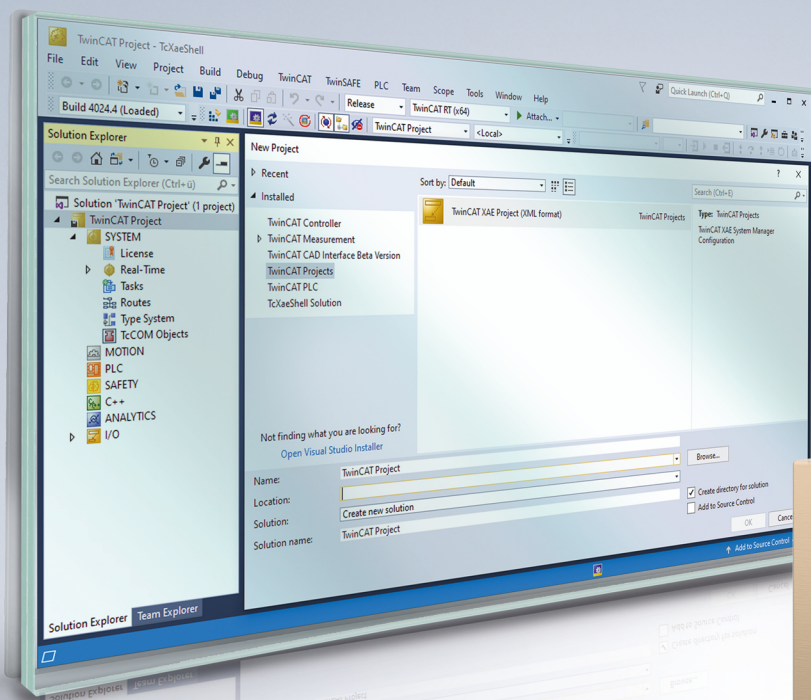


BECKHOFF New Automation Technology

Handbuch | DE

TF8540

TwinCAT 3 | Plastic Processing Framework



Inhaltsverzeichnis

1	Vorwort.....	5
1.1	Hinweise zur Dokumentation	5
1.2	Sicherheitshinweise	6
1.3	Hinweise zur Informationssicherheit	7
2	Einleitung.....	8
3	PLC-Temperaturregler	10
3.1	Einleitung	10
3.2	Übersicht.....	11
3.3	Bausteine	13
3.3.1	FB_TempCtrlMainBody_TcPfw()	13
3.3.2	FB_TempCtrlMainBody_TcPfw_TC3().....	14
3.3.3	FB_TempCtrlEnableZone_TcPfw().....	16
3.3.4	FB_TempCtrlStandByZone_TcPfw().....	17
3.3.5	FB_TempCtrlState_TcPfw().....	17
3.3.6	FB_TempCtrlCallback_TcPfw().....	19
3.3.7	FB_FcMainBody_TcPfw()	19
3.4	Utilities.....	20
3.4.1	Stand Alone.....	20
3.4.2	FB_xL3403_TcPfw()	27
3.4.3	FB_TempCtrlAdaptFm33xx_TcPfw().....	28
3.4.4	FB_TempCtrlClearSupply_TcPfw().....	29
3.4.5	FB_TempCtrlClearZones_TcPfw().....	29
3.4.6	FB_TermCoeRead_TcPfw().....	29
3.4.7	FB_TermCoeWrite_TcPfw().....	31
3.4.8	FB_TermRegRead_TcPfw().....	32
3.4.9	FB_TermRegWrite_TcPfw().....	34
3.5	Aufzählungen	35
3.5.1	E_TcPfw_TctrlPowerTerminal.....	35
3.5.2	E_TcPfw_TctrlOutSelect	36
3.5.3	E_TcPfw_TempSensType	36
3.5.4	E_TcPfw_TerminalType.....	37
3.6	Strukturen.....	40
3.6.1	Mapping	40
3.6.2	ST_TcPfw_TempMparamFromHmi_Itf	46
3.6.3	ST_TcPfw_TempPparamFromHmi_Itf.....	56
3.6.4	ST_TcPfw_TempToHmi_Itf.....	57
3.6.5	ST_TcPfw_SupplyParam	58
3.6.6	ST_TcPfw_TempCtrl_Itf.....	60
3.6.7	ST_TcPfw_PowerMeasurement_Cfg.....	64
3.7	Knowledge Base	65
3.7.1	Inbetriebnahme	65
3.7.2	Alarmhandling	75
3.7.3	Heizstromüberwachung	77
3.7.4	Parameter Speichern/Laden	79

3.7.5	Versorgungsgruppen.....	82
3.7.6	FAQs	84
3.7.7	Globale Variablen.....	87
4	PLC-Zeitschaltuhr	92
4.1	Übersicht	92
4.2	Bausteine	92
4.2.1	FB_ClockTimer_TcPfw().....	92
4.2.2	FB_ClockTimerParamSave_TcPfw()	92
4.2.3	FB_ClockTimerParamLoad_TcPfw()	93
4.3	Strukturen.....	94
4.3.1	ST_TcPfw_ClockTimerItf	94
4.3.2	ST_TcPfw_ClockTimerCam.....	95
4.4	Knowledge Base	95
4.4.1	Inbetriebnahme	95
4.4.2	Globale Variablen.....	97
4.4.3	FAQs	98
5	PLC-Alarmvisualisierung	99
5.1	Übersicht	99
5.2	Bausteine	99
5.2.1	FB_MsgAppend_TcTvA().....	99
5.2.2	FB_MsgClearPending_TcTvA()	100
5.2.3	FB_MsgClearSignal_TcTvA()	101
5.2.4	FB_MsgDeactivate_TcTvA().....	102
5.2.5	FB_MsgGarbageCollect_TcTvA().....	102
5.2.6	FB_MsgUpdateTime_TcTvA()	103
5.3	Strukturen.....	104
5.3.1	ST_TcTvA_Alarm_Itf.....	104
5.4	Knowledge Base	105
5.4.1	Inbetriebnahme	105
5.4.2	Globale Variablen.....	106
5.4.3	FAQs	106

1 Vorwort

1.1 Hinweise zur Dokumentation

Diese Beschreibung wendet sich ausschließlich an ausgebildetes Fachpersonal der Steuerungs- und Automatisierungstechnik, das mit den geltenden nationalen Normen vertraut ist.

Zur Installation und Inbetriebnahme der Komponenten ist die Beachtung der Dokumentation und der nachfolgenden Hinweise und Erklärungen unbedingt notwendig.

Das Fachpersonal ist verpflichtet, für jede Installation und Inbetriebnahme die zu dem betreffenden Zeitpunkt veröffentlichte Dokumentation zu verwenden.

Das Fachpersonal hat sicherzustellen, dass die Anwendung bzw. der Einsatz der beschriebenen Produkte alle Sicherheitsanforderungen, einschließlich sämtlicher anwendbaren Gesetze, Vorschriften, Bestimmungen und Normen erfüllt.

Disclaimer

Diese Dokumentation wurde sorgfältig erstellt. Die beschriebenen Produkte werden jedoch ständig weiter entwickelt.

Wir behalten uns das Recht vor, die Dokumentation jederzeit und ohne Ankündigung zu überarbeiten und zu ändern.

Aus den Angaben, Abbildungen und Beschreibungen in dieser Dokumentation können keine Ansprüche auf Änderung bereits gelieferter Produkte geltend gemacht werden.

Marken

Beckhoff®, TwinCAT®, TwinCAT/BSD®, TC/BSD®, EtherCAT®, EtherCAT G®, EtherCAT G10®, EtherCAT P®, Safety over EtherCAT®, TwinSAFE®, XFC®, XTS® und XPlanar® sind eingetragene und lizenzierte Marken der Beckhoff Automation GmbH.

Die Verwendung anderer in dieser Dokumentation enthaltenen Marken oder Kennzeichen durch Dritte kann zu einer Verletzung von Rechten der Inhaber der entsprechenden Bezeichnungen führen.

Patente

Die EtherCAT-Technologie ist patentrechtlich geschützt, insbesondere durch folgende Anmeldungen und Patente:

EP1590927, EP1789857, EP1456722, EP2137893, DE102015105702

mit den entsprechenden Anmeldungen und Eintragungen in verschiedenen anderen Ländern.



EtherCAT® ist eine eingetragene Marke und patentierte Technologie lizenziert durch die Beckhoff Automation GmbH, Deutschland

Copyright

© Beckhoff Automation GmbH & Co. KG, Deutschland.

Weitergabe sowie Vervielfältigung dieses Dokuments, Verwertung und Mitteilung seines Inhalts sind verboten, soweit nicht ausdrücklich gestattet.

Zuwendungen verpflichten zu Schadenersatz. Alle Rechte für den Fall der Patent-, Gebrauchsmuster- oder Geschmacksmustereintragung vorbehalten.

1.2 Sicherheitshinweise

Sicherheitsbestimmungen

Beachten Sie die folgenden Sicherheitshinweise und Erklärungen!
Produktspezifische Sicherheitshinweise finden Sie auf den folgenden Seiten oder in den Bereichen Montage, Verdrahtung, Inbetriebnahme usw.

Haftungsausschluss

Die gesamten Komponenten werden je nach Anwendungsbestimmungen in bestimmten Hard- und Software-Konfigurationen ausgeliefert. Änderungen der Hard- oder Software-Konfiguration, die über die dokumentierten Möglichkeiten hinausgehen, sind unzulässig und bewirken den Haftungsausschluss der Beckhoff Automation GmbH & Co. KG.

Qualifikation des Personals

Diese Beschreibung wendet sich ausschließlich an ausgebildetes Fachpersonal der Steuerungs-, Automatisierungs- und Antriebstechnik, das mit den geltenden Normen vertraut ist.

Erklärung der Symbole

In der vorliegenden Dokumentation werden die folgenden Symbole mit einem nebenstehenden Sicherheitshinweis oder Hinweistext verwendet. Die Sicherheitshinweise sind aufmerksam zu lesen und unbedingt zu befolgen!

GEFAHR

Akute Verletzungsgefahr!

Wenn der Sicherheitshinweis neben diesem Symbol nicht beachtet wird, besteht unmittelbare Gefahr für Leben und Gesundheit von Personen!

WARNUNG

Verletzungsgefahr!

Wenn der Sicherheitshinweis neben diesem Symbol nicht beachtet wird, besteht Gefahr für Leben und Gesundheit von Personen!

VORSICHT

Schädigung von Personen!

Wenn der Sicherheitshinweis neben diesem Symbol nicht beachtet wird, können Personen geschädigt werden!

HINWEIS

Schädigung von Umwelt oder Geräten

Wenn der Hinweis neben diesem Symbol nicht beachtet wird, können Umwelt oder Geräte geschädigt werden.



Tipp oder Fingerzeig

Dieses Symbol kennzeichnet Informationen, die zum besseren Verständnis beitragen.

1.3 Hinweise zur Informationssicherheit

Die Produkte der Beckhoff Automation GmbH & Co. KG (Beckhoff) sind, sofern sie online zu erreichen sind, mit Security-Funktionen ausgestattet, die den sicheren Betrieb von Anlagen, Systemen, Maschinen und Netzwerken unterstützen. Trotz der Security-Funktionen sind die Erstellung, Implementierung und ständige Aktualisierung eines ganzheitlichen Security-Konzepts für den Betrieb notwendig, um die jeweilige Anlage, das System, die Maschine und die Netzwerke gegen Cyber-Bedrohungen zu schützen. Die von Beckhoff verkauften Produkte bilden dabei nur einen Teil des gesamtheitlichen Security-Konzepts. Der Kunde ist dafür verantwortlich, dass unbefugte Zugriffe durch Dritte auf seine Anlagen, Systeme, Maschinen und Netzwerke verhindert werden. Letztere sollten nur mit dem Unternehmensnetzwerk oder dem Internet verbunden werden, wenn entsprechende Schutzmaßnahmen eingerichtet wurden.

Zusätzlich sollten die Empfehlungen von Beckhoff zu entsprechenden Schutzmaßnahmen beachtet werden. Weiterführende Informationen über Informationssicherheit und Industrial Security finden Sie in unserem <https://www.beckhoff.de/secguide>.

Die Produkte und Lösungen von Beckhoff werden ständig weiterentwickelt. Dies betrifft auch die Security-Funktionen. Aufgrund der stetigen Weiterentwicklung empfiehlt Beckhoff ausdrücklich, die Produkte ständig auf dem aktuellen Stand zu halten und nach Bereitstellung von Updates diese auf die Produkte aufzuspielen. Die Verwendung veralteter oder nicht mehr unterstützter Produktversionen kann das Risiko von Cyber-Bedrohungen erhöhen.

Um stets über Hinweise zur Informationssicherheit zu Produkten von Beckhoff informiert zu sein, abonnieren Sie den RSS Feed unter <https://www.beckhoff.de/secinfo>.

2 Einleitung

Das Plastics Processing Framework erleichtert den Aufbau von PLC-Projekten für Kunststoffmaschinen. Ziel des Plastics Processing Frameworks ist es, die Basiselemente, die in jeder Kunststoffmaschine vorhanden sind, in einem Paket zu vereinen.

Im Einzelnen werden folgende Themen abgedeckt:

- Temperaturregelung
 - Autotune
 - Laden/ Speichern von Parametern
- Meldungsausgabe
- Filterbausteine
- Zeitschaltuhren

Die vorliegende Dokumentation teilt sich in drei verschiedene Module:

- Im ersten Bereich wird der Regler genauer beschrieben.
 - Funktion des Reglers
 - Lade- und Speicherfunktionen
 - Filterfunktionen
- Im zweiten Bereich wird eine Wochenzeitschaltuhr vorgestellt, um den Regler auch zeitgesteuert zu aktivieren.
- Der letzte Bereich beschreibt Funktionen, um ein Meldehandling zu realisieren
 - Meldungen in eine Liste hinzufügen bzw. aus einer Liste entfernen,
 - Meldungen aktivieren bzw. deaktivieren

Jedes Modul gliedert sich wie folgt:

- Übersicht: Gibt einen Überblick zu diesem Modul.
- Bausteine: Es werden die einzelnen Bausteine zu diesem Modul erklärt.
- Strukturen: Es werden die einzelnen Strukturen zu diesem Baustein erklärt.
- Knowledge Base: Es sind hier häufig gestellte Fragen sowie eine Inbetriebnahmeanleitung zu finden.

● **Softwarestand**

i Die Dokumentation beruht auf dem Softwarestand 1.0.9 der PfwLib_Processing.lib. In älteren Versionen der PfwLib_Processing.lib kann nicht garantiert werden, dass alle hier erklärten Funktionen implementiert sind.

i Knowledge Base

Hier werden alle vorhandenen Funktionen, Bausteine und Datentypen dieses Framework-Teils aufgelistet. Antworten zu häufig gestellten Fragen sowie Hinweise zum Einsatz des Frameworks, zu Inbetriebnahme und Problemanalyse sowie Beispielprojekte finden Sie in der Knowledge Base. Beachten Sie die Hinweise zur Dokumentation.

Einige der hier aufgeführten Komponenten sind nicht für die Benutzung durch eine Applikation vorgesehen. Ihr Vorhandensein, Interface und Verhalten wird dementsprechend nicht garantiert. Da ein TwinCAT PLC Framework jedoch strikt offen ist, besteht keine Möglichkeit, diese internen Komponenten zu verbergen. Es sollte jedoch unbedingt darauf verzichtet werden, diese entsprechend mit (internal use only) oder (not recommended) gekennzeichneten Komponenten aus einer Applikation heraus direkt zu benutzen. Sollte eine dieser Komponenten für Sie von praktischem Nutzen sein, nehmen Sie bitte Kontakt mit unserem Support auf. Wir werden dann die Möglichkeit prüfen, ob Ihnen ein Baustein unabhängig von der Library zur eigenverantwortlichen Verfügung gestellt werden kann.

Sollte die Library Bausteine, Typen oder Konstanten enthalten, die in der Dokumentation nicht aufgeführt werden, handelt es sich um nicht freigegebene Elemente, die Gegenstand der aktuellen Softwarepflege und -entwicklung sind. Diese Elemente dürfen auf keinen Fall in einer Applikation direkt verwendet werden, da sie in der Regel noch nicht getestet sind.

Das Framework macht eine Reihe von Hardware- und Software-Voraussetzungen.

3 PLC-Temperaturregler

3.1 Einleitung

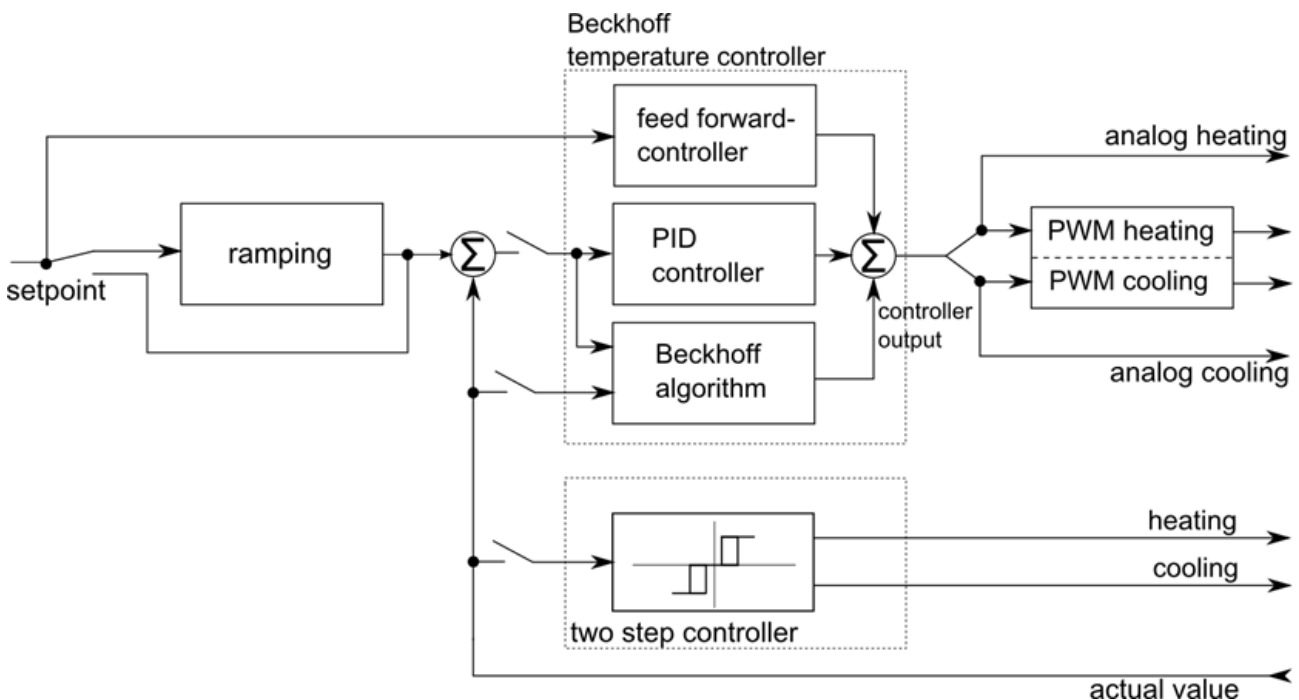
Temperaturregelung des Plastics Processing Frameworks

Der Temperaturregler des Plastics Processing Frameworks wurde auf

- ein schnelles Anregeln,
- überschwingfreies Führungsverhalten,
- schnelles Ausregeln von Störungen,
- optimales Heiz- und Kühlverhältnis,
- selbstständige Parameterermittlung und
- exaktes Führungsverhalten während der Produktion

optimiert. Diese Anforderungen sind mit einem normalen PID-Regler nicht zu realisieren, wodurch neben dem PID-Regler weitere Algorithmen implementiert wurden.

Das folgende Bild zeigt schematisch den Aufbau des Reglers.



Ermittlung der Streckenparameter:

- Findet über das Wendetangentenverfahren statt.
- Getrennte Regelparameterermittlung für das Heizen und Kühlen.
- Ermittlung des Vorsteueranteils (Idle-Load):
 - Mindert das Überschwingen bei Sollwertänderung.
 - Parameterermittlung muss im eingeregelteten Zustand durchgeführt werden.
 - Dauert nur einige Sekunden und beeinflusst nicht die Ist-Temperatur.

Das Regelverhalten:

- Große Sollwertsprünge: Regelt der Beckhoff Algorithmus. Nach dem Erreichen der Solltemperatur wird der PID-Regler hinzu geschaltet. Durch die optimal aufeinander abgestimmten Regler-Kombinationen ist ein schnelles und überschwingfreies Ausregeln möglich.
- Kleine Sollwertsprünge: Werden durch einen modifizierten PID-Regler geregelt, welcher auf den Umgang mit großen Totzeiten optimiert wurde.

- Exaktes Führungsverhalten: Um ein exaktes Führungsverhalten während der Produktion zu garantieren, sind neben dem PID-Regler weitere Stabilisierungsmaßnahmen, wie zum Beispiel die Berücksichtigung von Transport- und Friktionsenergie, getroffen worden.

Weitere Besonderheiten des Reglers:

- Unterstützung üblicher Thermoelemente und Pt-Sensoren.
- Error-Heating: Ermöglicht das Produzieren auch bei Temperaturfühlerdefekt.
- Extruder-Kompensation: Berechnet die Scherleistung und den Materialtransport in den Prozesswert mit ein.
- Zoning: Verteilung der Einschaltzeiten innerhalb eines PWM-Zyklus vermeidet unnötige Leistungsspitzen.
- Leistungskontrolle Heizbänder: Misst und überprüft die Heizleistung der Heizbänder.
- I/O-Swap: Ist ein Ein- bzw. Ausgangskanal defekt, so kann dieser vom Bediener im laufenden Betrieb vom einen zum anderen Kanal umgelenkt werden.
- Der Temperaturregler übernimmt die gesamte Klemmenkommunikation, sodass der Regler lediglich parametrieren muss.
- Es stehen Lade- und Speicherroutinen zur separaten Speicherung der Produkt- und Maschinendaten zur Verfügung.
- Es stehen verschiedene Filterfunktionen (z.B. HMI-Filter) zur Verfügung.
- Es stehen Funktionen zum Alarmhandling bereit.
- Die Bibliothek überwacht:
 - den Betriebszustand der Klemmen,
 - das Autotuning,
 - ob die aktuelle Temperatur bestimmte Grenzwerte überschreitet,
 - ob zur Heizleistungsausgabe eine angemessene Temperaturänderung vorliegt.

Nützliche Hinweise:

- Beispielprogramm: siehe Knowledge Base unter [Inbetriebnahme \[► 95\]](#).
- Inbetriebnahmeanleitung: siehe [Knowledge Base \[► 65\]](#).
- Wie binde ich weitere Funktionen ein (Leistungsmessung, Scope-Funktionen usw.): siehe Knowledge Base in den [FAQs \[► 84\]](#).

3.2 Übersicht

Bausteine im Temp.-Ctrl. Ordner

Name	Beschreibung
FB_TempCtrlMainBody_TcPfw [► 13]	Zyklisch aufzurufender Framework-Baustein.
FB_TempCtrlEnableZone_TcPfw [► 16]	Zonen der Temperaturregelung werden in den aktiven oder passiven Zustand geschaltet.
FB_TempCtrlStandByZone_TcPfw [► 17]	Zonen der Temperaturregelung werden in den StandBy- oder Aktiv-Zustand geschaltet.
FB_TempCtrlState_TcPfw [► 17]	Dieser Baustein ermittelt eine Reihe von Status-Informationen einer Zone.
FB_TempCtrlCallback_TcPfw [► 19]	Dieser Baustein prüft einmalig den Typ der verknüpften Klemme und parametrieren sie entsprechend dem vorgegebenen Sensortyp.

Utilities

Name	Beschreibung
FB_TempCtrlAdaptFm33xx_TcPfw [► 28]	Die E/A-Daten von FM3312 oder FM3332 Feldbusmodulen werden an die E/A-Strukturen der Library angepasst.
FB_TempCtrlClearSupply_TcPfw [► 29]	Die Daten der Versorgungsgruppen der Temperaturregelung werden initialisiert.
FB_TempCtrlClearZones_TcPfw [► 29]	Die Daten der Regler-Zonen der Temperaturregelung werden initialisiert.
FB_TermCoeRead_TcPfw [► 29]	Baustein wird für den Lesezugriff auf EL-Klemmen genutzt.
FB_TermCoeWrite_TcPfw [► 31]	Baustein wird für den Schreibzugriff auf EL-Klemmen genutzt.
FB_TermRegRead_TcPfw [► 32]	Baustein wird für den Lesezugriff auf KL-Klemmen genutzt.
FB_TermRegWrite_TcPfw [► 34]	Baustein wird für den Schreibzugriff auf KL-Klemmen genutzt.
FUN_TempCtrlSensorTypeCode_TcPfw	Die Funktion liefert zu einem textuellen Sensortyp die numerische Übersetzung.
FUN_TempCtrlSensorTypeName_TcPfw	Die Funktion liefert zu einem numerischen Sensortyp die textuelle Übersetzung.

Framework-Bausteine im StandAlone Ordner

Name	Beschreibung
FB_TempParamLoad_TcPfw [► 25]	Dieser Baustein liest die Parameter einer Zone aus einer Datei.
FB_TempParamSave_TcPfw [► 22]	Dieser Baustein schreibt die Parameter einer Zone in eine Datei.
FB_TempParamSaveP_TcPfw [► 24]	Dieser Baustein schreibt die Parameter einer Zone in eine Datei.

Datentypen: Aufzählungen

Name	Beschreibung
E_TcPfw_TctrlOutSelect [► 36]	Kennungen für die Auswahl des Ausgabesignals einer Zone der Temperaturregelung.
E_TcPfw_TempSensType [► 36]	Kennungen für die unterstützten Typen von Temperatursensoren.
E_TcPfw_TerminalType [► 37]	Grundlagen des Frameworks: Kennungen für die unterstützten Typen von E/A-Klemmen.

Datentypen: Strukturtypen

Name	Beschreibung
ST_TcPfw_FM3332_Input [▶ 45]	Eine solche Struktur enthält die Eingangsdaten eines FM3312 oder FM3332 Feldbusmoduls.
ST_TcPfw_SupplyParam [▶ 58]	Eine solche Struktur enthält die Parameter und Laufzeitdaten einer Versorgungsgruppe.
ST_TcPfw_TempCtrl_Itf [▶ 60]	Zustand und Aktivität einer Zone der Temperaturregelung.
ST_TcPfw_TempCtrlInput [▶ 41]	Eingangsdaten einer Zone der Temperaturregelung.
ST_TcPfw_TempCtrlOutput [▶ 40]	Ausgabedaten einer Zone der Temperaturregelung.
ST_TcPfw_TempMparamFromHmi_Itf [▶ 46]	Maschinendaten einer Zone der Temperaturregelung.
ST_TcPfw_TempPparamFromHmi_Itf [▶ 56]	Produktdaten einer Zone der Temperaturregelung.
ST_TcPfw_TempToHmi_Itf [▶ 57]	Visualisierungsdaten einer Zone der Temperaturregelung.

3.3 Bausteine

3.3.1 FB_TempCtrlMainBody_TcPfw()

FB_TempCtrlMainBody_TcPfw	
— ConfigEnable	BOOL
— tCycle	LREAL
— Looptest_Enable	BOOL
— Callback_Enable	BOOL
— Simu_Enable	BOOL
— Simu_DisCharge	BOOL

Dieser Baustein muss zwingend in der Applikation aufgerufen werden. Er organisiert intern die komplette Temperaturregelung.

Syntax

```
VAR_INPUT
ConfigEnable    : BOOL;
tCycle          : LREAL;
Looptest_Enable: BOOL;
Callback_Enable: BOOL;
Simu_Enable     : BOOL:=FALSE;
Simu_DisCharge  : BOOL:=FALSE;
END_VAR
```

 **Eingänge**

Name	Typ	Beschreibung
ConfigEnable	BOOL	Wenn TRUE, ist die Konfiguration gültig.
tCycle	LREAL	Zykluszeit
Looptest_Enable	BOOL	Eine Strommessung wird durchgeführt.
Callback_Enable	BOOL	Prüft einmalig (durch Aufruf von FB_TempCtrlCallback_TcPfw()) den Typ der verknüpften Klemme und parametrier sie entsprechend dem vorgegebenen Sensortyp.
Simu_Enable	BOOL	Wird für interne Simulationszwecke verwendet.
Simu_DisCharge	LREAL	Wird für interne Simulationszwecke verwendet.

Verhalten des Bausteins:

Bei jedem Aufruf prüft der Baustein die globale Variable bPfw_UseTempControl. Ist diese Variable TRUE und signalisiert ConfigEnable eine gültige Konfiguration, wird der Baustein aktiv:

- Im ersten Zyklus ruft der Baustein einen internen Baustein vom Typ FB_internal_tmpCtrlInitlinks_TcPfw() auf, um die von der Temperaturregelung verwendeten Strukturen zu initialisieren.
- Wenn Looptest_Enable gesetzt ist, wird eine Strommessung durchgeführt.
- Wenn Callback_Enable gesetzt ist, wird ein Baustein vom Typ FB_TempCtrlCallback_TcPfw() aufgerufen. Dieser Baustein prüft einmalig die Type der verknüpften Klemme und parametriert sie entsprechend dem vorgegebenen Sensortyp.
- Für jede Zone der Regelung werden folgende Aktivitäten durchgeführt:
 - Wenn in aaaPfwTempMparamFromHmi das Signal Update gesetzt ist, werden bei Bedarf die eingetragenen Werte auf die zulässigen Wertebereiche begrenzt und in die Regelung übernommen. Update wird gelöscht.
 - Wenn in aaaPfwTempPparamFromHmi das Signal Update gesetzt ist, werden bei Bedarf die eingetragenen Werte auf die zulässigen Wertebereiche begrenzt und in die Regelung übernommen. Update wird gelöscht.
 - Wenn InUse in aaaPfwTempToHmi auf TRUE gesetzt ist, werden folgende Schritte durchgeführt:
 - Es wird ein FB_CTRL_TempController() Baustein aus der TcTempCtrl.LIB Library aufgerufen.
 - In out_PfwTempCtrlOutput werden YPWMPos, YPWMNeg, YDigPos, YDigNeg und Yanalog aktualisiert.
 - Verschiedene Signale (aaaTempFault_Reset, Autotune in aaaPfwTempMparamFromHmi usw.) steuern das Zurücksetzen von Störzuständen oder aktivieren das Autotuning.
 - Wenn vom Reglerbaustein oder aus der E/A-Schnittstelle ein Problem gemeldet wird, werden die entsprechenden Events aktiviert.
 - Diverse Daten in aaaPfwTempToHmi werden aktualisiert.
- Wenn die Isttemperatur mindestens einer Zone unterhalb von fAbsoluteLow in aaaPfwTempMparamFromHmi liegt, wird aaaTempAlarm_AbsoluteLow signalisiert.
- Wenn die Isttemperatur mindestens einer Zone oberhalb von fAbsoluteHigh in aaaPfwTempPparamFromHmi liegt, wird aaaTempAlarm_AbsoluteHigh signalisiert.



Am Ende des Zyklus wird aaaTempFault_Reset automatisch gelöscht.



Wenn Callback_Enable nicht gesetzt ist, wird kein Baustein vom Typ FB_TempCtrlCallback_TcPfw() aufgerufen. In diesem Fall ist von der Applikation sicher zu stellen, dass die E/A-Elektronik zum Sensortyp passt.

3.3.2 FB_TempCtrlMainBody_TcPfw_TC3()

FB_TempCtrlMainBody_TcPfw_TC3	
—	iParamLoadCheck <i>I_ParamLoadCheck</i>
—	ConfigEnable <i>BOOL</i>
—	tCycle <i>LREAL</i>
—	Looptest_Enable <i>BOOL</i>
—	Callback_Enable <i>BOOL</i>
—	Simu_Enable <i>BOOL</i>
—	Simu_DisCharge <i>BOOL</i>
—	Scope_TempCtrlVariables <i>FB_Scope_TempCtrlVariables</i>

Dieser Baustein muss zwingend zyklisch in der Applikation aufgerufen werden. Er organisiert intern die komplette Temperaturregelung.

Syntax

```
VAR_INPUT
  iParamLoadCheck:I_ParamLoadCheck;
  ConfigEnable      : BOOL;
  tCycle            : LREAL;
  Looptest_Enable  : BOOL;
  Callback_Enable   : BOOL;
  Simu_Enable       : BOOL:=FALSE;
  Simu_DisCharge    : BOOL:=FALSE;
  Scope_TempCtrlVariables: FB_Scope_TempCtrlVariables;
END_VAR
```

 **Eingänge**

Name	Typ	Beschreibung
iParamLoadCheck	I_ParamLoadCheck	Optionales Interface zum (externen) Speichern der Temperaturzonen.
ConfigEnable	BOOL	Wenn TRUE, ist die Konfiguration gültig
tCycle	LREAL	Zykluszeit
Looptest_Enable	BOOL	Eine Strommessung wird durchgeführt.
Callback_Enable	BOOL	Prüft einmalig (durch Aufruf von FB_TempCtrlCallback_TcPfw()) den Typ der verknüpften Klemme und parametriert sie entsprechend dem vorgegebenen Sensortyp.
Simu_Enable	BOOL	Wird für interne Simulationszwecke verwendet.
Simu_DisCharge	BOOL	Wird für interne Simulationszwecke verwendet.
Scope_TempCtrlVariables	FB_Scope_TempCtrlVariables	Anzeige der einzelnen Temperaturzonen und deren internen Variablen.

Verhalten des Bausteins:

Bei jedem Aufruf prüft der Baustein die globale Variable bPfw_UseTempControl. Ist diese Variable TRUE und signalisiert ConfigEnable eine gültige Konfiguration, wird der Baustein aktiv:

- Im ersten Zyklus ruft der Baustein einen internen Baustein vom Typ FB_internal_tmpCtrlInitlinks_TcPfw() auf, um die von der Temperaturregelung verwendeten Strukturen zu initialisieren.
- Wenn Looptest_Enable gesetzt ist, wird eine Strommessung durchgeführt.
- Wenn Callback_Enable gesetzt ist, wird ein Baustein vom Typ FB_TempCtrlCallback_TcPfw() aufgerufen. Dieser Baustein prüft einmalig die Type der verknüpften Klemme und parametriert sie entsprechend dem vorgegebenen Sensortyp.
- Für jede Zone der Regelung werden folgende Aktivitäten durchgeführt:
 - Wenn in aaaPfwTempMparamFromHmi das Signal Update gesetzt ist, werden bei Bedarf die eingetragenen Werte auf die zulässigen Wertebereiche begrenzt und in die Regelung übernommen. Update wird gelöscht.
 - Wenn in aaaPfwTempPparamFromHmi das Signal Update gesetzt ist, werden bei Bedarf die eingetragenen Werte auf die zulässigen Wertebereiche begrenzt und in die Regelung übernommen. Update wird gelöscht.
 - Wenn InUse in aaaPfwTempToHmi auf TRUE gesetzt ist, werden folgende Schritte durchgeführt:
 - Es wird ein FB_CTRL_TempController() Baustein aus der TcTempCtrl.LIB Library aufgerufen.

- In out_PfwTempCtrlOutput werden YPWMPos, YPWMNeg, YDigPos, YDigNeg und Yanalog aktualisiert.
- Verschiedene Signale (aaaTempFault_Reset, Autotune in aaaPfwTempMparamFromHmi usw.) steuern das Rücksetzen von Stöorzuständen oder aktivieren das Autotuning.
- Wenn vom Reglerbaustein oder aus der E/A-Schnittstelle ein Problem gemeldet wird, werden die entsprechenden Events aktiviert.
- Diverse Daten in aaaPfwTempToHmi werden aktualisiert.
- Wenn die Isttemperatur mindestens einer Zone unterhalb von fAbsoluteLow in aaaPfwTempMparamFromHmi liegt, wird aaaTempAlarm_AbsoluteLow signalisiert.
- Wenn die Isttemperatur mindestens einer Zone oberhalb von fAbsoluteHigh in aaaPfwTempPparamFromHmi liegt, wird aaaTempAlarm_AbsoluteHigh signalisiert.

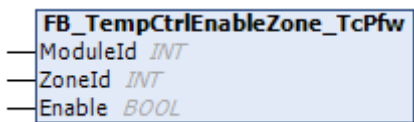


Am Ende des Zyklus wird aaaTempFault_Reset automatisch gelöscht.



Wenn Callback_Enable nicht gesetzt ist, wird kein Baustein vom Typ FB_TempCtrlCallback_TcPfw() aufgerufen. In diesem Fall ist von der Applikation sicher zu stellen, dass die E/A-Elektronik zum Sensortyp passt.

3.3.3 FB_TempCtrlEnableZone_TcPfw()



Einer oder mehrere Bausteine dieses Typs werden aus einem Baustein der Applikation aufgerufen, um einzelne Zonen, Zonengruppen oder alle Zonen der Temperaturregelung in den aktiven oder passiven Zustand zu schalten.

Syntax

```
VAR_INPUT
ModuleId: INT:= -1;
ZoneId : INT:= -1;
Enable : BOOL:= TRUE;
END_VAR
```

Eingänge

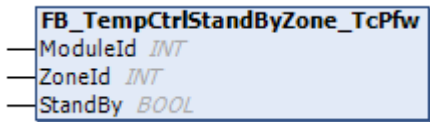
Name	Typ	Beschreibung
ModuleId	INT	ModuleId der zu steuernden Temperaturzonen.
Zoneld	INT	ZonenId der zu steuernden Temperaturzonen.
Enable	BOOL	Neuer Zustand der Temperaturzonen.

Verhalten des Bausteins:

- Um eine Zone in einer Gruppe zu steuern, sind sowohl der ModuleId als auch der Zoneld anzugeben.
- Um alle Zonen in einer Gruppe zu steuern, ist der ModuleId anzugeben. Als Zoneld wird hier der Wert 0 verwendet.
- Um alle Zonen in allen Gruppen zu steuern, ist sowohl für ModuleId als auch für Zoneld der Wert 0 zu verwenden.

In der/den so ausgewählten Zonen wird ST_TcPfw_TempCtrl_If.Enable mit Enable aktualisiert.

3.3.4 FB_TempCtrlStandByZone_TcPfw()



Einer oder mehrere Bausteine dieses Typs werden aus einem Baustein der Applikation aufgerufen, um einzelne Zonen, Zonengruppen oder alle Zonen der Temperaturregelung in den StandBy- oder Aktiv-Zustand zu schalten.

Syntax

```

VAR_INPUT
ModuleId : INT:= -1;
ZoneId : INT:= -1;
StandBy : BOOL:= TRUE;
END_VAR
    
```

Eingänge

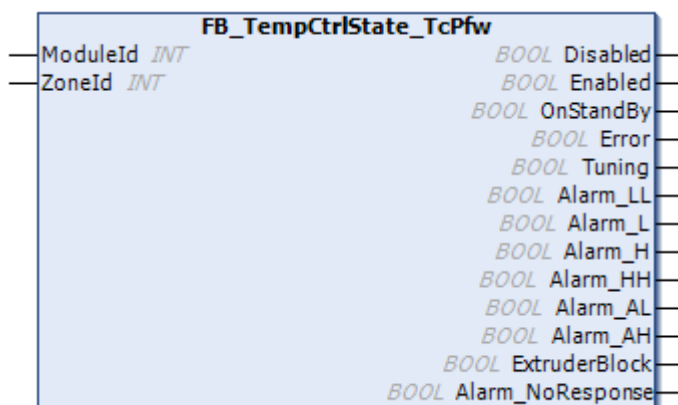
Name	Typ	Beschreibung
ModuleId	INT	ModuleId der zu steuernden Temperaturzonen.
Zoneld	INT	ZonenId der zu steuernden Temperaturzonen.
StandBy	BOOL	Neuer Zustand der Temperaturzonen.

Verhalten des Bausteins:

- Um eine Zone in einer Gruppe zu steuern, sind sowohl der ModuleId als auch der Zoneld anzugeben.
- Um alle Zonen in einer Gruppe zu steuern, ist der ModuleId anzugeben. Als Zoneld wird hier der Wert 0 verwendet.
- Um alle Zonen in allen Gruppen zu steuern, ist sowohl für ModuleId als auch für Zoneld der Wert 0 zu verwenden.

In der/den so ausgewählten Zonen wird ST_TcPfw_TempCtrl_Itf.SelSetpoint mit StandBy aktualisiert.

3.3.5 FB_TempCtrlState_TcPfw()



Dieser Baustein ermittelt eine Reihe von Status-Informationen einer Zone.

Syntax

```

VAR_INPUT
ModuleId: INT:= -1;
ZoneId: INT:= -1;
END_VAR
VAR_OUTPUT
    
```

```

Disabled      : BOOL;
Enabled       : BOOL;
OnStandBy     : BOOL;
Error         : BOOL;
Tuning        : BOOL;
Alarm_LL      : BOOL;
Alarm_L       : BOOL;
Alarm_H       : BOOL;
Alarm_HH      : BOOL;
Alarm_AL      : BOOL;
Alarm_AH      : BOOL;
ExtruderBlock : BOOL;
Alarm_NoResponse : BOOL;
END_VAR
    
```

 **Eingänge**

Name	Typ	Beschreibung
ModuleId	INT	ModuleId der zu steuernden Temperaturzonen.
ZoneId	INT	ZoneId der zu steuernden Temperaturzonen.

 **Ausgänge**

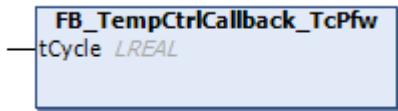
Name	Typ	Beschreibung
Disabled	BOOL	TRUE wenn die Zone nicht freigegeben ist.
Enabled	BOOL	TRUE wenn die Zone freigegeben ist.
OnStandBy	BOOL	TRUE wenn die Zone freigegeben und auf absenken geschaltet ist.
Error	BOOL	TRUE wenn die Zone in einem Fehlerzustand ist.
Tuning	BOOL	TRUE wenn die automatische Tuningfunktion der Zone aktiv ist.
Alarm_LL	BOOL	TRUE wenn die Ist-Temperatur der Zone die äußere negative Toleranzschwelle unterschreitet.
Alarm_L	BOOL	TRUE wenn die Ist-Temperatur der Zone die innere negative Toleranzschwelle unterschreitet.
Alarm_H	BOOL	TRUE wenn die Ist-Temperatur der Zone die innere positive Toleranzschwelle unterschreitet.
Alarm_HH	BOOL	TRUE wenn die Ist-Temperatur der Zone die äußere positive Toleranzschwelle überschreitet.
Alarm_AL	BOOL	TRUE wenn die Ist-Temperatur der Zone die negative absolute Alarmschwelle unterschreitet.
Alarm_AH	BOOL	TRUE wenn die Ist-Temperatur der Zone die positive absolute Alarmschwelle überschreitet.
ExtruderBlock	BOOL	TRUE wenn die Ist-Temperatur der Zone die negative absolute Alarmschwelle unterschreitet und dies zum Abschalten des Extruders führt.
Alarm_NoResponse	BOOL	TRUE wenn die Ist-Temperatur der Zone keine angemessene Reaktion auf die Heizleistung zeigt. Mögliche Ursache ist z. B. ein nicht richtig angebrachter Sensor oder ein nicht auf anderem Wege erkennbarer Defekt der Heizung.

Verhalten des Bausteins:

In jedem Zyklus aktualisiert der Baustein den Status einer oder mehrerer Zonen. Dabei wird das Verhalten im Detail durch den Aufruf festgelegt:

- Wird der Baustein mit nicht bekannter ModuleId>0 und/oder einem nicht bekannten ZoneId>0 aufgerufen oder ist aaaPfwTempToHmi[.].InUse=FALSE, sind alle Ausgänge FALSE.
- Ist ModuleId<>0 und ZoneId=0, werden die Stati aller Zonen des Moduls mit einer ODER-Bedingung verknüpft.
- Ist ModuleId<>0 und ZoneId<>0, wird der Status der ausgewählten Zone gemeldet.

3.3.6 FB_TempCtrlCallback_TcPfw()



Dieser Baustein prüft einmalig den Typ der verknüpften Klemme und parametriert sie entsprechend dem vorgegebenen Sensortyp.

Syntax

```
VAR_INPUT
    tCycle: LREAL;
END_VAR
```

Eingänge

Name	Typ	Beschreibung
tCycle	LREAL	Zykluszeit

Verhalten des Bausteins:

Anhand von aaaPfwTempMparamFromHmi[..].TempSensTerm erkennt der Baustein, ob die eingesetzte E/A-Elektronik Register-Kommunikation oder azyklische CoE-Kommunikation unterstützt. Ist dies der Fall wird der Klemmentyp aus der Baugruppe ausgelesen und überprüft. Stimmt der Hardware-Typ mit der Software-Einstellung überein wird die Baugruppe auf den in aaaPfwTempMparamFromHmi[..].SensorType angegebenen Sensor-Typ eingestellt.

3.3.7 FB_FcMainBody_TcPfw()



Dieser Baustein muss von der Applikation nach dem TempCtrl_FB_TempCtrlMainBody_TcPfw aufgerufen werden. Der Baustein organisiert das zeitweise Einschalten der Kühlung unabhängig von der Reglerausgabe.

Syntax

```
VAR_INPUT
    Activate:    BOOL;
END_VAR
VAR_IN_OUT
    TempToHmi  : ST_TcPfw_TempToHmi_Itf;
    TempCtrl   : ST_TcPfw_TempCtrl_Itf;
    TempOut    : ST_TcPfw_TempCtrlOutput;
    Mparam     : ST_TcPfw_TempMparamFromHmi_Itf;
END_VAR
VAR_OUTPUT
    Error      : BOOL;
    ErrorID    : BOOL; (* not used yet *)
END_VAR
```

Eingänge

Name	Typ	Beschreibung
Activate	BOOL	Nur bei einem TRUE wird die Zwangskühlung wirklich ausgegeben.

 **Ein-/Ausgänge**

Name	Typ	Beschreibung
TempToHmi	ST_TcPfw_TempToHmi_Itf	Hier ist eine Referenz auf die Daten zum HMI der Zone bereitzustellen.
TempCtrl	ST_TcPfw_TempCtrl_Itf	Hier ist eine Referenz auf die Laufzeitdaten der Zone bereitzustellen.
TempOut	ST_TcPfw_TempCtrlOutput	Hier ist eine Referenz auf die Ausgabeschnittstelle der Zone bereitzustellen.
Mparam	ST_TcPfw_TempMparamFromHmi_Itf	Hier ist eine Referenz auf die Maschinenparameter der Zone bereitzustellen.

 **Ausgänge**

Name	Typ	Beschreibung
Error	BOOL	Zeigt an, wenn etwas falsch konfiguriert ist.
ErrorID	BOOL	Nicht verwendet.

Verhalten des Bausteins:

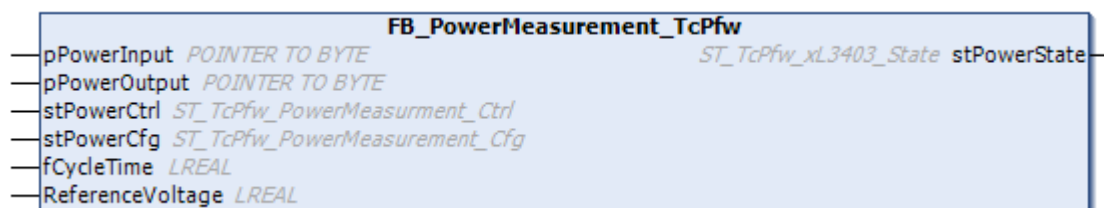
Voraussetzung für eine einwandfreie Funktion ist, dass die Zone "InUse" ist und eine Kühlung besitzt. Des Weiteren muss in den Maschinenparametern dieser Zone fc_Enable aktiviert sein und realistische Zeiten für die Kühlzeit fc_OnTime und die Pausenzeit fpwmOffTime eingestellt worden sein.

Ist die fpwmOffTime abgelaufen wird für die Zeit fc_OnTime abzüglich der schon ausgegebenen Kühlleistung die Kühlung aktiv.

3.4 Utilities

3.4.1 Stand Alone

3.4.1.1 FB_PowerMeasurement_TcPfw()



Dieser Baustein muss zwingend in der Applikation aufgerufen werden. Er organisiert intern die komplette Temperaturregelung.

Syntax

```

VAR_INPUT
    pPowerInput   : POINTER TO BYTE;
    pPowerOutput  : POINTER TO BYTE;
    stPowerCtrl   : ST_TcPfw_PowerMeasurment_Ctrl;
    stPowerCfg    : ST_TcPfw_PowerMeasurement_Cfg;
    fCycleTime    : LREAL:=0.025;
END_VAR
VAR_OUTPUT
    stPowerState  : ST_TcPfw_xL3403_State;
END_VAR
    
```

Eingänge

Name	Typ	Beschreibung
pPowerInput	POINTER TO BYTE	Pointer auf die Inputstruktur der Leistungsmessklemme.
pPowerOutput	POINTER TO BYTE	Pointer auf die Outputstruktur der Leistungsmessklemme.
stPowerCtrl	ST_TcPfw_PowerMeasurement_Ctrl	Ermöglicht das separate Auslesen der aktuellen Spannung.
stPowerCfg	ST_TcPfw_PowerMeasurement_Cfg	Konfiguration der Leistungsmessklemme.
fCycleTime	LREAL	Übergabe der Zykluszeit für diesen Baustein.

Ausgänge

Name	Typ	Beschreibung
stPowerState	ST_TcPfw_xL3403_State	Hier wird die aktuelle Leistung, der aktuelle Strom und Fehlerzustände zurückgemeldet.

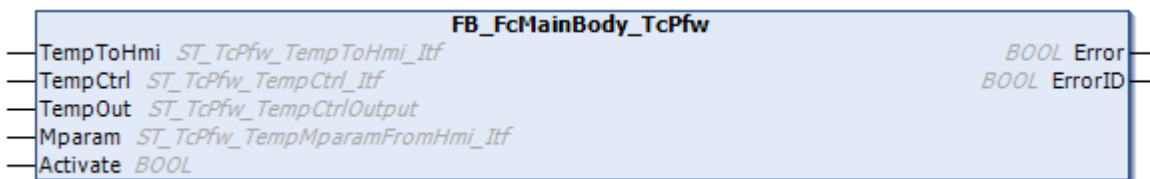
Verhalten des Bausteins:

Dieser Baustein muss von der Applikation zyklisch aufgerufen werden. Der Baustein bekommt von der Applikation das Mapping-Interface als Pointer über pPowerInput und pPowerOutput übergeben. Abhängig von der ausgewählten Leistungsmessklemme in stPowerCfg werden die Pointer-Adressen interpretiert. Der Baustein verteilt abhängig von der Versorgungsgruppe die gemessenen Leistungen auf die einzelnen SupplyLines.

i Die Pointer-Adresse und der hinterlegte Klemmentyp müssen zu jeder Zeit übereinstimmen. Andernfalls gibt es falsche Speicherzugriffe.

i Bei der Leistungsmessung mit der EL3403 ist die erhöhte Filterzeit mit zu berücksichtigen.

3.4.1.2 FB_FcMainBody_TcPfw()



Dieser Baustein muss von der Applikation nach dem TempCtrl_FB_TempCtrlMainBody_TcPfw aufgerufen werden. Der Baustein organisiert das zeitweise Einschalten der Kühlung unabhängig von der Reglerausgabe.

Syntax

```

VAR_INPUT
    Activate:    BOOL;
END_VAR
VAR_IN_OUT
    TempToHmi:  ST_TcPfw_TempToHmi_Itf;
    TempCtrl   :  ST_TcPfw_TempCtrl_Itf;
    TempOut    :  ST_TcPfw_TempCtrlOutput;
    Mparam     :  ST_TcPfw_TempMparamFromHmi_Itf;
END_VAR
VAR_OUTPUT
    
```

```
Error      : BOOL;
ErrorID    : BOOL; (* not used yet *)
END_VAR
```

 **Eingänge**

Name	Typ	Beschreibung
Activate	BOOL	Nur bei einem TRUE wird die Zwangskühlung wirklich ausgegeben.

 **Ein-/Ausgänge**

Name	Typ	Beschreibung
TempToHmi	ST_TcPfw_TempToHmi_Itf	Hier ist eine Referenz auf die Daten zum HMI der Zone bereitzustellen.
TempCtrl	ST_TcPfw_TempCtrl_Itf	Hier ist eine Referenz auf die Laufzeitdaten der Zone bereitzustellen.
TempOut	ST_TcPfw_TempCtrlOutput	Hier ist eine Referenz auf die Ausgabeschnittstelle der Zone bereitzustellen.
Mparam	ST_TcPfw_TempMparamFromHmi_Itf	Hier ist eine Referenz auf die Maschinenparameter der Zone bereitzustellen.

 **Ausgänge**

Name	Typ	Beschreibung
Error	BOOL	Zeigt an, wenn etwas falsch konfiguriert ist.
ErrorID	BOOL	Nicht verwendet.

Verhalten des Bausteins:

Voraussetzung für eine einwandfreie Funktion ist, dass die Zone "InUse" ist und eine Kühlung besitzt. Des Weiteren muss in den Maschinenparametern dieser Zone fc_Enable aktiviert sein und realistische Zeiten für die Kühlzeit fc_OnTime und die Pausenzeit fpwmOffTime eingestellt worden sein. Ist die fpwmOffTime abgelaufen, wird für die Zeit fc_OnTime abzüglich der schon ausgegebenen Kühlleistung die Kühlung aktiv.

3.4.1.3 FB_TempParamSave_TcPfw()



Dieser Baustein schreibt die Parameter einer Zone in eine Datei. Zum Lesen der Datei ist ein FB_TempParamLoad_TcPfw Baustein zu verwenden.



Anstatt dieses Bausteins kann man alternativ ein Speichern über das SaveDelay in den Maschinenparametern, den Produktparametern oder den SupplyLines aufrufen.

Syntax

```
VAR_INPUT
Execute      : BOOL;
PathName     : STRING(80);
PreFix       : STRING(20);
END_VAR
```

```

VAR_IN_OUT
  Mparam      : ST_TcPfw_TempMparamFromHmi_Itf;
  Pparam      : ST_TcPfw_TempPparamFromHmi_Itf;
END_VAR
VAR_OUT
  Done        : BOOL;
  Error       : BOOL;
  ErrorId     : DINT;
  sFileName   : STRING(80);
END_VAR
    
```

 **Eingänge**

Name	Typ	Beschreibung
Execute	BOOL	Eine steigende Flanke an diesem Eingang startet den Vorgang. Bei einem FALSE an Execute werden alle Ausgänge gelöscht. Dabei wird sichergestellt, dass sie für mindestens einen Zyklus anstehen.
PathName	STRING	Hier ist der zu verwendende Pfadname bereitzustellen.
PreFix	STRING	Zu verwendender Prefix vor dem Dateinamen.

 **Ein-/Ausgänge**

Name	Typ	Beschreibung
Mparam	ST_TcPfw_TempMparamFromHmi_Itf	Hier ist eine Referenz auf die Maschinenparameter der Zone bereitzustellen.
Pparam	ST_TcPfw_TempPparamFromHmi_Itf	Hier ist eine Referenz auf die Produktparameter der Zone bereitzustellen.

 **Ausgänge**

Name	Typ	Beschreibung
Done	BOOL	Ein TRUE signalisiert hier die erfolgreiche Abarbeitung des Kommandos.
Error	BOOL	Ein TRUE signalisiert hier das Auftreten eines Problems bei der Abarbeitung des Kommandos.
ErrorId	DINT	Wenn ein Fehler aufgetreten ist, wird hier eine kodierte Information über die Art des Problems bereitgestellt.
sFileName	STRING	Name der gespeicherten Datei.

Verhalten des Bausteins:

Bei einer steigenden Flanke an Execute bildet der Baustein aus PathName, dem textuellen Namen der Zone und STRING-Konstanten einen Dateinamen.

Beispiel:

Aus PathName:='C:\Parameter\' und Mparam.ZoneName:='Zone1' wird 'C:\Parameter\Tctrl_Zone1.par' gebildet.

Die Parameter werden in einem kodierten Binärformat geschrieben, dass nicht mit einem Texteditor bearbeitet werden kann. Durch die Kodierung ist das Format weitgehend unempfindlich gegen Versionsunterschiede. In der Regel sind Dateien auch dann lesbar, wenn sie von älteren oder jüngeren Versionen der Library geschrieben wurden.

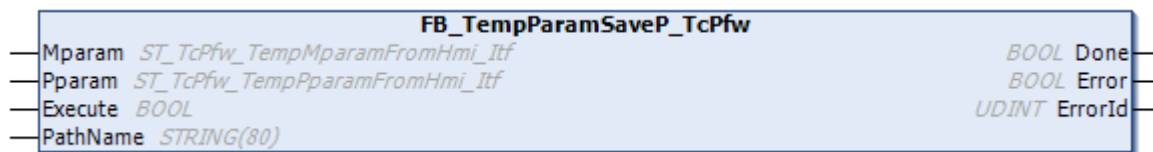
HINWEIS

Sollen die Produktparameter unabhängig von den Maschinendaten der Zone gespeichert werden, sind je ein FB_TempParamSave_TcPfw() Baustein und ein FB_TempParamSaveP_TcPfw() Baustein zu verwenden. Um ein gegenseitiges Überschreiben der namensgleichen Dateien zu vermeiden, sind die Pfadnamen unterschiedlich zu wählen. Beim System-Start sind zunächst die Maschinendaten und anschließend die Produktparameter mit FB_TempParamLoad_TcPfw() Bausteinen zu laden.

HINWEIS

Bei einem Versionswechsel der Library können neue Parameter hinzukommen. Diese werden mit Default-Werten aufgefüllt, deren Wirkung nicht immer das gewünschte Verhalten erzeugt.

3.4.1.4 FB_TempParamSaveP_TcPfw()



Dieser Baustein schreibt die Produktparameter einer Zone in eine Datei. Zum Lesen der Datei ist ein FB_TempParamLoad_TcPfw Baustein zu verwenden.

Syntax

```

VAR_INPUT
    Execute : BOOL;
    PathName: STRING(80);
END_VAR
VAR_IN_OUT
    Mparam : ST_TcPfw_TempMparamFromHmi_Itf;
    Pparam : ST_TcPfw_TempPparamFromHmi_Itf;
END_VAR
VAR_OUT
    Done : BOOL;
    Error : BOOL;
    ErrorId : DINT;
END_VAR
    
```

Eingänge

Name	Typ	Beschreibung
Execute	BOOL	Eine steigende Flanke an diesem Eingang startet den Vorgang. Bei einem FALSE an Execute werden alle Ausgänge gelöscht. Dabei wird sichergestellt, dass sie für mindestens einen Zyklus anstehen.
PathName	STRING	Hier ist der zu verwendende Pfadname bereitzustellen.

Ein-/Ausgänge

Name	Typ	Beschreibung
Mparam	ST_TcPfw_TempMparamFromHmi_Itf	Hier ist eine Referenz auf die Maschinenparameter der Zone bereitzustellen.
Pparam	ST_TcPfw_TempPparamFromHmi_Itf	Hier ist eine Referenz auf die Produktparameter der Zone bereitzustellen.

Ausgänge

Name	Typ	Beschreibung
Done	BOOL	Ein TRUE signalisiert hier die erfolgreiche Abarbeitung des Kommandos.
Error	BOOL	Ein TRUE signalisiert hier das Auftreten eines Problems bei der Abarbeitung des Kommandos.
ErrorId	BOOL	Wenn ein Fehler aufgetreten ist, wird hier eine kodierte Information über die Art des Problems bereitgestellt.

Verhalten des Bausteins:

Bei einer steigenden Flanke an Execute bildet der Baustein aus PathName, dem textuellen Namen der Zone und STRING-Konstanten einen Dateinamen.

Beispiel:

Aus PathName:='C:\Parameter\' und Mparam.ZoneName:='Zone1' wird 'C:\Parameter\Tctrl_Zone1.par' gebildet.

Die Parameter werden in einem kodierten Binärformat geschrieben, dass nicht mit einem Texteditor bearbeitet werden kann. Durch die Kodierung ist das Format weitgehend unempfindlich gegen Versionsunterschiede. In der Regel sind Dateien auch dann lesbar, wenn sie von älteren oder jüngeren Versionen der Library geschrieben wurden.

HINWEIS

Sollen die Produktparameter unabhängig von den Maschinendaten der Zone gespeichert werden, sind je ein FB_TempParamSave_TcPfw() Baustein und ein FB_TempParamSaveP_TcPfw() Baustein zu verwenden. Um ein gegenseitiges Überschreiben der namensgleichen Dateien zu vermeiden, sind die Pfadnamen unterschiedlich zu wählen. Beim System-Start sind zunächst die Maschinendaten und anschließend die Produktparameter mit FB_TempParamLoad_TcPfw() Bausteinen zu laden.

HINWEIS

Bei einem Versionswechsel der Library können neue Parameter hinzukommen. Diese werden mit Default-Werten aufgefüllt, deren Wirkung nicht immer das gewünschte Verhalten erzeugt.

3.4.1.5 FB_TempParamLoad_TcPfw()



Dieser Baustein liest die Parameter einer Zone aus einer Datei. Zum Schreiben der Datei ist ein FB_TempParamSave_TcPfw Baustein zu verwenden.

Syntax

```

VAR_INPUT
    Execute      : BOOL;
    PathName     : STRING(80);
    ProductParam : BOOL:=FALSE;
END_VAR
VAR_IN_OUT
    Mparam      : ST_TcPfw_TempMparamFromHmi_Itf;
    Pparam      : ST_TcPfw_TempPparamFromHmi_Itf;
END_VAR
VAR_OUT
    Done        : BOOL;
    Error       : BOOL;
    ErrorId     : DINT;
END_VAR
    
```

 **Eingänge**

Name	Typ	Beschreibung
Execute	BOOL	Eine steigende Flanke an diesem Eingang startet den Vorgang. Bei einem FALSE an Execute werden alle Ausgänge gelöscht. Dabei wird sichergestellt, dass sie für mindestens einen Zyklus anstehen.
PathName	STRING	Hier ist der zu verwendende Pfadname bereitzustellen.
ProductParam	BOOL	Bei einem TRUE werden die Produktdaten geladen.

 **Ein-/Ausgänge**

Name	Typ	Beschreibung
Mparam	ST_TcPfw_TempMparamFromHmi_Itf	Hier ist eine Referenz auf die Maschinenparameter der Zone bereitzustellen.
Pparam	ST_TcPfw_TempPparamFromHmi_Itf	Hier ist eine Referenz auf die Produktparameter der Zone bereitzustellen.

 **Ausgänge**

Name	Typ	Beschreibung
Done	BOOL	Ein TRUE signalisiert hier die erfolgreiche Abarbeitung des Kommandos.
Error	BOOL	Ein TRUE signalisiert hier das Auftreten eines Problems bei der Abarbeitung des Kommandos.
ErrorId	DINT	Wenn ein Fehler aufgetreten ist, wird hier eine kodierte Information über die Art des Problems bereitgestellt.

Verhalten des Bausteins:

Bei einer steigenden Flanke an Execute bildet der Baustein aus PathName, dem textuellen Namen der Zone und STRING-Konstanten einen Dateinamen.

Beispiel:

Aus PathName:='C:\Parameter\' und Mparam.ZoneName:='Zone1' wird 'C:\Parameter\Tctrl_Zone1.par' gebildet.

Die Parameter werden in einem kodierten Binärformat gelesen, das nicht mit einem Texteditor bearbeitet werden kann. Durch die Kodierung ist das Format weitgehend unempfindlich gegen Versionsunterschiede. In der Regel sind Dateien auch dann lesbar, wenn sie von älteren oder jüngeren Versionen der Library geschrieben wurden.

HINWEIS

Sollen die Produktparameter unabhängig von den Maschinendaten der Zone gespeichert werden, sind je ein FB_TempParamSave_TcPfw() Baustein und ein FB_TempParamSaveP_TcPfw() Baustein zu verwenden. Um ein gegenseitiges Überschreiben der namensgleichen Dateien zu vermeiden, sind die Pfadnamen unterschiedlich zu wählen. Beim System-Start sind zunächst die Maschinendaten und anschließend die Produktparameter mit FB_TempParamLoad_TcPfw() Bausteinen zu laden.

HINWEIS

Bei einem Versionswechsel der Library können neue Parameter hinzukommen. Diese werden mit Default-Werten aufgefüllt, deren Wirkung nicht immer das gewünschte Verhalten erzeugt.

3.4.2 FB_xL3403_TcPfw()



Dieser Baustein bereitet die durch eine xL3403 ermittelten Daten (Spannung, Leitung) auf und stellt diese dann der Applikation zur Verfügung.



Von der Applikation sollte der Baustein FB_PowerMeasurement_TcPfw() aufgerufen werden. Dieser Baustein ruft intern FB_xL3403_TcPfw() auf.

Syntax

```

VAR_INPUT
    I_ratio      : LREAL:=1.0;
    CycleTime   : LREAL:=0.025;
    ReadVoltage  : BOOL:=FALSE;
    Using_EL    : BOOL:=FALSE;
END_VAR
VAR_IN_OUT
    Input       : ST_TcPfw_xL3403_Input;
    Output      : ST_TcPfw_xL3403_Output;
    State       : ST_TcPfw_xL3403_State;
END_VAR
    
```

Eingänge

Name	Typ	Beschreibung
I_ratio	LREAL	Hier muss das Übersetzungsverhältnis der Stromwandler eingetragen werden.
CycleTime	LREAL	Zykluszeit mit der dieser Baustein aufgerufen wird.
ReadVoltage	BOOL	Ein TRUE bewirkt das Auslesen der aktuell gemessenen Spannung statt der Leistung.
Using_EL	BOOL	Ein TRUE teilt dem Baustein mit, dass es sich um eine EL-Klemme handelt.

Ein-/Ausgänge

Name	Typ	Beschreibung
Input	ST_TcPfw_xL3403_Input	Stellt die Eingangsdaten der Klemme bereit.
Output	ST_TcPfw_xL3403_Output	Stellt die Ausgangsdaten der Klemme bereit.
State	ST_TcPfw_xL3403_State	Gibt den Status der Klemme sowie die aufbereiteten Daten an die Applikation zurück.

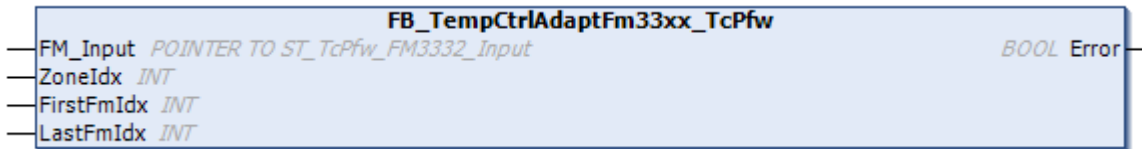
Verhalten des Bausteins:

Meldet die Klemme keinen Fehler, so werden in jedem Zyklus die aufbereiteten Eingangsdaten bereitgestellt.



Bei der Leistungsmessung mit der EL3403 ist die erhöhte Filterzeit mit zu berücksichtigen.

3.4.3 FB_TempCtrlAdaptFm33xx_TcPfw()



Die E/A-Daten von FM3312 oder FM3332 Feldbusmodulen werden an die E/A-Strukturen der Library angepasst.

Dieser Baustein muss zwingend in der Applikation aufgerufen werden. Er organisiert intern die komplette Temperaturregelung.

Syntax

```

VAR_INPUT
FM_Input : POINTER TO ST_TcPfw_FM3332_Input;
ZoneIdx : INT;
FirstFmIdx: INT;
LastFmIdx : INT;
END_VAR
VAR_OUTPUT
Error : BOOL;
END_VAR
    
```

Eingänge

Name	Typ	Beschreibung
FM_Input	POINTER TO ST_TcPfw_FM3332_Input	Die Adresse einer Einzelvariable oder eines Arrays vom Typ ST_TcPfw_FM3332_Input.
Zoneldx	INT	Der Index der Zone, der das Prozessabbild zugeordnet werden soll.
FirstFmIdx	INT	Der erste Index des als FM_Input bereitgestellten Prozessabbilds.
LastFmIdx	INT	Der letzte Index des als FM_Input bereitgestellten Prozessabbilds.

Ausgänge

Name	Typ	Beschreibung
Error	BOOL	Hier werden eventuelle Probleme mit Aufrufparametern oder Zonenparametern signalisiert.

Verhalten des Bausteins:

Wenn einer der Aufrufparameter außerhalb des zulässigen Bereichs liegt, wird dies mit Error gemeldet. Weiterhin wird Error gemeldet, wenn zwar die Aufrufparameter korrekt sind, die adressierte Zone in ihren Parametern aber eine nicht zulässige Einstellung in aaaPfwTempMparamFromHmi[Zoneldx].TermChannel aufweist.

Hier sind zwei Modi nutzbar:

- Wenn in FM_Input die Adresse des einzigen ST_TcPfw_FM3332_Input Prozessabbilds der Applikation oder innerhalb eines Arrays die Adresse des für diese Zone zuständigen Prozessabbilds bereitgestellt wird, ist als TermChannel der Index (1..32) des Eingangs im Prozessabbild anzugeben.
- Wenn in FM_Input die Adresse des ersten ST_TcPfw_FM3332_Input Prozessabbilds eines Arrays bereitgestellt wird, ist als TermChannel der Index des Eingangs im Prozessabbild-Array anzugeben. Dieser Index ist 1..32 für die Eingänge des ersten Moduls, 33..64 für die Eingänge des zweiten Moduls usw.

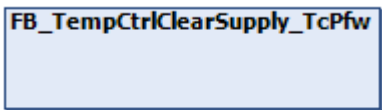
HINWEIS

Bei Modulen mit weniger als 32 Eingängen wird das gleiche Abbild benutzt. Die in der Hardware des Moduls nicht implementierten Kanäle bleiben dann unbenutzt, werden aber bei der oben beschriebenen Festlegung des Eingangs-Index mitgezählt.

Andernfalls werden die Daten des ST_TcPfw_FM3332_Input Prozessabbilds in das ST_TcPfw_TempCtrlInput Prozessabbild der Zone konvertiert:

- Der Prozesswert für die Isttemperatur ist kompatibel und wird umkopiert.
- Aus dem DpState wird ein EL_SnsWcState für die Verbindungsüberwachung abgeleitet.
- Das zum Messkanal gehörenden Bit in OpenCircuit[...] wird als SNS_Overrange Bit in KL_SnsState eingeblendet.
- Das zum Messkanal gehörenden Bit in Backvoltage[...] wird als SNS_GeneralError Bit in KL_SnsState eingeblendet.

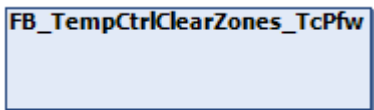
3.4.4 FB_TempCtrlClearSupply_TcPfw()



Ein Baustein dieses Typs ist in der Initialisierungsphase der Applikation vor dem Bereitstellen der Parameter einmalig aufzurufen.

Der Baustein initialisiert die Daten der Versorgungsgruppen der Temperaturregelung.

3.4.5 FB_TempCtrlClearZones_TcPfw()

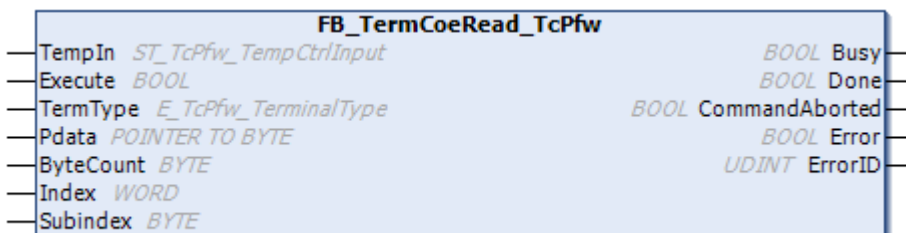


Ein Baustein dieses Typs ist in der Initialisierungsphase der Applikation vor dem Bereitstellen der Parameter einmalig aufzurufen.

Der Baustein initialisiert die Daten aller Zonen der Temperaturregelung:

- aaaPfwTempPparamFromHmi
- aaaPfwTempMparamFromHmi
- aaaPfwTempToHmi
- aaaTempCtrl
- out_PfwTempCtrlOutput

3.4.6 FB_TermCoeRead_TcPfw()



Ein Baustein dieses Typs wird von FB_TempCtrlCallback_TcPfw für den Lesezugriff auf EL-Klemmen genutzt.

Syntax

```

VAR_INPUT
  Execute      : BOOL;
  TermType     : E_TcPfw_TerminalType:=eTcPfwTermT_NoTerminal;
  Pdata        : POINTER TO BYTE:=0;
  ByteCount    : BYTE:=0;
  Index        : WORD:=0;
  Subindex     : BYTE:=0;
END_VAR
VAR_IN_OUT
  TempIn       : ST_TcPfw_TempCtrlInput;
END_VAR
VAR_OUTPUT
  Busy         : BOOL:=FALSE;
  Done         : BOOL:=FALSE;
  CommandAborted: BOOL:=FALSE;
  Error        : BOOL:=FALSE;
  ErrorID      : UDINT:=0;
END_VAR
    
```

 **Eingänge**

Name	Typ	Beschreibung
Execute	BOOL	Eine steigende Flanke startet den Vorgang. Eine fallende Flanke löscht alle Ausgänge.
TermType	E_TcPfw_TerminalType	Der kodierte Typ der angesprochenen Klemme.
Pdata	POINTER TO BYTE	Die Zieladresse für die gelesenen Daten.
ByteCount	BYTE	Die Größe der zu lesenden Daten in Bytes.
Index	WORD	Die CoE Adresse entsprechend dem Objektverzeichnis der Klemme.
Subindex	BYTE	Die CoE Adresse entsprechend dem Objektverzeichnis der Klemme.

 **Ein-/Ausgänge**

Name	Typ	Beschreibung
TempIn	ST_TcPfw_TempCtrlInput	Das Eingangs-Prozessabbild der Klemme.

 **Ausgänge**

Name	Typ	Beschreibung
Busy	BOOL	Die andauernde Aktivität des Bausteins wird hier signalisiert.
Done	BOOL	Der erfolgreiche Abschluss des Vorgangs wird hier gemeldet.
CommandAborted	BOOL	Ein TRUE hier signalisiert den Abbruch des Vorgangs.
Error	BOOL	Ein TRUE hier signalisiert das Auftreten eines Problems.
ErrorID	UDINT	Im Fehlerfall wird hier eine kodierte Information bereitgestellt.

Verhalten des Bausteins:

Eine steigende Flanke an Execute veranlasst den Baustein zu einer Reihe von Überprüfungen.

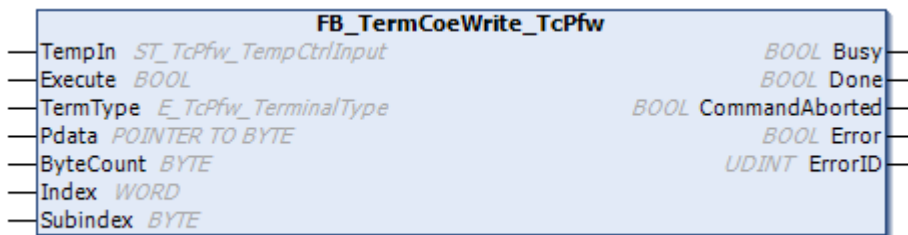
- Der Subindex muss im Bereich 0 bis 127 (inklusive) liegen.
- ByteCount muss größer als 0 sein.
- Pdata muss ungleich 0 sein.
- TermType muss eine EL Klemme kennzeichnen. Nur diese unterstützen den hier verwendeten Kommunikationsmechanismus CoE.

Ist eine der genannten Bedingungen nicht erfüllt, wird ein Fehler gemeldet. Andernfalls wird der Zugriff an die Klemme übermittelt. Das Ergebnis der Übermittlung wird an den Ausgängen bereitgestellt.

i Neben den oben genannten Fehlermöglichkeiten kann es bei der Übertragung zu Problemen kommen. Weiterhin kann die Klemme ein Problem (Adressierung, Werte, Zugriffsart) melden.

i Ein CoE Zugriff setzt voraus, dass ST_TcPfw_TempCtrlInput.EL_AdsAddr verknüpft ist.

3.4.7 FB_TermCoeWrite_TcPfw()



Ein Baustein dieses Typs wird von FB_TempCtrlCallback_TcPfw für den Schreibzugriff auf EL-Klemmen genutzt.

Syntax

```

VAR_INPUT
Execute      : BOOL;
TermType     : TcPfw_TerminalType:=eTcPfwTermT_NoTerminal;
Pdata        : POINTER TO BYTE:=0;
ByteCount    : BYTE:=0;
Index        : WORD:=0;
Subindex     : BYTE:=0;
END_VAR
VAR_IN_OUT
TempIn       : ST_TcPfw_TempCtrlInput;
END_VAR
VAR_OUTPUT
Busy         : BOOL:=FALSE;
Done         : BOOL:=FALSE;
CommandAborted: BOOL:=FALSE;
Error        : BOOL:=FALSE;
ErrorID      : UDINT:=0;
END_VAR
    
```

Eingänge

Name	Typ	Beschreibung
Execute	BOOL	Eine steigende Flanke startet den Vorgang. Eine fallende Flanke löscht alle Ausgänge.
TermType	_TcPfw_TerminalType	Der kodierte Typ der angesprochenen Klemme.
Pdata	POINTER TO BYTE	Die Zieladresse für die gelesenen Daten.
ByteCount	BYTE	Die Größe der zu lesenden Daten in Bytes.
Index	WORD	Die CoE Adresse entsprechend dem Objektverzeichnis der Klemme.
Subindex	BYTE	Die CoE Adresse entsprechend dem Objektverzeichnis der Klemme.

Ein-/Ausgänge

Name	Typ	Beschreibung
TempIn	ST_TcPfw_TempCtrlInput	Das Eingangs-Prozessabbild der Klemme.

Ausgänge

Name	Typ	Beschreibung
Busy	BOOL	Die andauernde Aktivität des Bausteins wird hier signalisiert.
Done	BOOL	Der erfolgreiche Abschluss des Vorgangs wird hier gemeldet.
CommandAborted	BOOL	Ein TRUE hier signalisiert den Abbruch des Vorgangs.
Error	BOOL	Ein TRUE hier signalisiert das Auftreten eines Problems.
ErrorID	UDINT	Im Fehlerfall wird hier eine kodierte Information bereitgestellt.

Verhalten des Bausteins:

Eine steigende Flanke an Execute veranlasst den Baustein zu einer Reihe von Überprüfungen:

- Der Subindex muss im Bereich 0 bis 127 (inklusive) liegen.
- ByteCount muss größer als 0 sein.
- Pdata muss ungleich 0 sein.
- TermType muss eine EL Klemme kennzeichnen. Nur diese unterstützen den hier verwendeten Kommunikationsmechanismus CoE.

Ist eine der genannten Bedingungen nicht erfüllt, wird ein Fehler gemeldet. Andernfalls wird der Zugriff an die Klemme übermittelt. Das Ergebnis der Übermittlung wird an den Ausgängen bereitgestellt.

i Neben den oben genannten Fehlermöglichkeiten kann es bei der Übertragung zu Problemen kommen. Weiterhin kann die Klemme ein Problem (Adressierung, Werte, Zugriffsart) melden.

i Ein CoE Zugriff setzt voraus, dass ST_TcPfw_TempCtrlInput.EL_AdsAddr verknüpft ist.

3.4.8 FB_TermRegRead_TcPfw()



Ein Baustein dieses Typs wird von FB_TempCtrlCallback_TcPfw für den Lesezugriff auf EL-Klemmen genutzt.

Syntax

```

VAR_INPUT
    Execute      : BOOL;
    TermType     : E_TcPfw_TerminalType:=eTcPfwTermT_NoTerminal;
    Select       : INT:=-1;
    CycleTime    : LREAL:=0.025;
END_VAR
VAR_IN_OUT
    Ctrl         : USINT;
    State        : USINT;
    InData       : INT;
    OutData      : INT;
END_VAR
VAR_OUTPUT
    RegData      : WORD:=0;
    Busy         : BOOL:=FALSE;
    
```



```

Done          : BOOL:=FALSE;
CommandAborted: BOOL:=FALSE;
Error        : BOOL:=FALSE;
ErrorID      : UDINT:=0;
END_VAR
    
```

 **Eingänge**

Name	Typ	Beschreibung
Execute	BOOL	Eine steigende Flanke startet den Vorgang. Eine fallende Flanke löscht alle Ausgänge.
TermType	E_TcPfw_TerminalType	Der kodierte Typ der angesprochenen Klemme.
Select	INT	Die Register-Adresse der Klemme.
CycleTime	LREAL	Die Zykluszeit der aufrufenden Task.

 **Ein-/Ausgänge**

Name	Typ	Beschreibung
Ctrl	USINT	Eine Referenz auf ST_TcPfw_TempCtrlOutput.KL_SnsCtrl der Klemme.
State	USINT	Eine Referenz auf ST_TcPfw_TempCtrlInput.KL_SnsState der Klemme.
InData	INT	Eine Referenz auf ST_TcPfw_TempCtrlInput.KL_SnsData der Klemme.
OutData	INT	Eine Referenz auf ST_TcPfw_TempCtrlOutput.KL_SnsData der Klemme.

 **Ausgänge**

Name	Typ	Beschreibung
RegData	WORD	Bei erfolgreicher Ausführung wird hier der gelesene Registerinhalt bereitgestellt.
Busy	BOOL	Die andauernde Aktivität des Bausteins wird hier signalisiert.
Done	BOOL	Der erfolgreiche Abschluss des Vorgangs wird hier gemeldet.
CommandAborted	BOOL	Ein TRUE hier signalisiert den Abbruch des Vorgangs.
Error	BOOL	Ein TRUE hier signalisiert das Auftreten eines Problems.
ErrorID	UDINT	Im Fehlerfall wird hier eine kodierte Information bereitgestellt.

Verhalten des Bausteins:

Eine steigende Flanke an Execute veranlasst den Baustein zu einer Reihe von Überprüfungen:

- Select muss im Bereich 0 bis 63 (inklusive) liegen.
- Es darf keine andere Register-Kommunikation mit dieser Klemme aktiv sein.
- TermType muss eine KL-Klemme kennzeichnen. Nur diese unterstützen den hier verwendeten Kommunikationsmechanismus.

Ist eine der genannten Bedingungen nicht erfüllt, wird ein Fehler gemeldet. Andernfalls wird der Zugriff an die Klemme übermittelt. Das Ergebnis der Übermittlung wird an den Ausgängen bereitgestellt.

i Neben den oben genannten Fehlermöglichkeiten kann es bei der Übertragung zu Problemen kommen. Weiterhin kann die Klemme ein Problem (Adressierung, Werte, Zugriffsart) melden.

i Ein Register-Zugriff setzt voraus, dass alle Elemente mit Namensanfang "KL_" in ST_TcPfw_TempCtrlInput verknüpft ist.

3.4.9 FB_TermRegWrite_TcPfw()



Ein Baustein dieses Typs wird von FB_TempCtrlCallback_TcPfw für den Schreibzugriff auf KL-Klemmen genutzt.

Syntax

```

VAR_INPUT
    Execute      : BOOL;
    TermType     : E_TcPfw_TerminalType:=eTcPfwTermT_NoTerminal;
    Select       : INT:=-1;
    RegData      : WORD:=0;
    CycleTime    : LREAL:=0.01;
END_VAR
VAR_IN_OUT
    Ctrl         : USINT;
    State        : USINT;
    InData       : INT;
    OutData      : INT;
END_VAR
VAR_OUTPUT
    Busy         : BOOL:=FALSE;
    Done         : BOOL:=FALSE;
    CommandAborted : BOOL:=FALSE;
    Error: BOOL  : :=FALSE;
    ErrorID      : UDINT:=0;
END_VAR
    
```

Eingänge

Name	Typ	Beschreibung
Execute	BOOL	Eine steigende Flanke startet den Vorgang. Eine fallende Flanke löscht alle Ausgänge.
TermType	E_TcPfw_TerminalType	Der kodierte Typ der angesprochenen Klemme.
Select	INT	Die Register-Adresse der Klemme.
RegData	WORD	Hier ist der zu schreibende Registerinhalt bereitzustellen.
CycleTime	LREAL	Die Zykluszeit der aufrufenden Task.

Ein-/Ausgänge

Name	Typ	Beschreibung
Ctrl	USINT	Eine Referenz auf ST_TcPfw_TempCtrlOutput.KL_SnsCtrl der Klemme.
State	USINT	Eine Referenz auf ST_TcPfw_TempCtrlInput.KL_SnsState der Klemme.
InData	INT	Eine Referenz auf ST_TcPfw_TempCtrlInput.KL_SnsData der Klemme.
OutData	INT	Eine Referenz auf ST_TcPfw_TempCtrlOutput.KL_SnsData der Klemme.

 **Ausgänge**

Name	Typ	Beschreibung
Busy	BOOL	Die andauernde Aktivität des Bausteins wird hier signalisiert.
Done	BOOL	Der erfolgreiche Abschluss des Vorgangs wird hier gemeldet.
CommandAborted	BOOL	Ein TRUE hier signalisiert den Abbruch des Vorgangs.
Error	BOOL	Ein TRUE hier signalisiert das Auftreten eines Problems.
ErrorID	UDINT	Im Fehlerfall wird hier eine kodierte Information bereitgestellt.

Verhalten des Bausteins:

Eine steigende Flanke an Execute veranlasst den Baustein zu einer Reihe von Überprüfungen:

- Select muss im Bereich 0 bis 63 (inklusive) liegen.
- Es darf keine andere Register-Kommunikation mit dieser Klemme aktiv sein.
- TermType muss eine KL-Klemme kennzeichnen. Nur diese unterstützen den hier verwendeten Kommunikationsmechanismus.

Ist eine der genannten Bedingungen nicht erfüllt, wird ein Fehler gemeldet. Andernfalls wird der Zugriff an die Klemme übermittelt. Das Ergebnis der Übermittlung wird an den Ausgängen bereitgestellt.



Neben den oben genannten Fehlermöglichkeiten kann es bei der Übertragung zu Problemen kommen. Weiterhin kann die Klemme ein Problem (Adressierung, Werte, Zugriffsart) melden.



Ein Registerzugriff setzt voraus, dass alle Elemente mit Namensanfang "KL_" in ST_TcPfw_TempCtrlInput verknüpft ist.

3.5 Aufzählungen

3.5.1 E_TcPfw_TctrlPowerTerminal

Diese Aufzählung definiert Kennungen für die unterstützten Typen von E/A-Klemmen. Diese Kennungen werden in ST_TcPfw_PowerMeasurement_Cfg verwendet.

Syntax

```

TYPE E_TcPfw_TctrlPowerTerminal:
(* last modification: 16.03.2009 *)
(
NoTerminal,
KL3403,
EL3403,
EL3773,
EL3x64,
EL3413,
EL3443,
EL3446,
EL3453,
Simulation:=1000,
Customized:=10000
);
END_TYPE
    
```

Werte

Name	Beschreibung
NoTerminal	Keine Leistungsmessklemme angeschlossen.
KL3403	Busklemme, 3-Kanal-Analog-Eingang, Leistungsmessung, 500 V AC, 1 A, 16 Bit
EL3403	EtherCAT-Klemme, 3-Kanal-Analog-Eingang, Leistungsmessung, 500 V AC, 1 A, 16 Bit
EL3773	EtherCAT-Klemme, 3-Kanal-Analog-Eingang, Multifunktion, 500 V AC/DC, 1 A, 16 Bit, 10 kSps, Oversampling
EL3x64	EL3064 - EtherCAT-Klemme, 4-Kanal-Analog-Eingang, Spannung, 0...10 V, 12 Bit, single-ended EL3164 - EtherCAT-Klemme, 4-Kanal-Analog-Eingang, Spannung, 0...10 V, 16 Bit, single-ended
EL3413	EtherCAT-Klemme, 3-Kanal-Analog-Eingang, Leistungsmessung, 690 V AC, 1/5 A, 16 Bit, galvanisch getrennt
EL3443	EtherCAT-Klemme, 3-Kanal-Analog-Eingang, Leistungsmessung, 480 V AC/DC, 1 A, 24 Bit
EL3446	EtherCAT-Klemme, 6-Kanal-Analog-Eingang, Strom, 1 A, 24 Bit, verteilte Leistungsmessung
EL3453	EtherCAT-Klemme, 3-Kanal-Analog-Eingang, Leistungsmessung, 690 V AC, 0,1/1/5 A, 24 Bit, galvanisch getrennt
Simulation	Leistungsmessung über Simulation.
Customized	Untergrenze des kundenspezifischen Bereiches.

3.5.2 E_TcPfw_TctrlOutSelect

Diese Aufzählung definiert Kennungen für die Auswahl des Ausgabesignals einer Zone der Temperaturregelung. Diese Kennungen werden in ST_TcPfw_TempMparamFromHmi_Itf verwendet.

Syntax

```

TYPE E_TcPfw_TctrlOutSelect:
(* last modification: xx.xx.200x *)
(
eTcPfwTcOut_NoSignal,
eTcPfwTcOut_PWM,
eTcPfwTcOut_Sign,
eTcPfwTcOut_2step
);
END_TYPE
    
```

Werte

Name	Beschreibung
eTcPfwTcOut_NoSignal	Es wird keines der Ausgangssignale ausgewählt.
eTcPfwTcOut_PWM	Es wird das PWM-Signal ausgewählt, das von der Reaktion des PID-Reglers abgeleitet wird.
eTcPfwTcOut_Sign	Es wird ein Schaltsignal ausgewählt, das vom Vorzeichen der Reglerausgabe abgeleitet wird.
eTcPfwTcOut_2step	Es wird ein Schaltsignal ausgewählt, dass von einem hysteresebefaheten Zweipunktregler (Schmitt-Trigger) aus der Regelabweichung gebildet wird.

3.5.3 E_TcPfw_TempSensType

Diese Aufzählung definiert Kennungen für die unterstützten Typen von Temperatursensoren. Diese Kennungen werden in ST_TcPfw_TempMparamFromHmi_Itf verwendet.

Syntax

```

TYPE E_TcPfw_TempSensType:
(* last modification: 17.02.2011 *)
(
    
```

```
eTcPfwTempSensT_NoSensor,
eTcPfwTempSensT_TC_B,
eTcPfwTempSensT_TC_E,
eTcPfwTempSensT_TC_J,
eTcPfwTempSensT_TC_K,
eTcPfwTempSensT_TC_L,
eTcPfwTempSensT_TC_N,
eTcPfwTempSensT_TC_R,
eTcPfwTempSensT_TC_S,
eTcPfwTempSensT_TC_T,
eTcPfwTempSensT_TC_U,

eTcPfwTempSensT_PT_100:=100,
eTcPfwTempSensT_NI_100,
eTcPfwTempSensT_NI_120,
eTcPfwTempSensT_PT_200,
eTcPfwTempSensT_PT_500,
eTcPfwTempSensT_PT_1000,
eTcPfwTempSensT_NI_1000,

eTcPfwTempSensT_Customized:=10000
);
END_TYPE
```

Werte

Name	Beschreibung
eTcPfwTempSensT_NoSensor	Diese Kennung erscheint in der Konfiguration von nicht benutzten Temperaturreglerzonen.
eTcPfwTempSensT_TC_B	Typ B Sensor: 600°C bis 1800°C
eTcPfwTempSensT_TC_E	Typ B Sensor: -100°C bis 1000°C
eTcPfwTempSensT_TC_J	Typ B Sensor: -100°C bis 1200°C
eTcPfwTempSensT_TC_K	Typ B Sensor: -100°C bis 1370°C
eTcPfwTempSensT_TC_L	Typ B Sensor: -25°C bis 900°C
eTcPfwTempSensT_TC_N	Typ B Sensor: -100°C bis 1300°C
eTcPfwTempSensT_TC_R	Typ B Sensor: 0°C bis 1700°C
eTcPfwTempSensT_TC_S	Typ B Sensor: 0°C bis 1700°C
eTcPfwTempSensT_TC_T	Typ B Sensor: -100°C bis 400°C
eTcPfwTempSensT_TC_U	Typ B Sensor: -25°C bis 600°C
eTcPfwTempSensT_PT_100	PT100 Sensor: -200°C bis 850°C
eTcPfwTempSensT_NI_100	NI100 Sensor: -60°C bis 250°C.
eTcPfwTempSensT_NI_120	NI120 Sensor: -60°C bis 320°C.
eTcPfwTempSensT_PT_200	PT200 Sensor: -200°C bis 850°C
eTcPfwTempSensT_PT_500	PT500 Sensor: -200°C bis 850°C
eTcPfwTempSensT_PT_1000	PT1000 Sensor: -200°C bis 850°C
eTcPfwTempSensT_NI_1000	NI1000 Sensor: -60°C bis 250°C
eTcPfwTempSensT_Customized	Diese und alle höheren Zahlenwerte markieren applikationsspezifische Temperatursensoren.

3.5.4 E_TcPfw_TerminalType

Diese Aufzählung definiert Kennungen für die unterstützten Typen von E/A-Klemmen. Diese Kennungen werden in ST_TcPfw_TempMparamFromHmi_Itf verwendet.

Syntax

```
TYPE E_TcPfw_TerminalType:
(* last modification: 16.03.2009 *)
(
eTcPfwTermT_NoSensor,

eTcPfwTermT_KL_RangeLow:=1000,
eTcPfwTermT_KL300x, (* +/-10V *)
eTcPfwTermT_KL301x, (* 0..20mA *)
```

```
eTcPfwTermT_KL302x, (* 4..20mA *)
eTcPfwTermT_KL304x, (* 0..20mA *)
eTcPfwTermT_KL305x, (* 4..20mA *)
eTcPfwTermT_KL306x, (* 0..10V *)
eTcPfwTermT_KL310x, (* +/-10V *)
eTcPfwTermT_KL311x, (* 0..20mA *)
eTcPfwTermT_KL312x, (* 4..20mA *)
eTcPfwTermT_KL313x, (* +/-10V *)
eTcPfwTermT_KL314x, (* 0..20mA *)
eTcPfwTermT_KL315x, (* 4..20mA *)
eTcPfwTermT_KL317x, (* 0..2V *)
eTcPfwTermT_KL318x, (* +/-2V *)
eTcPfwTermT_KL340x, (* +/-10V *)
eTcPfwTermT_KL344x, (* 0..20mA *)
eTcPfwTermT_KL346x, (* 0..10V *)
eTcPfwTermT_KL331x, (* thermo couple *)
eTcPfwTermT_KL_RangeHigh,

eTcPfwTermT_EL_RangeLow:=2000,
eTcPfwTermT_EL331x, (* thermo couple *)
eTcPfwTermT_EL320x, (* PT100/PT1000 *)
eTcPfwTermT_EL316x, (* 0..10V *)
eTcPfwTermT_EL_RangeHigh,

eTcPfwTermT_XX_RangeLow:=3000,
eTcPfwTermT_FM33xx, (* thermo couple *)
eTcPfwTermT_EM8908, (* multi signal backplane *)
eTcPfwTermT_XX_RangeHigh,

eTcPfwTermT_Customized:=10000
);
END_TYPE
```

Werte

Name	Beschreibung
eTcPfwTermT_NoSensor	Keine E/A-Elektronik vorhanden, die Zone wird in Simulation betrieben.
eTcPfwTermT_KL_RangeLow	Untergrenze der Kennungen für K-Bus Klemmen.
eTcPfwTermT_KL300x	K-Bus Klemmen für den Anschluss von 1 oder 2 ±10 Volt Signalen.
eTcPfwTermT_KL301x	K-Bus Klemmen für den Anschluss von 1 oder 2 0..20 Milliampere Signalen.
eTcPfwTermT_KL302x	K-Bus Klemmen für den Anschluss von 1 oder 2 4..20 Milliampere Signalen.
eTcPfwTermT_KL304x	K-Bus Klemmen für den Anschluss von 1, 2 oder 4 0..20 Milliampere Signalen.
eTcPfwTermT_KL305x	K-Bus Klemmen für den Anschluss von 1, 2 oder 4 4..20 Milliampere Signalen.
eTcPfwTermT_KL306x	K-Bus Klemmen für den Anschluss von 1, 2 oder 4 0..10 Volt Signalen.
eTcPfwTermT_KL310x	K-Bus Klemmen für den Anschluss von 2 ±10 Volt Signalen.
eTcPfwTermT_KL311x	K-Bus Klemmen für den Anschluss von 2 0..20 Milliampere Signalen.
eTcPfwTermT_KL312x	K-Bus Klemmen für den Anschluss von 2 4..20 Milliampere Signalen.
eTcPfwTermT_KL313x	K-Bus Klemmen für den Anschluss von 2 ±10 Volt Signalen.
eTcPfwTermT_KL314x	K-Bus Klemmen für den Anschluss von 2 0..20 Milliampere Signalen.
eTcPfwTermT_KL315x	K-Bus Klemmen für den Anschluss von 2 4..20 Milliampere Signalen.
eTcPfwTermT_KL317x	K-Bus Klemmen für den Anschluss von 2 0..2 Volt Signalen.
eTcPfwTermT_KL318x	K-Bus Klemmen für den Anschluss von 2 ±2 Volt Signalen.
eTcPfwTermT_KL340x	K-Bus Klemmen für den Anschluss von 4 oder 8 ±10 Volt Signalen.
eTcPfwTermT_KL344x	K-Bus Klemmen für den Anschluss von 4 oder 8 0..20 Milliampere Signalen.
eTcPfwTermT_KL346x	K-Bus Klemmen für den Anschluss von 4 oder 8 0..10 Volt Signalen.
eTcPfwTermT_KL331x	K-Bus Klemmen für den direkten Anschluss von 1, 2 oder 4 Thermoelementen.
eTcPfwTermT_KL_RangeHigh	Obergrenze der Kennungen für K-Bus Klemmen.
eTcPfwTermT_EL_RangeLow	Untergrenze der Kennungen für EtherCAT-Klemmen.
eTcPfwTermT_EL331x	EtherCAT-Klemmen für den direkten Anschluss von 1, 2 oder 4 Thermoelementen.
eTcPfwTermT_EL320x	EtherCAT-Klemmen für den direkten Anschluss von PT100- oder PT1000-Sensoren.
eTcPfwTermT_EL316x	EtherCAT-Klemmen für den Anschluss von analogen Eingängen.
eTcPfwTermT_EL_RangeHigh	Obergrenze der Kennungen für EtherCAT-Klemmen.
eTcPfwTermT_XX_RangeLow	Untergrenze der Kennungen für sonstige Baugruppen.
eTcPfwTermT_FM33xx	Profibus-Modul für den direkten Anschluss von 12 bzw. 32 Thermoelementen. Hinweis: Es ist ein Anpassbaustein aufzurufen.
eTcPfwTermT_EM8908	I/O-Board für Spritzgießmaschinen.
eTcPfwTermT_XX_RangeHigh	Obergrenze der Kennungen für sonstige Baugruppen.
eTcPfwTermT_Customized	Der Temperatureingang wird von der Applikation versorgt.

3.6 Strukturen

3.6.1 Mapping

3.6.1.1 ST_TcPfw_TempCtrlOutput

Diese Struktur enthält die Ausgangsdaten einer Zone für die E/A Verknüpfung.

HINWEIS

Die mit dem SystemManager hergestellte Verknüpfung mit den Variablen der E/A-Klemme muss zum eingestellten Typ in aaaPfwTempMparamFromHmi[...].TempSensTerm passen.

Syntax

```

TYPE ST_TcPfw_TempCtrlOutput :
(* last modification: 15.07.2008 *)
STRUCT
(*
=====
temperature zone output data
see cnv_TempCtrlOutput_TcPfw for format definition
=====
*)
Htr_Analog : INT;
KL_SnsData : INT;
KL_SnsCtrl : USINT;
Htr_PwmPos : BOOL;
Htr_PwmNeg : BOOL;
Htr_DigPos : BOOL;
Htr_DigNeg : BOOL;
SelOutNeg : BOOL;
SelOutPos : BOOL;
NoHighAlarm: BOOL;
END_STRUCT
END_TYPE

```

Parameter

Name	Typ	Beschreibung
Htr_Analog	INT	Der Leistungsbedarf der Zone als Integer-Zahl im Bereich ± 32767 .
KL_SnsData	INT	Bei Verwendung einer Klemme aus der Typenfamilie KL331x ist der Prozesswert "Daten Aus" hierher zu verknüpfen. Diese Verbindung wird für die Register-Kommunikation genutzt.
KL_SnsCtrl	USINT	Bei Verwendung einer Klemme aus der Typenfamilie KL331x ist der Prozesswert "Kontrolle" hierher zu verknüpfen. Diese Verbindung wird für die Register-Kommunikation genutzt.
Htr_PwmPos	BOOL	Das PWM-Heizsignal. Sollte nicht mehr verwendet werden, stattdessen bitte SelOutNeg verwenden.
Htr_PwmNeg	BOOL	Das PWM-Kühlsignal. Sollte nicht mehr verwendet werden, stattdessen bitte SelOutPos verwenden.
Htr_DigPos	BOOL	Das digitale Heizsignal. Sollte nicht mehr verwendet werden, stattdessen bitte SelOutNeg verwenden.
Htr_DigNeg	BOOL	Das digitale Kühlsignal. Sollte nicht mehr verwendet werden, stattdessen bitte SelOutPos verwenden.
SelOutNeg	BOOL	Das durch aaaPfwTempMparamFromHmi[...].OutputSel_C ausgewählte Kühlsignal.
SelOutPos	BOOL	Das durch aaaPfwTempMparamFromHmi[...].OutputSel_H ausgewählte Heizungssignal.
NoHighAlarm	BOOL	Dieser Ausgabe ist TRUE solange die Temperatur der Zone den Schwellwert aaaPfwTempMparamFromHmi[...].AbsoluteHigh nicht überschreitet.

3.6.1.2 ST_TcPfw_TempCtrlInput

Diese Struktur enthält die Eingangsdaten einer Zone für die E/A Verknüpfung.

HINWEIS

Die mit dem SystemManager hergestellte Verknüpfung mit den Variablen der E/A-Klemme muss zum eingestellten Typ in aaaPfwTempMparamFromHmi[..].TempSensTerm passen.

Syntax

```
TYPE ST_TcPfw_TempCtrlInput :
(* last modification: 11.01.2008 *)
STRUCT
(*)
=====
temperature zone input data
see cnv_TempCtrlInput_TcPfw for format definition
=====
*)
KL_SnsData      : INT;
EL_SnsData      : INT;
EL_SnsState     : UINT;
KL_SnsState     : USINT;
EL_SnsUnderrun  : BOOL;
EL_SnsOverrun   : BOOL;
EL_SnsError     : BOOL;
EL_SnsWcState   : BOOL;
EL_AdsAddr      : ST_TcPfw_AdsAddr;
END_STRUCT
END_TYPE
```

Parameter

Name	Typ	Beschreibung
KL_SnsData	INT	Bei Verwendung einer Klemme aus der Typenfamilie KL331x ist der Prozesswert "Daten Ein" hierher zu verknüpfen. Diese Verbindung wird für die Ermittlung der Ist-Temperatur und für die Register-Kommunikation genutzt.
EL_SnsData	INT	Bei Verwendung einer Klemme aus der Typenfamilie EL331x ist der Prozesswert "State" hierher zu verknüpfen. Diese Verbindung wird für die Ermittlung der Ist-Temperatur genutzt.
EL_SnsState	UINT	Bei Verwendung einer Klemme aus der Typenfamilie EL331x ist der Prozesswert "State" hierher zu verknüpfen. Diese Verbindung wird für die Überwachung des Klemmen-Betriebszustands genutzt.
KL_SnsState	USINT	Bei Verwendung einer Klemme aus der Typenfamilie KL331x ist der Prozesswert "Status" hierher zu verknüpfen. Diese Verbindung wird für die Diagnose und für die Register-Kommunikation genutzt.
EL_SnsUnderrun	BOOL	Bei Verwendung einer Klemme aus der Typenfamilie EL331x ist das Signal "Underrange" hierher zu verknüpfen. Diese Verbindung wird für die Diagnose genutzt.
EL_SnsOverrun	BOOL	Bei Verwendung einer Klemme aus der Typenfamilie EL331x ist das Signal "Overrange" hierher zu verknüpfen. Diese Verbindung wird für die Diagnose genutzt.
EL_SnsError	BOOL	Bei Verwendung einer Klemme aus der Typenfamilie EL331x ist das Signal "Error" hierher zu verknüpfen. Diese Verbindung wird für die Diagnose genutzt.
EL_SnsWcState	BOOL	Bei Verwendung einer Klemme aus der Typenfamilie EL331x ist das Prozesswert "WcState" hierher zu verknüpfen. Diese Verbindung wird für die Verbindungsüberwachung genutzt.
EL_AdsAddr	ST_TcPfw_AdsAddr	Bei Verwendung einer Klemme aus der Typenfamilie EL331x ist der Prozesswert "AdsAddr" hierher zu verknüpfen. Diese Verbindung wird für die CoE-Kommunikation genutzt.

3.6.1.3 Leistungsmessung**3.6.1.3.1 ST_TcPfw_EL3773_Input**

Eine solche Struktur enthält die Eingangsdaten für die Leistungsmessung.

Syntax

```

TYPE ST_TcPfw_EL3773_Input:
(* location PfwLib_TempControl.PRO *)
(* last modification: 08.09.2010 *)
STRUCT
    uiStatusU1    : UINT;
    iVoltageU1    : ARRAY[1..cnOversampling] OF INT;
    uiStatusU2    : UINT;
    iVoltageU2    : ARRAY[1..cnOversampling] OF INT;
    uiStatusU3    : UINT;
    iVoltageU3    : ARRAY[1..cnOversampling] OF INT;
    uiStatusI1    : UINT;
    iCurrentI1    : ARRAY[1..cnOversampling] OF INT;
    uiStatusI2    : UINT;
    iCurrentI2    : ARRAY[1..cnOversampling] OF INT;
    uiStatusI3    : UINT;
    iCurrentI3    : ARRAY[1..cnOversampling] OF INT;

```

```

SampleCount : UINT;
WcState     : BOOL;
InputToggle : BOOL;
State       : UINT;

DcOutputShift: UDINT;
DcInputShift : UDINT;

AdsAddr      : ST_TcPfw_AdsAddr;

END_STRUCT
END_TYPE
    
```

Parameter

Name	Typ	Beschreibung
uiStatusU1	UINT	Status des ersten Spannungskanals
iVoltageU1	ARRAY	Spannung des ersten Kanals
uiStatusU2	UINT	Status des zweiten Spannungskanals
iVoltageU2	ARRAY	Spannung des zweiten Kanals
uiStatusU3	UINT	Status des dritten Spannungskanals
iVoltageU3	ARRAY	Spannung des dritten Kanals
uiStatusI1	UINT	Status des ersten Stromkanals
iCurrentI1	ARRAY	Stromwert des ersten Stromkanals
uiStatusI2	UINT	Status des zweiten Stromkanals
iCurrentI2	ARRAY	Stromwert des zweiten Stromkanals
uiStatusI3	UINT	Status des dritten Stromkanals
iCurrentI3	ARRAY	Stromwert des dritten Stromkanals
SampleCount	UINT	Der SampleCounter wird mit jedem Prozessdatenzyklus um eine Einheit hoch gezählt. Durch den CycleCounter kann die übergeordnete Steuerung eine Kontrolle vornehmen, ob evtl. ein Datensatz ausfiel oder doppelt übertragen wurde. Dann ist i.d.R. die DC-ShiftZeit der Klemme anzupassen.
WcState	BOOL	Muss mit der "WcState" Variablen der EL-Klemme verknüpft werden. Hierüber wird erkannt, ob die Prozessdaten, die von der Klemme kommen, in Ordnung sind.
InputToggle	BOOL	Die Variable InputToggle zählt, ob ein neues gültiges Telegramm empfangen wurde. Nach jedem erfolgreichen Zyklus wird der Wert um eins hochgezählt.
State	UINT	Muss mit der "State" Variablen der EL-Klemme verknüpft werden. Hierüber wird der aktuelle Zustand der Klemme zurückgegeben.
DcOutputShift	UDINT	DcOutputShift ist die Zeit für die Ausgabe der Prozessdaten an den Antrieb, d.h. für den Zeitverzug zwischen Berechnung und Wirkung dieser Daten.
DcInputShift	UDINT	DcInputShift ist die Zeit, die benötigt wird, um Statusinformationen, wie z. B. die Istposition eines Antriebs, in die Steuerung zu übertragen. Also die Zeit zwischen der Erfassung und der Auswertung dieser Daten.
AdsAddr	ST_TcPfw_AdsAddr	Bei Verwendung der EL3773 Klemme ist der Prozesswert "AdsAddr" hierher zu verknüpfen. Diese Verbindung wird für die CoE-Kommunikation genutzt.

3.6.1.3.2 ST_TcPfw_xL3403_Input

Eine solche Struktur enthält die Eingangsdaten für die Leistungsmessung.

Syntax

```

TYPE ST_TcPfw_xL3403_Input:
(* location PfwLib_TempControl.PRO *)
(* last modification: 08.09.2010 *)
STRUCT
  KL_DataIn      : ARRAY[1..3] OF INT;
  KL_State       : ARRAY[1..3] OF USINT;

  EL_Current     : ARRAY[1..3] OF DINT;
  EL_Voltage     : ARRAY[1..3] OF DINT;
  EL_Power       : ARRAY[1..3] OF DINT;
  EL_NoZeroCross : ARRAY[1..3] OF BOOL;
  EL_WcState     : BOOL;
  EL_State       : UINT;
  EL_AdsAddr     : ST_TcPfw_AdsAddr;
END_STRUCT
END_TYPE
    
```

Parameter

Name	Typ	Beschreibung
KL_DataIn	ARRAY OF INT	Muss mit "Daten ein" verknüpft werden. Hierüber wird mit den Klemmen kommuniziert.
KL_State	ARRAY OF USINT	Muss mit der "Status" Variablen der KL-Klemme verknüpft werden. Hierüber wird der Status der Klemme übergeben.
EL_Current	ARRAY OF DINT	Muss mit der "Current" Variablen der EL-Klemme verknüpft werden.
EL_Voltage	ARRAY OF DINT	Muss mit der "Voltage" Variablen der EL-Klemme verknüpft werden.
EL_Power	ARRAY OF DINT	Muss mit der "Active Power" Variablen der EL-Klemme verknüpft werden. In dieser Variablen ist die Leistung hinterlegt.
EL_NoZeroCross	ARRAY OF BOOL	Muss mit "Missing Zero Crossing" der EL-Klemme verknüpft werden.
EL_WcState	BOOL	Muss mit der "WcState" Variablen der EL-Klemme verknüpft werden. Hierüber wird erkannt, ob die Prozessdaten, die von der Klemme kommen, in Ordnung sind.
EL_State	UINT	Muss mit der "State" Variablen der EL-Klemme verknüpft werden. Hierüber wird der aktuelle Zustand der Klemme zurückgegeben.
EL_AdsAddr	ST_TcPfw_AdsAddr	Bei Verwendung der EL3403 Klemme ist der Prozesswert "AdsAddr" hierher zu verknüpfen. Diese Verbindung wird für die CoE-Kommunikation genutzt.

3.6.1.3.3 ST_TcPfw_xL3403_Output

Eine solche Struktur enthält die Ausgangsdaten für die Leistungsmessung.

Syntax

```

TYPE ST_TcPfw_xL3403_Output :
(* location PfwLib_TempControl.PRO *)
(* last modification: 08.09.2010 *)
STRUCT
  KL_Ctrl      : ARRAY[1..3] OF USINT;
END_STRUCT
END_TYPE
    
```

Parameter

Name	Typ	Beschreibung
KL_Ctrl	ARRAY OF USINT	Muss mit "Daten Aus" in der Klemme verknüpft werden. Hierüber wird definiert, welche Prozessdaten die Klemme zur Verfügung stellt soll.

Beispiel: 0 -> Scheinleistung.

3.6.1.3.4 ST_TcPfw_FM3332_Input

Eine solche Struktur enthält die Eingangsdaten eines FM3312 oder FM3332 Feldbusmoduls.



Die mit dem SystemManager hergestellte Verknüpfung mit den Variablen des E/A-Moduls muss zum eingestellten Typ in `aaaPfwTempMparamFromHmi[...].TempSensTerm` passen.



Um das Prozessabbild eines FM33xx Moduls auswerten zu können, muss es auf die Prozessabbilder der Zonen verteilt und konvertiert werden. Dazu ist ein Baustein vom Typ `FB_TempCtrlAdaptFm33xx_TcPfw()` zu verwenden.

Syntax

```

TYPE ST_TcPfw_FM3332_Input:
(* last modification: 16.03.2009 *)
STRUCT
DpState      : USINT;
ExtDiagFlag  : BOOL;
(**)
Kanal_Daten  : ARRAY[1..32] OF UINT;
(**)
OpenCircuit  : ARRAY[0..3] OF SINT;
Backvoltage  : ARRAY[0..3] OF SINT;
END_STRUCT
END_TYPE
    
```

Parameter

Name	Typ	Beschreibung
DpState	USINT	Der ProfiBus DP State. Dieser Status dient zur Überwachung der Kommunikation und des Gerätezustands.
ExtDiagFlag	BOOL	reserviert
Kanal_Daten	ARRAY OF UINT	Die Istwerte von bis zu 32 Kanälen.
OpenCircuit	ARRAY OF SINT	In vier Bytes á 8 Bit sind Alarmsignale für bis zu 32 Kanäle zusammengefasst. Gemeldetes Problem: Drahtbruch im Messkreis.
Backvoltage	ARRAY OF SINT	In vier Bytes á 8 Bit sind Alarmsignale für bis zu 32 Kanäle zusammengefasst. Gemeldetes Problem: Fremdspannung im Messkreis.

3.6.1.3.5 ST_TcPfw_xL3403_State

Eine solche Struktur enthält die Ergebnisse der Leistungsmessung und stellt dies der Applikation bereit.

Syntax

```

TYPE ST_TcPfw_xL3403_State :
(* location PfwLib_TempControl.PRO *)
(* last modification: 08.09.2010 *)
STRUCT
    Power: ARRAY[1..3] OF LREAL;
    Voltage: ARRAY[1..3] OF LREAL;
    Current: ARRAY[1..3] OF LREAL;
    LineError: ARRAY[1..3] OF BOOL;

    SubType      : INT;
    ErrorID      : INT;
    LatchedErrID : INT;
    LatchedErr   : BOOL;
    Error        : BOOL;
    Ready        : BOOL;
END_STRUCT
END_TYPE
    
```

Parameter

Name	Typ	Beschreibung
Power	ARRAY OF LREAL	Aufbereitete Leistung zur Weiterverarbeitung.
Voltage	ARRAY OF LREAL	Aufbereitete Spannung zur Weiterverarbeitung.
Current	ARRAY OF LREAL	Aufbereiteter Strom zur Weiterverarbeitung.
LineError	ARRAY OF BOOL	Ein TRUE signalisiert, dass die Klemme sich in einem Fehlerzustand befindet.
SubType	INT	Der Klemmensubtyp, dieser beeinflusst den Umrechnungsfaktor.
ErrorID	INT	Über diese FehlerID wird eine detaillierte Information zum Fehler geliefert. Die Fehlercodes können in den Konstanten nachgeschlagen werden.
LatchedErrID	INT	Hier wird die letzte aktive ErrorID auch nach einem Fehlerreset gespeichert.
LatchedErr	BOOL	Latched Error = TRUE bleibt auch nach einem Fehlerreset aktiv. Dieser muss aktiv zurückgesetzt werden, d.h. LatchedError = FALSE.
Error	BOOL	Ein TRUE signalisiert, dass der Baustein sich in einem Fehlerzustand befinden. Durch ein TRUE am Reset-Eingang des Bausteins FB_xL3403_TcPfw.htm ist es möglich, den Baustein zurückzusetzen.
Ready	BOOL	Signalisiert bei einer KL-Klemme eine abgeschlossene Wandlung.

3.6.2 ST_TcPfw_TempMparamFromHmi_Itf

Eine solche Struktur enthält die Maschinendaten einer Zone.

Syntax

```

TYPE ST_TcPfw_TempMparamFromHmi_Itf :
(* last modification: 20.12.2010 *)
STRUCT
(*
=====
temperature zone machine parameters see cnv_TempMparamFromHmi_TcPfw for format definition
=====
*)
ZoneName: STRING(79);

AbsoluteHigh      : LREAL;
AbsoluteLow       : LREAL;
ExtruderComp      : LREAL;
KpCool            : LREAL;
KpHeat            : LREAL;
TdCool            : LREAL;
TdHeat            : LREAL;
TnCool            : LREAL;
TnHeat            : LREAL;
TvCool            : LREAL;
TvHeat            : LREAL;
Overshoot         : LREAL;
Tracking_Td       : LREAL;
Ramping_Rate      : LREAL; (* starting with cnv_TempMparamFromHmi_TcPfw=9 *)
Ramping_RateC     : LREAL; (* starting with cnv_TempMparamFromHmi_TcPfw=15 *)
Ramping_Tolerance : LREAL; (* starting with cnv_TempMparamFromHmi_TcPfw=9 *)
dTmax             : LREAL;
SensorOffset      : LREAL;
SettlingTime      : LREAL;
SupplyLoad_Cooler : LREAL;
SupplyLoad_Heater : LREAL;
SupplyLoad_Tolerance : LREAL;
TuneEnd           : LREAL;
TuneKp            : LREAL;
TuneTd            : LREAL;
TuneTn            : LREAL;
TuneTv            : LREAL;
TuneTrackingTd    : LREAL:=0.0; (* starting with cnv_TempMparamFromHmi_TcPfw=15 *)
    
```

```

TuneY                : LREAL;

L_LoadIdle           : LREAL;
Weighting_C          : LREAL;

ErrorHeatingFactor   : LREAL:=0.0; (* starting with V1.0.8: will define default heating in error sta
te *)
fPwmStdMaxOnTime     : LREAL;
fPwmMaxOnTime        : ARRAY[cnv_TempCtrl_SetpointFirst..cnv_TempCtrl_SetpointLast] OF LREAL:=0.0;
fPwmMinOnTime        : LREAL:=0.0;
fc_OnTime             : LREAL:=0.0; (* starting with cnv_TempMparamFromHmi_TcPfw=15 *)
fc_OffTime           : LREAL:=0.0; (* starting with cnv_TempMparamFromHmi_TcPfw=15 *)
ActTempGain          : LREAL:=1.0; (* starting with cnv_TempMparamFromHmi_TcPfw=18 *)
ActTempOffset        : LREAL:=0.0; (* starting with cnv_TempMparamFromHmi_TcPfw=18 *)
SaveDelay            : DINT:=-1; (* starting with cnv_TempMparamFromHmi_TcPfw=17 *)

OutputSel_H          : E_TcPfw_TctrlOutSelect:=eTcPfwTcOut_PWM;
OutputSel_C          : E_TcPfw_TctrlOutSelect:=eTcPfwTcOut_PWM;
TempSensTerm         : E_TcPfw_TerminalType :=eTcPfwTermT_NoTerminal;
SensorType           : E_TcPfw_TempSensType:=eTcPfwTempSensT_NoSensor;
App_HmiType          : INT:=0;
TermChannel          : INT;
ExtruderId           : INT;
ModuleId             : INT;
ZoneId               : INT;
SupplyId             : INT;

CJ_CompMode          : INT; (* starting with cnv_TempMparamFromHmi_TcPfw=8 *)
CJ_CompZone          : INT; (* starting with cnv_TempMparamFromHmi_TcPfw=8 *)
TermIdx              : INT; (* used to connect the zone to a terminal *)
HeaterSwapIdx        : INT; (* used for I/O re-location of selected heater signal *)
CoolerSwapIdx        : INT; (* used for I/O re-location of selected cooler signal *)
nPwmFactorC          : INT;
eTuningMethod        : E_TcPfw_TctrlTuningMethod;

InUse                : BOOL;
UseCooling           : BOOL;
ExtruderCompEna     : BOOL;
TuneCooling          : BOOL;
Autotune             : BOOL;
StartReTune          : BOOL;
Enable               : BOOL;
Update              : BOOL;
EnaExtruderBlock    : BOOL;
NoFanWhileTrackDown : BOOL;
Ena_TuneIdleLoad     : BOOL;
LoopTestUpdate       : BOOL:=FALSE;
EnableErrorHeating   : BOOL:=FALSE; (* starting with V1.0.8: will enable default heating in error st
ate *)
ReadBack             : BOOL:=FALSE;
TuneExtruderComp     : BOOL;
TuneHeaterLoad       : BOOL:=FALSE; (* tuning heater power monitoring *)
OpenLoopHeating      : BOOL:=FALSE; (* starting with cnv_TempMparamFromHmi_TcPfw=15 *)

fc_Enable            : BOOL:=FALSE; (* starting with cnv_TempMparamFromHmi_TcPfw=15 *)
HibernateI_Cool      : BOOL:=FALSE; (* starting with cnv_TempMparamFromHmi_TcPfw=15 *)
HibernateI_Heat      : BOOL:=FALSE; (* starting with cnv_TempMparamFromHmi_TcPfw=15 *)
bSavingParams        : BOOL:=FALSE; (* *)
bLoadParams          : BOOL:=FALSE; (* *)
bHighPrecision       : BOOL;
bDisableTerminalCom  : BOOL;
bReset               : BOOL;
END_STRUCT
END_TYPE

```

Variable	Überwachung	Aktion	Regelung	Konfiguration	Auto-tuning	Beschreibung
AbsoluteHigh	x					Überschreitet die Ist-Temperatur der Zone diesen Grenzwert, wird ein Alarm ausgelöst.
AbsoluteLow	x					Unterschreitet die Ist-Temperatur der Zone diesen Grenzwert, wird ein Alarm ausgelöst.
ActTempGain				x		Skalierung für die aktuelle Temperatur (für eine nachträglichen Kalibrierung)
ActTempOffset				x		Offset für die aktuelle Temperatur (für eine nachträglichen Kalibrierung)
Autotune		x				Ein TRUE hier aktiviert das Autotuning (automatische Parameterermittlung) der Zone. Um das Autotuning erfolgreich durchzuführen, muss die Zone stabil sein, InUse, Enable und mindestens einen Temperaturhub von 40 °C durchlaufen können.
App_HmiType						Stellt eine Nummerierung dar, die nur von der Applikation genutzt wird, aber von der Bibliothek gespeichert wird. Hierüber sollen Gruppenbildungen (Heißkanal, Zylinder1, Zylinder2 usw.) möglich gemacht werden.
bDisableTerminalCom		x				Deaktiviert die Klemmenkommunikation für diese Zone.
bHighPrecision			x			Die Regelung der Ist-Temperatur erfolgt hochgenau. Dadurch kann das Aufheizen mehr Zeit in Anspruch nehmen.
bReset		x				Führt in dieser Zone ein Reset aus.
bSavingParams		x				Hierüber wird signalisiert, dass Maschinendaten gespeichert werden.
bLoadParams		x				Ein TRUE löst das Laden von Maschinenparametern aus.
CJ_CompMode						Aktiviert die externe Kompensation bei Thermoelementen.
CJ_CompZone						Zone, welche die zu kompensierende Temperatur misst.
CoolerSwapIdx				x		Definiert die Ausgabe von Lüfter-Schaltsignalen über die umlenkbare E/A-Ebene.

Variable	Überwachung	Aktion	Regelung	Konfiguration	Auto-tuning	Beschreibung
dTmax					x	Hier wird beim Autotuning die maximale Anstiegsgeschwindigkeit festgehalten. Der ermittelte Wert wird in °C/s wiedergegeben.
Enable				x		Ein TRUE gibt die Freigabe für ein aktives Heizen oder Kühlen.
EnableError Heating		x				Ein TRUE hier aktiviert die Ausgabe von Heizleistung bei gestörter Temperatur-Sensorik. Das heißt, sobald ein Sensorfehler auftritt, wird die Temperatur ausgegeben, die zum Halten der aktuellen Temperatur notwendig ist. Diese Funktion ist nur dann voll funktionsfähig, wenn vorher das Ena_TuneldleLoad erfolgreich durchgeführt wurde.
EnaExtruder Block						reserviert
Ena_TuneldleLoad		x				Ein TRUE aktiviert die Parameterermittlung für das "IdleLoad". Bei dieser Optimierung wird das System in keiner Weise angeregt.
ErrorHeating Factor				x		Dieser Parameter beeinflusst die Ausgabe von Heizleistung bei gestörter Temperatur-Sensorik. Er kann Werte zwischen 0% und 100% annehmen, wobei 100% die maximale Leistung ist, um die Zone auf der aktuellen Temperatur zu halten.
eTuningMethod						eTcPfwTcTun_StepResponse: Ist der Standard autotune Vorgang. Die Parameter werden über eine Sprungantwort ermittelt.
eTuningMethod						eTcPfwTcTun_OscillationTest: In Vorbereitung
ExtruderComp						Dieser Parameter kompensiert die Friktions- und Transportenergie in einer Zone.

Variable	Überwachung	Aktion	Regelung	Konfiguration	Auto-tuning	Beschreibung
ExtruderCompEna						Mit einem TRUE wird die Extruderkompensation aktiviert. Bei Einschalten der Strecke wird automatisch die passende Energie in der jeweiligen Zone bereitgestellt, wodurch ein Einschwingen des Reglers minimiert wird. Drehzahländerungen im Bereich von 20% (bezogen auf die Abgleichdrehzahl) werden problemlos kompensiert, wobei bei großen Drehzahländerungen (z. B. Produktwechsel) eine erneute Kompensation notwendig ist.
ExtruderId						reserviert
fc_Enable			x			Aktivierung einer fluiden Zwangskühlung. Um das Fluid unter dem Heizband nicht zu überhitzen, muss in bestimmten Zeitabständen das Fluid zirkulieren. Durch die Aktivierung erfolgt automatisch eine Verrrechnung der Störung in den Regler.
fc_OnTime			x			Für diese Zeitdauer (in Sekunden) ist die Zwangskühlung aktiv (der Kühlausgang wird geschaltet). Wird während der "fc_OffTime" Phase über die Reglerausgabe gekühlt, wird dies verrechnet.
fc_OffTime			x			Für diese Zeitdauer (in Sekunden) ist die Zwangskühlung inaktiv, kann jedoch jederzeit über die Reglerausgabe aktiviert werden.
fPwmMinOnTime			x			Über diesen Faktor kann die Mindest-PWM-Einschaltzeit in Bezug auf die Zykluszeit festgelegt werden. Es ist ein Wert zwischen 0.1 und 1.0 einzugeben.
fPwmMaxOnTime			x			Über diesen Faktor kann die maximale PWM-Einschaltzeit in Bezug auf die Zykluszeit festgelegt werden. Es ist ein Wert zwischen 0.05 und $0.75 * PwmMaxOn$ einzugeben. Jedem Sollwert im Array ist eine PwmMaxOnTime zugeordnet. Bei Null wird die PwmMaxOnTime aus der SupplyLine verwendet.

Variable	Überwachung	Aktion	Regelung	Konfiguration	Auto-tuning	Beschreibung
fPwmStdMaxOnTime			x			Über diesen Faktor kann die Maximale-PWM Einschaltzeit in Bezug auf die Zykluszeit festgelegt werden. Es ist ein Wert zwischen 0.05 und 0.75*PwmMaxOn einzugeben. Diese Variable ist aktiv, wenn auf den std. Sollwert geregelt wird. Bei Null wird die PWMMaxOnTime aus der SupplyLine verwendet.
HeaterSwapIdx				x		Definiert die Ausgabe von Heizungs-Schaltsignalen über die umlenkbare E/A-Ebene.
Hibernatel_Heat						Ein TRUE lässt den I-Anteil der Heizungsregelung einfrieren.
Hibernatel_Cool						Ein TRUE lässt den I-Anteil des Kühlreglers einfrieren.
InUse				x		Die Zone wird durch TRUE zu einem aktiven Bestandteil der Temperaturregelung. Bei FALSE wird die Zone auch bei gesetztem Enable und einem Einschalten der Gruppe nicht aktiv. Sie wird zu keinem Zeitpunkt und aus keinem Anlass einen Fehler signalisieren und weder bei der Lastverteilung noch bei einer optionalen Strommessung berücksichtigt.
KpCool			x			Der Parameter für den P-Anteil des Temperaturreglers. Dieser Parameter sollte durch ein Autotuning ermittelt werden.
KpHeat			x			Der Parameter für den P-Anteil des Temperaturreglers. Dieser Parameter sollte durch ein Autotuning ermittelt werden.
L_LoadIdle			x			Dieser Parameter stellt den Grundleistungsbedarf dar. Ein gut eingestellter IdleLoad ermöglicht ein überschwingfreies Erreichen der Solltemperatur sowie für ein gutes Verhalten beim Error-Heating. Dieser Parameter sollte durch ein Idle-Tune ermittelt werden.
LooptestUpdate						reserviert.

Variable	Überwachung	Aktion	Regelung	Konfiguration	Auto-tuning	Beschreibung
ModuleId				x		Dieser Parameter ordnet der Zone eine Temperaturgruppe zu. Eine Reihe von Funktionen (z.B. Einschalten, Absenkung usw.) werden innerhalb einer Temperaturgruppe organisiert und gesteuert.
NoFanWhileTrackDown				x		Wenn dieser Parameter gesetzt ist, wird eine vorhandene Kühlung nicht genutzt, um bei einer Verringerung der Vorgabe die Solltemperatur zu erreichen.
nPwmFactorC			x			Mit diesem Faktor wird die PWM-Zykluszeit multipliziert, um eine angemessene Zykluszeit beim Kühlen zu realisieren.
OutputSel_C				x		Diese Parameter bestimmen, welches der angebotenen Signale für die Heizung ausgewählt wird.
OutputSel_H				x		Diese Parameter bestimmen, welches der angebotenen Signale für die (optionale) Kühlung ausgewählt wird.
Overshoot					x	Hier wird beim Autotuning die Höhe des Überschießens festgehalten. Hierüber können Rückschlüsse auf die Dynamik des Systems gezogen werden.
OpenloopHeating		x				Hat eine Zone keinen Sensor, kann über einen festen Stellwert die Zone gesteuert geheizt werden.
Ramping_Rate			x		x	Gibt an, mit welcher Steigung der Reglersollwert den eingegebenen Sollwert beim Heizen erreichen soll. Die Eingabe ist in °C/min vorzunehmen Dieser Parameter sollte durch ein Autotuning ermittelt werden.
Ramping_RateC			x		x	Gibt an, mit welcher Steigung der Reglersollwert den eingegebenen Sollwert beim Kühlen erreichen soll. Die Eingabe ist in °C/min vorzunehmen Dieser Parameter sollte durch ein Autotuning ermittelt werden.

Variable	Überwachung	Aktion	Regelung	Konfiguration	Auto-tuning	Beschreibung
Ramping_Tolerance			x			Gibt an, ab wann eine Sollwertänderung über eine Rampe erhöht werden soll. Es ist zu empfehlen Sollwertänderungen von 5 bis 10 °C über das Ramping anzufahren. Sollte die automatisch ermittelte Rampe eine zu kleine Steigung haben, kann diese über den Parameter Ramping_Rate verändert werden.
ReadBack						reserviert.
SaveDelay				x		Zeit in ns. Der Wert wird kontinuierlich heruntergezählt. Erreicht der Wert 0, wird das Speichern aktiviert. -1 bedeutet Ruhezustand.
SensorOffset						Hier kann ein bei der Ermittlung der Ist-Temperatur zu verwendender Offset vorgegeben werden. Die Vorgabe hat in °C zu erfolgen.
SensorType				x		Diese Parameter bestimmen, welche der unterstützten Sensortypen für die Erfassung der Ist-Temperatur verwendet werden.
SettlingTime	x		x		x	Dieser Parameter wird an diversen Stellen benutzt, um das Zeitverhalten der Zone zu berücksichtigen.
StartReTune					x	Sollte im Betrieb der Regler nicht ideal regeln, besteht die Möglichkeit einer nachträglichen Selbstoptimierung. Dies kann während der Produktion erfolgen.
SupplyId				x		Dieser Parameter ordnet die Zone einer Versorgungsgruppe zu. Sie wird einige Parameter für die PWM-Ausgabe von dieser Gruppe übernehmen. Eine Reihe von Funktionen (z.B. Lastverteilung, Strommessung usw.) werden innerhalb einer Versorgungsgruppe organisiert und synchronisiert. Standardmäßig gibt es 4 verschiedene Versorgungsgruppen, wobei 1 bis 3 für die Phasen 1 bis 3 zu verwenden sind. In der Versorgungsgruppe 4 können zwei oder dreiphasige Heizbänder eingegeben werden.

Variable	Überwachung	Aktion	Regelung	Konfiguration	Auto-tuning	Beschreibung
SupplyLoad_Cooler						Die Kühlleistung der Zone in Watt.
SupplyLoad_Heater	x		x		x	Die Heizleistung der Zone in Watt. Bei einer (optionalen) Überwachung der Heizleistung ist dies der Sollwert.
SupplyLoad_Tolerance	x					Überschreitet die Abweichung der gemessenen Heizleistung diese Toleranz, wird ein Alarm ausgelöst. Die Tolleranz wird in % zwischen 0.0 und 100.0 angegeben. Wird hier 0.0 eingestellt oder der FB_TempCtrlMainBody_TcPfw() Baustein mit Looptest_Enable:=FALSE aufgerufen, erfolgt keine Überwachung.
TdCool			x			Der Parameter für den D-Anteil (Dämpfungszeit) des Temperaturreglers. Dieser Parameter sollte durch ein Autotuning ermittelt werden.
TdHeat			x			Der Parameter für den D-Anteil (Dämpfungszeit) des Temperaturreglers. Dieser Parameter sollte durch ein Autotuning ermittelt werden.
TempSensTerm				x		Diese Parameter bestimmen, welche der unterstützten E/A-Klemmen für die Erfassung der Ist-Temperatur verwendet wird.
TermChannel				x		Bei mehrkanaligen Klemmen ist hier der Kanal innerhalb der Klemme anzugeben.
TermIdx				x		Definiert das Umlenken der Ist-Temperatur -Erfassung.
TnCool			x			Der Parameter für den I-Anteil des Temperaturreglers. Dieser Parameter sollte durch ein Autotuning ermittelt werden.
TnHeat			x			Der Parameter für den I-Anteil des Temperaturreglers. Dieser Parameter sollte durch ein Autotuning ermittelt werden.

Variable	Überwachung	Aktion	Regelung	Konfiguration	Auto-tuning	Beschreibung
Tracking_Td			x			Dieser Parameter wird für die Regelung gerade von großen Sollwertänderungen verwendet. Dieser Parameter sollte durch ein Autotuning ermittelt werden.
TuneCooling					x	Nur wenn hier ein TRUE eingetragen ist, wird beim Autotuning das Abkühlverhalten ausgewertet.
TuneEnd					x	Dieser Anteil in Prozent vom Temperatursollwert wird beim Autotuning ausgenutzt.
TuneExtruderComp			x			Ein TRUE hier berechnet die ExtruderComp.
TuneHeaterLoad			x			Ein TRUE hier berechnet die Heizleistung dieser Zone.
Tune_IdleLoad			x			Ein TRUE hier berechnet den IdleLoad dieser Zone während des Autotunings.
TuneKp					x	Der P-Anteil des Autotuning-Mechanismus. Mit diesem Faktor wird der P-Anteil beim Autotuning gewichtet.
TuneTd					x	Der D-Anteil des Autotuning-Mechanismus. Mit diesem Faktor wird die Verzögerung des D-Anteils beim Autotuning gewichtet.
TuneTn					x	Der I-Anteil des Autotuning-Mechanismus. Mit diesem Faktor wird der I-Anteil beim Autotuning gewichtet.
TuneTv					x	Der D-Anteil des Autotuning-Mechanismus. Mit diesem Faktor wird der D-Anteil beim Autotuning gewichtet.
TuneTrackingTd					x	Der D-Anteil des Autotuning-Mechanismus. Mit diesem Faktor wird der D-Anteil des Beckhoff-Algorithmus eingestellt. Ein Wert zwischen Null und eins bedeutet aggressiver Anregeln; Ein Wert größer Eins bedeutet vorsichtiger Anregeln. Default ist Null.
TuneY					x	Dieser Anteil in Prozent von der verfügbaren Heizleistung wird beim Autotuning verwendet.

Variable	Überwachung	Aktion	Regelung	Konfiguration	Auto-tuning	Beschreibung
TvCool			x			Der Parameter für den D-Anteil (Vorhaltezeit) des Temperaturreglers. Dieser Parameter sollte durch ein Autotuning ermittelt werden.
TvHeat			x			Der Parameter für den D-Anteil (Vorhaltezeit) des Temperaturreglers. Dieser Parameter sollte durch ein Autotuning ermittelt werden.
Update		x				Mit einem TRUE signalisiert die Bedienoberfläche hier, dass sie Werte in dieser Struktur verändert hat. Das Framework wird diese Werte überprüfen, bei Bedarf anpassen und übernehmen.
UseCooling				x		Nur wenn hier ein TRUE eingetragen ist, gibt der Regler an die Kühlung aus.
Weighting_C			x		x	Gewichtungsfaktor, worüber die Kühlparameter für die Regelung ermittelt werden.
Zoneld				x		Dieser Parameter nummeriert die Zonen innerhalb der Maschine durch. Der Zahlenwert darf nur in einer einzigen Zone verwendet werden.
ZoneName				x		Der textuelle Name der Zone. Beispiel: 'Ext_1' oder 'Kopf_5'.

3.6.3 ST_TcPfw_TempPparamFromHmi_Itf

Eine solche Struktur enthält die Produktdaten einer Zone der Temperaturregelung.

Syntax

```

TYPE ST_TcPfw_TempPparamFromHmi_Itf :
(* last modification: 11.11.2008 *)
STRUCT
(*
see cnv_TempPparamFromHmi_TcPfw for format definition
Attention: HMI access via address
*)
Setpoint: LREAL;
StandbySetpoint: LREAL;
Setpoints : ARRAY[cnv_TempCtrl_SetpointFirst..cnv_TempCtrl_SetpointLast] OF LREAL; (* AST: supporting selectable setpoints *)
Threshold_PP : LREAL;
Threshold_P : LREAL;
Threshold_M : LREAL;
Threshold_MM : LREAL;

Openloop_Output: LREAL;

SaveDelay: DINT:= -1;

Update : BOOL;
bLoadParams : BOOL:=FALSE; (* *)

```



```
bSavingParams : BOOL:=FALSE; (* *)
END_STRUCT
END_TYPE
```

Parameter

Name	Typ	Beschreibung
Setpoint	LREAL	Der Temperatursollwert der Zone.
StandbySetpoint	LREAL	Dieser Wert wird im Standby von der Zone als Temperatursollwert verwendet.
Setpoints	ARRAY OF LREAL	Hierüber können weitere Sollwerte vorgegeben werden, die einfach über einen Index umzuschalten sind. Die Nummer des Indexes ist in TempCtrl.SelectSetpoint vorzugeben.
Threshold_PP	LREAL	Die äußere positive Toleranzgrenze.
Threshold_P	LREAL	Die innere positive Toleranzgrenze.
Threshold_M	LREAL	Die innere negative Toleranzgrenze.
Threshold_MM	LREAL	Die äußere negative Toleranzgrenze.
Openloop_Output	LREAL	Stellwertausgabe, wenn keine Ist-Temperatur vorhanden ist.
SaveDelay	DINT	Speicherverzögerung in µs. Nach einer von der Applikation auf diese Variable geschriebenen Zeit, wird das Speichern der Istwerte ausgelöst. (Ist der Wert Null, wird gespeichert; bei -1 ist das System in Ruhestellung; ist der Wert größer Null, wird nach Ablauf der Zeit gespeichert.)
Update	BOOL	Mit einem TRUE signalisiert die Bedienoberfläche hier, dass sie Werte in dieser Struktur verändert hat. Das Framework wird diese Werte überprüfen, bei Bedarf anpassen und übernehmen.
bLoadParams	BOOL	Über dieses Flag wird das Speichern der Parameter ausgelöst.
bSavingParams	BOOL	Signalisiert der Applikation, dass gerade Daten gespeichert werden.

3.6.4 ST_TcPfw_TempToHmi_Itf

Eine solche Struktur enthält die Visualisierungsdaten einer Zone der Temperaturregelung.

Syntax

```
TYPE ST_TcPfw_TempToHmi_Itf :
(* last modification: 25.09.2008 *)
STRUCT
ActualTemp : LREAL;
SupplyMatch : LREAL;
ActCurrent : LREAL;
FileErrId : DINT;
ErrorId : UINT;
ModuleId : INT;
PowerLevel : INT;
ZoneId : INT;
Cooling : BOOL;
Enable : BOOL;
Error : BOOL;
FileErr : BOOL;
Heating : BOOL;
InUse : BOOL;
OnStandBy : BOOL;
TuningActive : BOOL;
TuningDone : BOOL;
IdleLoadActive : BOOL;
IdleLoadDone : BOOL;
LoopTestActive : BOOL;
ExtruderCompActive : BOOL;
ExtruderCompDone : BOOL;
(*
*)
END_STRUCT
END_TYPE
```

Parameter

Name	Typ	Beschreibung
ActualTemp	LREAL	Die Ist-Temperatur der Zone.
SupplyMatch	LREAL	Gibt das Verhältnis zwischen gemessener aktueller Leistung und der vorgegebenen Sollleistung wieder.
ActCurrent	LREAL	Aktueller Strom.
FileErrId	DINT	Im Speicher-/ Ladefall der Zone wird hier eine kodierte Information bereitgestellt. Jedoch nur, wenn das Speichern und Laden über die Maschinen und Produktparameter ausgeführt werden.
ErrorId	UINT	Im Fehlerfall wird hier eine kodierte Information bereitgestellt. Die Umsetzung der Fehlernummer in einen Klartext kann in den globalen Variablen eingesehen werden. Fehlernummern, die nicht dort aufgeführt sind, sind in der Regel allgemeine Beckhoff-Fehlernummern von unterlagerten Bausteinen (meistens ADS Fehler).
ModuleId	INT	Diese Id gibt die Gruppe an, der diese Zone zugeordnet ist.
PowerLevel	INT	Dieser Wert gibt den vom Regler vorgegebenen Leistungswert in % wieder.
ZonId	INT	Diese Id gibt die Einordnung der Zone innerhalb ihrer Gruppe wieder.
Cooling	BOOL	Ein TRUE hier signalisiert, dass die Zone aktiv kühlt.
Enable	BOOL	Ein TRUE hier signalisiert, dass der Regler für die Zone freigegeben ist.
Error	BOOL	Ein TRUE hier signalisiert, dass in der Zone ein Regler-, Autotuning- oder Hardware-Fehler aufgetreten ist.
FileErr	BOOL	Ein TRUE hier signalisiert, dass im Speicher-/ Ladefall der Zone ein Fehler aufgetreten ist.
Heating	BOOL	Ein TRUE hier signalisiert, dass die Zone aktiv heizt.
InUse	BOOL	Ein TRUE hier signalisiert, dass die Zone ein aktiver Bestandteil der aktuellen Konfiguration ist. Voraussetzung ist, dass in dem Maschinenparametern ModuleId<>0; ZonId<>0; SupplyId<>0 und InUse:=TRUE
OnStandBy	BOOL	Ein TRUE hier signalisiert, dass die Zone auf den Standby-Sollwert geschaltet wurde.
TuningActive	BOOL	Während des Autotunings der Zone ist dieses Signal TRUE.
TuningDone	BOOL	Hier wird ein erfolgreiches Autotuning der Zone gemeldet.
IdleLoadActive	BOOL	Während des IdleLoad-Tunings der Zone ist dieses Signal TRUE.
IdleLoadDone	BOOL	Hier wird ein erfolgreiches IdleLoad- Tuning der Zone gemeldet.
LooptestActive	BOOL	Ein TRUE hier signalisiert, dass die Leistungsmessung aktiv ist.
ExtruderCompActive	BOOL	Signalisiert, dass die automatische Berechnung der Extruderkompensation aktiv ist.
ExtruderCompDone	BOOL	Signalisiert, dass die automatische Berechnung der Extruderkompensation erfolgreich gewesen ist.

3.6.5 ST_TcPfw_SupplyParam

Diese Struktur beschreibt die Parameter und Laufzeitdaten einer Versorgungseinheit.

Syntax

```

TYPE ST_TcPfw_SupplyParam :
(* last modification: 28.05.2008 *)
STRUCT
(*
see cnv_SupplyParam_TcPfw for format definition
*)
(*
temperature controller pwm setup

```

```
*)
fPwmCycleTime : LREAL; (* will be updated to all temperature zones in supply group *)
fPwmMinOnTime : LREAL; (* will be updated to all temperature zones in supply group *)
fPwmMaxOnTime : LREAL; (* will be updated to all temperature zones in supply group *)
fPwmMaxOnC : LREAL;
fPwmMaxRampLoad : LREAL; (* in kW, will be updated to all temperature zones in supply group *)
)

fActSupplyLoad : LREAL;
fActSupplyCurrent: LREAL;
fSupplyLoad : LREAL;
fSupplyMatch : LREAL;

nPwmFactorC : INT:=1; (* will be updated to all temperature zones in supply group *)

(*
internal
*)
fUsedLoad : LREAL;
fUsedLoad_H : LREAL;
fUsedLoad_C : LREAL;
tpwmtimer : LREAL:=0.0;

SaveDelay : DINT:=-1;
FileErrId : DINT;

refresh_H : BOOL;

Unsaved : BOOL;
bSavingParams : BOOL:=FALSE; (* *)
bLoadParams : BOOL:=FALSE; (* *)
FileErr : BOOL:=FALSE; (* *)
END_STRUCT
END_TYPE
```

Parameter

Name	Typ	Beschreibung
fPwmCycleTime	LREAL	Die Zykluszeit (in Sekunden) des PWM-Signalgenerators.
fPwmMinOnTime	LREAL	Der minimale Einschaltanteil vom PWM-Signal. Bereich 0.1 bis 1.0
fPwmMaxOnTime	LREAL	Die maximale Einschaltzeit vom PWM-Signal. Bereich 0.1 bis 1.0
fPwmMaxOnC	LREAL	Die maximale Einschaltzeit des Kühlausganges vom PWM-Signal. Bei Null wird die fPwmMaxOnTime verwendet. Bereich 0.1 bis 1.0
fPwmMaxRampLoad	LREAL	reserviert
fActSupplyLoad	LREAL	Wenn eine Leistungsmessung durchgeführt wird, kann die aktuelle Leistung hier abgelesen werden.
fActSupplyCurrent	LREAL	Wenn eine Leistungsmessung durchgeführt wird, kann der aktuell gemessene Strom hier abgelesen werden.
fSupplyLoad	LREAL	Die vorausberechnete Gesamtleistung der Versorgungsgruppe.
fSupplyMatch	LREAL	Das Verhältnis von fActSupplyLoad zu fSupplyLoad.
nPwmFactorC	INT	Der Multiplikator für die Kühl-PWM-Zykluszeit. $T_{pwm_cool} := T_{pwm_heat} * nPwmFactorC$.
fUsedLoad	LREAL	reserviert
fUsedLoad_H	LREAL	reserviert
fUsedLoad_C	LREAL	reserviert
tpwmtimer	LREAL	Zähler für die PWM-Zyklen der Heizungsregelung.
SaveDelay	DINT	Speicherverzögerung in μs . Nach einer von der Applikation auf diese Variable geschriebenen Zeit, wird das Speichern ausgelöst. (Ist der Wert Null, wird gespeichert; bei -1 ist das System in Ruhestellung; ist der Wert größer Null, wird nach Ablauf der Zeit gespeichert.)
FileErrId	DINT	Im Speicher-/ Ladefall der Zone wird hier eine kodierte Information bereitgestellt. Jedoch nur, wenn das Speichern und Laden über die Maschinen- und Produktparameter ausgeführt werden.
refresh_H	BOOL	Trigger-Signale für Heizungs- und Kühlungs-PWM-Zyklen.
Unsaved	BOOL	Signalisiert der Applikation, dass sich aus der Bibliothek heraus Parameter verändert haben.
bSavingParams	BOOL	Signalisiert der Applikation, dass gerade Daten gespeichert werden.
bLoadParams	BOOL	Über dieses Flag wird das Speichern der Parameter ausgelöst.
FileErr	BOOL	Im Speicher-/ Ladefall der Zone wird hier angezeigt, wenn der Speicher- oder Ladevorgang fehlgeschlagen ist.

Die Reglerausgabe wird in der Regel als PWM-Signal dem Heizband bereitgestellt. Es ist möglich, für jede Phase einer Versorgungsgruppe eine eigene PWM-Konfiguration bereit zu stellen.

3.6.6 ST_TcPfw_TempCtrl_Itf

Eine solche Struktur enthält die Visualisierungsdaten einer Zone der Temperaturregelung.

Syntax

```

TYPE ST_TcPfw_TempCtrl_Itf :
(* last modification: 01.10.2010 *)
STRUCT
(*
=====
temperature zone internal data
see cnv_TempCtrl_Itf_TcPfw for format definition
=====
*)
stRtData: ST_TcPfw_TempCtrl_RtData;

ZoneName: STRING(79);

Heater_SupplyLoad : LREAL;
Setpoint          : LREAL;
Threshold_PP      : LREAL;

```

```
Threshold_P      : LREAL;
Threshold_M      : LREAL;
Threshold_MM     : LREAL;
StandbySetpoint  : LREAL;

tempEnergy       : LREAL;
Openloop_Output  : LREAL;

PrevSameSupply   : INT;
NextSameSupply   : INT;
EvtIdx_Autotune  : INT;
EvtIdx_Hardware  : INT;
TempTermInit     : INT;
SelectSetpoint   : INT:=0; (* AST: supporting selectable setpoints *)

SelSetpoint      : BOOL;
Cmd_TuneHeaterLoad : BOOL;
Sema_Update      : BOOL;
Sema_Used        : BOOL;
LoopTest         : BOOL;
LoopTest_Inv     : BOOL;
Enable           : BOOL;
Alarm_LowLow     : BOOL;
Alarm_Low        : BOOL;
Alarm_High       : BOOL;
Alarm_HighHigh   : BOOL;
Alarm_AbsoluteLow : BOOL;
Alarm_AbsoluteHigh : BOOL;
Alarm_NoResponse : BOOL;
Force_Heating    : BOOL:=FALSE;
Force_Cooling    : BOOL:=FALSE;

Fault            : BOOL:=FALSE; (* starting with V1.0.9 *)
END_STRUCT
END_TYPE
```

Parameter

Name	Typ	Beschreibung
stRtData	ST_TcPfw_TempCtrl_RtData	Die Laufzeitdaten der Zone.
ZoneName	STRING	Reserviert, nicht garantiert.
Heater_SupplyLoad	LREAL	Die Summen-Heizleistung aller Zonen derselben Versorgungsgruppe.
Setpoint	LREAL	Die Solltemperatur der Zone. (Kopie des Parameters aus ST_TcPfw_TempPparamFromHmi_Itf).
Threshold_PP	LREAL	Die äußere positive Toleranzgrenze der Zone. (Kopie des Parameters aus ST_TcPfw_TempPparamFromHmi_Itf).
Threshold_P	LREAL	Die innere positive Toleranzgrenze der Zone. (Kopie des Parameters aus ST_TcPfw_TempPparamFromHmi_Itf).
Threshold_M	LREAL	Die innere negative Toleranzgrenze der Zone. (Kopie des Parameters aus ST_TcPfw_TempPparamFromHmi_Itf).
Threshold_MM	LREAL	Die äußere negative Toleranzgrenze der Zone. (Kopie des Parameters aus ST_TcPfw_TempPparamFromHmi_Itf).
StandbySetpoint	LREAL	Die Absenk-Solltemperatur der Zone. (Kopie des Parameters aus ST_TcPfw_TempPparamFromHmi_Itf).
tempEnergy	LREAL	Zwischenvariable für die Energieberechnung.
Openloop_Output	LREAL	Kopie des vorgegebenen Stellgrades aus den Produktparametern.
PrevSameSupply	INT	Reserviert für interne Nutzung.
NextSameSupply	INT	Reserviert für interne Nutzung.
EvtIdx_Autotune	INT	Reserviert für Blow Molding Framework.
EvtIdx_Hardware	INT	Reserviert für Blow Molding Framework.
TempTermInit	INT	Reserviert für den FB_TempCtrlCallback_TcPfw() Baustein.
SelectSetpoint	INT	Auswahl des Sollwerts aus dem Array Setpoints. Liegt der Wert außerhalb des Wertebereichs, ist Setpoint aus den Produktparametern aktiv.
SelSetpoint	BOOL	Der Umschaltung des wirksamen Sollwerts der Zone. Ein TRUE wählt den StandbySetpoint aus, ein FALSE den Setpoint.
Cmd_TuneHeaterLoad	BOOL	Reserviert für automatische Messung der Heizleistung.
Sema_Update	BOOL	reserviert
Sema_Used	BOOL	reserviert
LoopTest	BOOL	Die Strommessung für diese Zone ist aktiv. Die Heizung wird unabhängig von der Regelung für eine kurze Zeit eingeschaltet.
LoopTest_Inv	BOOL	Die Strommessung für eine andere Zone derselben Versorgungsgruppe ist aktiv. Die Heizung wird unabhängig von der Regelung für eine kurze Zeit ausgeschaltet.
Enable	BOOL	Ein TRUE hier signalisiert, dass der Regler für die Zone freigegeben ist.
Alarm_LowLow	BOOL	Die Zone ist ST_TcPfw_TempMparamFromHmi_Itf.InUse und die Ist-Temperatur der Zone liegt um mehr als die äußere negative Toleranzgrenze unter dem wirksamen Sollwert.

Name	Typ	Beschreibung
Alarm_Low	BOOL	Die Zone ist ST_TcPfw_TempMparamFromHmi_Itf.InUse und die Ist-Temperatur der Zone liegt um mehr als die innere negative Toleranzgrenze unter dem wirksamen Sollwert.
Alarm_High	BOOL	Die Zone ist ST_TcPfw_TempMparamFromHmi_Itf.InUse und die Ist-Temperatur der Zone liegt um mehr als die innere positive Toleranzgrenze über dem wirksamen Sollwert.
Alarm_HighHigh	BOOL	Die Zone ist ST_TcPfw_TempMparamFromHmi_Itf.InUse und die Isttemperatur der Zone liegt um mehr als die äußere positive Toleranzgrenze über dem wirksamen Sollwert.
Alarm_AbsoluteLow	BOOL	Die Zone ist ST_TcPfw_TempMparamFromHmi_Itf.InUse und die Isttemperatur der Zone liegt unter ST_TcPfw_TempMparamFromHmi_Itf.AbsoluteLow.
Alarm_AbsoluteHigh	BOOL	Die Zone ist ST_TcPfw_TempMparamFromHmi_Itf.InUse und die Isttemperatur der Zone liegt über ST_TcPfw_TempMparamFromHmi_Itf.AbsoluteHigh. Ist dieses Flag gesetzt, so wird in dieser Zone keine Heizleistung generiert.
Alarm_NoResponse	BOOL	Die Ist-Temperatur der Zone hat innerhalb angemessener Zeit nicht auf die Heizleistung reagiert.
Force_Heating	BOOL	Ein TRUE erzeugt für 100 Zyklen eine Heizleistungsausgabe mit 100% Stellgrad.
Force_Cooling	BOOL	Ein TRUE erzeugt für 100 Zyklen eine Kühlausgabe mit 100% Stellgrad.
Fault	BOOL	Fehler.

3.6.7 ST_TcPfw_PowerMeasurement_Cfg

Eine solche Struktur enthält die Konfigurationsdaten für die Leistungsmessung. Diese Daten werden nicht gespeichert.

Syntax

```

TYPE ST_TcPfw_xL3403_Input:
(* location PfwLib_TempControl.PRO *)
(* last modification: 08.09.2010 *)
STRUCT
  CycleTime      : LREAL:=0.025;
  I_ratio        : LREAL:=1.0;

  ePowerTerminal: E_TcPfw_TctrlPowerTerminal;
  TerminalIdx   : INT;
  TerminalSubIdx: INT;
END_TYPE

```


Parameter

Name	Typ	Beschreibung
CycleTime	LREAL	Zykluszeit der Task.
I_ratio	LREAL	Übersetzungsverhältnis der Stromwandler.
ePowerTerminal	E_TcPfw_TctrlPowerTerminal	Art der Leistungsmessklemme, die angeschlossen ist.
TerminalIdx	INT	Nummer der Supplygroup, welche ihr zugeordnet ist.
TerminalSubIdx	INT	Bei Klemmen, die mehr als drei Strommesskanäle enthalten (EL3446), muss angegeben werden, ob die Gruppe an den ersten drei Strommesseingängen (-> 1) oder an den hinteren drei Strommesseingängen (-> 2) angeschlossen ist.

3.7 Knowledge Base

3.7.1 Inbetriebnahme

3.7.1.1 Global

Inbetriebnahme

Die Bibliothek PfwLib_TempControl.lib ist in sechs wesentliche Strukturen aufgeteilt.

- ST_TcPfw_TempCtrlInput
In dieser Struktur sind alle verknüpfbaren Eingangsvariablen enthalten.
- ST_TcPfw_TempCtrlOutput
In dieser Struktur sind alle verknüpfbaren Ausgangsvariablen enthalten.
- ST_TcPfw_TempMparamFromHmi_Itf
In dieser Struktur sind alle Maschinenparameter hinterlegt.
- ST_TcPfw_TempPparamFromHmi_Itf
In dieser Struktur sind alle Produktparameter hinterlegt.
- ST_TcPfw_SupplyParam
In dieser Struktur werden Einstellungen zur PWM-Ausgabe getätigt.
- ST_TcPfw_TempToHmi_Itf
Diese Struktur gibt die aktuellen Zustände des Temperaturreglers wieder.

Für eine erfolgreiche Inbetriebnahme des Temperaturreglers muss die Bibliothek PfwLib_TempControl.lib in das Projekt eingebunden werden. Anschließend sind folgende Schritte durchzuführen:

Anlegen einer Instanz des Funktionsbausteins FB_TempCtrlMainBody_TcPfw und Deklaration der Variablen

Variable	Kurzbeschreibung	Beispielwert
ConfigEnable	Signalisiert die Gültigkeit der Parameter.	TRUE
Callback_Enable	Gibt die Hintergrund-Kommunikation mit den E/A-Klemmen frei.	TRUE
Looptest_Enable	Aktiviert die Leistungsüberwachung der Heizbänder.	FALSE
tCycle	Zykluszeit (Es ist eine Zykluszeit ungleich der Netzfrequenz zu empfehlen).	0.025s
Simu_Enable	Starten einer Simulation.	FALSE
Simu_DisCharge	Simulation auf den Ausgangswert zurücksetzen.	FALSE
Simu_DisCharge	Simulation auf den Ausgangswert zurücksetzen.	FALSE



Die Temperaturbibliothek beinhaltet zwei Simulationen:

- Eine Pt2-Strecke, die aktiv ist, wenn der Parameter `aaaPfwTempMparamFromHmi[..].TempSensTerm` eine 0 enthält (empfohlene Simulation für den Anwender).
- Wird der Parameter **Simu_Enable** gesetzt, wird ein physikalisches Modell eines Extruderzylinders aktiv. Diese Simulation ist hauptsächlich für Entwicklungszwecke vorgesehen und sollte vom Anwender nicht aktiviert werden.

Konstantendefinition

Variable	Kurzbeschreibung	Typ	Beispielwert
cnPfwTempCtrlFirst	Nummer der ersten Zone (in der Regel 1).	INT	1
cnPfwTempCtrlLast	Nummer der letzten Zone.	INT	32
cnPfwAppSupplyFirst	Nummer der ersten Versorgungsgruppe.	INT	1
cnPfwAppSupplyLast	Nummer der letzten Versorgungsgruppe.	INT	3
cnPfwTempTrendFirst	cnPfwTempTrendFirst und cnPfwTempTrendLast geben die Anzahl an Sampels zur Trenderfassung an.	INT	1
cnPfwTempTrendLast	cnPfwTempTrendFirst und cnPfwTempTrendLast geben die Anzahl an Sampels zur Trenderfassung an.	INT	100
cnPfwTempTrend_sPH	Periodendauer der Abtastzeit in ms.	INT	
cnPfwBoolOutSwapFirst	Anfangsindex des Arrays out_SwappedDigitalOut.	INT	1
cnPfwBoolOutSwapLast	Endindex des Arrays out_SwappedDigitalOut.	INT	2 (Wenn nicht in Benutzung)
cnPfwBoolInSwapFirst	Anfangsindex des Arrays in_SwappedDigitalIn.	INT	1
cnPfwBoolInSwapLast	Endindex des Arrays in_SwappedDigitalIn.	INT	2 (Wenn nicht in Benutzung)
cnPfwScopeSampleFirst	cnPfwScopeSampleFirst und cnPfwScopeSampleLast geben die Anzahl an Sampels zur Scopeerfassung an.	INT	
cnPfwScopeSampleLast	cnPfwScopeSampleFirst und cnPfwScopeSampleLast geben die Anzahl an Sampels zur Scopeerfassung an.	INT	
bPfw_UseTempControl	Die Rückführung ist aktiv und damit das Auslesen von Istwerten möglich.	BOOL	TRUE
bPfw_UseRtScope	Die Erfassung von Scopedaten ist aktiv.	BOOL	FALSE
bPfw_UseTempTrend	Die Erfassung von Trenddaten ist aktiv.	BOOL	FALSE
cnst_pfw_selRelAlarm	Über ein TRUE wird der ausgewählte Sollwert für die relativen Alarmer verwendet. Bei einem FALSE wird der intern verrampte Sollwert verwendet.	BOOL	FALSE
cnPfwLoopTestTimer	Zeit in ms in der die Leistungsmessung misst.	LREAL	0.4 s
cnPfwLoopTestCycle	Zeit in ms die für den Regler zu Verfügung steht, bevor die nächste Zone leistungstechnisch überprüft wird.	LREAL	9.999 s
cnst_PfwParamFilePath_CE	Ordnerpfad auf einem CE-Betriebssystem.	STRING	
cnst_PfwParamFilePath_XP	Ordnerpfad auf einem XP/ Win7 Betriebssystem.	STRING	
cnst_PfwSubDir_Logging	Unterordner für Logdateien.	STRING	
cnst_PfwSubDir_Product	Unterordner für Produktdateien.	STRING	
cnst_PfwSubDir_Machine	Unterordner für Maschinendateien.	STRING	
cnst_PfwSubDir_Supply	Unterordner für Supply-Dateien.	STRING	
bPfw_UseEnergyRecording	Aktivierung einer theoretischen Energieerfassung.	BOOL	FALSE
bPfwRunOn_WinCE	Ein TRUE aktiviert den Ordnerpfad für ein CE-Betriebssystem.	BOOL	TRUE



Bitte beachten Sie, dass die Konstanten der Zeitschaltuhr sowie der Alarmvisualisierung mit zu berücksichtigen sind.



Eine Heizzone ist in dieser Bibliothek ein Stellglied (Solid State Relais – SSR oder Schütz), ein oder mehrere Heizbänder und ein Temperatursensor. Gilt eine verrechnete/ errechnete Temperatur als Ist-Größe, so muss dies in der Applikation erfolgen.

Beispielprojekt

Anhand des Beispielprojektes soll das Laden und Speichern von Produkt- und Maschinenparametern sowie das allgemeine Handling mit der Temperaturbibliothek näher erklärt werden. Zum Starten der Applikation ist es notwendig, dass die PfwLib_TempControl.lib eingebunden wird und der Pfad zum Speichern und Laden der Parameter angepasst wird.

3.7.1.2 Applikation

Inbetriebnahme (Parametrierung)

In der folgenden Tabelle sind alle Parameter aufgeführt, die zum Initialisieren des Temperaturreglers notwendig sind.

Nachdem Änderungen an den Parametern der Strukturen aaaPfwTempMparamFromHmi, aaaPfwTempPparamFromHmi, stPfwSupplyLineCfg vorgenommen wurden, müssen diese über die jeweilige Struktur gespeichert werden. Hierzu sind folgende Variablen in jeder Struktur vorgesehen:

- **SaveDelay:** Wird hier eine Zahl in ms eingetragen, zählt die Bibliothek von dieser Zahl nach Null. Bei Null wird die Datei gespeichert und bei -1 ist das Speichern inaktiv.
- **SaveParam:** Löst ein sofortiges Speichern aus.
- **LoadParam:** Lädt die Parameter aus dem vorher definierten Pfad.

Während des Speicherns wird immer eine Backup Datei erzeugt, so dass die Gültigkeit der Daten zu jeder Zeit sichergestellt ist.

In der Struktur aaaPfwTempMparamFromHmi[...] werden die Maschinendaten initialisiert:

Variable	Kurzbeschreibung	Beispielwert	Typ
ZoneName	Name der Zone	Z1	STRING
AbsoluteHigh	Temperaturobergrenze für Alarm und Abschaltung	250 °C	LREAL
AbsoluteLow	Temperaturuntergrenze für Alarm und Abschaltung	40 °C	LREAL
ExtruderComp	TRUE: aktiv; FALSE: inaktiv	FALSE	BOOL
KpCool	P-Verstärkung vom PID-Regler (Heizen)	... %/°C	LREAL
KpHeat	P-Verstärkung vom PID-Regler (Kühlen)	.. %/°C	LREAL
TdCool	Der D-Anteil (Dämpfungszeit) des Temperaturreglers (Kühlen)	... s	LREAL
TdHeat	Der D-Anteil (Dämpfungszeit) des Temperaturreglers (Heizen)	... s	LREAL
TnCool	Nachstellzeit vom PID-Regler (Heizen)	... s	LREAL
TnHeat	Nachstellzeit vom PID-Regler (Kühlen)	... s	LREAL
TvCool	Der D-Anteil (Vorhaltezeit) des Temperaturreglers (Kühlen)	... s	LREAL
TvHeat	Der D-Anteil (Vorhaltezeit) des Temperaturreglers (Heizen)	... s	LREAL
Overshoot	Überschwingen nach der Selbstoptimierung	... °C	LREAL
Tracking_Td	Zeitkonstante für die Regelung bei großen Sollwertänderungen	... s	LREAL
Ramping_Rate	Rampensteigung in °C/min	... °C/min	LREAL
Ramping_Tolerance	Bis zu welchem Sollwertsprung soll eine Rampe verwendet werden?	5 °C	LREAL
dTmax	max. Steigung der Strecke bei einer Sprungantwort	... °C/s	LREAL
SensorOffset	Offsettemperatur vom Temperatursensor	0 °C	LREAL
SettlingTime	streckenspezifische Zeitbasis	5 s	LREAL
SupplyLoad_Cooler	Kühlleistung	100 W	LREAL
SupplyLoad_Heater	Heizleistung	1000 W	LREAL
SupplyLoad_Tolerance	Weicht die gemessene Heizleistung von dieser Toleranz ab, wird ein Alarm ausgelöst.		LREAL
TuneEnd	Endtemperatur der Selbstoptimierung in Bezug auf Setpoint [%]	80%	LREAL
TuneKp	Einstellfaktor für den Kp-Wert	01. Feb	LREAL
TuneTd	Einstellfaktor für den Td-Wert	0.2	LREAL
TuneTn	Einstellfaktor für den Tn-Wert	2	LREAL
TuneTv	Einstellfaktor für den Tv-Wert	0.5	LREAL
TuneY	Leistungsausgabe bei der Selbstoptimierung	100%	LREAL
TuneTrackingTd	Einstellfaktor für den TrackingTd-Wert	0	LREAL
L_LoadIdle	Ableitungswiderstand zur Umgebung	2 W/°C	LREAL
ErrorHeatingFactor	Heizfaktor bei ausgeschalteter Regelung	20 %	LREAL
fPwmStdMaxOnTime	Maximale Heizungseinschaltdauer bei Auswahl des Std. Setpoints. Bei Null wird die MaxPWMOntime der SupplyLine verwendet. Wertebereich 0.0...1.0	0.0	LREAL
fPwmMaxOnTime	Maximale Heizungseinschaltdauer bei Auswahl einer der Setpoints im Array. Bei Null wird die MaxPWMOntime der SupplyLine verwendet. Wertebereich 0.0...1.0	20 %	LREAL

Variable	Kurzbeschreibung	Beispielwert	Typ
fPwmMinOnTime	Minimale Heizungseinschaltdauer bei Auswahl des Std. Setpoints. Bei Null wird die PwmMinOnTime der SupplyLine verwendet. Wertebereich 0.0...1.0	20 %	LREAL
Weighting_C	Faktor zwischen den ermittelten Heiz- und Kühlparametern		LREAL
fc_OnTime	Einschaltdauer der Zwangskühlung		LREAL
fc_OffTime	Dauer für die normale Regelung		LREAL
OutputSel_H	Heizsignalauswahl (siehe E_TcPfw_TctrlOutSelect)	1	INT
OutputSel_C	Kühlsignalauswahl (siehe E_TcPfw_TctrlOutSelect)	3	INT
TempSensTerm	Sensorklemmentyp muss von E_TcPfw_TerminalType ausgewählt werden.		
SensorType	Typ des Sensors muss von E_TcPfw_TempSensType ausgewählt werden.		
TermChannel	Sensorklemmenkanal	1	INT
ExtruderId	Nummer des Extruders	1	INT
ModuleId	Ordnet Zonen eine Temperaturgruppe zu	1	INT
Zoneld	Nummerierung der Zone einer Maschine	1	INT
SupplyId	Versorgungsgruppe, um die Leistung gleichmäßig auf die drei Phasen zu verteilen	2	INT
CJ_CompMode			
CJ_CompZone			
SensTermSwapIdx			
HeaterSwapIdx			
CoolerSwapIdx			
InUse	TRUE: Zone in Benutzung; FALSE: Zone nicht in Benutzung Die Zone ist nur aktiv, wenn ST_TcPfw_TempToHmi_Itf[x].InUse zurückmeldet. Vorraussetzung ist: ModuleId<>0; Zoneld<>0; SupplyId<>0	TRUE	
UseCooling	Zone besitzt eine Kühlung.	TRUE	BOOL
ExtruderCompEna	Aktivierung der Extruderkompensation	FALSE	BOOL
TuneCooling	Selbstoptimierung für das Kühlen	FALSE	BOOL
Tune_IdleLoad	Selbstoptimierung für die IdleLoad Ermittlung. Dieses würde im Anschluss an die Heizleistungsermittlung erfolgen.	TRUE	BOOL
Autotune	Selbstoptimierung für das Heizen	FALSE	BOOL
Enable	Zone ist aktiv	TRUE	BOOL
Update	Die Bedienoberfläche signalisiert, dass Werte in dieser Struktur verändert wurden.	FALSE	BOOL
EnaExtruderBlock	reserviert	FALSE	BOOL
NoFanWhileTrackDown	Beim Herunterkühlen keinen Lüfter verwenden.	FALSE	BOOL
Ena_TuneIdleLoad	Berechnung des Ableitungswiderstands aktivieren.	FALSE	BOOL
LooptestUpdate	Aktivieren der Leistungsmessung in dieser Zone.	FALSE	BOOL

Variable	Kurzbeschreibung	Beispielwert	Typ
EnableErrorHeating	Aktivieren, dass das Heizband mit einer vordefinierten Leistung (ErrorHeatingFactor) versorgt wird.	FALSE	BOOL
NoFanWhileTrackDown	reserviert	FALSE	BOOL
Ena_TuneldleLoad	reserviert	FALSE	BOOL
OpenloopHeating	Gestellte Ausgabe mit einer vordefinierten Ausgabe	FALSE	BOOL
fc_Enable	Aktivierung der Zwangskühlung	FALSE	BOOL
Hibernatel_Cool	Einfrieren des I-Anteils zum Heizen	FALSE	BOOL
hibernatel_Heat	Einfrieren des I-Anteils zum Kühlen	FALSE	BOOL
bSavingParams	Signal, dass gerade Maschinendaten gespeichert werden.		BOOL
bLoadParams	Lädt die Maschinendaten	FALSE	BOOL
ReadBack	reserviert	FALSE	BOOL

In der Struktur aaaPfwTempPparamFromHmi[...] werden die Produktdaten initialisiert:

Variable	Kurzbeschreibung	Beispielwert	Typ	Anmerkung
Setpoint	Sollwert bei Betrieb	180 °C	LREAL	muss parametrieren werden
StandbySetpoint	Sollwert im Standby	60 °C	LREAL	muss parametrieren werden
Threshold_M	Die innere negative Toleranzgrenze, bezogen auf den Sollwert (Schwelle für den Zweipunktregler)	-5 °C	LREAL	muss parametrieren werden
Threshold_MM	Die äußere negative Toleranzgrenze, bezogen auf den Sollwert	-10 °C	LREAL	muss parametrieren werden
Threshold_P	Die innere positive Toleranzgrenze, bezogen auf den Sollwert (Schwelle für den Zweipunktregler)	+5 °C	LREAL	muss parametrieren werden
Threshold_PP	Die äußere positive Toleranzgrenze, bezogen auf den Sollwert	+10 °C	LREAL	muss parametrieren werden
Update	Die Bedienoberfläche signalisiert, dass Werte in dieser Struktur verändert wurden, bezogen auf den Sollwert.		BOOL	optional
Variable	Kurzbeschreibung	Beispielwert	Typ	Anmerkung

In der Struktur stPfwSupplyLineCfg[...] werden die Einstellungen für die PWM initialisiert und der Zusammenhang zwischen einer Zone und dem Versorgungsnetz hergestellt.

Diese Information ist lediglich für die Heizleistungsüberwachung und das Zooning wichtig. Wird beides nicht verwendet, reicht es, das Array mit einem Element zu initialisieren.

Kunststoffmaschinen sind zum Teil so aufgebaut, dass sie mehrere Temperierungsgruppen enthalten (Extruder1..n, Heizkanäle, usw.). Diese werden zum Teil getrennt versorgt. Diese Versorgungseinheiten werden im Framework Supply-Groups genannt. Jede Supply-Group besteht aus vier Supply-Lines (Phase 1..3 und mehrphasige Elemente) . An eine Supply-Line können mehrere Heizzonen angeschlossen werden. Die erste Supply-Line einer Gruppe stellt die Versorgung für die Heizzonen dar, die zwischen L1 und dem Neutralleiter angeklemt sind. Die Zweite für L2 und N und die Dritte für L3 und N. Zur vierten Supply-Line gehören alle Heizzonen, die zwischen zwei Außenleitern in Stern oder Dreieck geschaltet werden.

Supply-Group	1				2				...	n			
Supply-Line	1	2	3	4	5	6	7	8	...	(n*4)-3	(n*4)-2	(n*4)-1	(n*4)-0

Variable	Kurzbeschreibung	Beispielwert		Parametrierungs-grad
fPwmCycleTime	PWM Zykluszeit	0.1 s	LREAL	ja
fPwmMaxOnTime	PWM Höchsteinschaltdauer (bezogen auf die Zykluszeit)	0.9 für 90%	LREAL	ja
fPwmMinOnTime	PWM Mindesteinschaltdauer (bezogen auf die Zykluszeit)	0.1 für 10%	LREAL	ja
fPwmMaxOnC	PWM Höchsteinschaltdauer (bezogen auf die Zykluszeit) für das Kühlen	0.1 für 10%	LREAL	ja
fPwmMaxRampLoad	reserviert		LREAL	optional
fActSupplyLoad	Wenn eine Leistungsmessung durchgeführt wird, kann die aktuelle Leistung hier abgelesen werden.	... W	LREAL	optional
fActSupplyCurrent	Wenn eine Leistungsmessung durchgeführt wird, kann der aktuell gemessene Strom hier abgelesen werden.	... W	LREAL	optional
fSupplyLoad	Die vorausberechnete Gesamtleistung der Versorgungsgruppe.	... W	LREAL	optional
fSupplyMatch	Das Verhältnis von fActSupplyLoad zu fSupplyLoad		LREAL	optional
nPwmFactorC	Bei einer PWM-Kühlung wird nPwmFactorC mit "fPwmCycleTime" multipliziert (Grund: Kühlungen werden oft über Schütze gesteuert).	1	INT	muss parametrieren werden

In der Struktur aaaPfwTempToHmi[...] werden folgende Anzeigewerte zum HMI zurückgeschrieben:

Variable	Kurzbeschreibung	
ActualTemp	Aktuelle Temperatur der Zone	LREAL
SupplyMatch	Aktuell gemessenes Leistungsverhältnis (Istleistung/Solleistung)	LREAL
ActCurrent	Aktuell gemessener Strom von der Zone	LREAL
FileErrId	Fehlernummer, wenn das Laden oder Speichern der Maschinen oder Produktdaten fehlschlägt.	DINT
ErrorId	Fehlernummer	WORD
ModuleId	Modulnummer	INT
PowerLevel	Heizleistungsausgabe des Reglers	LREAL
ZonId	Zonenummer	INT
Cooling	Die Zone kühlt gerade.	BOOL
Enable	Die Zone regelt aktive auf ihren Sollwert.	BOOL
Error	Die Zone heizt gerade.	BOOL
Heating	Die Zone befindet sich im Fehler.	BOOL
FileErr	Die Zone konnte die Maschinen- oder Produktparameter nicht erfolgreich laden oder speichern.	BOOL
InUse	Zone wird verwendet. In den Maschinenparametern muss: ModuleId<>0; ZonId<>0; SupplyId<>0 und InUse:=TRUE	BOOL
OnStandBy	Standby-Temperatur ist als Sollwert aktiv.	BOOL
TuningActive	Selbstoptimierung aktiv	BOOL
TuningDone	Selbstoptimierung erfolgreich	BOOL
IdleLoadActive	Berechnung des Ableitungswiderstands aktiv	BOOL
IdleLoadDone	Berechnung des Ableitungswiderstands erfolgreich	BOOL
LooptestActive	Aktivierung der Leistungskontrolle der einzelnen Heizbänder	BOOL
ExtruderCompActive	Rückmeldung, dass die Extruderkompensation eingelernt wird.	BOOL
ExtruderCompDone	Rückmeldung, dass die Extruderkompensation erfolgreich eingelernt wurde.	BOOL

3.7.1.3 Mapping

System-Manager

Für jeden Temperaturfühler müssen folgende Variablen verknüpft werden:

- EL-Klemme (EL33xx):

Klemmen-variable	PLC-Variable	Bemerkung
Value	in_PfwTempCtrlInput[...].EL_Sns_Data	Prozesswert
Underrange	in_PfwTempCtrlInput[...].EL_SnsUnderrun	Der untere Temperaturbereich des Messfühlers wurde unterschritten.
Overrange	in_PfwTempCtrlInput[...].EL_SnsOverrun	Der obere Temperaturbereich des Messfühlers wurde überschritten.
Error	in_PfwTempCtrlInput[...].EL_SnsError	Der Kanal einer Klemme liefert einen Fehler.
WcState	in_PfwTempCtrlInput[...].EL_SnsWcState	Bei dem Wert 0 sind die Daten der Klemme gültig; Bei dem Wert 1 sind die Daten ungültig. Die Klemmenvariable muss auf jeden von dieser Klemme benutzten PLC-Kanal verknüpft werden (Mehrfachverknüpfung).
AdsAddr	in_PfwTempCtrlInput[...].EL_AdsAddr	Adresse zur Kommunikation mit der Klemme. Die Klemmenvariable muss auf jeden von dieser Klemme benutzten PLC-Kanal verknüpft werden (Mehrfachverknüpfung).
State	in_PfwTempCtrlInput[...].EL_SnsState	Allgemeiner Klemmenstatus Die Klemmenvariable muss auf jeden von dieser Klemme benutzten PLC-Kanal verknüpft werden (Mehrfachverknüpfung).

- KL-Klemme (KL33xx):

Klemmen-variable	PLC-Variable	Beschreibung
Data In	in_PfwTempCtrlInput[...].KL_SnsData	Dateneingang
State	in_PfwTempCtrlInput[...].KL_SnsState	Status der Klemme
Data out	out_PfwTempCtrlOutput[...].KL_SnsData	Datenausgabe
Ctrl	out_PfwTempCtrlOutput[...].KL_SnsCtrl	Zur Registerkommunikation

- Über aaaPfwTempMparamFromHmi[...].OutputSel_C wird in der PLC ein Kühlsignal ausgewählt, welches durch out_PfwTempCtrlOutput[...].SelOutNeg mit dem System-Manager verknüpft werden muss.
- Über aaaPfwTempMparamFromHmi[...].OutputSel_H wird in der PLC ein Heizsignal ausgewählt, welches durch out_PfwTempCtrlOutput[...].SelOutPos mit dem System-Manager verknüpft werden muss.

3.7.1.4 Selbstoptimierung

Nachdem die Bibliothek eingebunden und alle Parameter vordefiniert sind, sollte eine Selbstoptimierung durchgeführt werden. Dazu sind folgende Parameter relevant:

- **TuneCooling:** Hat die Zone eine Kühlung (UseCooling muss TRUE sein und die Ausgabeart muss festgelegt werden OutputSel_H), so können für die Kühlung separate Parameter ermittelt werden.
- **Tune_IdleLoad:** Ein TRUE löst während der Selbstoptimierung die Ermittlung des Vorsteueranteils aus.
- **TuneY:** Gibt die Höhe der Heizleistungsausgabe während der Selbstoptimierung an.
- **TuneEnd:** Nach TuneEnd/100* Sollwert wird die Stellwertausgabe abgeschaltet und das Überschwingen ermittelt. Es soll garantiert werden, dass nach dem Überschwingen die Isttemperatur unterhalb der Solltemperatur ist.

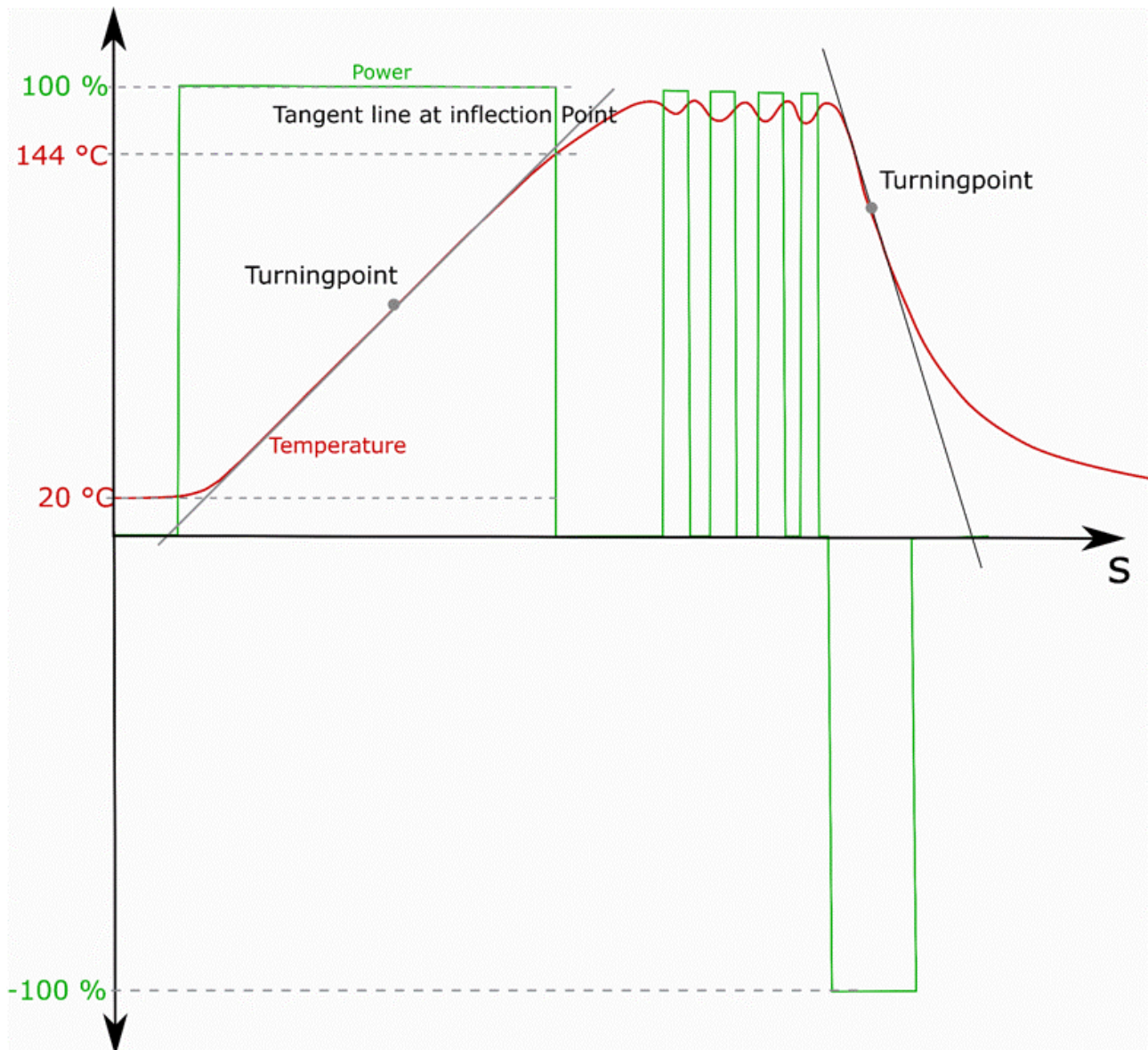
Über das Signal `ST_TcPfw_TempMparamFromHmi_Itf[...].Autotune` wird die Selbstoptimierung gestartet. Anschließend wird `ST_TcPfw_TempToHmi_Itf[...].TuningActive` TRUE. Falls dies nicht der Fall ist, wurde mit Fehler abgebrochen. Nachdem das Autotune erfolgreich war, wird dieses in `ST_TcPfw_TempToHmi_Itf[...].TuningDone` über ein TRUE angezeigt. Die Selbstoptimierung kann jederzeit abgebrochen werden, indem man `ST_TcPfw_TempMparamFromHmi_Itf[...].Autotune` auf FALSE setzt.

Voraussetzung für ein erfolgreiche Selbstoptimierung ist, dass der Regler aktiv ist (`ST_TcPfw_TempMparamFromHmi_Itf[...].Enable`) und auf StandBy-Temperatur stabil ausgegelt hat.

Es ist zu empfehlen, dass die Ist-Temperatur in der Nähe der Umgebungstemperatur liegt. Die Differenz zwischen aktueller Temperatur und Arbeitspunkt-Temperatur sollte mindestens 40.0°C betragen.

Nach dem Aktivieren des Autotunes wird für eine gewisse Zeit die Temperatur beobachtet, um zu überprüfen, ob diese stabil ist. Anschließend wird ein Stellwert in Höhe von `TuneY` ausgegeben. Schneidet die Ist-Temperatur

`ST_TcPfw_TempMparamFromHmi_Itf[...].TuneEnd*ST_TcPfw_TempPparamFromHmi_Itf[...].Setpoint`, wird die Sollwertausgabe abgeschaltet und das Überschwingen beobachtet.

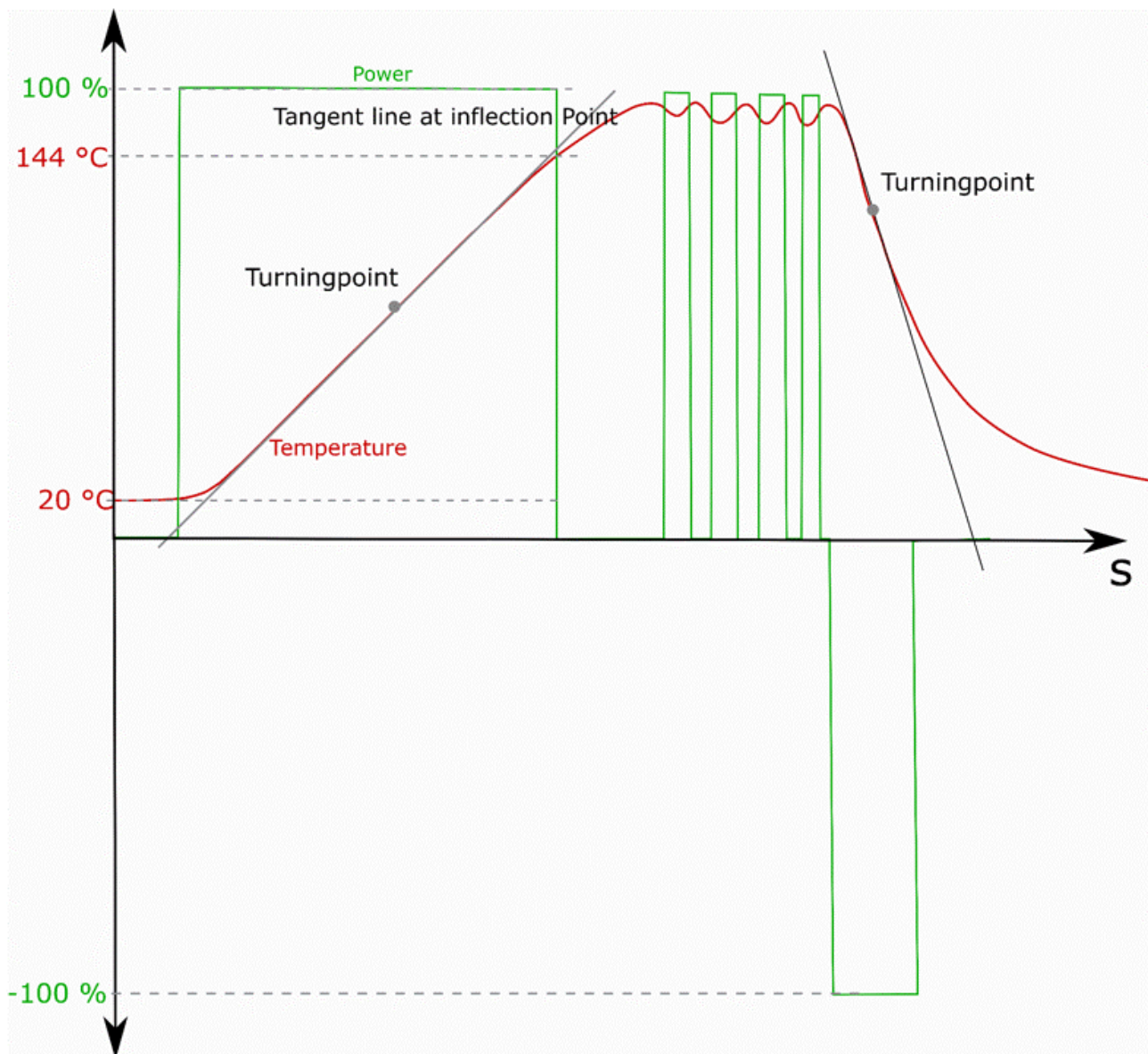


3.7.2 Alarmhandling

Übersicht über die Alarmer des Plastics Processing Frameworks

Im Temperaturregler werden folgende Alarmer ständig überwacht:

- Absolute Temperaturen (high und low):
 - befindet sich die Ist-Temperatur unterhalb von `aaaPfwTempMparamFromHmi[...].AbsoluteLow`, so wird das globale Flag `aaaTempAlarm_AbsoluteLow` auf TRUE gesetzt. Sobald die Ist-Temperatur die Schwelle überschreitet, wird `aaaTempAlarm_AbsoluteLow` wieder FALSE.
 - befindet sich die Ist-Temperatur oberhalb von `aaaPfwTempMparamFromHmi[...].AbsoluteHigh`, so wird das globale Flag `aaaTempAlarm_AbsoluteHigh` auf TRUE gesetzt. Sobald die Ist-Temperatur die Schwelle unterschreitet, wird `aaaTempAlarm_AbsoluteHigh` wieder FALSE. Sobald `aaaTempAlarm_AbsoluteHigh` aktiv ist, wird die entsprechende Zone abgeschaltet. Nach dem Unterschreiten der Grenze geht die Zone wieder in den geregelten Betrieb über.
- Relative Temperaturen (in zwei Bändern um den Sollwert):
 - befindet sich die Ist-Temperatur unterhalb von `aaaPfwTempPparamFromHmi[...].Threshold_MM`, so wird das globale Flag `aaaTempAlarm_LowLow` auf TRUE gesetzt. Sobald die Ist-Temperatur die Schwelle überschreitet, wird `aaaTempAlarm_LowLow` wieder FALSE.
 - befindet sich die Ist-Temperatur unterhalb von `aaaPfwTempPparamFromHmi[...].Threshold_M`, so wird das globale Flag `aaaTempAlarm_Low` auf TRUE gesetzt. Sobald die Ist-Temperatur die Schwelle überschreitet, wird `aaaTempAlarm_Low` wieder FALSE.
 - befindet sich die Ist-Temperatur oberhalb von `aaaPfwTempPparamFromHmi[...].Threshold_PP`, so wird das globale Flag `aaaTempAlarm_HighHigh` auf TRUE gesetzt. Sobald die Ist-Temperatur die Schwelle unterschreitet, wird `aaaTempAlarm_HighHigh` wieder FALSE.
 - befindet sich die Ist-Temperatur oberhalb von `aaaPfwTempPparamFromHmi[...].Threshold_P`, so wird das globale Flag `aaaTempAlarm_High` auf TRUE gesetzt. Sobald die Ist-Temperatur die Schwelle unterschreitet, wird `aaaTempAlarm_High` wieder FALSE.
- Ist die Konstante `cnst_pfw_selRelAlarm` TRUE, beziehen sich die relativen Alarme auf die eingegebene Solltemperatur, andernfalls auf die intern verrampte Solltemperatur.



- Fehlermeldungen von der Klemme:
 - Werden über `ST_TcPfw_TempToHmi_Itf[...].Error` und `ST_TcPfw_TempToHmi_Itf[...].ErrorId` weitergegeben.
 - Die zugehörige Zone wird daraufhin abgeschaltet.
 - Durch Aktivieren des Errorheating in den Maschinenparametern wird die Zone weiter auf Temperatur gehalten.
- Fehlermeldungen beim Autotune:
 - Werden über `ST_TcPfw_TempToHmi_Itf[...].Error` und `ST_TcPfw_TempToHmi_Itf[...].ErrorId` weitergegeben.
- Fehlverhalten beim Aufheizen:
 - Reagiert der Istwert einer Zone beim Hochheizen nicht auf die Sollwertänderung, so wird ein Error ausgegeben und die Zone schaltet ab.

3.7.3 Heizstromüberwachung

Im Plastics Processing Framework wird die Heizbandüberwachung „Looptest“ genannt. Diese ermöglicht folgende Fehlfunktionen einer Heizzone zu erkennen:

- Zustand der Stellglieder
 - Durchlegierte Solid State Relais (SSR)

- „Klebende“ Schütze
- Zustand der Heizbänder
 - Teil- und Gesamtausfall
 - Leistungsverluste durch Alterungserscheinungen

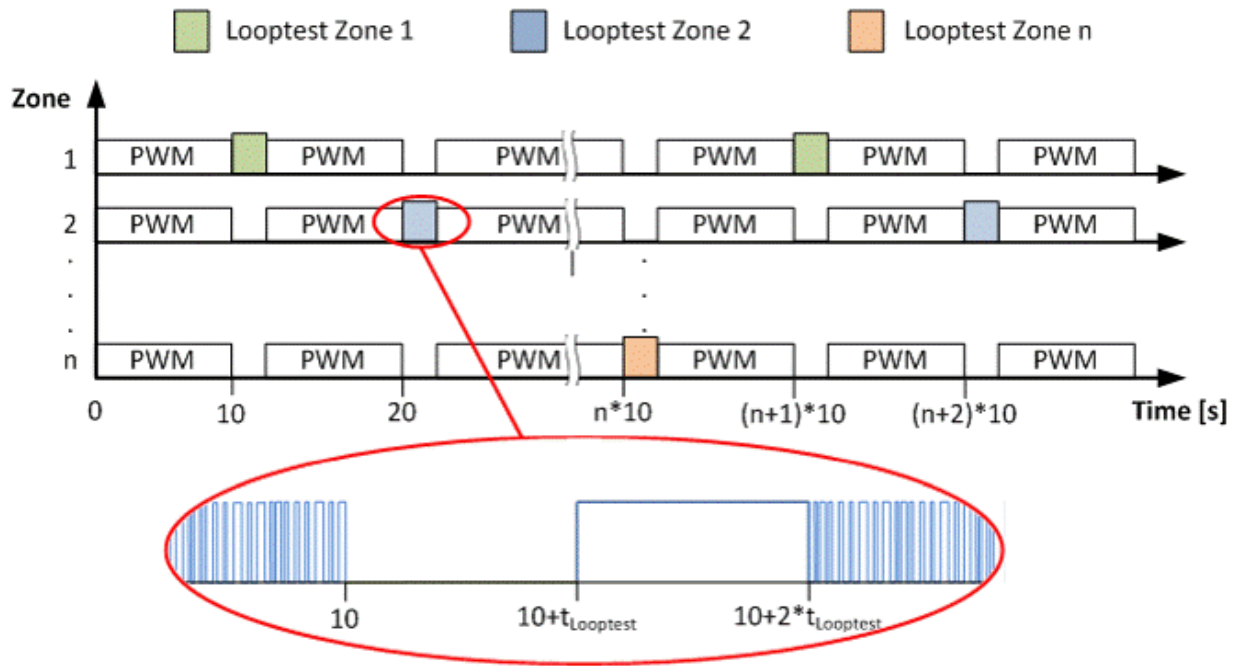
Kunststoffmaschinen sind zum Teil so aufgebaut, dass sie mehrere Temperierungsgruppen enthalten (Extruder1..n, Heizkanäle, usw.). Diese werden zum Teil getrennt versorgt. Diese Versorgungseinheiten werden im Framework Supply-Groups genannt. Jede Supply-Group besteht aus vier Supply-Lines und einer Leistungsmessklemme. An eine Supply-Line können mehrere Heizzonen angeschlossen werden. Jede Heizzone beinhaltet ein Stellglied (Solid State Relais – SSR oder Schütz), ein oder mehrere Heizbänder und einen Temperatursensor. Die erste Supply-Line einer Gruppe stellt die Versorgung für die Heizzonen dar, die zwischen L1 und dem Neutralleiter angeklemt sind. Die Zweite für L2 und N und die Dritte für L3 und N. Zur vierten Supply-Line gehören alle Heizzonen, die zwischen zwei Außenleitern in Stern oder Dreieck geschaltet werden.

Supply-Group	1				2				...	n			
Supply-Line	1	2	3	4	5	6	7	8	...	(n*4)-3	(n*4)-2	(n*4)-1	(n*4)-0

In den Parametern einer Heizzone lässt sich nun einstellen, in welcher Supply-Line sich die Zone befindet und dadurch zu welcher Supply-Group sie gehört. Des Weiteren lässt sich in den Parametern einstellen, welche Nennleistung das Heizelement besitzt. Diese Angaben sind notwendig, damit der Looptest die Zone zuordnen, testen und seine Messung interpretieren kann.

Im Plastic Processing Framework gibt es pro Supply-Group einen Looptest. Dieser durchläuft alle Heizzonen. Findet er eine Zone, die zu seiner Supply-Group gehört, startet er eine Messung. Dazu werden alle Elemente, die zu dieser Gruppe gehören, abgeschaltet und die Leistung bestimmt. Diese sollte null sein. Ist dies nicht der Fall, ist ein Stellglied defekt. Nach dem Ablauf einer Zeit $t_{Looptest}$ (Standardeinstellung 200ms) wird das Heizelement für die gleiche Zeit eingeschaltet und die Leistung bestimmt. Diese wird mit der Nennleistung der Heizzone verglichen. Weicht sie um einen eingestellten Prozentwert ab, wird ein Fehler angezeigt. Im Anschluss ist für 10 Sekunden lang die Regelung aktiv. Die selbst verursachte zusätzliche Heizleistung wird im Regler mitberücksichtigt. Danach wird die nächste Zone derselben Supply-Group gesucht.

In folgender Abbildung soll dies an einem Beispiel verdeutlicht werden. Es wird angenommen, eine Anlage hat nur eine Supply-Group, beispielsweise einen Extruder mit n Zonen. Der Looptest ist aktiv und findet Zone 1. Wie zuvor beschrieben, werden zunächst alle Zonen abgeschaltet und die Leistung gemessen. Anschließend wird Zone 1 für $t_{Looptest}$ eingeschaltet und die Leistung wieder gemessen. In einem Zyklus von 10 Sekunden wird die nächste Zone nach der gleichen Vorgehensweise getestet.



Die EL3403 berechnet die Leistung standardmäßig über ein Intervall von 10 Perioden. Die errechnete Leistung wird anschließend mittels Tief- und Hochpass gefiltert. Zusätzlich besitzen Solid State Relais, besonders jene die im Nulldurchgang schalten, eine nicht unerhebliche Schaltzeit. Dies führt dazu, dass zum Teil erst nach 500ms eine verwendbare Leistung gemessen wird. Bei einem Intervall von einer Periode konnte die Zeit auf 200ms verringert werden. Trotzdem können unter Umständen Messfehler von um die 10% auftreten. Wird eine entsprechend längere Messzeit (z.B. 400ms) vorgehalten, steigt die Genauigkeit erheblich. Ein Vorteil der xL3403 Klemmen ist, dass die Berechnung der Leistung innerhalb der Klemme stattfindet und so keine PLC-Rechenzeit benötigt wird.

Die Implementierung:

Folgende Punkte müssen für die Implementation beachtet werden:

- FB_PowerMeasurement_TcPfw: Dieser Baustein muss entsprechend der Klemmenanzahl instanziiert werden.
- ST_TcPfw_PowerMeasurement_Cfg: Muss als ARRAY entsprechend der Klemmenzahl angelegt werden.
- ST_TcPfw_PowerMeasurement_Ctrl: Muss als ARRAY entsprechend der Klemmenzahl angelegt werden.
- ST_TcPfw_xL3403_State: Muss als ARRAY entsprechend der Klemmenzahl angelegt werden. Auch dann, wenn keine xL3403 im Einsatz ist.

Die von der Bibliothek bereitgestellten Mapping-Strukturen für das Mapping zur Leistungsmessklemme müssen verwendet werden und dem FB_PowerMeasurement_TcPfw Baustein als Adresse übergeben werden.

HINWEIS

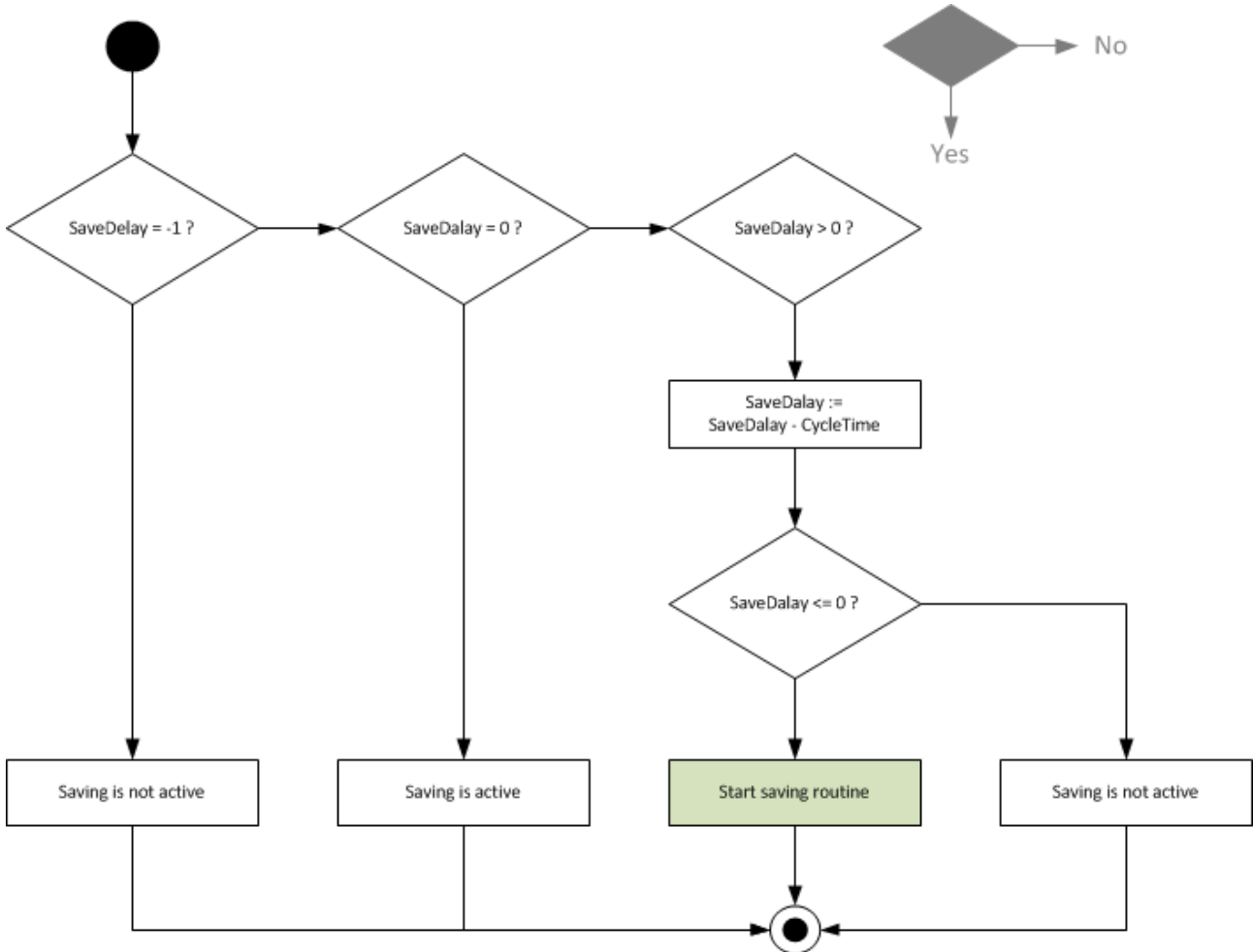
Die Pointer-Adresse und der hinterlegte Klemmentyp müssen zu jeder Zeit übereinstimmen. Andernfalls gibt es falsche Speicherzugriffe.

3.7.4 Parameter Speichern/Laden

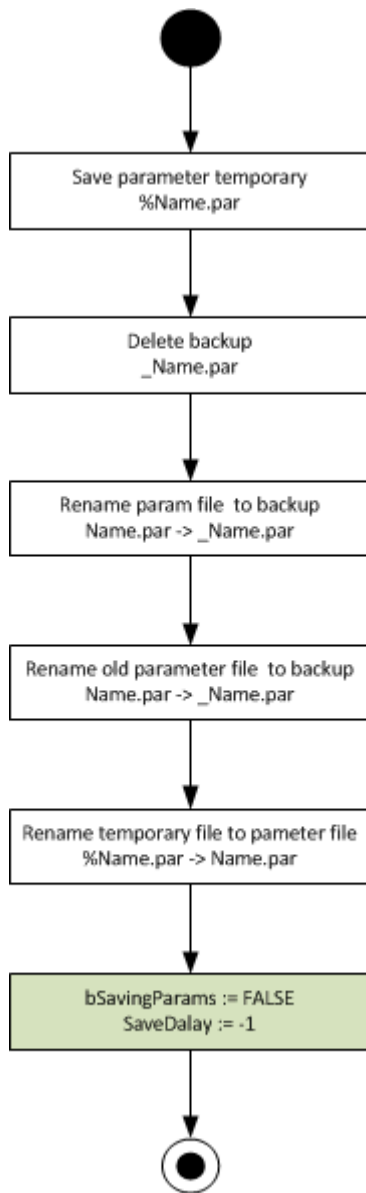
Die Strukturen ST_TcPfw_TempMparamFromHmi_Itf, ST_TcPfw_TempPparamFromHmi_Itf, ST_TcPfw_SupplyParam beinhalten Parameter, die remanent gespeichert werden müssen. Hierfür sind in jeder Struktur folgende Variablen hinterlegt:

- **SaveDelay:** Zeitverzögertes Speichern. Nach einer Parameteränderung kann hier über eine Zahl in Millisekunden ein zeitverzögertes Speichern durchgeführt werden. Ausgehend von dem eingetragenen Wert, zählt die Bibliothek automatisch diese Zahl auf Null herunter. Bei Null wird intern die Speicherroutine angestoßen und bei -1 ist die Speicherverzögerung inaktiv.
- **bLodParams:** Über ein TRUE können die Parameter der entsprechenden Struktur geladen werden
- **bSavingParams:** Rückmeldung, dass Parameter gerade gespeichert werden.

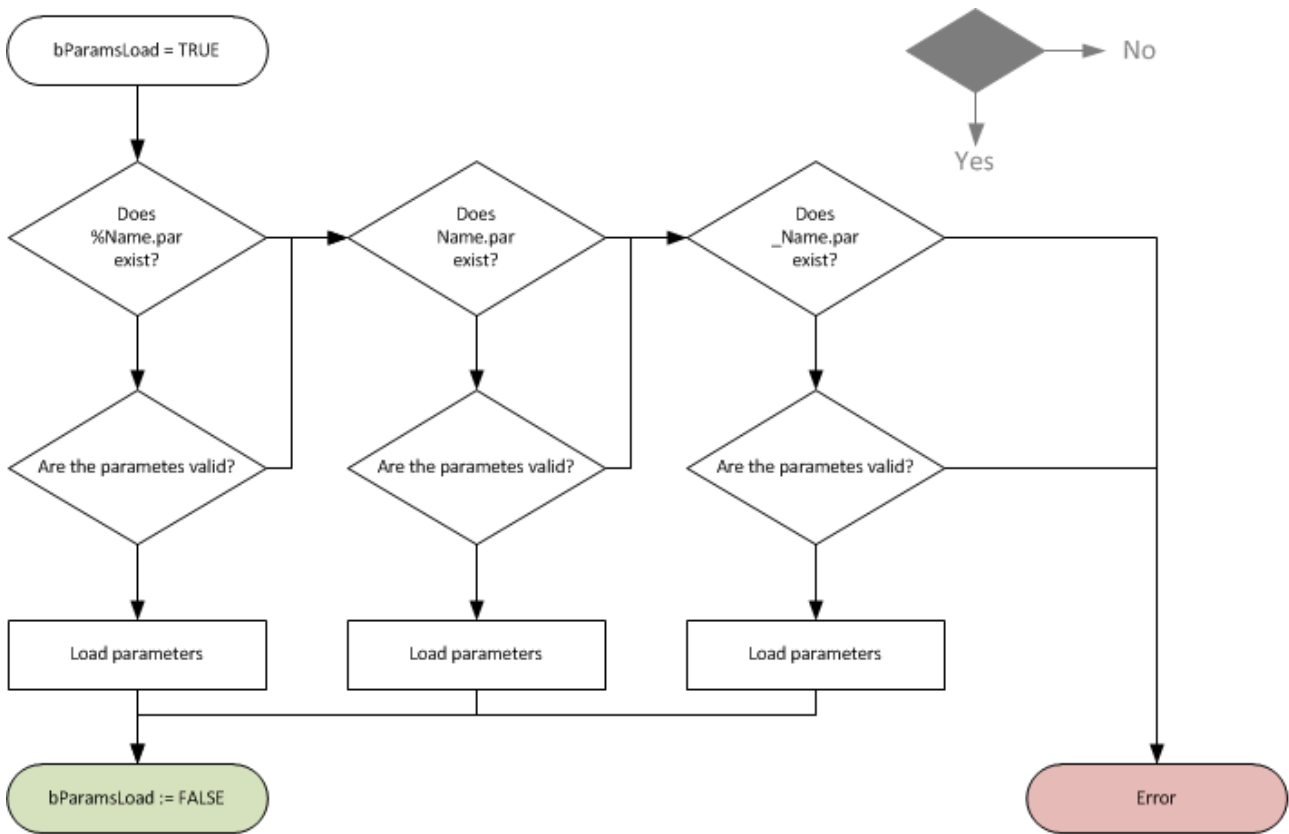
Das folgende Ablaufdiagramm soll deutlich machen, wie der Speichervorgang abläuft.



Beim Speichern, wird erst einmal eine temporäre Datei angelegt. Anschließend wird die Backupdatei gelöscht und die Originaldatei in eine Backupdatei umgewandelt. Als letzter Schritt wird die temporäre Datei in eine "Originaldatei" umgewandelt. Durch diesen Mechanismus ist gewährleistet, dass immer eine vollständig abgespeicherte Parameterdatei vorhanden ist.



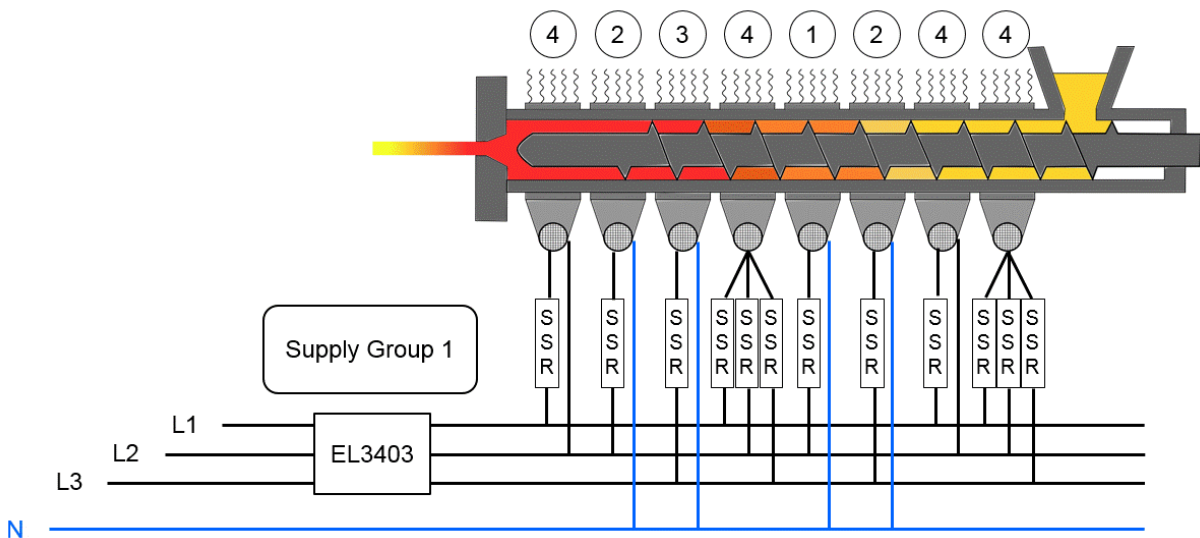
Beim Laden eines Produktes über bLoadParams wird über folgenden Ablauf geprüft, ob die abgespeicherte Datei konsistent ist.



3.7.5 Versorgungsgruppen

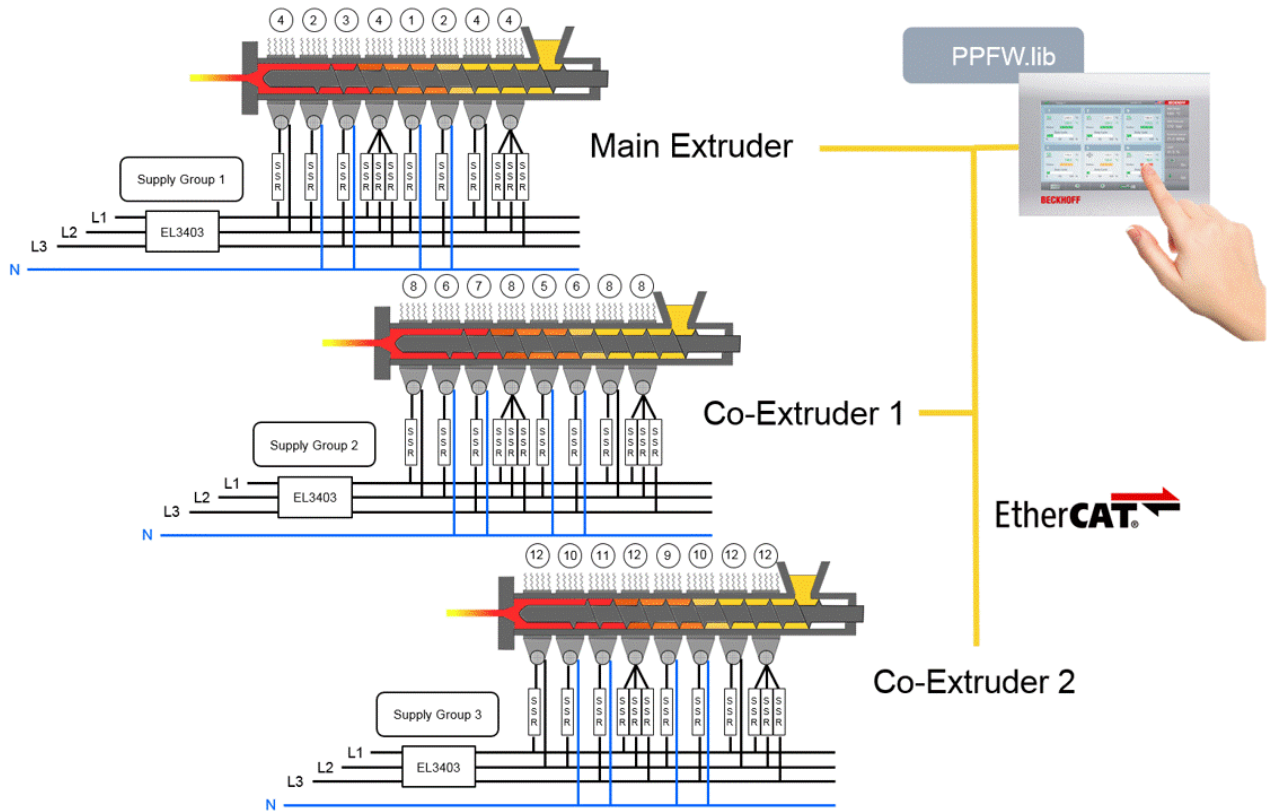
Eine Supply Line stellt eine Versorgungseinheit dar. Für die Heizleistungsüberwachung und für das verteilte Einschalten (Zoning) muss bekannt sein, welche Zone an welcher Versorgungsleitung angeschlossen ist.

In der Regel besteht die Versorgung aus ein, zwei oder mehr Phasen eines Drehstromnetzes. Heizbänder die zwischen Phase1 und Neutraleiter angeschlossen sind, müssen eine 1 in dem MParam.SupplyLine bekommen. Heizbänder die zwischen Phase 2 und Neutraleiter angeschlossen sind, bekommen in den Mparam.SupplyLine eine 2 zugewiesen. Heizbänder die zwischen Phase 3 und Neutraleiter angeschlossen sind, bekommen in den Mparam.SupplyLine eine 3 zugewiesen. Mehrphasige Heizbänder (zwei- oder dreiphasige) bekommen eine 4 zugewiesen.



Heizband	Supply Line
L1-N	1
L2-N	2
L3-N	3
Mehrphasen (L1 - L2; L2-L3; L1-L2-L3; usw)	4

Sollte es mehr als eine Versorgung für die Heizungen geben, so muss mindestens für jede Versorgung eine Leistungsmessklemme vorgesehen sein. Dies kann zum Beispiel bei mehreren Extrudern, die an einer Steuerung betrieben werden, der Fall sein.



Die Nummerierung der anschließenden Versorgungsgruppen (Supply groups) ist fortlaufend. D.h. Supply group 2 hat Supply Line 5,6,7,8 zugewiesen.

Supply-Group	1				2				...	n			
Supply-Line	1	2	3	4	5	6	7	8	...	$(n*4)-3$	$(n*4)-2$	$(n*4)-1$	$(n*4)-0$

Die Konstante `cnPfwAppSupplyLast` muss mindestens dem größten Wert der `Mparam.SupplyLine` entsprechen. Über diese Konstante wird die Größe des Arrays der `TempCtrl_ST_TcPfw_SupplyParam` definiert. Die `SupplyParams` stellen wesentliche Eigenschaften der Ausgabegenerierung dar, wie die PWM-Zykluszeit, mindest- und maximale Einschaltzeit.

3.7.6 FAQs

Name	Beschreibung
FAQ#1 [▶ 84]	Welche Hardware-Voraussetzungen sind zu erfüllen?
FAQ#2 [▶ 84]	Welche Software-Voraussetzungen sind zu erfüllen?
FAQ#3 [▶ 84]	Kann eine Zone bei Sensor-Ausfall heizen?
FAQ#4 [▶ 85]	Wie kann ich die Ausgabe von Heiz- und Kühlsignalen umleiten?
FAQ#5 [▶ 85]	Wie kann ich die Erfassung der Ist-Temperatur umleiten?
FAQ#6 [▶ 85]	Wie funktioniert das Idle-Load Tuning?
FAQ#7 [▶ 85]	Wie funktioniert die Extruder-Kompensation?
FAQ#8 [▶ 86]	Was soll ich bei Fehlfunktion tun?

FAQ#1: Hardware-Voraussetzungen

Ein IPC mit x86-Architektur und Hardware FPU. Rechenleistungsbedarf und Feldbusunterstützung sind in jedem Fall abzuklären. Beispiele:

- Hutschienen-PCs der Typen CX1010, CX1020 oder CX1030.
- Hutschienen-PCs der Typen CX50xx.
- Hutschienen-PCs der Typen CX51xx.
- Hutschienen-PCs der Typen CX20xx.
- Schaltschrank-PCs der Familien C41xx, C61xx, C62xx, C69xx.
- Panel-PCs der Familien CP62xx, CP72xx, CP67xx, CP22xx, CP32xx und CP37xx.
- Andere PC-Familien sind in der Regel ebenfalls geeignet. Dies ist im Einzelfall zu klären.

Alternativ ein IPC mit iXP-Architektur und FPU-Emulation. Hier ergeben sich bei der Zonenzahl und/oder der Zykluszeit möglicherweise Einschränkungen.

- Hutschienen-PCs der Typen CX9010 oder CX9020.
- Panel-PCs der Familien CP66xx und CP26xx.

Eine Feldbus-Architektur mit geeigneter Leistungsfähigkeit:

- Vorzugsweise EtherCAT
- Alternativ einer der folgenden Feldbusse (es sind Zykluszeiten von 10ms oder kleiner anzustreben):
 - LightBus
 - RT-Ethernet
 - Profibus mit 12MBaud

FAQ#2: Software-Voraussetzungen

Ein Microsoft Betriebssystem des Typs Windows CE 6, Windows Embedded Compact 7, Windows XP, Windows XPe oder Microsoft Windows Embedded Standard 7, 32 Bit.

Ein lauffähiges und lizenziertes TwinCAT System, freigegeben für mindestens TwinCAT PLC.

Eine lizenzierte Kopie der Plastics Temperature Control Framework Library PfwLib_TempControl.LIB, Version V1.0.1 oder höher.

Ein beigelegtes Plastics Framework AppExtension Projekt PfwLib_TempControlAppExtension.PRO dessen Version mit der Version der Plastics Temperature Control Framework Library übereinstimmt.

FAQ#3: Heizen bei Sensor-Ausfall

Beim Ausfall eines Temperatursensors kann es erforderlich sein, die Maschine weiter zu betreiben. In einem solchen Fall kann die Library dazu veranlasst werden, den Leistungsbedarf der Zone zu schätzen und das entsprechende Heizsignal auszugeben. Dabei sind eine Reihe von Vorbedingungen zu erfüllen:

- Als Signaltyp muss eTcPfwTcOut_PWM eingestellt sein.
- Die Zone muss vorher vollständig in Betrieb genommen sein. Vor allem muss das IdleLoad Tuning mit ausreichender Genauigkeit erfolgt sein.
- Diese Option sollte nicht in benachbarten Zonen aktiviert werden. Ansonsten steigt durch die unregelmäßige Ausgabe der Leistung das Risiko einer Überhitzung der Heizbänder und des Materials.

HINWEIS

Diese Option darf auf keinen Fall in Zonen aktiviert werden, die im Betrieb überschüssige Energie (Reibung, Einleitung aus Nachbarzonen) aus dem Prozess erhalten.

Um die Option zu aktivieren ist in den Maschinendaten EnableErrorHeating auf TRUE zu setzen. Durch einen Wert im Bereich 0...1 in ErrorHeatingFactor kann festgelegt werden, welcher Anteil der geschätzten Leistung ausgegeben werden soll.



Wenn die Solltemperatur der Zone geändert wird, muss ErrorHeatingFactor nicht angepasst werden. Die Schätzung wird die erforderliche Leistung automatisch anpassen.

FAQ#4: Umleiten von Heiz- und Kühl-Signalen

Falls ein digitaler Ausgang der SPS defekt ist, so kann der Bediener diesen digitalen Ausgang auf einen anderen freien Ausgang auflegen und anschließend im HMI das entsprechende Signal auf den anderen Ausgang umlenken. Um dieses Feature nutzen zu können, sind verschiedene Einstellungen in der Temperaturbibliothek vorzunehmen. Da die Inbetriebnahme der E/A-Umleitung etwas aufwändiger ist, wenden Sie sich in diesem Fall an den Support. Wenn Sie dieses Feature nicht nutzen möchten, sind die Einstellungen in der Schritt für Schritt Inbetriebnahme zu übernehmen.

FAQ#5: Umleiten der Ist-Temperatur-Erfassung

Falls ein analoger Eingang der SPS defekt ist, so kann der Bediener diesen analogen Eingang auf einen anderen freien Eingang auflegen und anschließend im HMI das entsprechende Signal auf den anderen Eingang umlenken. Um dieses Feature nutzen zu können, sind verschiedene Einstellungen in der Temperaturbibliothek vorzunehmen. Da die Inbetriebnahme der E/A-Umleitung etwas aufwändiger ist, wenden Sie sich in diesem Fall an den Support. Wenn Sie dieses Feature nicht nutzen möchten, sind die Einstellungen in der Schritt für Schritt Inbetriebnahme zu übernehmen.

FAQ#6: Das Idle-Load Tuning

Voraussetzung für die Ermittlung der Grundlast ist:

- das System muss stabil sein (Regeldifferenz von kleiner 1°C)
- der Sollwert muss größer 70°C sein
- es dürfen keine Störungen auf das System wirken (drehende Schnecke).

Bei der Grundlastermittlung wird der Istwert nicht angeregt. Es finden lediglich interne Berechnungen statt. Die Ermittlung dauert mehrere Sekunden, wobei anschließend der ermittelte Wert automatisch in die Regelung mit einfließt.

FAQ#7: Extruder-Kompensation

Voraussetzung für die Ermittlung der Extruder-Kompensation:

- das System muss stabil sein (Regeldifferenz von kleiner 1°C)
- die Schneckendrehzahl muss die Arbeitsdrehzahl erreicht haben

Nachdem die Extruder-Kompensation ermittelt wurde, ist diese nicht aktiv. Erst wenn der Parameter ExtruderCompEna aus den Maschinenparametern ein TRUE erhält, wird dieser auch wirksam. Bei einer korrekten Extruder-Kompensation verhält sich beim Ein- bzw. Ausschalten der Schnecke die Isttemperatur wesentlich ruhiger als ohne Kompensation.

FAQ#8: Verhalten bei Fehlfunktion

Machen Sie von den Maschinen- und Produktparametern Screenshots.

Zeichnen Sie mittels dem Scope das Verhalten auf.

3.7.7 Globale Variablen

Error Codes

Name	hex	dez	Beschreibung	Error-heating
dwTcPfwTempErrNoError	16#0000	0	Kein Fehler.	
dwTcPfwTempErrIllegalValue	16#0706	1798	Parameter hat einen nicht zulässigen Wert. Beispiel: TempSensTerm ist nicht zulässig. Der ausgewählte Sensor passt nicht zur ausgewählten Klemme.	x
dwTcPfwTempErrBusy	16#0708	1800	Klemme ist bereits aktiv. FB_TempCtrlCallback_TcPfw() meldet diesen Fehler, wenn es den Kommunikationsweg belegt vorfindet.	x
dwTcPfwTempTuneErr_NoSigType	16#1001	4097	Beim Autotuning wird dieser Fehler gemeldet, wenn OutputSel_H kein aktives Signal auswählt.	
dwTcPfwTempTuneErr_Parameter	16#1002	4098	Beim Autotuning wird dieser Fehler gemeldet, wenn die Tuning-Parameter nicht zulässig sind.	
dwTcPfwTempTuneErr_NoTravel	16#1003	4099	Beim Autotuning wird dieser Fehler gemeldet, wenn der Sollwertsprung <25°C ist.	
dwTcPfwTempTuneErr_NoSettling	16#1004	4100	Beim Autotuning wird dieser Fehler gemeldet, wenn die Ist-Temperatur der Zone vor dem Heizversuch nicht stabil ist (Ist-Temperatur schwangt um >2°C).	
dwTcPfwTempTuneErr_Aborted	16#1005	4101	Dieser Fehler wird gemeldet, wenn das Autotuning durch einen Eingriff des Bedieners abgebrochen wurde.	
dwTcPfwTempTuneErr_ShortOfPoints	16#1006	4102	Beim Heiz- oder Kühlversuch wurde eine zu geringe Anzahl von Messpunkten ermittelt.	
dwTcPfwTempTuneErr_NoTop	16#1007	4103	Bei der Auswertung der Messpunkte eines Heiz- oder Kühlversuchs wurde kein Wendepunkt (= Punkt maximaler Steigung) ermittelt.	
dwTcPfwTempTuneErr_NoResponse	16#1008	4104	Die Messwerte sind nicht auswertbar. Die Zone hat nicht reagiert oder die Messwerte sind stark gestört.	
dwTcPfwTempErrNotSupported	16#4107	16647	FB_TempCtrlCallback_TcPfw() meldet diesen Fehler, wenn die verwendete Klemme eine erforderliche Funktionalität nicht unterstützt. Beispiel: Die verknüpfte Klemme ist nicht vom angegebenen Typ.	
dwTcPfwTempErrSnsUnderrun	16#4450	17488	Die Klemme meldet eine Unterschreitung des Messbereichs.	x
dwTcPfwTempErrSnsOverrun	16#4451	17489	Die Klemme meldet eine Überschreitung des Messbereichs.	x
dwTcPfwTempErrSnsHardwareFailed	16#4464	17508	Die Klemme meldet eine interne Störung.	x
dwTcPfwTempErrDisconnected	16#4FF0	20464	Die Verbindung zur Klemme ist unterbrochen. Nicht alle Feldbusse und E/A-Geräte unterstützen die Überwachung der Verbindung.	x
dwTcPfwTempErr_NoResponse	16#4FF1	20465	Ist-Temperatur reagiert nicht auf Heizung.	x
dwTcPfwTempErrNotOperational	16#4FF2	20466	Die Klemme ist nicht in einem betriebsbereiten Zustand.	x

Name	hex	dez	Beschreibung	Error-heating
dwTcPfwTempErrIoSwapCollision	16#4FF3	20467	E/A-Umleitung verursacht eine Kollision. Die Variable SensTermSwapIdx in den Maschinenparametern ist vermutlich nicht richtig konfiguriert.	
dwTcPfwTempErrAdsSwapCollision	16#4FF4	20468	TermChannel-Umleitung verursacht eine ADS-Kollision. Die Variable SensTermSwapIdx in den Maschinenparametern ist vermutlich nicht richtig konfiguriert.	
dwTcPfwTempErrInheritedFault	16#4FF5	20469	Hardware-Problem in einem anderen Kanal.	
dwTcPfwTempErrOverCurrent	16#4FFE	20478	Diese Fehler wird generiert, wenn die Abweichung zwischen Ist-Heizleistung und Sollheizleistung größer als die vorgegebene Toleranz ist. Siehe Leistungsmessung	
dwTcPfwTempErrUnderCurrent	16#4FFF	20479	reserviert.	x
dwTcPfwTempErrSnsCommFailed	16#5000	20480	reserviert.	x

Errorheating: bei aktivem Errorheating wird bei den mit x gekennzeichneten Fehlern weiter Heizleistung ausgegeben und die Zone auf Temperatur gehalten.

Deklarierte globale Konstanten

Name	Beschreibung
cnv_SupplyParam_TcPfw	Aktuelle Versionskennung der ST_TcPfw_SupplyParam Struktur.
cnv_TempCtrl_Itf_TcPfw	Aktuelle Versionskennung der ST_TcPfw_TempCtrl_Itf Struktur.
cnv_TempCtrlInput_TcPfw	Aktuelle Versionskennung der ST_TcPfw_TempCtrlInput Struktur.
cnv_TempCtrlOutput_TcPfw	Aktuelle Versionskennung der ST_TcPfw_TempCtrlOutput Struktur.
cnv_TempMparamFromHmi_TcPfw	Aktuelle Versionskennung der ST_TcPfw_TempMparamFromHmi_Itf Struktur.
cnv_TempMparamFileVers	Nur für Stand Alone Betrieb: Aktuelle Versionskennung der Struktur der optionalen Parameterdatei.
cnv_TempPparamFromHmi_TcPfw	Aktuelle Versionskennung der ST_TcPfw_TempPparamFromHmi_Itf Struktur.
cnv_TempToHmi_TcPfw	Aktuelle Versionskennung der ST_TcPfw_TempToHmi_Itf Struktur.

Nicht deklarierte globale Konstanten

Name	Beschreibung	Typ	Empfehlungswert/Maximalwerte
cnPfwTempCtrlFirst	Anfangsindex, um die Größe des Arrays festzulegen und damit die Anzahl der Zonen, die geregelt werden können.	INT	1 (empfohlen)
cnPfwTempCtrlLast	Endindex, um die Größe des Arrays festzulegen und damit die Anzahl der Zonen, die geregelt werden können.	INT	512 (abhängig von der CPU)
cnPfwScopeSampleFirst	Anfangsindex, um die Größe des Arrays festzulegen und damit die Anzahl der Speicherpunkte im Scope.	INT	1 (empfohlen)
cnPfwScopeSampleLast	Endindex, um die Größe des Arrays festzulegen und damit die Anzahl der Speicherpunkte im Scope.	INT	32767 (Maximal)
cnPfwTempTrendFirst	Anfangsindex, um die Größe des Arrays festzulegen und damit die Anzahl der Speicherpunkte im Trend.	INT	1 (empfohlen)
cnPfwTempTrendLast	Endindex, um die Größe des Arrays festzulegen und damit die Anzahl der Speicherpunkte im Trend.	INT	32767 (Maximal)
cnPfwAppSupplyFirst	Anfangsindex, um die Größe des Arrays festzulegen und damit die Anzahl der Versorgungsgruppen (in der Regel 4; 1=Phase 1, 2=Phase 2, 3=Phase 3 und 4 für mehrphasige Heizbänder).	INT	1
cnPfwAppSupplyLast	Endindex, um die Größe des Arrays festzulegen und damit die Anzahl der Versorgungsgruppen (in der Regel 4; 1=Phase 1, 2=Phase 2, 3=Phase 3 und 4 für mehrphasige Heizbänder).	INT	4
cnPfwBoolOutSwapFirst	Anfangsindex des Arrays out_SwappedDigitalOut (E/A-Umlenkung).	INT	1
cnPfwBoolOutSwapLast	Endindex des Arrays out_SwappedDigitalOut (E/A-Umlenkung).	INT	2 (wenn nicht in Benutzung)
cnPfwBoolInSwapFirst	Anfangsindex des Arrays in_SwappedDigitalIn (E/A-Umlenkung).	INT	1
cnPfwBoolInSwapLast	Endindex des Arrays in_SwappedDigitalIn (E/A-Umlenkung).	INT	2 (wenn nicht in Benutzung)

Globale Variablen des Frameworks

Name	Beschreibung
aaaPfwTempMparamFromHmi	Die Maschinenparameter der Zonen.
aaaPfwTempPparamFromHmi	Die Produktparameter der Zonen.
aaaPfwTempToHmi	Die Interfaces der Zonen zum HMI.
aaaTempAlarm_AbsoluteHigh	Sammelmeldung: Überschreitung der Alarmschwelle durch mindestens eine Zone.
aaaTempAlarm_AbsoluteLow	Sammelmeldung: Unterschreitung der Alarmschwelle durch mindestens eine Zone.
aaaTempAlarm_High	Sammelmeldung: Überschreitung der positiven inneren Toleranzschwelle durch mindestens eine Zone.
aaaTempAlarm_HighHigh	Sammelmeldung: Überschreitung der positiven äußeren Toleranzschwelle durch mindestens eine Zone.
aaaTempAlarm_Low	Sammelmeldung: Überschreitung der negativen inneren Toleranzschwelle durch mindestens eine Zone.
aaaTempAlarm_LowLow	Sammelmeldung: Überschreitung der negativen äußeren Toleranzschwelle durch mindestens eine Zone.
aaaTempFault_Reset	Sammelbefehl: Rücksetzen eventueller Fehlerzustände für alle Zonen.
bPfwTempLinksInitDone	reserviert
in_PfwTempCtrlInput	Die Eingangsprozessabbilder der Zonen.
out_PfwTempCtrlOutput	Die Ausgangsprozessabbilder der Zonen.
stPfwSupplyLineCfg	Die Parameter der Versorgungsgruppen.
stPfwTempCtrl	Die Laufzeitdaten der Zonen.
out_SwappedDigitalOut	Für die Umlenkung von digitalen Ausgangssignalen. Dieses Array stellt die Schnittstelle zum Verknüpfen der digitalen Ausgänge bereit. (E/A-Umlenkung)
in_SwappedDigitalIn	Für die Umlenkung von digitalen Eingangssignalen. Dieses Array stellt die Schnittstelle zum Verknüpfen der digitalen Eingänge bereit. (E/A-Umlenkung)

4 PLC-Zeitschaltuhr

4.1 Übersicht

Über die Wochenzeitschaltuhr ist es möglich, automatisch Zonen in verschiedene Betriebszustände zu überführen. Damit die eingestellten Zeiten auch nach einem Neustart vorhanden sind, werden Speicher- und Laderoutinen bereitgestellt.

Die folgende Tabelle gibt einen Überblick über die bereitgestellten Funktionsbausteine und ihre Bedeutung.

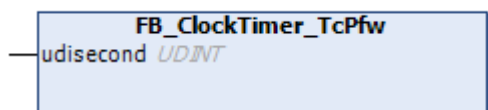
Name	Beschreibung
FB_ClockTimer_TcPfw() [▶ 92]	Schaltet die Ausgänge entsprechend der eingestellten Zeit ein bzw. aus.
FB_ClockTimerParamSave_TcPfw() [▶ 92]	Speichert die Zeitschaltuhreinstellungen ab.
FB_ClockTimerParamLoad_TcPfw() [▶ 93]	Lädt die Zeitschaltuhreinstellungen.

Datentypen: Strukturtypen

Name	Beschreibung
ST_TcPfw_ClockTimerCam [▶ 95]	Ist eine Unterstruktur der Struktur ST_TcPfw_ClockTimerItf. Hierüber wird die Einschalt- und Abschaltzeit eingegeben.
ST_TcPfw_ClockTimerItf [▶ 94]	Die Struktur enthält die Schnittstellen zur Einstellung der Zeitschaltuhr.

4.2 Bausteine

4.2.1 FB_ClockTimer_TcPfw()



In diesem Baustein ist die eigentliche Zeitschaltuhr implementiert. Er muss zyklisch aufgerufen werden und aktiviert bzw. deaktiviert in der eingestellten Zeit den entsprechenden Ausgang ST_TcPfw_ClockTimerItf.Q.

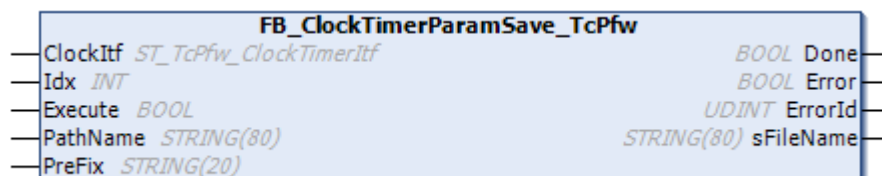
Syntax

```
VAR_INPUT
    udisecond      : UDINT:=0;
END_VAR
```

Eingänge

Name	Typ	Beschreibung
udisecond	UDINT	Zeit der Task in Mikrosekunden

4.2.2 FB_ClockTimerParamSave_TcPfw()



Dieser Baustein speichert die aktuellen Einstellungen der Zeitschaltuhr als Binärdatei ab.

Syntax

```
VAR_IN_OUT
  ClockItf : ST_TcPfw_ClockTimerItf;
END_VAR
VAR_INPUT
  Idx: INT;
  Execute : BOOL:=FALSE;
  PathName : STRING(80);
END_VAR
VAR_OUTPUT
  Done : BOOL:=FALSE;
  Error : BOOL:=FALSE;
  ErrorId : UDINT:=0;
END_VAR
```

 **Eingänge**

Name	Typ	Beschreibung
Idx	INT	Nummer der Zeitschaltuhr
Execute	BOOL	Mit steigender Flanke wird der Speichervorgang gestartet.
PathName	STRING	Pfadname, wo die Zeitschaltuhreinstellungen gespeichert werden sollen.

 **Ein-/Ausgänge**

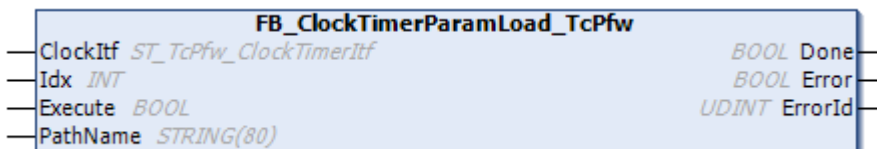
Name	Typ	Beschreibung
ClockItf	ST_TcPfw_ClockTime rtf	

 **Ausgänge**

Name	Typ	Beschreibung
Done	BOOL	Rückmeldung, dass das Speichern erfolgreich war.
Error	BOOL	Fehler beim Speichern der Zeitschaltuhreinstellungen.
ErrorId	UDINT	Fehlernummer

Durch eine positive Flanke am Execute Eingang wird der Speichervorgang aktiviert. Nach einem erfolgreichen Speichervorgang liegt für einen Zyklus ein Done an. Bei nicht erfolgreichem Laden wird ein Fehler mit Fehlernummer zurückgegeben.

4.2.3 FB_ClockTimerParamLoad_TcPfw()



Dieser Baustein lädt die abgespeicherten Einstellungen der Zeitschaltuhr.

Syntax

```
VAR_IN_OUT
  ClockItf: ST_TcPfw_ClockTimerItf;
END_VAR
VAR_INPUT
  Idx : INT;
  Execute : BOOL:=FALSE;
  PathName: STRING(80);
```

```

END_VAR
VAR_OUTPUT
  Done      : BOOL:=FALSE;
  Error     : BOOL:=FALSE;
  ErrorId   : UDINT:=0;
END_VAR

```

Eingänge

Name	Typ	Beschreibung
Idx	INT	Nummer der Zeitschaltuhr
Execute	BOOL	Mit steigender Flanke wird der Ladevorgang gestartet.
PathName	STRING	Pfadname, wo die Zeitschaltuhreinstellungen geladen werden können.

Ein-/Ausgänge

Name	Typ	Beschreibung
ClockItf	ST_TcPfw_ClockTimeItf	Aktuelle Einstellungen der Zeitschaltuhr als Binärdatei.

Ausgänge

Name	Typ	Beschreibung
Done	BOOL	Rückmeldung, dass das Laden erfolgreich war.
Error	BOOL	Fehler beim Laden der Zeitschaltuhreinstellungen.
ErrorId	UDINT	Fehlernummer

Verhalten des Bausteins:

Durch eine positive Flanke am Execute Eingang wird der Ladevorgang aktiviert. Nach einem erfolgreichen Ladevorgang liegt für einen Zyklus ein Done an. Bei nicht erfolgreichem Laden wird ein Fehler mit Fehlernummer zurückgegeben.

4.3 Strukturen

4.3.1 ST_TcPfw_ClockTimerItf

Diese Struktur muss in den globalen Variablen der Applikation deklariert werden. Die Struktur enthält die Schnittstellen zur Einstellung der Zeitschaltuhr.

In dieser Struktur wird der Wochentag ausgewählt sowie die Zeit; wann die Zeitschaltuhr aktiv ist und wann diese inaktiv ist.

Syntax

```

TYPE ST_TcPfw_ClockTimerItf:
(* location PfwLib_Common.PRO *)
(* last modification: 08.07.2008 *)
STRUCT
(*
see cnv_ClockTimerItf_TcPfw for format information
*)
  Q : ARRAY[cnPfwClockTimerCamFirst..cnPfwClockTimerCamLast] OF BOOL:=FALSE;

  Day: ARRAY[1..7,cnPfwClockTimerCamFirst..cnPfwClockTimerCamLast] OF ST_TcPfw_ClockTimerCam;
END_STRUCT
END_TYPE

```

Parameter

Name	Typ	Beschreibung
Q	ARRAY OF BOOL	Ist ein eindimensionales Array, in dem die Anzahl der Ausgänge pro Zeitschaltuhr angegeben ist. Hierüber kann eine Zone bis zu 9 verschiedene Betriebszustände pro Tag einnehmen.
Day	ARRAY OF ST_TcPfw_ClockTimerCam	Ist ein zweidimensionales Array, in dem die erste Zahl den Wochentag (1=Montag, 2=Dienstag, ..., 7=Sonntag) angibt und die zweite Zahl die Nummer der Zeitschaltuhr.

4.3.2 ST_TcPfw_ClockTimerCam

Ist eine Unterstruktur der Struktur ST_TcPfw_ClockTimerItrf. Hierüber wird die Einschalt- und Abschaltzeit eingegeben.

Syntax

```

TYPE ST_TcPfw_ClockTimerCam:
(* location PfwLib_Common.PRO *)
(* last modification: 05.06.2008 *)
STRUCT
(*
see cnv_ClockTimerCam_TcPfw for format information
*)
    On : ARRAY[1..3] OF INT;
    Off : ARRAY[1..3] OF INT;
END_STRUCT
END_TYPE
    
```

Parameter

Name	Typ	Beschreibung
On	ARRAY OF INT	Nach der Uhrzeit, welche in On durch den Bediener eingegeben worden ist, wird der entsprechende Ausgang aktiv. Dabei steht an erster Stelle On[1] die Stunde mit dem Wertebereich 0...23, an zweiter Stelle On[2] die Minute mit dem Wertebereich 0...59 und an dritter Stelle On[3] die Sekunde mit dem Wertebereich 0...59.
Off	ARRAY OF INT	Nach der Uhrzeit, welche in Off durch den Bediener eingegeben ist, worden ist, wird der entsprechende Ausgang inaktiv. Dabei steht an erster Stelle Off[1] die Stunde mit dem Wertebereich 0...23, an zweiter Stelle Off[2] die Minute mit dem Wertebereich 0...59 und an dritter Stelle Off[3] die Sekunde mit dem Wertebereich 0...59.

4.4 Knowledge Base

4.4.1 Inbetriebnahme

In dieser Inbetriebnahmeanleitung wird davon ausgegangen, dass der Temperaturregler vollständig inbetriebgenommen ist.

Im ersten Schritt ist es notwendig, dass ein Array vom Typ ST_TcPfw_ClockTimerItrf in den globalen Variablen erzeugt wird. Die Array-Größe gibt die Anzahl der Wochenzeitschaltuhren wieder. Anschließend sind folgende Konstanten anzulegen:

Konstantendefinition

Variable	Kurzbeschreibung	Beispiel-wert	Maximal-werte
cnPfwAppClockTimerFirst	Anfangsindex, um die Größe des Arrays festzulegen und damit die Anzahl der Wochenzeitschaltuhren.	1	1
cnPfwAppClockTimerLast	Endindex, um die Größe des Arrays festzulegen und damit die Anzahl der Wochenzeitschaltuhren.	2	Durch Rechenleistung und Speicher-platz begrenzt
cnPfwClockTimerCamFirst	Anfangsindex, um die Größe des Arrays festzulegen und damit die Anzahl der Kanäle pro Wochenzeitschaltuhr.	1	1
cnPfwClockTimerCamLas3	Endindex, um die Größe des Arrays festzulegen und damit die Anzahl der Kanäle pro Wochenzeitschaltuhr.	3	9

Parametrierung

Um der Zeitschaltuhr die korrekte Uhrzeit zu übergeben, muss folgender Strukturaufbau beachtet werden:

Die Struktur ST_TcPfw_ClockTimerItf teilt sich in Q und Day, wobei Day ein zweidimensionales Array ist. Der erste Index gibt die Nummer des Wochentags an (1=Montag, 2=Dienstag, 3=Mittwoch, 4=Donnerstag, 5=Freitag, 6=Samstag, 7=Sonntag) und der zweite Index die Nummer des entsprechenden Kanals wieder. Dieses zweidimensionale Array gliedert sich wiederum in ein On-Array und ein Off-Array. Sowohl On-Array als auch Off-Array besitzt drei Elemente für:

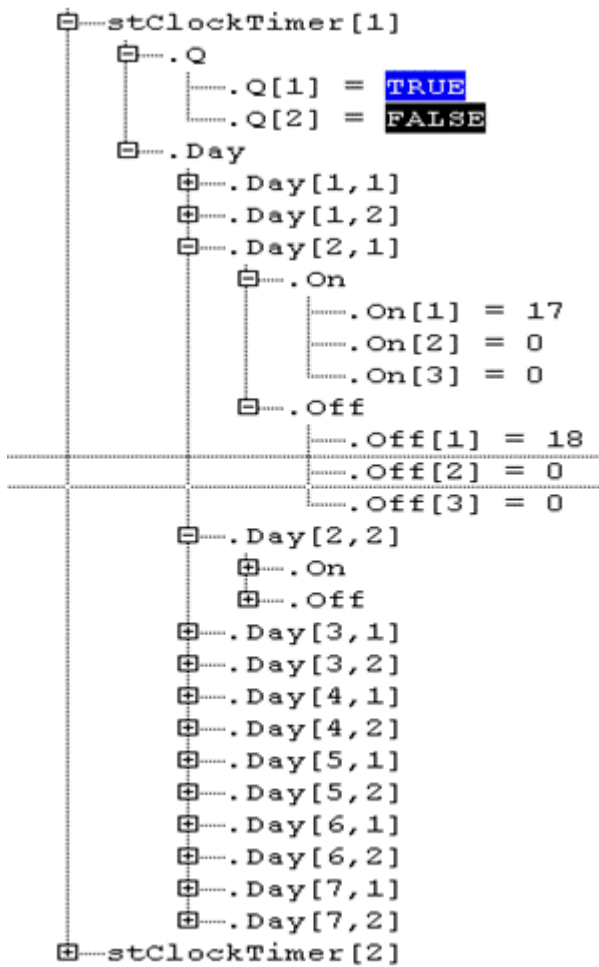
- On/Off[1]=Stunde im Wertebereich 0...23,
- On/Off[2]=Minute im Wertebereich 0...59,
- On/Off[3]=Sekunde im Wertebereich 0...59.

Hierüber wird die Einschalt- und Ausschaltuhrzeit festgelegt und der Ausgang Q entsprechend gesetzt.

Parametrierung

Als Beispielprojekt ist eine sehr einfache Applikation vorbereitet worden.

- Alle Programmzeilen, die sich auf die Wochenzeitschaltuhr beziehen, sind mit (*ClockTimer*) gekennzeichnet.
- Werden im laufenden Programm in das Array verschiedene Zeiten eingegeben, so wird zu den entsprechenden Zeiten der entsprechende Ausgang aktiv.
- **Beispiel:**
 - Am Dienstag von 17:00 Uhr bis 18:00 Uhr Standby / der Ausgang stClockTimer[1].Q[1] muss in der Applikation den Regler aktivieren.



4.4.2 Globale Variablen

Error Codes

Es sind keine eigenen Error Codes festgelegt. Wird ein Error Code ausgegeben, so muss dieser im Information System nachgeschlagen werden, da es sich um einen Fehler einer anderen Bibliothek handelt.

Deklarierte globale Konstanten

Name	Beschreibung
cnv_ClockTimerParamFileVers	Aktuelle Versionskennung
cnv_ClockTimerItf_TcPfw	Aktuelle Versionskennung der ST_TcPfw_ClockTimerItf Struktur.
cnv_ClockTimerCam_TcPfw	Aktuelle Versionskennung der ST_TcPfw_ClockTimerCam Struktur.

Nicht deklarierte globale Konstanten

Name	Beschreibung	Maximalwerte
cnPfwClockTimerCamFirst	Anfangsindex, um die Größe des Arrays festzulegen und damit die Anzahl der Kanäle pro Wochenzeitschaltuhr.	1
cnPfwClockTimerCamLast	Endindex, um die Größe des Arrays festzulegen und damit die Anzahl der Kanäle pro Wochenzeitschaltuhr.	Durch Rechenleistung und Speicherplatz begrenzt
cnPfwAppClockTimerFirst	Anfangsindex, um die Größe des Arrays festzulegen und damit die Anzahl der Zeitschaltuhren.	1
cnPfwAppClockTimerLast	Endindex, um die Größe des Arrays festzulegen und damit die Anzahl der Zeitschaltuhren.	9

4.4.3 FAQs

Name	Beschreibung
FAQ#1 [► 98]	Welche Uhrzeiteingaben sind zulässig?
FAQ#2 [► 98]	Wie wird die Uhrzeit berechnet? Wird Sommer/Winterzeit berücksichtigt? Werden Schaltjahre berücksichtigt?
FAQ#3 [► 98]	Wie aktiviere ich eine Uhrzeit über mehrere Tage?
FAQ#4 [► 98]	Kann ich über eine Zeitschaltuhr mehrere Schaltzeiten vorgeben?

FAQ#1: Uhrzeiteingaben

Es ist eine 24h Uhrzeitangabe notwendig, das heißt:

- Stunde: Wertebereich 0...23
- Minute: Wertebereich 0...59
- Sekunde: Wertebereich 0...59

FAQ#2: Wie wird die Uhrzeit berechnet?

Die Uhrzeit wird aus der Uhrzeit im Rechner ermittelt.

Es ist wichtig, dass diese Uhrzeit richtig eingestellt ist.

Da der Rechner eine Sommer-/ Winterzeit-Umstellung erkennt, wird diese auch in der Steuerung erkannt.

Da der Rechner die Schaltjahre kennt, werden diese auch in der Steuerung erkannt.

FAQ#3: Uhrzeit über mehrere Tage aktivieren

In der Struktur ST_TcPfw_ClockTimerCam wird unter **On** die Startuhrzeit eingetragen und unter **Off** muss die Uhrzeit 23:59:59 eingetragen werden. Am darauffolgenden Tag wird unter **On** 00:00:00 eingetragen und unter **Off** wann wieder abgeschaltet werden soll. In diesem Fall wird die Heizung nicht um Mitternacht für eine Sekunde ausgeschaltet, sondern bleibt aktiv.

FAQ#4: Über eine Zeitschaltuhr mehrere Schaltzeiten vorgeben

Es ist möglich, pro Wochenzeitschaltuhr 9 Kanäle anzulegen. Für jeden Kanal wird eine Einschalt- und eine Ausschaltzeit angegeben sowie ein Ausgang.

5 PLC-Alarmvisualisierung

5.1 Übersicht

Für das Anzeigen, Deaktivieren, Löschen und Verwalten von Meldungen (Alarmer, Warnungen usw.) werden fertige Bausteine zur Verfügung gestellt. Die Temperaturregelung selbst stellt bei einem Fehler lediglich eine ErrorID sowie ein Error-Flag zur Verfügung.

Meldungen werden durch den Baustein FB_MsgAppend_TcTvA() in ein Array des Typs ST_TcTvA_Alarm_Itf (Alarmhistorie) eingegeben und aktiviert. Ist die Ursache einer Meldung behoben, kann über **Reset** diese Meldung deaktiviert werden. Die Deaktivierung einer Meldung findet über den Baustein FB_MsgDeactivate_TcTvA() statt. Anschließend kann die Meldung durch Aufruf des Bausteins FB_MsgClearPending_TcTvA() in der Alarmhistorie gelöscht werden. Damit in der Alarmhistorie durch gelöschte Meldungen, keine Leerzeilen entstehen, muss der Baustein FB_MsgGarbageCollect_TcTvA() zyklisch aufgerufen werden.

Datentypen: Funktionsbausteine

Name	Beschreibung
FB_MsgAppend_TcTvA() [▶_99]	Aktiviert eine aktive Fehlermeldung im Array des Typs ST_TcTvA_Alarm_Itf.
FB_MsgClearPending_TcTvA() [▶_100]	Dieser Baustein löscht inaktive Fehler aus der Fehlerhistorie (Array des Typs ST_TcTvA_Alarm_Itf).
FB_MsgClearSignal_TcTvA() [▶_101]	Dieser Baustein löscht die Meldung ab.
FB_MsgDeactivate_TcTvA() [▶_102]	Deaktiviert eine nicht aktive Fehlermeldung im Array des Typs ST_TcTvA_Alarm_Itf.
FB_MsgGarbageCollect_TcTvA() [▶_102]	Muss zyklisch aufgerufen werden und sortiert die Fehlerhistorie (Array des Typs ST_TcTvA_Alarm_Itf) neu.
FB_MsgUpdateTime_TcTvA() [▶_103]	Dieser Baustein ermittelt die Windows-Systemzeit und gibt diese an die Applikation zurück.

Datentypen: Strukturtypen

Name	Beschreibung
ST_TcTvA_Alarm_Itf [▶_104]	Mit dem Anlegen eines Arrays dieser Struktur ergibt sich die Alarmhistorie.

5.2 Bausteine

5.2.1 FB_MsgAppend_TcTvA()

FB_MsgAppend_TcTvA	
— stTime	<i>TIMESTRUCT</i>
— Idx1	<i>INT</i>
— Idx2	<i>INT</i>
— ErrorId	<i>DINT</i>
— Prio	<i>INT</i>
— Active	<i>BOOL</i>
— IdxFirst	<i>INT</i>
— IdxLast	<i>INT</i>
— sync_idx	<i>INT</i>
— pAlarmBuffer	<i>POINTER TO ST_TcTvA_Alarm_Itf</i>

Dieser Baustein hängt an ein Array des Typs ST_TcTvA_Alarm_Itf aktive Alarmer an. Um in der Visualisierung möglichst viele Informationen zum Fehler zu erhalten, können der Fehlermeldung die unten aufgeführten Parameter mitgegeben werden.

Syntax

```

VAR_IN_OUT
  stTime      : Timestruct;
END_VAR
VAR_INPUT
  Idx1        : INT:=0;
  Idx2        : INT:=0;
  ErrorId     : DINT:=0;
  Prio        : INT:=0;
  Active      : BOOL:=TRUE;

  IdxFIRST   : INT;
  IdxLast    : INT;
  sync_idx   : INT:=0;
  pAlarmBuffer: POINTER TO ST_TcTvA_Alarm_Itf;
END_VAR
    
```

 **Eingänge**

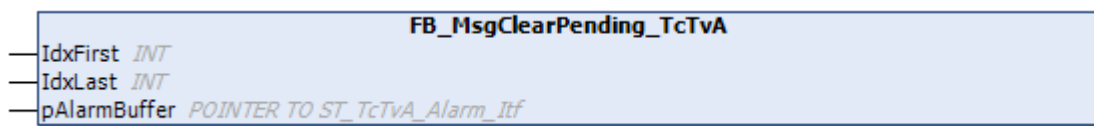
Name	Typ	Beschreibung
Idx1	INT	Zonennummer oder Antriebsnummer der fehlerhaften Zone.
Idx2	INT	Modulnummer der fehlerhaften Zone.
ErrorId	DINT	Die Fehlernummer kann ggf. auch als Klartextmeldung aus einer .xml Datei gelesen werden.
Prio	INT	Die Priorität des Fehlers, wobei 3=Alarm 2=Warning 1=Note 0=Empty ist.
Active	BOOL	Über Active wird angezeigt, dass der Fehler Aktive ist.
IdxFirst	INT	Der erste Index im Alarmpuffer.
IdxLast	INT	Der letzte Index im Alarmpuffer.
sync_idx	INT	Die Adresse des Alarmpuffers mit dem Typ ST_TcTvA_Alarm_Itf.
pAlarmBuffer	POINTER TO ST_TcTvA_Alarm_Itf	Die Adresse des Alarmpuffers mit dem Typ ST_TcTvA_Alarm_Itf.

 **Ein-/Ausgänge**

Name	Typ	Beschreibung
stTime	TIMESTRUCT	Die Zeit, wann der Fehler auftrat.

Nach einem erfolgreichen Aufruf dieses Bausteins muss sich an der ersten freien Stelle in der Alarmhistorie ein weiterer Eintrag befinden.

5.2.2 FB_MsgClearPending_TcTvA()



Dieser Baustein löscht alle inaktiven Meldungen aus der Alarmhistorie.

Syntax

```

VAR_INPUT
  IdxFIRST   : INT;
  IdxLast    : INT;
  pAlarmBuffer: POINTER TO ST_TcTvA_Alarm_Itf;
END_VAR
    
```

 **Eingänge**

Name	Typ	Beschreibung
IdxFirst	INT	Der erste Index im Alarmpuffer.
IdxLast	INT	Der letzte Index im Alarmpuffer.
pAlarmBuffer	POINTER TO ST_TcTvA_Alarm_Itf	Die Adresse des Alarmpuffers mit dem Typ ST_TcTvA_Alarm_Itf.

5.2.3 FB_MsgClearSignal_TcTvA()

FB_MsgClearSignal_TcTvA

- IdxFirst *INT*
- IdxLast *INT*
- pAlarmBuffer *POINTER TO ST_TcTvA_Alarm_Itf*
- Idx1 *INT*
- Idx2 *INT*
- ErrorId *DINT*
- Prio *INT*

Dieser Baustein deaktiviert den booleschen Ausdruck ST_TcTvA_Alarm_Itf.Signal.

Syntax

```

VAR_INPUT
  IdxFirst: INT;
  IdxLast: INT;
  pAlarmBuffer: POINTER TO ST_TcTvA_Alarm_Itf;

  Idx1 : INT:=0;
  Idx2 : INT:=0;
  ErrorId: DINT:=0;
  Prio : INT:=0;
END_VAR
    
```

 **Eingänge**

Name	Typ	Beschreibung
IdxFirst	INT	Der erste Index im Alarmpuffer.
IdxLast	INT	Der letzte Index im Alarmpuffer.
pAlarmBuffer	POINTER TO ST_TcTvA_Alarm_Itf	Die Adresse des Alarmpuffers mit dem Typ ST_TcTvA_Alarm_Itf.
Idx1	INT	Zonenummer der fehlerhaften Zone.
Idx2	INT	Modulnummer der fehlerhaften Zone.
ErrorId	DINT	Die Fehlernummer
Prio	INT	Die Priorität des Fehlers, wobei 3=Alarm 2=Warning 1=Note 0=Empty.

Verhalten des Bausteins:

Steht im Alarmpuffer ein Alarm mit prio:=3 an so setzt dieser Alarm zusätzlich das Signal ST_TcTvA_Alarm_Itf.Signal:=TRUE. Hierdurch kann zum Beispiel eine Hupe angesteuert werden. Über das Aufrufen dieses Bausteins werden alle ST_TcTvA_Alarm_Itf.Signal:=FALSE, wodurch die Meldung ausgeschaltet wird.

5.2.4 FB_MsgDeactivate_TcTvA()



Dieser Baustein deaktiviert eine nicht aktive Meldung in der Alarmhistorie (Typ: ST_TcTvA_Alarm_Itf). Dabei wird die Meldung nicht aus der Alarmhistorie gelöscht.

Syntax

```

VAR_INPUT
Idx1      : INT:=0;
Idx2      : INT:=0;
ErrorId   : DINT:=0;
Prio      : INT:=0;

IdxFirst  : INT;
IdxLast   : INT;
pAlarmBuffer: POINTER TO ST_TcTvA_Alarm_Itf;
END_VAR
    
```

🔴 Eingänge

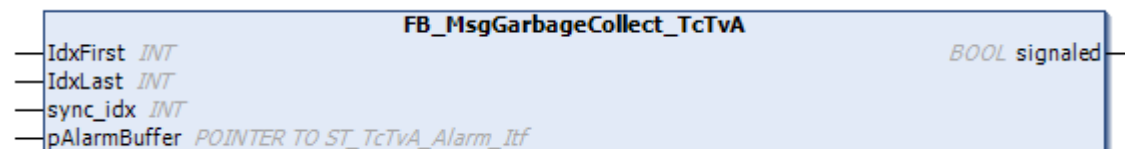
Name	Typ	Beschreibung
Idx1	INT	Zonenummer der fehlerhaften Zone.
Idx2	INT	Modulnummer der fehlerhaften Zone.
ErrorId	DINT	Die Fehlernummer
Prio	INT	Die Priorität des Fehlers, wobei 3=Alarm 2=Warning 1=Note 0=Empty.
IdxFirst	INT	Der erste Index im Alarmpuffer.
IdxLast	INT	Der letzte Index im Alarmpuffer.
pAlarmBuffer	POINTER TO ST_TcTvA_Alarm_Itf	Die Adresse des Alarmpuffers mit dem Typ ST_TcTvA_Alarm_Itf.

Verhalten des Bausteins:

Zu jedem FB_MsgAppend_TcTvA() Baustein gehört auch ein FB_MsgDeactivate_TcTvA(), der die Meldung zurücksetzt. Damit sichergestellt ist, dass exakt die richtige Meldung zurückgesetzt wird, muss die Deaktivierung mit den gleichen Eigenschaften (Idx1, Idx2, ErrorID, Prio) wie die Aktivierung aufgerufen werden.

Nach einer erfolgreichen Deaktivierung ist ST_TcTvA_Alarm_Itf.active:=FALSE. Die Variable ST_TcTvA_Alarm_Itf.Bitmap hängt von ST_TcTvA_Alarm_Itf.Prio und ob ST_TcTvA_Alarm_Itf.active:=TRUE oder FALSE ist ab.

5.2.5 FB_MsgGarbageCollect_TcTvA()



Dieser Baustein prüft, ob in der Alarmhistorie (Array vom Typ ST_TcTvA_Alarm_Itf) Leerzellen enthalten sind und rückt nachfolgende Meldungen auf.

Syntax

```
VAR_INPUT
  IdxFirst : INT;
  IdxLast : INT;
  sync_idx : INT:=0;
  pAlarmBuffer: POINTER TO ST_TcTvA_Alarm_Itf;
END_VAR
VAR_OUTPUT
  signaled : BOOL;
END_VAR
```

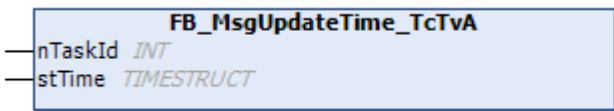
 **Eingänge**

Name	Typ	Beschreibung
IdxFirst	INT	Der erste Index im Alarmpuffer.
IdxLast	INT	Der letzte Index im Alarmpuffer.
sync_idx	INT	Reserviert; mit 0 wird automatisch im Array nach Leerstellen gesucht.
pAlarmBuffer	POINTER TO ST_TcTvA_Alarm_Itf	Die Adresse des Alarmpuffers mit dem Typ ST_TcTvA_Alarm_Itf .

 **Ausgänge**

Name	Typ	Beschreibung
signaled	BOOL	Signalisiert, ob eine Meldung vorliegt.

5.2.6 FB_MsgUpdateTime_TcTvA()



Dieser Baustein ermittelt die Windows-Systemzeit und gibt diese an die Applikation zurück.

```
VAR_INPUT
  nTaskId: INT;
END_VAR
VAR_IN_OUT
  stTime : TIMESTRUCT;
END_VAR
```

 **Eingänge**

Name	Typ	Beschreibung
nTaskId	INT	Task in der das Alarmhandling stattfindet.

 **Ein-/Ausgänge**

Name	Typ	Beschreibung
stTime	TIMESTRUCT	Systemzeit von Windows.

5.3 Strukturen

5.3.1 ST_TcTvA_Alarm_Itf

Mit dem Anlegen eines Arrays dieser Struktur ergibt sich die Alarmhistorie. Jede neue, aktive und inaktive Meldung wird in diesem Array zwischengespeichert.

Syntax

```

TYPE ST_TcTvA_Alarm_Itf :
(* last modification: 04.10.2008 *)
STRUCT
(*
=====
message data
see cnv_ItfStructType_TvA for format definition
=====
*)
sTime: STRING(23);

ErrorId: DINT:=0;

Count : INT:=0;
Bitmap : INT:=0; (* 5=AlarmInaktiv 4=AlarmAktiv 3=WarningInaktiv 2=WarningAktiv 1=Note 0=Empty *)
Prio : INT:=0; (* 3=Alarm 2=Warning 1=Note 0=Empty *)
Idx1 : INT:=0; (* module or drive no. *)
Idx2 : INT:=0; (* zone no. *)
i_align: ARRAY[1..3]OF INT;

Active : BOOL:=FALSE;
Pending: BOOL:=FALSE;
Signal : BOOL:=FALSE;
b_align: ARRAY[1..5]OF BOOL;
(**)
END_STRUCT
END_TYPE

```

Parameter

Name	Typ	Beschreibung
sTime	STRING	Die Zykluszeit (in Sekunden) des PWM-Signalgenerators.
ErrorId	DINT	Fehlernummer (hierüber kann eine Klartextmeldung aus einer .XML-Datei ausgelesen werden).
Count	INT	Nummer des Alarmspeichers.
Bitmap	INT	Hierüber kann der Status in einer Fehlermeldung visualisiert werden. Dabei ist 5=Alarm inaktiv, 4=Alarm aktiv, 3=Warning inaktiv, 2=Warning aktiv, 1=Note, 0=Empty.
Prio	INT	Gibt die Art des Alarms wieder. Es wird zwischen Alarm=3, Warning=2, Note=1, Empty=0 unterschieden.
Idx1	INT	Hier soll die Modul- oder Drivenummer angegeben werden (kann aber beliebig verwendet werden).
Idx2	INT	Hier soll die Zonennummer angegeben werden (kann aber beliebig verwendet werden).
i_align	ARRAY OF INT	INTEGER Alignment.
Active	BOOL	Über den Aufruf des Bausteins FB_MsgAppend_TcTvA() wird ein Fehler in die Alarmhystorie hinzugefügt und über dieses Bool aktiviert. Ist der Fehler behoben, wird durch den Baustein FB_MsgDeactivate_TcTvA() dieses Bool deaktiviert.
Pending	BOOL	Alarmer die nicht mehr Aktive sind (Active:=FALSE) können über den Baustein FB_MsgClearPending_TcTvA() deaktiviert werden.
Signal	BOOL	Bei Meldungen mit Prio:=5 wird zusätzlich Signal:=TRUE, wodurch ein Horn aktiv geschaltet werden kann. Durch den Aufruf von FB_MsgClearSignal_TcTvA() wird Signal:=FALSE.
b_align	ARRAY OF BOOL	BOOL Alignment.

5.4 Knowledge Base

5.4.1 Inbetriebnahme

Es gibt zwei Applikationsbeispiele für die Inbetriebnahme des Alarmhandlings:

- minimale Applikation, um nur das Alarmhandling zu beschreiben.
- Applikation mit kompletter Temperaturbibliothek für das Alarmhandling der Temperaturbibliothek.

In beiden Applikationen sind wichtige Stellen für das Alarmhandling durch (*TVAlarm*) gekennzeichnet.

Grundsätzlicher Aufbau der Applikation Example_TvAlarm zum Alarmhandling

Das Projekt besitzt folgende Arrays:

- **stVisuTempAlarm:**
Dieses Array ist vom Typ ST_VisuAlarm (eine Applikations-spezifische Struktur) und dient als Puffer, falls es mehrere Programmteile gibt, die gleichzeitig einen Fehler erzeugen (z.B. mehrere Zonen an einem Extruder), sowie zum Voreinstellen von prio, ldx1 und ldx2 zu jeder Fehlernummer.
- **aaaPfwTempMessageHmi:**
Dieses Array ist vom Typ ST_TcTvA_Alarm_Itf und ist das Array, welches für die Visualisierung benutzt werden muss. Hier werden die einzelnen Fehler der Reihenfolge nach aufgelistet.
- **FB_InitAlarmMessage:**
In diesem Baustein wird jeder Zelle des Array stVisuTempAlarm genau ein Fehlercode zugewiesen.
- **F_initTempMsg:**
Diese Funktion wird im Baustein FB_InitAlarmMessage aufgerufen und ist für die Initialisierung des Fehlercodes im Array stVisuTempAlarm zuständig.
- **FB_MsgCheck:**
Dieser Baustein muss zyklisch aufgerufen werden. Von hier aus werden sämtliche Bibliotheksbausteine aufgerufen. Das heißt:
 - aktive Meldungen werden in das Meldesystem als aktiv eingetragen,
 - inaktive Meldungen werden in das Meldesystem als inaktiv eingetragen,
 - inaktive Meldungen können gelöscht werden.
- **FB_MsgCheckTemp:**
In diesem Baustein werden alle Meldungen, die in diesem Zyklus aktiv geworden sind, an den entsprechenden Stellen im Array stVisuTempAlarm eingetragen.

Da in diesem Beispiel die PfwLib_Processing.lib-Bibliothek eingebunden wurde, mussten verschiedene Konstanten für die Temperaturregelung deklariert werden, welche jedoch für das eigentliche Alarm-Handling unwichtig sind und nur dazu dienen, dass das Projekt übersetzt werden kann. Im Quellcode sind diese Stellen entsprechend gekennzeichnet.

Grundsätzlicher Aufbau der Applikation Example_TvAlarm_TempCtrl zum Alarmhandling

Im Vergleich zum oberen Beispiel ist dieses Alarmhandling durch die verschiedenen Zonen der Temperaturregelung und durch die vielen möglichen Fehlermeldungen wesentlich komplexer.

Das Projekt besitzen folgende Arrays:

- **stVisuTempAlarm:**
Dieses Array ist vom Typ ST_VisuAlarm (eine Applikations-spezifische Struktur) und dient als Puffer, falls in einem Zyklus mehrere Alarme aktiv werden. In dem Array hat man die Möglichkeit 30 verschiedene Fehler für eine Zone aufzunehmen.
- **aaaPfwTempMessageHmi:**
Dieses Array ist vom Typ ST_TcTvA_Alarm_Itf und muss für die Visualisierung benutzt werden.
- **FB_InitAlarmMessage:**
In diesem Baustein wird jeder Zelle eines Arrays stVisuTempAlarm genau ein Fehlercode zugewiesen.

- **F_initTempMsg:**
Diese Funktion wird im Baustein FB_InitAlarmMessage aufgerufen und ist für die Initialisierung des Fehlercodes im Array stVisuTempAlarm zuständig.
- **FB_MsgCheck:**
Dieser Baustein muss zyklisch aufgerufen werden. Von hier aus werden sämtliche Bibliotheksbausteine aufgerufen. Das heißt
 - aktive Meldungen werden in das Meldesystem als aktiv eingetragen,
 - inaktive Meldungen werden in das Meldesystem als inaktiv eingetragen,
 - inaktive Meldungen können gelöscht werden.
- **FB_MsgCheckTemp:**
In diesem Baustein werden alle Meldungen, die in diesem Zyklus aktiv geworden sind, an den entsprechenden Stellen im Array stVisuTempAlarm eingetragen.

5.4.2 Globale Variablen

Error Codes

Es sind keine eigenen Error Codes festgelegt. Wird ein Error Code ausgegeben, so muss dieser im Informationssystem von Beckhoff nachgeschlagen werden, da es sich um einen Fehler einer anderen Bibliothek handelt.

Name	Beschreibung
cnv_ItfStructType_TvA	Aktuelle Versionskennung der <u>ST_TcTvA_Alarm_Itf</u> Struktur.

5.4.3 FAQs

Name	Beschreibung
FAQ#1 [▶ 106]	Wird zwischen Alarm, Warnung und Information unterschieden?
FAQs [▶ 106]	Können auch andere Errors, außer die aus bestimmten Bibliotheken, mitgelockt werden?

FAQ#1: Wird zwischen Alarm, Warnung und Information unterschieden?

Es gibt drei verschiedene Meldestufen, welche unter ST_TcTvA_Alarm_Itf.Prio festgelegt werden können:

- prio:=3 bedeutet Alarm und aktiviert gleichzeitig ST_TcTvA_Alarm_Itf.Signal. Dieser Ausgang kann zum Beispiel mit einem Horn verknüpft werden.
- prio:=2 ist eine Warnung.
- prio:=1 ist eine Information.

FAQ#2: Können auch andere Errors außer die aus bestimmten Bibliotheken mitgelockt werden?

Wenn die zusätzlichen Error Codes mit in die Alarmverwaltung aufgenommen werden, ist es möglich, diese Meldungen auch anzuzeigen.

Mehr Informationen:
www.beckhoff.de/tf8540

Beckhoff Automation GmbH & Co. KG
Hülshorstweg 20
33415 Verl
Deutschland
Telefon: +49 5246 9630
info@beckhoff.com
www.beckhoff.com

