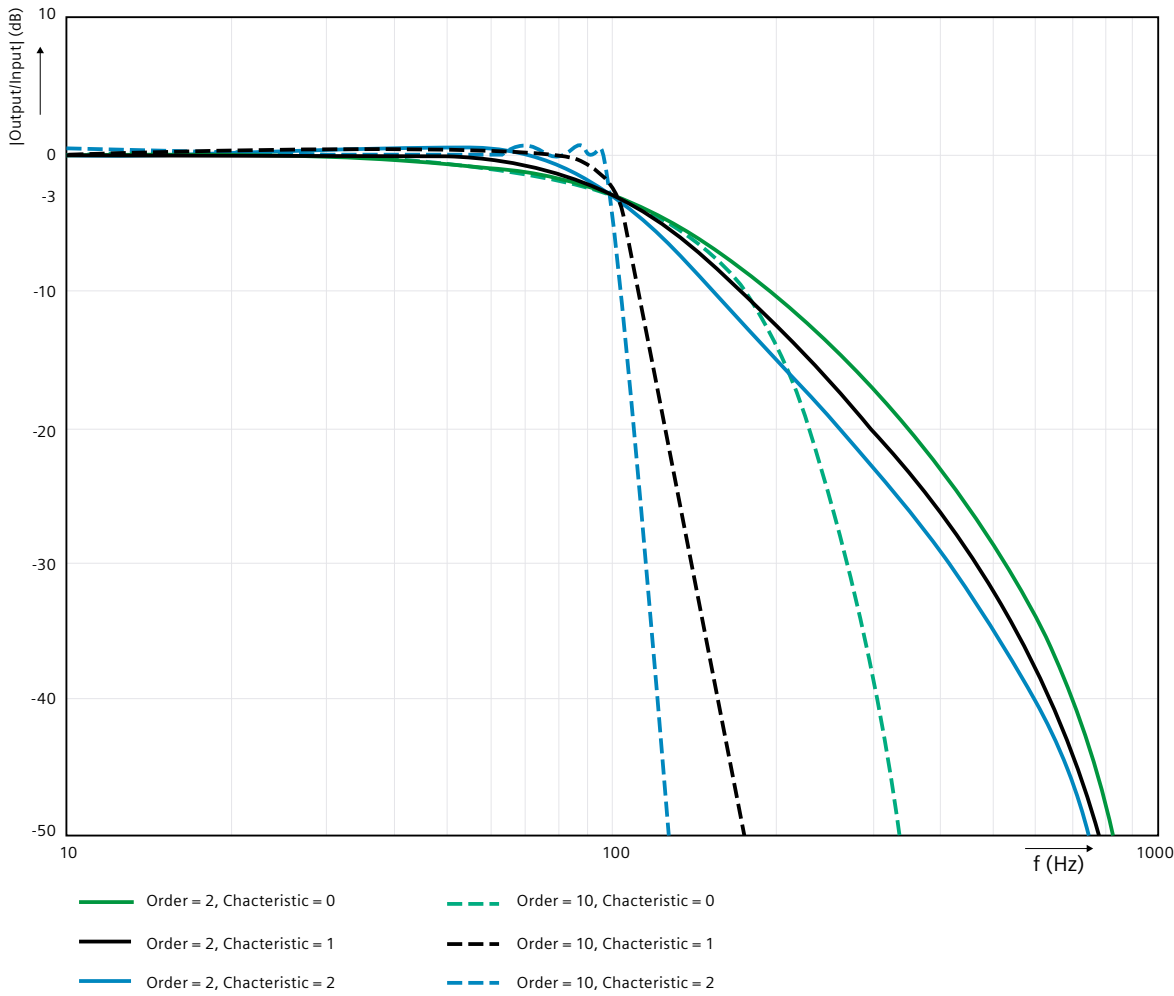
Questa agisce su:

- l'ondulazione della caratteristica di ampiezza nella banda passante
- la pendenza della caratteristica di ampiezza oltre la frequenza limite (quanto rapidamente aumenta lo smorzamento)
- la sovraelongazione nella risposta a gradino dopo un gradino di ingresso

Per Filter_Universal si possono configurare tre caratteristiche utilizzando la variabile Characteristic:

Caratteristica	Descrizione
0	Bessel: Questo filtro ha una caratteristica di ampiezza piatta nella banda passante. La pendenza della caratteristica di ampiezza oltre la frequenza limite è minore rispetto a quella dei filtri Butterworth e Chebyshev. La risposta a gradino ha solo una piccola sovraelongazione.
1	Butterworth: Questo filtro ha una caratteristica di ampiezza piatta nella banda passante. La pendenza della caratteristica di ampiezza oltre la frequenza limite è maggiore rispetto a quella dei filtri Butterworth e minore rispetto ai filtri Chebyshev. La sovraelongazione della risposta a gradino è maggiore rispetto a quella dei filtri Bessel e minore rispetto a quella dei filtri Chebyshev.
2	Chebyshev tipo I: Questo filtro ha un'ondulazione della caratteristica di ampiezza di 0.5 dB nella banda passante. La pendenza della caratteristica di ampiezza oltre la frequenza limite è maggiore rispetto a quella dei filtri Bessel e Butterworth. La risposta a gradino ha una sovraelongazione maggiore rispetto a quella dei filtri Bessel e Butterworth.

La figura seguente mostra l'effetto dei diversi valori di ordine e caratteristica sull'ampiezza del filtro passa basso:



Modifica dei parametri del filtro

Affinché l'algoritmo del filtro possa calcolare il valore di uscita, Filter_Universal deve prima rilevare i coefficienti di filtro sulla base dei parametri del filtro. Questa rilevazione viene avviata nelle seguenti situazioni:

- Con la prima esecuzione dopo la commutazione dello stato di funzionamento della CPU da STOP a RUN
- Con ogni modifica dei parametri del filtro
- Con l'esecuzione di "Carica i valori iniziali come valori attuali"

Come parte della rilevazione, vengono verificate le seguenti condizioni per i parametri del filtro:

- $0.0 < \text{Frequency} < 0.5 / \text{CycleTime.Value}$
- $0.0 \leq \text{Bandwidth} < 0.5 / \text{CycleTime.Value} - \text{Frequency}$
- $0 \leq \text{Type} \leq 3$
- $0 \leq \text{Characteristic} \leq 2$
- $0 \leq \text{Order} \leq 10$

Se una di queste condizioni non viene soddisfatta e contemporaneamente $\text{Reset} = \text{FALSE}$, l'algoritmo del filtro non può calcolare correttamente il valore di ingresso. In questo caso viene emesso un messaggio di errore e nel parametro Output viene emesso un valore sostitutivo finché tutti i parametri del filtro non hanno un valore corretto.

Quando tutti i valori sono corretti, i coefficienti del filtro vengono rilevati e salvati internamente per il calcolo dell'algoritmo del filtro.

La reazione del valore di ingresso in caso di modifiche valide dei parametri del filtro, dipende dal tipo di filtro:

- Per i filtri passa alto e passa banda ($\text{Type} = 1$ o 2) l'algoritmo del filtro viene impostato come se fosse in uno stato stazionario con $\text{Output} = 0.0$. Se il parametro Input rimane costante, il valore di uscita salta a $\text{Output} = 0.0$. Se il parametro Input cambia, il valore di uscita salta ad un valore corrispondente.
- Per i filtri passa basso ed elimina banda ($\text{Type} = 0$ o 3) l'algoritmo del filtro viene impostato in modo tale per cui è come se il filtro fosse in uno stato stazionario con $\text{Output} = \text{SubstituteOutput}$. La commutazione avviene bumpless.

Per le applicazioni sensibili al tempo occorre tenere conto che la rilevazione dei coefficienti del filtro richiede molto più tempo per l'esecuzione del calcolo da parte dell'algoritmo del filtro.

10.12.3.2 Inizializzazione dei valori di uscita

Il parametro Output viene inizializzato per la prima esecuzione dopo le seguenti azioni:

- Cambio dello stato operativo della CPU da STOP a RUN
- Esecuzione di "Carica i valori iniziali come valori attuali" con l'opzione "Tutti i valori"

Il primo valore del parametro Output dipende dal tipo di filtro:

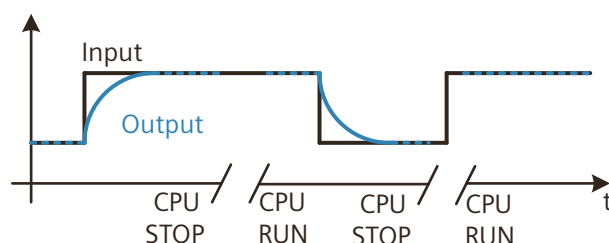
- Per i filtri passa alto e passa banda (Type = 1 o 2) il primo valore del parametro è Output = 0.0.
- Per i filtri passa basso ed elimina banda (Type = 0 o 3) è possibile configurare il primo valore del parametro Output tramite la variabile StartMode .

Ai richiami successivi Filter_Universal calcola il valore di uscita partendo da questo valore di inizializzazione e tenendo conto del valore di ingresso e dei parametri del filtro.

Per i filtri passa basso ed elimina banda sono possibili le seguenti impostazioni della variabile StartMode:

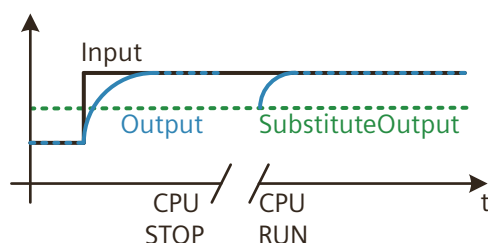
- StartMode = 0

Il parametro Output acquisisce il valore del parametro Input.



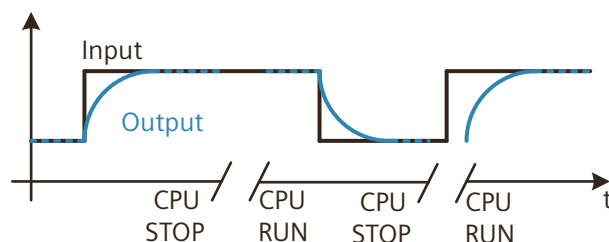
- StartMode = 1

Il parametro Output acquisisce il valore del parametro SubstituteOutput.



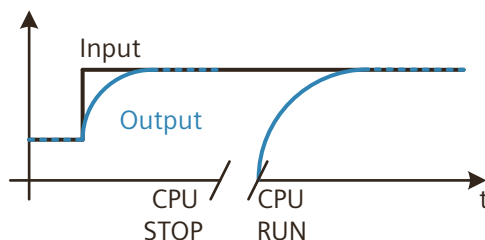
- StartMode = 2

Il parametro Output resta invariato.



- StartMode = 3

Il parametro Output acquisisce il valore 0.0.



Inoltre, per tutti i valori della variabile StartMode vale quanto segue:

- La variabile StartMode e i parametri del filtro non sono a ritenzione. Queste variabili vengono inizializzate con i valori di avvio dopo ogni commutazione dello stato di funzionamento della CPU da STOP a RUN. Assicurarsi che al primo richiamo dell'istruzione Filter_Universal, dopo la commutazione dello stato di funzionamento della CPU da STOP a RUN, i valori di queste variabili siano adatti ad ottenere il comportamento desiderato.
- Il valore selezionato tramite StartMode viene limitato al campo valori del tipo di dati REAL. Solo successivamente il valore viene emesso nel parametro Output.
- Se il valore selezionato tramite StartMode non è un valore REAL valido, viene emesso il valore di uscita sostitutivo nel parametro Output. Il valore di uscita sostitutivo viene configurato con la variabile ErrorMode e limitato al campo valori del tipo di dati REAL. Se il valore di uscita sostitutivo è a sua volta un valore REAL non valido, nel parametro Output viene emesso 0.0. Basandosi su questo valore di uscita sostitutivo, l'istruzione calcola il valore di uscita ai richiami successivi.
- Soltanto se è impostato il parametro Reset = FALSE e al contempo non è presente alcun errore con il messaggio ErrorBits $\geq 16\#0002\_0000$ la variabile StartMode ha effetto sul parametro Output. Se è impostato il parametro Reset = TRUE, il valore del parametro SubstituteOutput viene emesso nel parametro Output. In presenza di un errore con il messaggio ErrorBits $\geq 16\#0002\_0000$, nel parametro Output viene emesso il valore di uscita sostitutivo configurato nella variabile ErrorMode.

10.12.3.3 Valore finale nello stato stazionario

Se il valore di ingresso è costante, anche il valore di uscita di Filter_Universal dopo un certo tempo deve raggiungere un valore finale costante:

- Output = 0.0 per i filtri passa alto e passa banda (Type = 1 o 2)
- Output = Input per i filtri passa basso ed elimina banda (Type = 0 o 3)

La precisione limitata dei numeri in virgola mobile può far sì che non si raggiunga con esattezza il valore finale.

Questo si verifica più frequentemente nei filtri con ordine dispari (variabile Order = 1, 3, 5, 7 o 9) rispetto ai filtri con ordine pari.

Oltre alla scelta del filtro di ordine pari, per raggiungere il valore finale può essere utile impostare la variabile FinalValueMode = 1. Se il valore assoluto del valore di uscita non varia per più cicli, questa impostazione commuta il valore di uscita all'ultimo valore valido. Questa opzione è valida solo con un valore di ingresso costante.

L'utilizzo dell'opzione `FinalValueMode = 1` può quasi raddoppiare il tempo di calcolo dell'algoritmo del filtro. L'effetto sul tempo di esecuzione dipende dai parametri del filtro, dal valore di ingresso e dal tempo di ciclo. Per applicazioni sensibili al tempo si può verificare se è necessario utilizzare la variabile `FinalValueMode = 1` oppure se è sufficiente il comportamento con `FinalValueMode = 0`.

In base ai parametri del filtro, al valore di ingresso e al tempo di ciclo è possibile fare in modo che il valore finale venga raggiunto con precisione già con l'opzione `FinalValueMode = 0`, per cui l'opzione `FinalValueMode = 1` non è necessaria.

Esempio:

La tabella seguente mostra l'effetto della variabile `FinalValueMode` sul valore di uscita di un filtro passa basso con `Frequency = 120.0`, `Order = 1`, `Characteristic = 2` e `CycleTime.Value = 0.001`, con una gradino di ingresso da 1.0 a 0.0:

Tempo in secondi	Input	Output con <code>FinalValueMode = 0</code>	Output con <code>FinalValueMode = 1</code>
-0.001	1.0	1.0000000	1.0000000
0.000	0.0	0.7163693	0.7163693
0.001	0.0	0.3100007	0.3100007
...	0.0	...	...
0.075	0.0	-4.597156E-17	-4.597156E-17
0.076	0.0	4.597156E-17	4.597156E-17
0.077	0.0	-4.597156E-17	-4.597156E-17
0.078	0.0	4.597156E-17	0.0000000
0.079	0.0	-4.597156E-17	0.0000000
...	0.0	+/-4.597156E-17	0.0000000

10.12.3.4 Impiego nelle applicazioni sensibili al tempo

Il tempo di esecuzione di `Filter_Universal` dipende strettamente dalla sua configurazione. Con una configurazione standard (`CycleTime.EnableDetection = TRUE`, `FinalValueMode = 1`) non tutti i tipi di CPU sono in grado di eseguire un filtro di ordine superiore nel tempo di ciclo più rapido. Utilizzando `Filter_Universal` per frequenze di segnale più elevate, potrebbero essere necessari tali tempo di ciclo rapidi, ad esempio le frequenze di segnale fino a 2 kHz richiedono un tempo di ciclo massimo di 250 µsec.

Per ridurre il tempo di esecuzione di `Filter_Universal` con un'applicazione sensibile al tempo, si può ricorrere ai seguenti adattamenti della configurazione:

- Riduzione dell'ordine del filtro (variabile `Order`), se ciò consente di ottenere ancora il comportamento del filtro desiderato.
- Impostazione di `FinalValueMode` a 0, se ciò non comporta variazioni nel comportamento del filtro con valori di ingresso costanti rispetto a `FinalValueMode = 1`.
- Impostazione di `CycleTime.EnableDetection` a `FALSE` e impostazione manuale della variabile `CycleTime.Value`. Questo influisce solo sul tempo di esecuzione del primo richiamo o dopo la modifica dei parametri del filtro.

Anche lo stato operativo attuale di `Filter_Universal` incide sul tempo di esecuzione. La rilevazione dei coefficienti del filtro alla prima esecuzione o dopo la modifica dei parametri

del filtro richiede in tempo di esecuzione molto maggiore rispetto al calcolo da parte dell'algoritmo del filtro con parametri di filtro costanti.

10.12.3.5 Ambiente di richiamo e riconoscimento automatico del tempo di ciclo

Per il calcolo del valore di uscita Filter_Universal richiede un tempo di ciclo costante e quindi deve essere richiamato in un OB di schedulazione orologio. Con la configurazione standard, Filter_Universal riconosce automaticamente il tempo di ciclo dell'OB e lo salva nella variabile CycleTime.Value. Ciò avviene nelle seguenti situazioni:

- Con la prima esecuzione dopo la commutazione dello stato di funzionamento della CPU da STOP a RUN
- Con ogni modifica dei parametri del filtro
- Con l'esecuzione di "Carica i valori iniziali come valori attuali"

Se il riconoscimento automatico del tempo di ciclo non fornisce un risultato valido o se Filter_Universal non viene richiamato in un OB di schedulazione orologio non è possibile calcolare correttamente il parametro Output. In questo caso viene emesso un messaggio di errore e nel parametro Output viene emesso un valore sostitutivo fino a quando un OB di schedulazione orologio non riconosce un tempo di ciclo valido.

Tenere presente che una modifica del tasso di richiamo tramite richiami condizionati di Filter_Universal comporta delle discrepanze tra il tempo di ciclo rilevato e quello effettivo, influenzando di conseguenza il comportamento del filtro. Evitare pertanto i richiami condizionati di Filter_Universal.

I punti di arresto attivi o il caricamento di istantanee come valori attuali non influiscono sul rilevamento automatico del tempo di ciclo.

Se si disattiva il rilevamento automatico del tempo di ciclo impostando la variabile CycleTime.EnableDetection = FALSE, il tempo di ciclo deve essere calcolato manualmente nella variabile CycleTime.Value. Filter_Universal verifica la validità di Variable CycleTime.Value a ogni richiamo.

La disattivazione del rilevamento automatico del tempo di ciclo riduce il tempo di esecuzione di Filter_Universal e può quindi essere utile per le applicazioni sensibili al tempo.

Se il rilevamento automatico del tempo di ciclo è disattivato, Filter_Universal non emette alcun messaggio di errore quando viene richiamata al di fuori di un OB di schedulazione orologio. Il valore di uscita viene calcolato dall'algoritmo del filtro. I richiami al di fuori di un OB di schedulazione orologio possono avere effetti negativi sul comportamento del filtro perché il tempo di ciclo effettivo non è costante.

10.12.3.6 Comportamento in caso di reset

L'istruzione Filter_Universal si comporta, in funzione del parametro Reset, nel seguente modo:

- Se il parametro Reset = TRUE, il valore del parametro SubstituteOutput viene emesso nel parametro Output.
- Se il parametro Reset = FALSE, il valore emesso nel parametro Output viene calcolato con l'algoritmo del filtro.
- Se il parametro Reset di FALSE è impostato su TRUE, il valore del parametro Output commuta direttamente sul valore del parametro SubstituteOutput. Questa commutazione può essere repentina. Inoltre viene resettato il parametro ErrorBits.
- Se il parametro Reset di TRUE viene impostato su FALSE, il comportamento dipende dal tipo di filtro:
 - Per i filtri passa alto e passa banda (Type = 1 o 2) l'algoritmo del filtro viene impostato in modo tale per cui è come se il filtro fosse in uno stato stazionario con Output = 0.0. Se il parametro Input rimane costante, il valore di uscita salta a Output = 0.0. Se il parametro Input cambia, il valore di uscita salta ad un valore corrispondente.
 - Per i filtri passa basso ed elimina banda (Type = 0 o 3) l'algoritmo del filtro viene impostato in modo tale per cui è come se il filtro fosse in uno stato stazionario con Output = SubstituteOutput. La commutazione avviene bumpless.

10.12.3.7 Comportamento di abilitazione EN/ENO

Se una delle seguenti condizioni è soddisfatta, l'uscita di abilitazione ENO viene impostata su FALSE:

- L'ingresso di abilitazione EN è impostato su TRUE e il parametro Output è definito tramite un valore di uscita sostitutivo in presenza dei messaggi di errore $\text{ErrorBits} \geq 16\#0001\_0000$
- L'ingresso di abilitazione EN è impostato su FALSE.

Diversamente l'uscita di abilitazione ENO viene impostata su TRUE:

10.12.4 Parametro di ingresso Filter_Universal

Parametri	Tipo di dati	Preassegnazione	Descrizione
Input	REAL	0.0	Valore di ingresso
SubstituteOutput	REAL	0.0	SubstituteOutput viene impiegato come valore di uscita sostitutivo se Reset = TRUE o se è attualmente attiva una delle seguenti modalità: - ErrorMode = 1 - StartMode = 1
ErrorAck	BOOL	FALSE	Cancella i messaggi di errore. Il fronte FALSE -> TRUE ErrorBits viene resettato.
Reset	BOOL	FALSE	Resetta l'istruzione - Fronte FALSE -> TRUE ErrorBits viene resettato. - Finché Reset è impostato su TRUE, viene emesso il valore di uscita sostitutivo SubstituteOutput. - Finché Reset è impostato su FALSE, il calcolo del valore di uscita viene eseguito. - Fronte TRUE -> FALSE - Per i filtri passa alto e passa banda (tipo 1 o 2) l'algoritmo del filtro viene impostato come se il filtro si trovasse in uno stato stazionario con Output = 0.0. - Per i filtri passa alto e passa banda (tipo 0 o 3) l'algoritmo del filtro viene impostato come se il filtro si trovasse in uno stato stazionario con Output = SubstituteOutput.

10.12.5 Parametro di uscita Filter_Universal

Parametro	Tipo di dati	Preassegnazione	Descrizione
Output	REAL	0.0	Valore di uscita Il valore di uscita è a ritenzione.
ErrorBits	DWORD	16#0	Il parametro ErrorBits (Pagina 601) mostra quali messaggi di errore sono presenti. ErrorBits è a ritenzione e viene resettato in presenza di un fronte di salita in Reset o ErrorAck.
Error	BOOL	FALSE	Se Error è impostato su TRUE si è attualmente in presenza di almeno un errore.

10.12.6 Variabili statiche Filter_Universal

Variabile	Tipo di dati	Preassegnazione	Descrizione
Frequency	REAL	50.0	Frequenza limite per filtri passa basso e passa alto o frequenza media per filtri passa banda e elimina banda in Hz Campo di valori consentito: $0.5/CycleTime.Value > Frequency > 0.0$
Bandwidth	REAL	0.0	Larghezza di banda per filtri passa banda e elimina banda in Hz Campo di valori consentito: $0.5/CycleTime.Value - Frequency > Bandwidth \geq 0.0$
Type	INT	0	Tipo di filtro 0 = filtro passa basso 1 = filtro passa alto 2 = filtro passa banda 3 = filtro elimina banda Campo di valori ammesso: 0 - 3
Characteristic	INT	0	Caratteristica del filtro 0 = Bessel 1 = Butterworth 2 = Chebyshev con ondulazione 0.5 dB nella banda passante Campo di valori ammesso: 0 - 2
Order	INT	2	Ordine del filtro (con Order = 0 è Output = Input) Campo di valori ammesso: 0 - 10
ErrorMode	INT	2	Selezione del valore di uscita sostitutivo in caso di errori 0 = Input 1 = SubstituteOutput 2 = ultimo valore di uscita valido del filtro 3 = 0.0 Campo di valori ammesso: 0 - 3 Se il valore di ErrorMode non corrisponde al campo di valori consentito, ErrorMode = 2.
StartMode	INT	2	Selezione del primo valore di uscita per filtri passa basso ed elimina banda 0 = Input 1 = SubstituteOutput 2 = ultimo valore di uscita 3 = 0.0 Campo di valori ammesso: 0 - 3 Se il valore di StartMode non corrisponde al campo di valori consentito, StartMode = 2.
FinalValueMode	INT	1	Selezione del comportamento nello stato stazionario 0 = possibili discrepanze tra valore di uscita e valore finale 1 = il valore di uscita raggiunge esattamente valore finale Campo di valori ammesso: 0 - 1 Se il valore di FinalValueMode non corrisponde al campo di valori consentito, FinalValueMode = 1.

Variabile	Tipo di dati	Preasse- gnazio- ne	Descrizione
CycleTime	AuxFct_CycleTimeDe- tection	-	Dati del tempo di ciclo
CycleTime.Value	REAL	0.001	Tempo di ciclo in secondi (intervallo tra due richiami) Campo di valori ammesso: CycleTime.Value > 0.0
CycleTime.EnableDe- tection	BOOL	TRUE	Rilevamento automatico del tempo di ciclo - FALSE = disattivato - TRUE = attivato

10.12.7 Parametro ErrorBits

Se si verificano diversi errori contemporaneamente, i valori di ErrorBits vengono visualizzati in un'addizione binaria. Se viene visualizzato ad es. ErrorBits = 16#0000_0003 sono presenti contemporaneamente gli errori 16#0000_0001 e 16#0000_0002.

Nel caso di Filter_Universal gli errori emessi nel parametro ErrorBits sono suddivisi in due categorie:

- Errori con messaggi di errore ErrorBits < 16#0001_0000
Il valore di uscita può essere calcolato nonostante l'errore.
- Errori con messaggi di errore ErrorBits ≥ 16#0001_0000
L'errore impedisce il calcolo del valore di uscita. Viene emesso un valore di uscita sostitutivo.

Errore con messaggio di errore ErrorBits < 16#0001_0000

In presenza di uno o più errori con i messaggi di errore ErrorBits < 16#0001_0000, Filter_Universal ha la seguente reazione:

- Nonostante questo errore, il valore di uscita viene rilevato come indicato nel seguito:
 - Se Reset = FALSE, calcolo del valore di uscita con l'algoritmo del filtro
 - Se Reset = TRUE, emissione di SubstituteOutput
- Il parametro di uscita Error viene impostato.
- L'uscita di abilitazione ENO non viene modificata.

Il parametro di uscita Error viene eliminato non appena tutti gli errori sono stati eliminati.

ErrorBits (DW#16#...)	Descrizione
0000_0000	Nessun errore
0000_0001	Causa degli errore e reazione agli errori: Il parametro Output è stato limitato a -3.402823e+38 oppure +3.402823e+38. Rimedio: Se sull'uscita viene emesso il valore stabilito dalla funzione del filtro (Reset = FALSE e ErrorBits < 16#0001_0000), controllare il parametro Input. Se ErrorBits ≥ 16#0001_0000 e Reset = FALSE, il valore di uscita sostitutivo viene limitato al momento della relativa emissione. Successivamente, a seconda del valore impostato nella variabile ErrorMode:, controllare i seguenti parametri • Input • SubstituteOutput

ErrorBits (DW#16#...)	Descrizione
	Se Reset = TRUE, controllare il parametro SubstituteOutput.

Errore con messaggio di errore ErrorBits $\geq$ 16#0001_0000

In presenza di uno o più errori con i messaggi di errore ErrorBits $\geq$ 16#0001_0000, Filter_Universal ha la seguente reazione:

- Il valore di uscita non può essere rilevato come previsto. Viene invece emesso il valore di uscita sostitutivo.
- Il parametro di uscita Error viene impostato.
- L'uscita di abilitazione ENO viene impostata su FALSE.

Non appena tutti gli errori segnalati con i messaggi ErrorBits $\geq$ 16#0001_0000 sono stati eliminati, Filter_Universal ha le seguenti reazioni:

- Il valore di uscita viene rilevato come indicato nel seguito:
 - Se Reset = FALSE, calcolo del valore di uscita con l'algoritmo del filtro
 - Se Reset = TRUE, emissione di SubstituteOutput
- L'uscita di abilitazione ENO viene impostata su TRUE.

Il parametro di uscita Error viene eliminato non appena tutti gli errori sono stati eliminati.

ErrorBits (DW#16#...)	Descrizione		
0001_0000	Causa dell'errore: Il parametro SubstituteOutput o un'altra variabile utilizzata come valore di uscita non ha un valore REAL valido.		
	Reazione all'errore: L'uscita viene impostata su 0.0.		
	Rimedio: Accertarsi che la variabile, utilizzata come valore di uscita sia un valore REALvalido (≠NaN ad es. 16#7FFF_FFFF). La variabile utilizzata come valore di uscita è in funzione di Reset e ErrorMode:		
	Reset	ErrorMode	Valore di uscita
	FALSE	0	Input
0002_0000	FALSE	1	SubstituteOutput
	TRUE	-	SubstituteOutput
	Causa dell'errore: Il parametro Input non ha un valore REAL valido mentre viene eseguito il calcolo del valore di uscita (Reset = FALSE).		
	Reazione all'errore: Nel parametro Output viene emesso il valore di uscita sostitutivo configurato nella variabile ErrorMode. Con ErrorMode = 0, 0.0 viene impiegato come valore di uscita.		
	Rimedio: Accertarsi che il parametro Input sia un valore REAL valido (≠NaN, ad es. 16#7FFF_FFFF).		

ErrorBits (DW#16#...)	Descrizione
0004_0000	Causa dell'errore: Il calcolo del valore di uscita fornisce un valore REAL non valido per il parametro Output. Reazione all'errore: Nel parametro Output viene emesso il valore di uscita sostitutivo configurato nella variabile ErrorMode. Rimedio: Controllare tutte le variabili comprese nel calcolo del valore di uscita: • Input • Frequency • Bandwidth • Type • Characteristic • Order • CycleTime.Value Queste variabili hanno i seguenti valori validi. Con questa combinazione delle variabili, il calcolo del valore di uscita non riesce.
0008_0000	Causa dell'errore: Uno o più parametri del filtro hanno un valore non valido durante l'esecuzione del calcolo del valore di uscita (Reset = FALSE). Reazione all'errore: Nel parametro Output viene emesso il valore di uscita sostitutivo configurato nella variabile ErrorMode. Rimedio: Accertarsi che per i valori dei parametri del filtro siano soddisfatte le seguenti condizioni: • $0.0 < \text{Frequency} < 0.5 / \text{CycleTime.Value}$ • $0.0 \leq \text{Bandwidth} < 0.5 / \text{CycleTime.Value} - \text{Frequency}$ • $0 \leq \text{Type} \leq 3$ • $0 \leq \text{Characteristic} \leq 2$ • $0 \leq \text{Order} \leq 10$
0010_0000	Causa dell'errore: Il riconoscimento automatico del tempo di ciclo non è andato a buon fine perché Filter_Universal non viene richiamato in un OB di schedulazione orologio. Reazione all'errore: Nel parametro Output viene emesso il valore di uscita sostitutivo configurato nella variabile ErrorMode. Rimedio: Assicurarsi che Filter_Universal venga richiamata in un OB di schedulazione orologio. Informazione supplementare: Il rilevamento automatico del tempo di ciclo può essere disattivato impostando la variabile CycleTime.EnableDetection = FALSE. Il tempo di ciclo deve essere calcolato manualmente nella variabile CycleTime.Value. Il richiamo di Filter_Universal al di fuori di un OB di schedulazione orologio può avere effetti negativi sul comportamento del filtro perché il tempo di ciclo effettivo oscilla.
0020_0000	Causa dell'errore: La variabile (configurata con StartMode) per l'inizializzazione del parametro Output al primo richiamo dell'istruzione non ha un valore REAL valido. Reazione all'errore: Al primo richiamo dell'istruzione nel parametro Output viene emesso il valore di uscita sostitutivo configurato nella variabile ErrorMode. Ai richiami successivi Filter_Universal calcola il valore di uscita basandosi su questo valore di uscita sostitutivo. Rimedio: Accertarsi che la variabile utilizzata per l'inizializzazione del parametro Output sia un valore REAL valido ($\neq$ NaN ad es. 16#7FFF_FFFF). Se è impostato Reset = FALSE, l'inizializzazione diventa effettiva al primo richiamo dell'istruzione dopo la commutazione dello stato di funzionamento della CPU da STOP a RUN. La variabile utilizzata per l'inizializzazione del parametro Output è in funzione di StartMode: • StartMode = 1: SubstituteOutput • StartMode = 2: Output

ErrorBits (DW#16#...)	Descrizione
0040_0000	Causa dell'errore: La variabile CycleTime.Value durante l'esecuzione del calcolo del valore di uscita (Reset = FALSE) non ha un valore valido. Reazione all'errore: Nel parametro Output viene emesso il valore di uscita sostitutivo configurato nella variabile ErrorMode. Rimedio: Accertarsi che siano soddisfatte le seguenti condizioni: • $0.0 < \text{CycleTime.Value} \leq 3.402823\text{e}+38$ • CycleTime.Value è un valore REAL valido ($\neq$ NaN ad es. 16#7FFF_FFFF) Informazione supplementare: Per il calcolo automatico del valore della variabile CycleTime.Value, impostare la variabile CycleTime.EnableDetection su TRUE.
0080_0000	Causa dell'errore: Durante il rilevamento automatico del tempo di ciclo si è verificato un errore interno. Reazione all'errore: Nel parametro Output viene emesso il valore di uscita sostitutivo configurato nella variabile ErrorMode. Rimedio: Assicurarsi che Filter_Universal venga richiamato in un OB di schedulazione orologio. Se l'errore è ancora presente rivolgersi al SIMATIC Customer Support. Informazione supplementare: Il rilevamento automatico del tempo di ciclo può essere disattivato impostando la variabile CycleTime.EnableDetection = FALSE. Il tempo di ciclo deve essere calcolato manualmente nella variabile CycleTime.Value.

Indice analitico

C

CONT_C

- Funzionamento, 445
- Schema a blocchi, 447
- Parametri di ingresso, 448
- Parametri di uscita, 449

CONT_S

- Istruzione, 450
- Funzionamento, 451
- Schema a blocchi, 452
- Parametri di ingresso, 453
- Parametri di uscita, 454

F

- Filter_DT1, 580
- Filter_PT1, 554
- Filter_PT2, 567
- Filter_Universal, 594

O

Oggetti tecnologici

- PID_Compact, 78
- PID_3Step, 124
- PID_Temp, 166
- CONT_C, 216
- CONT_S, 220
- TCONT_CP, 223
- TCONT_S, 246

P

PID_3Step

- Istruzione, 322
- Parametri di ingresso, 331
- Parametri di uscita, 333
- Parametri di transito, 334
- Istruzione, 354
- Parametri di ingresso, 362
- Parametri di uscita, 364
- Variabili statiche, 365

PID_Compact

- Parametri di ingresso, 268
- Parametri di uscita, 269
- Parametri di transito, 270
- Istruzione, 300
- Parametri di ingresso, 303
- Parametri di uscita, 304
- Variabili statiche, 305

PID_Temp

- Regolazione in cascata, 201
- Applicazioni multizona, 208
- Modo di funzionamento, 391
- Parametro di ingresso, 397
- Parametri di uscita, 398
- Parametri di transito, 400
- Modo, 400
- Cascata, 400
- Parametri State e Mode PID_Temp, 428
- Parametro ErrorBits, 435
- Variabile ActivateRecoverMode, 438
- Variabile Warning, 439
- PwmPeriode, 441

Polyline, 497

PULSEGEN

- Istruzione, 455
- Funzionamento, 456

PULSEGEN

- Parametri di ingresso, 463
- Parametri di uscita, 464

R

RampFunction, 516

RampSoak, 529

- Regolatore software
- Progettazione, 43

S

Simbolo

- per confronto di valori, 53

SplitRange, 509

T

TCONT_CP

- Istruzione, [464](#)
- Funzionamento, [465](#)
- Parametri di ingresso, [478](#)
- Parametri di uscita, [479](#)
- Parametri di transito, [480](#)
- Variabili statiche, [480](#)

TCONT_S

- Istruzione, [486](#)
- Funzionamento, [488](#)
- Parametri di ingresso, [493](#)
- Parametri di uscita, [494](#)
- Parametri di transito, [494](#)
- Variabili statiche, [495](#)

V

Valori

- confronto, [53](#)