



DOKUMENTATION ISG-kernel

Funktionsbeschreibung Fehlermanagement

Kurzbezeichnung:
FCT-M7

© Copyright
ISG Industrielle Steuerungstechnik GmbH
STEP, Gropiusplatz 10
D-70563 Stuttgart
Alle Rechte vorbehalten
www.isg-stuttgart.de
support@isg-stuttgart.de

Dokumentation Version: 1.06
07.11.2024

Vorwort

Rechtliche Hinweise

Diese Dokumentation wurde sorgfältig erstellt. Die beschriebenen Produkte und der Funktionsumfang werden jedoch ständig weiterentwickelt. 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.

Qualifikation des Personals

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

Zur Installation und Inbetriebnahme ist die Beachtung der Dokumentation, der nachfolgenden Hinweise und Erklärungen unbedingt notwendig. Das Fachpersonal ist verpflichtet, für jede Installation und Inbetriebnahme die zum betreffenden Zeitpunkt veröffentlichte Dokumentation zu verwenden.

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

Weiterführende Informationen

Unter den Links (DE)

<https://www.isg-stuttgart.de/produkte/softwareprodukte/isg-kernel/dokumente-und-downloads>

bzw. (EN)

<https://www.isg-stuttgart.de/en/products/softwareproducts/isg-kernel/documents-and-downloads>

finden Sie neben der aktuellen Dokumentation weiterführende Informationen zu Meldungen aus dem NC-Kern, Onlinehilfen, SPS-Bibliotheken, Tools usw.

Haftungsausschluss

Änderungen der Software-Konfiguration, die über die dokumentierten Möglichkeiten hinausgehen, sind unzulässig.

Marken und Patente

Der Name ISG®, ISG kernel®, ISG virtuos®, ISG dirigent® und entsprechende Logos sind eingetragene und lizenzierte Marken der ISG Industrielle Steuerungstechnik GmbH.

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

Copyright

© ISG Industrielle Steuerungstechnik GmbH, Stuttgart, Deutschland.

Weitergabe sowie Vervielfältigung dieses Dokuments, Verwertung und Mitteilung seines Inhalts sind verboten, soweit nicht ausdrücklich gestattet. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte für den Fall der Patent-, Gebrauchsmuster oder Geschmacksmustereintragung vorbehalten.

Allgemeine- und Sicherheitshinweise

Verwendete Symbole und ihre Bedeutung

In der vorliegenden Dokumentation werden die folgenden Symbole mit nebenstehendem Sicherheitshinweis und Text verwendet. Die (Sicherheits-) Hinweise sind aufmerksam zu lesen und unbedingt zu befolgen!

Symbole im Erklärtext

- Gibt eine Aktion an.
- ⇒ Gibt eine Handlungsanweisung an.



GEFAHR

Akute Verletzungsgefahr!

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



VORSICHT

Schädigung von Personen und Maschinen!

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



Achtung

Einschränkung oder Fehler

Dieses Symbol beschreibt Einschränkungen oder warnt vor Fehlern.



Hinweis

Tipps und weitere Hinweise

Dieses Symbol kennzeichnet Informationen, die zum grundsätzlichen Verständnis beitragen oder zusätzliche Hinweise geben.



Beispiel

Allgemeines Beispiel

Beispiel zu einem erklärten Sachverhalt.



Programmierbeispiel

NC-Programmierbeispiel

Programmierbeispiel (komplettes NC-Programm oder Programmsequenz) der beschriebenen Funktionalität bzw. des entsprechenden NC-Befehls.



Versionshinweis

Spezifischer Versionshinweis

Optionale, ggf. auch eingeschränkte Funktionalität. Die Verfügbarkeit dieser Funktionalität ist von der Konfiguration und dem Versionsumfang abhängig.

Inhaltsverzeichnis

Vorwort	2
Allgemeine- und Sicherheitshinweise	3
1 Übersicht	6
2 Beschreibung	7
2.1 Melden und Aufzeichnen der Fehler	9
2.1.1 Unterdrücken von Warnungen	11
2.1.2 Ausgabebeispiele.....	12
2.2 Filtern von Fehlermeldungen	13
2.2.1 Programmierbeispiel Filtern von Fehlermeldungen	15
2.3 Aktivieren von Filterregeln.....	16
2.4 Ausgeben eigener Fehlermeldungen	17
2.4.1 Benutzerdefinierte Fehlerausgabe (#ERROR)	18
2.4.2 Ausgabe eigendefinierter Fehlermeldungen an HMI	21
2.5 Fehlerausgabe TwinCAT3	23
2.5.1 Ausgabe nur über SPS	25
2.5.2 Direkte Ausgabe an Eventlogger	25
2.5.2.1 Ausgabeformat-Eventlogger	27
2.5.3 Ausgabebeispiel an HMI	28
3 Parameter	29
3.1 Übersicht	29
3.2 Beschreibung	31
3.2.1 Hochlaufparameter	31
3.2.2 Kanalparameter	40
3.2.3 Achsparameter.....	45
3.3 CNC-Objekte	49
3.4 HLI-Parameter.....	52
4 Anhang	55
4.1 Anregungen, Korrekturen und neueste Dokumentation.....	55
Stichwortverzeichnis	56

Abbildungsverzeichnis

Abb. 1:	Übersicht Lesen und Auswerten der Fehlermeldung innerhalb der CNC.....	7
Abb. 2:	Ablauf und Aufbereitung der Fehlermeldungen in der CNC	8
Abb. 3:	Überblick der 3 Ausgabemöglichkeiten	9
Abb. 4:	Überblick der Ausgabemöglichkeiten mit Anbindung des SPS-Bausteins	10
Abb. 5:	Ausgabebeispiel - Protokolldatei	12
Abb. 6:	Meldung aus dem Kanal.....	13
Abb. 7:	Meldung einer Achse.....	13
Abb. 8:	Meldung aus dem Hochlauf.....	14
Abb. 9:	Aktivieren von Filterregeln	16
Abb. 10:	Ausgabe eigener Fehlertexte in Protokolldatei.....	17
Abb. 11:	Codeauszug – Ausgangslage.....	21
Abb. 12:	Codeauszug mit integriertem Fehlertext.....	22
Abb. 13:	Architektur- Übersicht TwinCAT-Eventlogger	23
Abb. 14:	Übersicht mit TwinCAT-Eventlogger	23
Abb. 15:	FB ChannelError()- Platzierung im System	25
Abb. 16:	Modifikation der DateiTcCncErrors.xml	26
Abb. 17:	Ausgabe in Beckhoff-Oberfläche	26
Abb. 18:	Eventlogger-Output	27
Abb. 19:	Json-Attribute Output.....	27
Abb. 20:	Bildschirmausgabe der Fehlermeldung an der Beckhoff-HMI	28
Abb. 21:	Bildschirmausgabe nach Doppelklick	28

1 Übersicht

Aufgabe

Die Funktionalität des Fehlermanagements umfasst folgende Bereiche:

- Ausgabeort und Umfang der Fehlerausgabe
- Filtern von Fehlermeldungen, sowohl global als auch kanal- oder achsspezifisch
- Zusätzliche Informationen in Fehlermeldungen hinzufügen
- Ausgabe benutzerdefinierter Fehlermeldungen durch den Anwender



Versionshinweis

Diese Funktionalität ist ab der CNC-Version V3.00.xx verfügbar.

Parametrierung

Die Parametrierung der Fehlerausgabe erfolgt in den Hochlaufparametern. Das Filtern und Ergänzen von Fehlermeldungen erfolgt abhängig von der Ebene, auf der der entsprechende Filter wirken soll.

Programmierung

Eine komplette Übersicht findet sich im Kapitel Parameter [▶ 29].

Eigene Fehlermeldungen können vom Anwender über den #ERROR [▶ 18]- Befehl im NC-Programm erzwungen werden.

Obligatorischer Hinweis zu Verweisen auf andere Dokumente

Zwecks Übersichtlichkeit wird eine verkürzte Darstellung der Verweise (Links) auf andere Dokumente bzw. Parameter gewählt, z.B. [PROG] für Programmieranleitung oder P-AXIS-00001 für einen Achsparameter.

Technisch bedingt funktionieren diese Verweise nur in der Online-Hilfe (HTML5, CHM), allerdings nicht in PDF-Dateien, da PDF keine dokumentenübergreifenden Verlinkungen unterstützt.

2 Beschreibung

Tritt ein Kanal-, Einzelachs- oder Plattformfehler auf, so wird eine entsprechende Meldung abgesetzt. Fehler können von einer CNC-Diagnoseinstanz aufbereitet und mit zusätzlicher Information ergänzt werden (Zugriff auf Datenbank / Dateisystem). Der Anwender kann das Verhalten (Filter) einzelner Fehlermeldungen parametrieren oder über die SPS einstellen. Die erweiterte Fehlermeldung wird über eine Diagnoseinstanz an das übergeordnete Diagnosesystem weitergemeldet.

An die steuerungsinternen weiteren Teilnehmer (z.B. SPS) wird das Auftreten eines Fehlers kompakt und zeitnah mit Fehler-ID, Fehlerklasse und Zeitpunkt gemeldet. Diese Teilnehmer können eine interne Fehlerreaktion unmittelbar ableiten.

Das Lesen und Aufbereiten der CNC-Fehlermeldungen wird standardmäßig durch die CNC durchgeführt.

Das Weitermelden der Fehler über das Kommunikationssystem an eine überlagerte Visualisierung wird spezifisch durch den Anwender eingebracht.

Verwaltung von Fehlermeldungen

Die Funktionalität des Fehlermanagements ermöglicht dem Anwender die Ausgabe von Fehlermeldungen zu steuern. Die CNC fungiert hierfür als Sammelstelle aller zur Verfügung stehender Informationen:

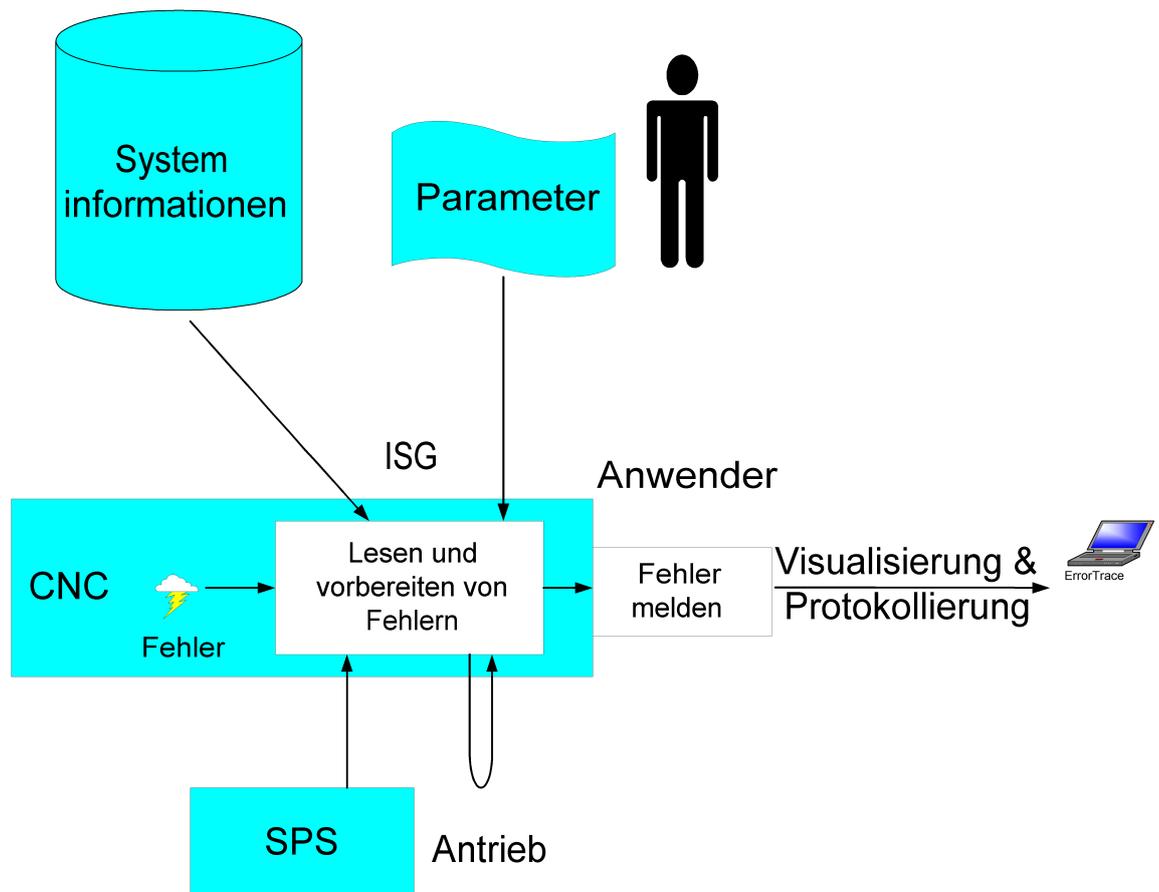


Abb. 1: Übersicht Lesen und Auswerten der Fehlermeldung innerhalb der CNC

Ablauf

- Erfassen und Sammeln aller Fehler der CNC
- Filter: Unterdrücken und Filtern der Fehler
- Add. Info: Hinzufügen von interner / externer Fehlerinformation
- Format: Aufbereitung und Formatierung des Fehlers
- Report & Log: Protokollierung der Fehler in Datei (LOG) -> Protokolldatei
- Report & Log: Melden der aufbereiteten Fehler (REPORT) -> Bildschirmausgabe

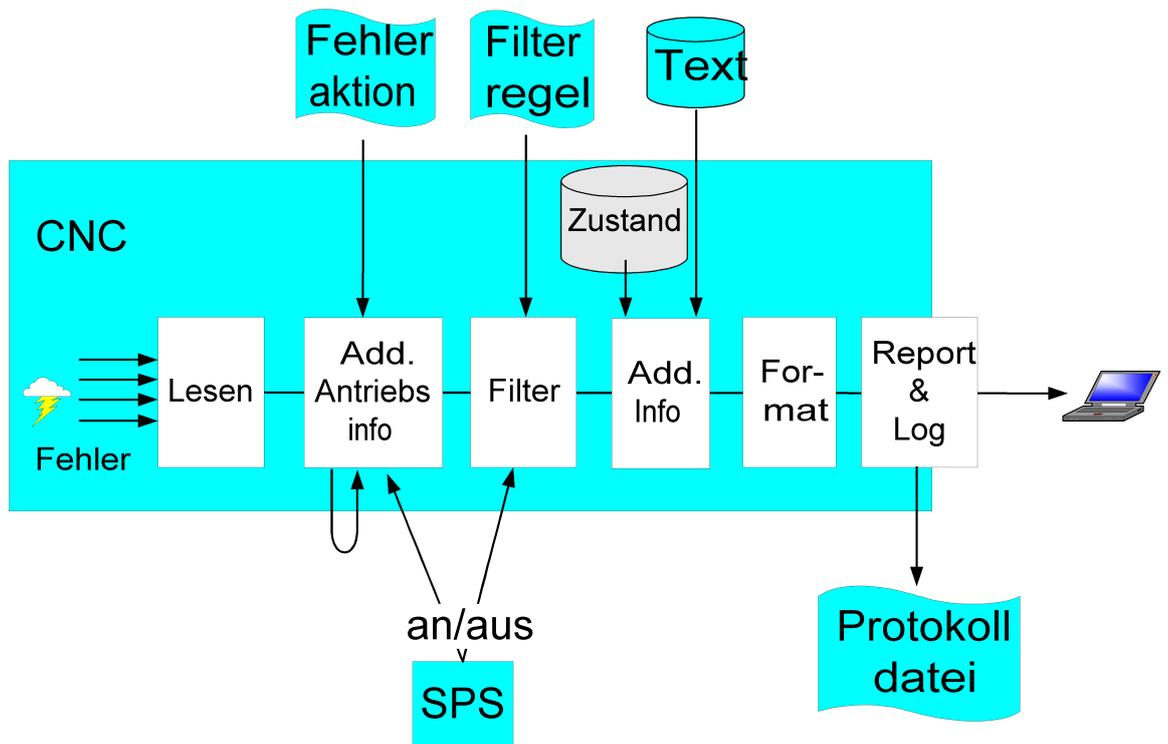


Abb. 2: Ablauf und Aufbereitung der Fehlermeldungen in der CNC

2.1 Melden und Aufzeichnen der Fehler

Die Ausgabe der Fehlermeldungen wird zentral über den Parameter P-STUP-00167 [▶ 32] festgelegt.



Hinweis

Alle gewünschten Optionen müssen bei P-STUP-00167 gesetzt werden.

Das ausschließliche setzen von NO_WARNINGS führt zu keiner Ausgabe!

Der Modus zur Steuerung der Ausgabe kann ebenfalls über CNC-Objekte gelesen und geschrieben werden.

- Lesen über `cnc_error_manager_mode_r` [▶ 50]
- Schreiben über `cnc_error_manager_mode_w` [▶ 50]

Ausgabemöglichkeiten

Grundsätzlich stehen folgende 3 parallele Ausgabemöglichkeiten zur Verfügung:

1. Print: Ausgabe über Bildschirm/Shell
2. Log: Ausgabe in Protokolldatei
3. Report: Applikationsspezifische Ausgabe

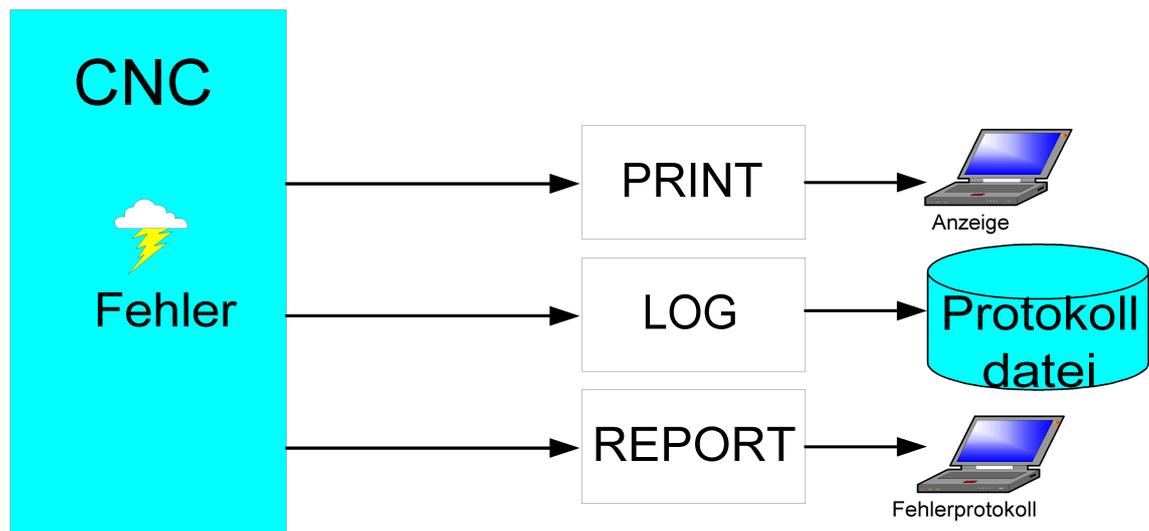


Abb. 3: Überblick der 3 Ausgabemöglichkeiten

Print

Je nach Applikation kann die Fehlerausgabe direkt auf den Bildschirm bzw. über die Shell erfolgen.

Protokollierung in Datei

Die Fehler werden in eine Protokolldatei protokolliert. Diese kann über den Parameter P-STUP-00170 [▶ 34] namentlich festgelegt und über P-STUP-00171 [▶ 34] die maximale Größe definiert werden. Sobald die Protokolldatei die definierte maximale Größe überschritten hat, wird die Originaldatei in eine Backupdatei kopiert und die Originaldatei gelöscht.

Applikationsspezifischer Report

Zusätzlich kann eine applikationsspezifische Ausgabe erfolgen.

„Unmittelbare“ Ausgabe an SPS

Neben der Ausgabe der aufbereiteten Fehlermeldung über PRINT/LOG/REPORT wird die Fehlermeldung "im Rohformat" unmittelbar nach Auftreten an die SPS gesendet. Diese Ausgabe kann über den Protokoll-Mode P-STUP-00167 [▶ 32] des Fehlermanagers bei Bedarf komplett abgeschaltet werden.

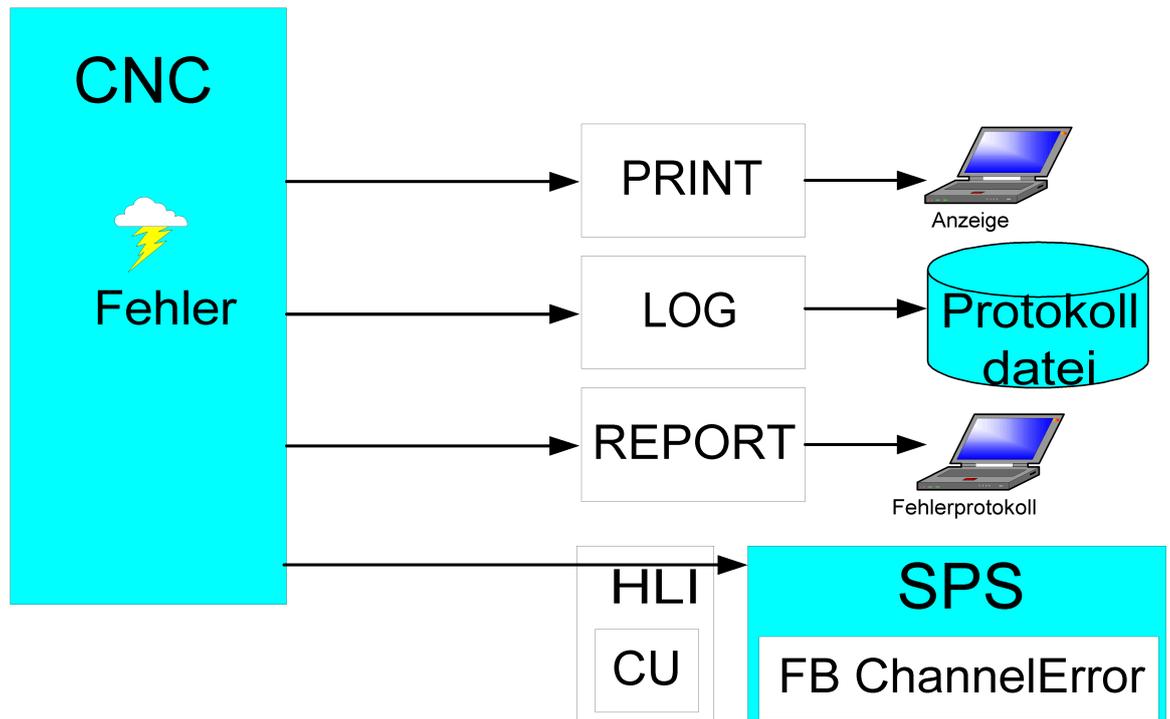


Abb. 4: Überblick der Ausgabemöglichkeiten mit Anbindung des SPS-Bausteins



Hinweis

Die Fehlerausgabe über die SPS kann nicht gefiltert werden.

Die entsprechenden Filter müssen in der SPS realisiert werden.

Falls die SPS keine weiteren Fehlermeldungen der CNC mehr auswerten muss, kann die Ausgabe an die SPS über die Control Unit (CU) auf dem HLI komplett entfallen.

```
error_protocol_mode PRINT | LOG | REPORT | SEND_NOT_TO_PLC
```

Ausgaben unter TwinCAT 3

In TwinCAT 3 ist der PRINT-Kanal an die Ausgabe zum Event-Logger angebunden:

2.1.1 Unterdrücken von Warnungen

Um den Endanwender nicht mit häufigen Warnungen zu irritieren, kann es von Vorteil sein, diese für den Anwender in der Anzeige zu unterdrücken, aber in der Protokolldatei (P-STUP-00170 [▶ 34]) zu dokumentieren.

Bei einem Reset der Steuerung wird standardmäßig eine Warnung erzeugt, diese kann aber über den Parameter P-STUP-00166 [▶ 31] unterdrückt werden. Hierfür muss die Einstellung des Protokoll-Modes geändert werden:



Programmierbeispiel

Protokoll-Mode: Filtern von Warnungen

Warnungen nur in Protokolldatei

```
error_protocol_mode LOG | PRINT | REPORT | PRINT_NO_WARNINGS | RE-  
PORT_NO_WARNINGS
```

Warnungen auch in Protokolldatei unterdrücken

```
error_protocol_mode LOG | PRINT | REPORT | PRINT_NO_WARNINGS | RE-  
PORT_NO_WARNINGS | LOG_NO_WARNINGS
```

Unterdrücken aller Warnungen: Dies beinhaltet auch das Unterdrücken der Ausgabe von Warnungen zur SPS.

```
error_protocol_mode LOG | PRINT | REPORT | NO_WARNINGS
```

Diese ist identisch zu:

```
error_protocol_mode LOG | PRINT | REPORT | PRINT_NO_WARNINGS | RE-  
PORT_NO_WARNINGS | LOG_NO_WARNINGS | SEND_TO_PLC_NO_WARNINGS
```



Programmierbeispiel

Protokoll-Mode: Filtern von Warnungen – TC3_EventLogger

Warnungen nur in Protokolldatei

```
error_protocol_mode LOG | PRINT | REPORT | TC3_EVENT_LOGGER |  
PRINT_NO_WARNINGS | REPORT_NO_WARNINGS
```

Warnungen auch in Protokolldatei unterdrücken

```
error_protocol_mode LOG | PRINT | REPORT | TC3_EVENT_LOGGER |  
PRINT_NO_WARNINGS | REPORT_NO_WARNINGS | LOG_NO_WARNINGS
```

Unterdrücken aller Warnungen: Dies beinhaltet auch das Unterdrücken der Ausgabe von Warnungen zur SPS.

```
error_protocol_mode LOG | PRINT | REPORT | TC3_EVENT_LOGGER | NO_WAR-  
NINGS
```

Diese ist identisch zu:

```
error_protocol_mode LOG | PRINT | REPORT | TC3_EVENT_LOGGER |  
PRINT_NO_WARNINGS | REPORT_NO_WARNINGS | LOG_NO_WARNINGS |  
SEND_TO_PLC_NO_WARNINGS
```

2.1.2 Ausgabebeispiele

Protokollierung in Datei

Die Ausgabe einer Fehlermeldung in der Protokolldatei sieht z.B. wie folgt aus:

```
<<-----
21.09.2018 16:03:13:571 (5616771)                               Version: V3.01.3061.3204
-----
Error   : 20232 - Unexpected 'ENDFOR'; it does not match the actual control block.
-----
Program : ..\prg\EventLogTest.nc
Path    : ..\prg\ (No: 65535)
File    : EventLogTest.nc
Block no: N10                                               Lineoffset Start/End: 5/13
Line    : N010 $ENDFOR
Position:          ^^^^^^^^
-----
Channel : (No.: 1)
Class   : SYNTAX (2)                                     Reaction : PROGRAM_ABORT (2)
=====
Config  : ZWEI_KANAL_KONFIGURIERUNG
Modul   : EW_FKT.C                               Line : 4017
BF-Type : DECODER (9)                             Commu: DEC_1 (42)           Multiple-ID: 0
Content : NC_PROGRAM (1)
----->>
```

Abb. 5: Ausgabebeispiel - Protokolldatei

2.2 Filtern von Fehlermeldungen

Die Ausgabe von Fehlermeldungen kann global, kanal- oder achsspezifisch gefiltert werden.

Filterebene erkennen

Auf welcher Ebene die Fehlermeldung oder Warnung gefiltert werden muss, ist über die Art der Meldung in der Protokolldatei ersichtlich.

Kanalfehlermeldungen oder -warnungen sind wie folgt zu erkennen:

```
<<-----
05.07.2021 08:14:56:273 (456)                               Version: V3.01.3077.6404
-----
Error   : 20418 - Kanalparameter: Spindelanzahl und Anzahl konf. Spindelachsen sti
-----
Channel : (No.: 1)
Value   : 17
Class   : WARNING (0)                                     Reaction : NO_REACTION (1)
=====
Value 1 : Actual value           : 17
Value 2 : Expected value        : 0
Value 3 : Corrected value       : 0
-----
Config  : EIN_KANAL_KONFIGURIERUNG
Modul   : DEC_ABL5.C                               Line : 6002
BF-Type : DECODER (9)                               Commu: DEC_1 (42)       Multiple-ID: 0
Content : MACHINE-DATA_SET (2)
----->>
```

Abb. 6: Meldung aus dem Kanal

Achsfehlermeldungen oder -warnungen sind wie folgt zu erkennen:

```
<<-----
05.07.2021 08:22:36:805 (397)                               Version: V3.01.3077.6404
-----
Error   : 110112 - Geschwindigkeitsoverride ueberschreitet Grenzwert.
-----
Value   : 1.235e+004                                     Axis      : Achse_1 (Axis-no: 1)
Max     : 2000
Class   : WARNING (0)                                     Reaction : NO_REACTION (1)
=====
Value 1 : Logical axis-number   : 1
Value 2 : Actual value          : 1.235e+004
Value 3 : Limiting value        : 2000
Value 4 : Corrected value       : 100
-----
Utility : Error 2045 - Fehlerhafter Index bei Arrayparameter.
Modul   : ISG_UTIL.C                               Line : 3066
-----
Config  : EIN_KANAL_KONFIGURIERUNG
Modul   : MDS_UTIL.C                               Line : 21987
BF-Type : AX_VERWALT (6)                               Commu: AXV (2)       Multiple-ID: 0
Content : MACHINE-DATA_SET (2)
----->>
```

Abb. 7: Meldung einer Achse

Eine Meldung aus dem Hochlauf hat weder einen Achs- noch einen Kanaleintrag:

```
<<-----
05.07.2021 08:20:36:015 (0)                               Version: V3.01.3077.6404
-----
Error   : 1001117 - Unbekanntes Listenelement in Liste enthalten.
-----
File    : hochtwin.lis                                   Fileoffset: 144
Line    : b_storage_size[0]                             0x200000
-----
Class   : WARNING (0)                                   Reaction : NO_REACTION (1)
=====
Utility : Error 1001117 - Unbekanntes Listenelement in Liste enthalten.
Modul   : INTPR.C                                       Line : 2096
-----
Config  : EIN_KANAL_KONFIGURIERUNG
Modul   : HOCH_UTI.C                                   Line : 582
BF-Type : SYS_ABL5 (18)                               Commu: (0)                                           Multiple-ID: 0
Content : INTERPRETE_FILE (6)
----->>
```

Abb. 8: Meldung aus dem Hochlauf

2.2.1 Programmierbeispiel Filtern von Fehlermeldungen

Die Warnung **ID 20048** soll im ersten Schritt exemplarisch gefiltert und im zweiten Schritt um einen Zusatztext erweitert werden.

Es handelt sich um eine Warnung im Kanal; folgendes ist zu parametrieren:



Programmierbeispiel

Filtern einer einzelnen Fehlermeldung

```
error_filter[0].reason 20048 (P-CHAN-00378)
(HIDE - keine Ausgabe der Meldung)
error_filter[0].conditional_action HIDE (P-CHAN-00381)
```

Mit diesen Einstellungen wird die Ausgabe der Warnung ID 20048 unterbunden.

Um diesen Filter für mehrere Meldungen wie beispielsweise den IDs 20048, 20622 und 21691 zu verwenden, müsste die Zeile wie folgt erweitert werden:

```
error_filter[0].reason 20048, 20622 , 21691
```

Es können maximal 5 Fehler-IDs in einem Filter angegeben werden.



Programmierbeispiel

Bestimmte Warnung zu Fehler verändern

```
error_filter[0].reason 20048 (P-CHAN-00378)
error_filter[0].action NONE (P-CHAN-00379 keine Aktion)

(Aktion: Ausgabe als Syntaxfehler)
error_filter[0].conditional_action F_SYNTAX (P-CHAN-00381)
```

Mit diesen Einstellungen wird aus der Warnung ID 20048 in der Ausgabe ein Fehler der Behebungs-kategorie 2.

2.3 Aktivieren von Filterregeln

Filterregeln können von der Oberfläche oder der SPS dynamisch aktiviert und deaktiviert werden. Dies ist auf Plattformebene, kanal- und achsspezifisch möglich.

Voraussetzung dafür ist, dass bei der Definition der Filterregel ein entsprechendes Bit im Datum „activation_bit“ gesetzt wird.

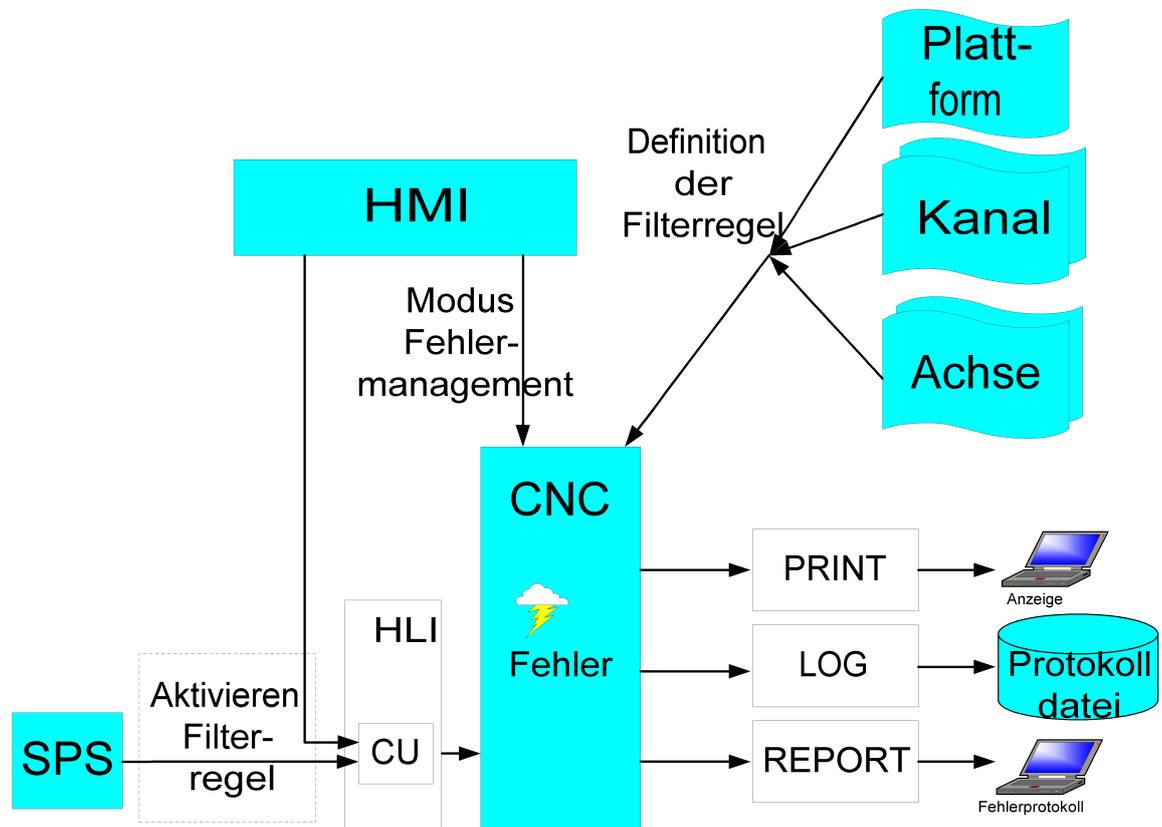


Abb. 9: Aktivieren von Filterregeln

Aktivierung über die Oberfläche

Voraussetzung dafür ist das Zusammenspiel von Parameter und CNC-Objekt.

Für die Nutzung sind folgende Paare zu betrachten:

- Plattform: P-STUP-00188 und cnc_error_filter_w [▶ 49]
- Kanalspezifisch: P-CHAN-00380 und mc_error_error_filter_w [▶ 49]
- Achsspezifisch: P-AXIS-00629 und ac_<i></i>_error_filter_w [▶ 50]

Aktivierung über SPS

Voraussetzung dafür ist das Zusammenspiel von Parameter und entsprechender Control Unit (CU).

Für die Nutzung sind folgende Paare zu betrachten:

- Plattform: P-STUP-00188 und der CU error_filter [▶ 52]
- Kanalspezifisch: P-CHAN-00380 und CU error_filter [▶ 53]
- Achsspezifisch: P-AXIS-00629 und CU error_filter [▶ 54]

2.4 Ausgeben eigener Fehlermeldungen

Der Anwender hat die Möglichkeit, im NC-Programm über den NC-Befehl #ERROR [▶ 18] eigene Fehlermeldungen auszugeben.

Die zugehörigen Fehlertexte werden in einer Textdatei hinterlegt, die über P-STUP-00169 [▶ 33] parametrisiert wird. Die Ausgabe erfolgt in die Protokolldatei, die über P-STUP-00170 [▶ 34] definiert wird.



Hinweis

Die in der Textdatei hinterlegten Fehler-IDs mit den zugehörigen Fehlertexten werden ausschließlich in die Protokolldatei geschrieben.

Der Bereich für mögliche Fehler-IDs ist 1 - 1000.



Programmierbeispiel

Kundenspezifischer Fehlertext in Protokolldatei

Inhalt der kundenspezifischen Fehlertextdatei:

```
455 Fehlertext zu ID455
```

Der Aufruf im NC-Programm ist dann wie folgt:

```
#ERROR [ID455 RC2 PV1=5 PV2=4.999 PM1=2 PM2=3]
```

Weitere Informationen zum #ERROR-Befehl : [PROG://Benutzdefinierte Fehlerausgabe [▶ 18]]

Die Ausgabe in der Protokolldatei:

```
<<-----
09.07.2021 06:33:48:225 (73934)                               Version: V3.01.3077.6404
-----
Error   : 455 - Fehlertext zu ID 455
-----
Program : -
Block no: N0                                               Lineoffset Start/End: 0/46
Position: ^
-----
Channel : (No.: 1)
Value   : 5
Class   : SYNTAX (2)                                     Reaction : PROGRAM_ABORT (2)
=====
Value 1 : Actual value           : 5
Value 2 : Error value            : 4.999
-----
Config  : EIN_KANAL_KONFIGURIERUNG
Modul   : STR_ERR.C                                       Line : 766
BF-Type : DECODER (9)                                     Commu: DEC_1 (42)      Multiple-ID: 0
Content : NC_PROGRAM (1)
----->>
```

Abb. 10: Ausgabe eigener Fehlertexte in Protokolldatei

Eigendefinierte Fehlermeldungen unter TwinCAT3 siehe: Ausgabe eigendefinierter Fehlermeldungen an HMI [▶ 21]

2.4.1 Benutzerdefinierte Fehlerausgabe (#ERROR)

Der NC-Befehl #ERROR erlaubt die Ausgabe anwenderdefinierter Fehlermeldungen, die von der übergeordneten Bedienung (GUI = Graphical User Interface) weiterverarbeitet werden. Zusätzliche Parameter bieten die Möglichkeit zur genaueren Spezifizierung des Fehlers.

Die Zuordnung zwischen Fehlernummer (ID) und Fehlertext erfolgt in einer anwenderspezifischen Datei (FCT-M7// Ausgeben eigener Fehlermeldungen [▶ 17]). Speicherort (Pfad) und Name dieser Datei werden im Parameter P-STUP-00169 [▶ 33] eingetragen.

Syntax:

```
#ERROR [ [ID=..] [RC=..] [MID=..] {PV<i>=..} {PM<i>=..} {PIV<i>=..} ]
```

ID=..	Fehlernummer: 1...1000: Der Zahlenwert bestimmt die auszugebende kundenspezifische Fehlernummer.
RC=..	Fehlerbehebungsklasse: 0: Warnung, Kein Übergang in Fehlerzustand. Fortsetzung der Programmbearbeitung. 2: Fehler, Übergang in Fehlerzustand. Per NC-RESET behebbar. 7: Fataler Fehler, Übergang in Zustand 'Systemfehler'. Neustart der Steuerung erforderlich.
MID=..	Mehrfach-ID. Zähler dient als Unterscheidungsmerkmal, wenn in einem NC-Programm mehrmals der #ERROR-Befehl mit der gleichen Fehlernummer (ID) verwendet wird. MID muss eine positive Ganzzahl sein.
PV<i>=>..	Es können maximal 5 ($1 \leq i \leq 5$) kundenspezifische Zahlenwerte (PV1...PV5) im Realformat in der Fehlermeldung mit ausgegeben werden. Ab V3.1.3080.14 bzw. V3.1.3107.48 können auch Strings ausgegeben werden, z.B. PV1="Test". Die maximale Länge ist auf 23 Zeichen beschränkt.
PM<j>=>..	Die maximal 5 ($1 \leq j \leq 5$) PM-Parameter (PM1...PM5) dienen dazu, die Bedeutung der PV-Parameter genauer zu spezifizieren. 0: IGNORE, Wert ohne Bedeutung 1: Grenzwert 2: Aktueller Wert 3: Fehlerhafter Wert 4: Erwarteter Wert 5: Korrigierter Wert 6: Logische Achsnummer 7: Antriebstyp 8: Logische Bedienelementnummer 9: Zustand 10: Transition 11: Sender 12: Klasse 13: Instanz 14: Identifikationsnummer 15: Status 16: Ringnummer 17: Satznummer 18: Unterer Grenzwert 19: Oberer Grenzwert 20: Startwert 21: Endwert
PIV<j>=>..	Die maximal 4 ($1 \leq j \leq 4$) PIV-Parameter (PIV1...PIV4) dienen zur Übermittlung zusätzlicher Informationen im Realformat.

Für nicht programmierte Parameter sind folgende Standardwerte gültig:

ID	1
RC	0
MID	0
PV1...PV5	0.0
PM1...PM5	1
PIV1...PIV4	0.0



Programmierbeispiel

Benutzerdefinierte Fehlerausgabe

```
; -----  
; Ausgabe der Warnung mit ID 100, Mehrfachkennung 10  
#ERROR [ID100 RC0 MID10]  
; ..  
; -----  
; Ausgabe des Fehlers mit ID 455 mit Parametern  
; Fehler 455 mit Parametern  
; Parameter 1 - aktueller Wert ist 1  
; Parameter 2 - fehlerhafte Wert ist 4.999  
#ERROR [ID455 RC2 PV1=5 PV2=4.999 PM1=2 PM2=3]  
; ..  
; Fehlerausgabe mit String ab V3.1.3080.14  
; Erwarteter Wert: Text-A  
; Fehlerhafter Wert Text-B  
#ERROR [ID123 RC2 PV1="Text-A" PM1=4 PV2="Text-B" PM2=3]  
;...  
; -----  
; Fataler Fehler 999  
#ERROR [ID999 RC7]
```

2.4.2 Ausgabe eigendefinierter Fehlermeldungen an HMI

Die Ausgabe von eigendefinierten Fehlermeldungen über den #ERROR [▶ 18] Befehl kann an der Oberfläche des TwinCAT3-Systems angezeigt werden. Der Bereich für die möglichen Fehler-IDs ist 1 - 1000.

Die Fehlertexte können mehrsprachig angelegt werden.



Hinweis

Eine Verknüpfung der angegebenen Fehlertexte aus der Datei P-STUP-00169 mit der Ausgabe an die HMI findet nicht statt.

Die erforderlichen Fehlertexte müssen in beiden Dateien integriert werden.

Die Fehlertexte für die Ausgabe an HMI müssen in der Datei

C:\TwinCAT\3.1\Target\Resource\TcCncUserEvents.xml

integriert werden.



Beispiel

Integration einer Meldung in TcCncUserEvents.xml

```
<Source>
<GUID>{3FD56AAE-0711-4359-89A5-3E1ECCCC634E}</GUID>
  <Id>650</Id>
  <Name>TwinCAT CNC User Errors</Name>
  <Event>
    <!-- User specific error messages: ID range 1 - 1000 -->
    <Id>1</Id>
    <Message LcId="1033">Error message ID1 (#ERROR[ID1])</Message>
    <Message LcId="1031">Fehlermeldung ID1 (#ERROR[ID1])</Message>
  </Event>
  <Event>
    <Id>2</Id>
    <Message LcId="1033">Error message ID2 (#ERROR[ID2])</Message>
    <Message LcId="1031">Fehlermeldung ID2 (#ERROR[ID2])</Message>
  </Event>
  <Event>
    <Id>1000</Id>
    <Message LcId="1033">Error message ID1000 (#ERROR[ID1000])</Message>
    <Message LcId="1031">Fehlermeldung ID1000 (#ERROR[ID1000])</Message>
  </Event>
</Source>
```

Abb. 11: Codeauszug – Ausgangslage

Exemplarisch wird in der Datei der Fehler mit der ID 455 integriert:

```
<Source>
<GUID>{3FD56AAE-0711-4359-89A5-3E1ECCCC634E}</GUID>
  <Id>650</Id>
  <Name>TwinCAT CNC User Errors</Name>
  <Event>
    <!-- User specific error messages: ID range 1 - 1000 -->
    <Id>1</Id>
    <Message LcId="1033">Error message ID1 (#ERROR[ID1])</Message>
    <Message LcId="1031">Fehlermeldung ID1 (#ERROR[ID1])</Message>
  </Event>
  <Event>
    <Id>2</Id>
    <Message LcId="1033">Error message ID2 (#ERROR[ID2])</Message>
    <Message LcId="1031">Fehlermeldung ID2 (#ERROR[ID2])</Message>
  </Event>
  <Event>
    <Id>455</Id>
    <Message LcId="1033">Error message ID455 (#ERROR[ID455])</Message>
    <Message LcId="1031">Fehlermeldung ID455 (#ERROR[ID455])</Message>
  </Event>
  <Event>
    <Id>1000</Id>
    <Message LcId="1033">Error message ID1000 (#ERROR[ID1000])</Message>
    <Message LcId="1031">Fehlermeldung ID1000 (#ERROR[ID1000])</Message>
  </Event>
</Source>
```

Abb. 12: Codeauszug mit integriertem Fehlertext



Versionshinweis

Ab TwinCAT-Version 4022.23 ist nach Integration des eigendefinierten Fehlertextes ein erneutes Aktivieren der Konfiguration erforderlich.

Bei TwinCAT-Versionen < 4022.23 ist ein Neustart des Rechners erforderlich

2.5 Fehlerausgabe TwinCAT3

Die Ausgabe von Fehlern unter TwinCAT3 erfolgt

- über die SPS oder
- direkt über den TwinCAT-Eventlogger.

Das nachfolgende Übersichtsbild verdeutlicht den Zusammenhang des Eventloggers:

Architecture: Sending / Receiving BECKHOFF

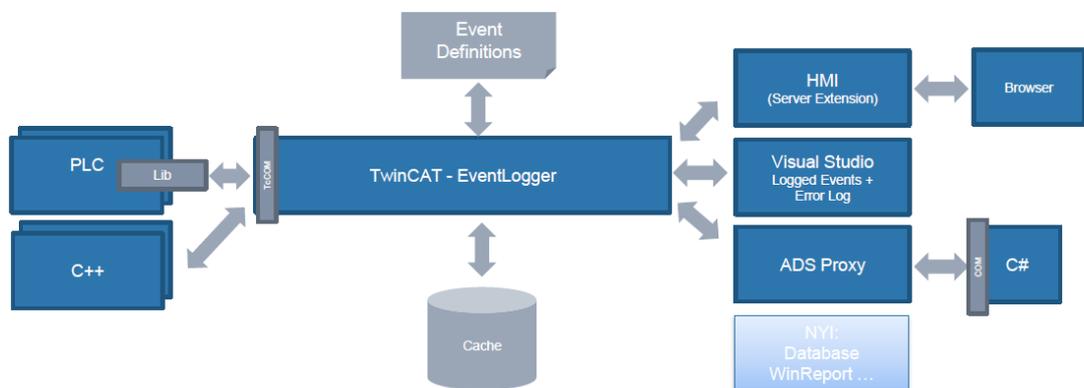


Abb. 13: Architektur- Übersicht TwinCAT-Eventlogger

Ausgaben unter TwinCAT 3

In TwinCAT 3 ist der PRINT-Kanal an die Ausgabe zum Event-Logger angebunden:

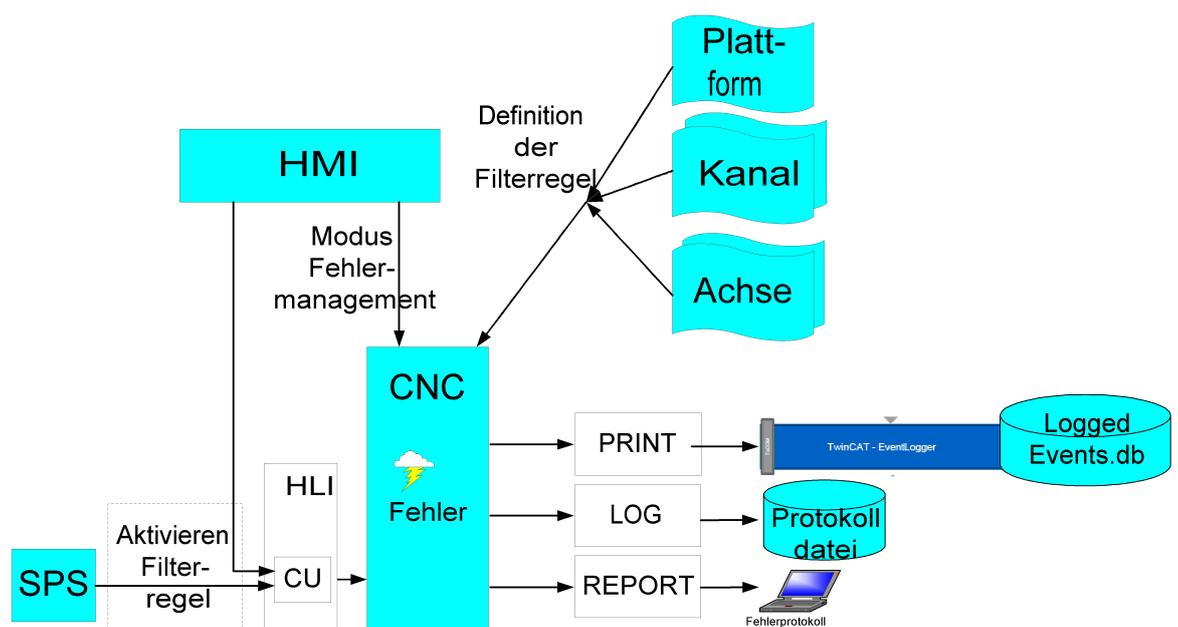


Abb. 14: Übersicht mit TwinCAT-Eventlogger

Standardmäßige Parametrierung

Standardmäßig erfolgt die Ausgabe der Fehler an die SPS mit dem Loggen über den ursprünglichen Funktionsbaustein ChannelError an den Eventlogger.

Bei einer Migration einer bestehenden CNC Konfiguration von TwinCAT2 nach TwinCAT3 ist hierfür in den Hochlaufparametern bezüglich des Fehlermanagements keine Anpassung in P-STUP-00167 [► 32] (error_protocol_mode) erforderlich.

Parameter	Parametername	Einstellung
P-STUP-00167	error_protocol_mode	PRINT LOG REPORT
P-STUP-00168	error_text_of_id	<TC3-Install>\components\mc\cnc\diagnostics\err_text_version_eng.txt
P-STUP-00169	error_text_user_of_id	nicht belegt
P-STUP-00170	error_log_file_name	<TC3-Install> \components\mc\cnc\diagnostics\error.log
P-STUP-00171	error_log_file_max_size	100000 # maximum length in bytes
P-STUP-00172	error_plc_wait_cycles	5 # Wartezyklen [IPO ticks]
P-STUP-00173	error_ao_name	(nicht belegt)

Fehlertexte

Die sprachspezifischen Fehlertexte werden bei einem Release der CNC über die Datei TcCncErrors.xml automatisch installiert.

2.5.1 Ausgabe nur über SPS

Soll die Ausgabe an den Event-Logger nur über den ChannelError() Funktionsbaustein (FB) der SPS erfolgen, so kann dies durch Schreiben des CNC-Objektes „log errors [▶ 51]“ angewählt werden. Die Aktivierung findet normalerweise beim Start der SPS statt, wenn diese das Loggen der Fehlermeldungen übernimmt.

Ausgabe über den FB ChannelError()

Werden die Fehler über die SPS-Schnittstelle an die SPS gemeldet, so können Fehler innerhalb der SPS ausgewertet werden und über einen Baustein FB ChannelError() an den Event-Logger gemeldet werden. Neben der Ausgabe des Fehlers kann die SPS hier evtl. unmittelbare Reaktionen auf einen individuellen Fehler ausführen.

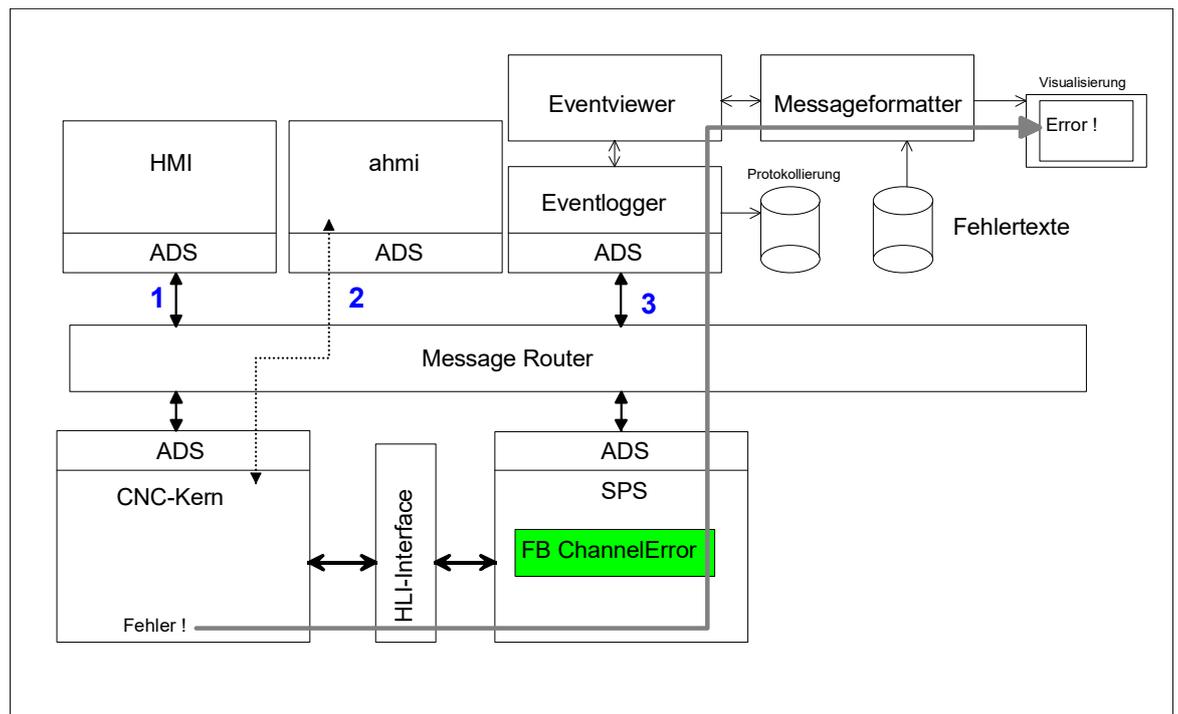


Abb. 15: FB ChannelError()- Platzierung im System

Diese Ausgabe ist identisch mit der von TwinCAT2.

2.5.2 Direkte Ausgabe an Eventlogger

Um die Ausgabe direkt an den „neuen“ Eventlogger zu nutzen muss in P-STUP-00167 [▶ 32] das Schlüsselwort TC3_EVENT_LOGGER angegeben werden. Die Fehler werden dann direkt von der CNC im neuen Format geloggt und nicht mehr über den SPS-Baustein ChannelError(). Eine Ausgabe der Fehler an die SPS erfolgt weiterhin.

```
error_protocol_mode PRINT | LOG | REPORT | TC3_EVENT_LOGGER
```

Falls die SPS keine Fehlermeldungen der CNC mehr auswerten muss kann die Ausgabe der Fehler an die SPS zu unterbunden werden durch:

```
error_protocol_mode PRINT | LOG | REPORT | TC3_EVENT_LOGGER |  
SEND_NOT_TO_PLC
```

Alternativ kann der Modus auch über das CNC-Objekt cnc_error_manager_mode_w [▶ 50] entsprechend gesetzt werden

Ausgabe von Meldungen aus dem NC-Programm an den Eventlogger

Funktionalität verfügbar ab CNC-Version V3.1.3080.11

Mit dem NC-Befehl #MSG kann der Anwender aus dem NC-Programm heraus Meldungen an den Eventlogger von TwinCAT3 senden. Die Meldungen werden kanalspezifisch ausgegeben.

Diese erfolgt über die Warnung ID 1035. Die Meldungen können sowohl direkt nach der Dekodierung als auch synchron zur Bearbeitung des Interpolators (SYN) ausgegeben werden.

Die Meldungen haben für den Eventlogger die Priorität INFO. Die Meldungen sind ohne Quittierung, wird eine Meldung mit einem Modus mit Quittierung (ACK oder SYN_ACK) programmiert, so wird die Warnung ID 1036 ausgegeben. Diese Meldung wird ohne Quittierungsanforderung versendet.

Der Anwender kann die Meldungen über die Programmierung explizit löschen.

```
#MSG SYN EVENT_LOGGER[""]
```

Bei CNC-Reset oder beim Herunterfahren der Steuerung werden die Meldungen ebenfalls gelöscht.



Programmierbeispiel

Meldungen an den Eventlogger

```
%Eventlogger_test.nc
N020 P1 = 123
N030 #MSG EVENT_LOGGER["Asynchrone Meldung %d", P1]
N040 #MSG SYN EVENT_LOGGER["Synchrone Meldung"]
      (Löschen der Meldungen)
N050 #MSG SYN EVENT_LOGGER[""]
M30
```

Um die nachfolgende Ausgabe in der Beckhoff-Oberfläche mit den dargestellten NC-Befehlszeilen zu erhalten, muss die Ressourcendatei TcCncErrors.xml modifiziert werden.

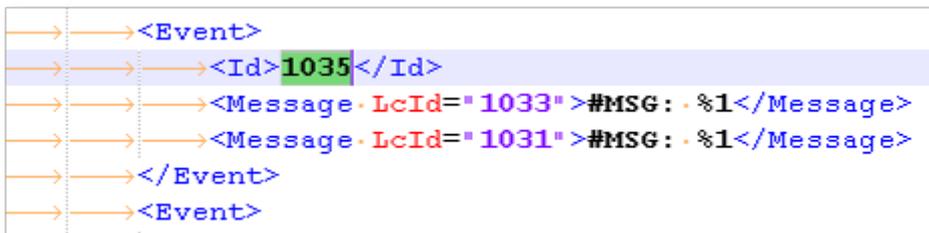


Abb. 16: Modifikation der Datei TcCncErrors.xml

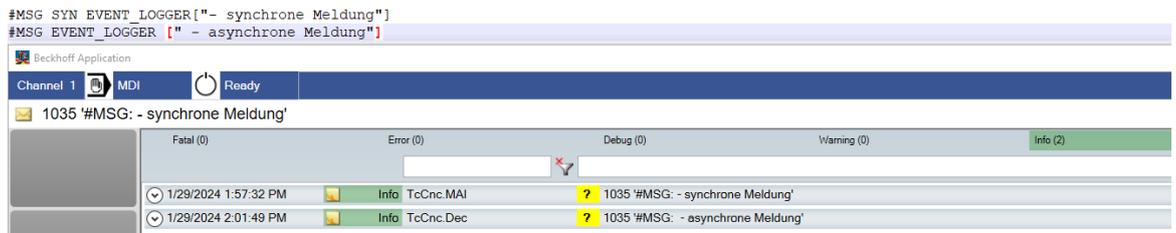


Abb. 17: Ausgabe in Beckhoff-Oberfläche

2.5.2.1 Ausgabeformat-Eventlogger

Im Visualstudio kann die Ausgabe des Eventloggers in einem separaten Fenster dargestellt werden:

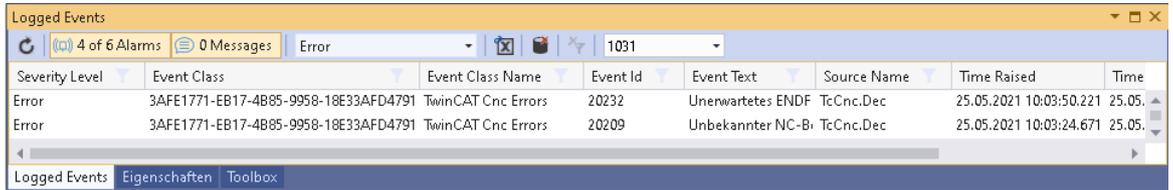


Abb. 18: Eventlogger-Output

Übersicht-Ausgabeformat des Eventloggers

Reitername		Bedeutung
Severity Level	Klassifizierung	Warning (0) / Error (1-6) / Critical (7)
EventClassName	Name der Fehlerklasse	TwinCAT Cnc Errors
SourceName	Fehlerquelle	TcCNC Config / Diag / PosCtrl / PathPrep / AxesMgr / Commu / DrvMgr / Dec / TRC / Filter / FileMgr / SAI / IPO / Manual / SysMgr / Application
EventId	ID des Events	CNC-spezifische ID der aufgetretenen Meldung
Text	Fehlertext	Multilingualer Fehlertext
Time Raised	Auslösezeitpunkt	Datum und Uhrzeit
Json Attributes	Json Attribute	

Json Attribute

Zusätzliche Parameter einer Fehlermeldung werden als Key/Value-Paare im Json-Format ausgegeben. Insbesondere stehen somit detailliertere Fehlerinformationen zum NC-Kanal oder Bezug zum NC-Programm zur Verfügung.

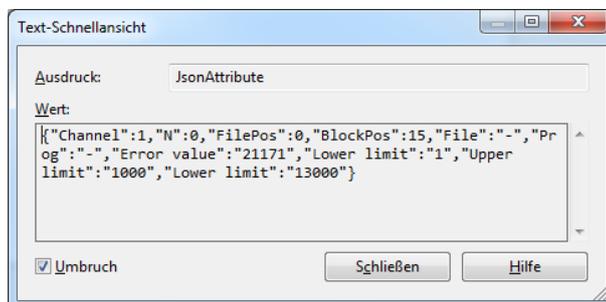


Abb. 19: Json-Attribute Output

2.5.3 Ausgabebeispiel an HMI

Die im Event-Logger aufgezeichneten Fehlermeldungen werden in der Oberfläche entsprechend dargestellt:

The screenshot shows the Beckhoff Application interface. At the top, there's a status bar with 'Channel 1', 'MDI', 'Error', 'TwinCAT', 'PLC', '09.07.2021', and the Beckhoff logo. Below this, a red banner displays the error message: **20131 'Unknown G-function.'** with a yellow question mark icon and the text 'Double click here'. The main area is divided into several sections: a table for axis positions, 'Kanal 1 Override' for feed rate (F 0) and spindle speed (S), and 'Technologie Daten Kanal 1' showing G and M codes. The bottom of the screen features a navigation bar with icons for Automatic, Manual, MDI, Mode, Start, Stop, Reset, GCodeView, Zerooffsets, Tools, and back.

AxisName	CurrentPosition (PCS) ActivePosition (PCS)	EndPosition (PCS) ActiveFeedrate	ActivePosition (ACS) ActivePosition (ACS)
X mm	10.00 10.00	10.00 0.00	10.00 10.00
Y mm	0.00 0.00	0.00 0.00	0.00 0.00
Z mm	0.00 0.00	0.00 0.00	0.00 0.00
X2 mm	0.00 0.00	0.00 0.00	0.00 0.00

Abb. 20: Bildschirmausgabe der Fehlermeldung an der Beckhoff-HMI

The screenshot shows the Beckhoff Application interface after a double-click. The error banner is still present. Below it, an event log is displayed with columns for severity, time, source, message, and status. The log shows three entries: two warnings and one error.

Fatal (0)	Error (1)	Debug (0)	Warning (2)	Info (0)	Undef (0)		
			7/9/2021 6:54:08 AM	WARNING	TcCnc.AxesMgr	110112 'Speed override exceeds limit.' [X]	?
			7/9/2021 6:55:52 AM	WARNING	TcCnc.Dec	20048 'Centre point coordinate within linear motion block ignored.'	? #
			7/9/2021 6:56:53 AM	ERROR	TcCnc.Dec	20131 'Unknown G-function.'	? #

Abb. 21: Bildschirmausgabe nach Doppelklick

3 Parameter

3.1 Übersicht

Hochlaufparameter

ID	Parameter	Beschreibung
P-STUP-00166	no_error_message_at_reset	Protokollierung eines CNC-Resets
P-STUP-00167	error_protocol_mode	Logging Modus
P-STUP-00168	error_text_of_id	Name der Datei für Fehlermeldungstexte
P-STUP-00169	error_text_user_of_id	Name der Datei für anwenderspezifische Fehlermeldungstexte
P-STUP-00170	error_log_file_name	Name der Error-Logdatei
P-STUP-00171	error_log_file_max_size	Maximale Größe der Error-Logdatei
P-STUP-00172	error_plc_wait_cycles	Wartezyklen vor Auswertung der SPS Aktivierung
P-STUP-00173	error_ao_name	Zusätzlicher Beschreibungstext (AO-Name)
P-STUP-00186	error_filter[i].reason	Identifikation des Fehlers (Ursache)
P-STUP-00187	error_filter[i].action	Fehleraktion, die ausgeführt werden soll
P-STUP-00188	error_filter[i].conditional_activation	Aktivierung über SPS / HMI
P-STUP-00189	error_filter[i].conditional_action	Aktion die erst nach Freigabe durch SPS / HMI ausgeführt werden soll
P-STUP-00190	error_filter[i].conditional_param	Zusätzlicher Vergleichsparameter im Falle einer Antriebsfehlermeldung
P-STUP-00191	error_filter[i].conditional_output	Individuelle zusätzliche Fehlerinformation zur Ausgabe
P-STUP-00200	error_text_cycles_of_id	Name der Datei für Fehlermeldungstexte der CNC-Zyklen

Kanalparameter

ID	Parameter	Beschreibung
P-CHAN-00338	mdi_log_file	Name der Handsatz-Logdatei
P-CHAN-00339	mdi_log_file_max_size	Maximale Größe der Handsatz-Logdatei
P-CHAN-00378	error_filter[i].reason	Identifikation des Fehlers (Ursache)
P-CHAN-00379	error_filter[i].action	Fehleraktion, die ausgeführt werden soll
P-CHAN-00380	error_filter[i].conditional_activation	Aktivierung über SPS / HMI
P-CHAN-00381	error_filter[i].conditional_action	Aktion die erst nach Freigabe durch SPS / HMI ausgeführt werden soll

ID	Parameter	Beschreibung
P-CHAN-00382	error_filter[i].conditional_param	Zusätzlicher Vergleichsparameter im Falle einer Antriebsfehlermeldung
P-CHAN-00383	error_filter[i].conditional_output	Individuelle zusätzliche Fehlerinformation zur Ausgabe

Achsparemeter

ID	Parameter	Beschreibung
P-AXIS-00627	error_filter[i].reason	Identifikation des Fehlers (Ursache)
P-AXIS-00628	error_filter[i].action	Fehleraktion, die ausgeführt werden soll
P-AXIS-00629	error_filter[i].conditional_activation	Aktivierung über SPS / HMI
P-AXIS-00630	error_filter[i].conditional_action	Aktion die erst nach Freigabe durch SPS / HMI ausgeführt werden soll
P-AXIS-00631	error_filter[i].conditional_param	Zusätzlicher Vergleichsparameter im Falle einer Antriebsfehlermeldung
P-AXIS-00632	error_filter[i].conditional_output	Individuelle zusätzliche Fehlerinformation zur Ausgabe

3.2 Beschreibung

3.2.1 Hochlaufparameter

P-STUP-00166	Protokollierung eines CNC-Resets als Events in der Fehlermeldungs Ausgabe
Beschreibung	Der Parameter definiert, ob der durch den Anwender ausgelöste CNC-Reset als Ereignis in das Protokoll der Fehlermeldungen mit aufgenommen wird. Bei einem CNC-Reset werden die bisher aufgetretenen Fehlermeldungen quittiert. Dies findet immer statt, unabhängig von der Einstellung von P-STUP-00166.
Parameter	no_error_message_at_reset
Datentyp	BOOLEAN
Datenbereich	0: ein CNC-Reset wird als Warnung ID 270076 in der Fehlermeldungs Ausgabe protokolliert 1 ein CNC-Reset wird nicht protokolliert
Dimension	----
Standardwert	0
Anmerkungen	

P-STUP-00167	Protokollmodus der Fehlerausgabe	
Beschreibung	Mit diesem Parameter kann die Ausgabe und der Umfang der Fehlerausgabe gesteuert werden.	
Parameter	error_protocol_mode	
Datentyp	STRING	
Datenbereich	Kennung	Bedeutung
	FILTER_OFF	Es werden keine Filter ausgewertet
	VERBOSE	Erweiterte interne Diagnose
	WITHOUT_ERROR_MANAGER	Direkte Ausgabe ohne Fehlermanagement
	PRINT	Ausgabe als Print ausführen
	LOG	Ausgabe auf Log-Datei protokollieren
	REPORT	Ausgabe auf Log-Datei protokollieren
	SEND_TO_PLC_NONE	Ausgabe an die SPS unterdrücken
	PRINT_EXTENDED	Erweiterte Print-Ausgabe
	LOG_EXTENDED	Erweiterte Log-Ausgabe
	REPORT_EXTENDED	Erweiterte applikationsspezifische Ausgabe
	PRINT_NO_WARNINGS	Warnungen werden in der Print-Ausgabe unterdrückt
	LOG_NO_WARNINGS	Warnungen werden in der Log-Ausgabe unterdrückt
	REPORT_NO_WARNINGS	Warnungen werden in der Report-Ausgabe unterdrückt
	SEND_TO_PLC_NO_WARNINGS	Warnungen an SPS unterdrücken
	STARTUP_NO_WARNINGS	Warnungen während Steuerungshochlauf unterdrücken
	NO_WARNINGS	Unterdrücken aller Warnungen
	TC3_EVENT_LOGGER	Ausgabe an TC3-Event-Logger
	TC3_EVENT_LOGGER_CONFIRMED	Ausgabe an TC3-Event-Logger, automatische Bestätigung (Zustand Confirmed) beim Löschen der Fehlermeldung
TC3_EVENT_LOGGER_NO_CONFIRMATION	Ausgabe an TC3-Event-Logger ohne eine Bestätigung (Confirmation) anzufordern	
Dimension	----	
Standardwert	LOG PRINT REPORT	
Anmerkungen	<p>Hinweis:</p> <p>Um beispielsweise Warnungen bei der Print-Ausgabe zu unterdrücken muss der gesamte Modus entsprechend gesetzt werden.</p> <p>error_protocol_mode LOG PRINT REPORT PRINT_NO_WARNINGS</p>	

P-STUP-00168	Name der Datei für Fehlermeldungstexte
Beschreibung	<p>Name der Datei, welche die zur ID (Fehlernummer) gehörigen Fehlermeldungstexte enthält. Diese können zur Ausgabe in die Log-Datei verwendet werden. Anhand dieser Datei erfolgt die Zuordnung einer Fehlernummer zum zugehörigen Fehlermeldungstext.</p> <p>Die Datei enthält für jede Fehler-ID eine Zeile nach folgendem Format :</p> <p><Error-ID> TABULATOR <Error-Text></p> <p>Wird keine Datei angegeben, so wird die Standarddatei 'err_text_version_eng.txt' angenommen.</p>
Parameter	error_text_of_id
Datentyp	STRING
Datenbereich	Maximal 256 Zeichen
Dimension	----
Standardwert	err_text_version_eng.txt
Anmerkungen	

P-STUP-00169	Name der Datei für anwenderspezifische Fehlermeldungstexte
Beschreibung	<p>Vergleichbar zu Standardfehlertexten (siehe P-STUP-00168) kann der Anwender in dieser Datei anwenderspezifische Texte angeben. Diese Texte werden für Fehler-IDs im Bereich [1;1000], die vom Anwender über den NC-Befehl #ERROR [► 18] selbst definieren kann, und für Fehler der McCOM-Schnittstellen verwendet. Anhand dieser Datei erfolgt die Zuordnung einer Fehlernummer zum zugehörigen anwenderspezifischen Fehlermeldungstext.</p> <p>Die Datei enthält für jede Fehler-ID eine Zeile nach folgendem Format :</p> <p><Error-ID> TABULATOR <Anwenderspezifischer-Error-Text></p> <p>Die Konfiguration des Dateinamens erfolgt mit relativer oder absoluter Pfadangabe. Weitere Informationen siehe auch (FCT-M7// Ausgeben eigener Fehlermeldungen [► 17]).</p>
Parameter	error_text_user_of_id
Datentyp	STRING
Datenbereich	Maximal 256 Zeichen
Dimension	----
Standardwert	*
Anmerkungen	<p>* Hinweis: Der Standardwert der Variablen ist ein Leerstring.</p> <p>Für die Fehlerwerte 292030- 292033 (ERR_KIN_TRAFO_CONFIG/ -INITIALIZE/ -FORWARD/ -BACKWARD) werden die zurückgegebenen Fehler-IDs der McCOM-Methoden aufgelöst.</p>

P-STUP-00170	Name der Error-Logdatei
Beschreibung	Name der Error-Logdatei (mit relativer oder absoluter Pfadangabe). Wird kein vollständiger Name angegeben, so wird keine Logdatei erzeugt und die Fehlermeldung ID 296000 ausgegeben. Wenn der Parameter nicht konfiguriert ist, dann wird die Error-Logdatei mit dem Standarddateinamen erzeugt.
Parameter	error_log_file_name
Datentyp	STRING
Datenbereich	Maximal 256 Zeichen
Dimension	----
Standardwert	<TwinCATInstallation>\Components\Mc\CNC\Diagnostics\error.log
Anmerkungen	Der Standarddateiname und der zugehörige Pfad sind applikationsabhängig.

P-STUP-00171	Maximale Größe der Error-Logdatei in Byte
Beschreibung	Der Parameter definiert die maximale Größe der Error-Logdatei.
Parameter	error_log_file_max_size
Datentyp	SGN32
Datenbereich	> 0 : Maximale Größe der Error-Logdatei. Wird diese Größe überschritten, so wird die Originaldatei in eine Backupdatei kopiert (Erweiterung <name>.bak) und der Inhalt der Originaldatei gelöscht. == 0 : Es wird keine Backupdatei angelegt.
Dimension	----
Standardwert	100000
Anmerkungen	

P-STUP-00172	Wartezyklen vor Auswertung der PLC Aktivierung
Beschreibung	Wartezyklen in CNC-Ticks nach Auftreten eines Fehlers, bis die Aktivierungsmaske der SPS für die Filterregeln ausgewertet wird.
Parameter	error_plc_wait_cycles
Datentyp	UNS32
Datenbereich	0 ... MAX(UNS32)
Dimension	----
Standardwert	-
Anmerkungen	

P-STUP-00173	Zusätzlicher Beschreibungstext (AO-Name)
Beschreibung	Beschreibungstext (Architecture Object), welcher bei einer Fehlermeldung zusätzlich angefügt wird.
Parameter	error_ao_name
Datentyp	STRING
Datenbereich	Maximal 83 Zeichen
Dimension	----
Standardwert	*
Anmerkungen	* Hinweis: Der Standardwert der Variablen ist ein Leerstring.

P-STUP-00186	Fehlerursache
Beschreibung	<p>Die einzelnen Fehlerkennungen können als Nummern oder Texte aufgelistet werden, wobei die gesamte Zeile folgender Syntax entsprechen muss:</p> <pre>(number text) { , (number text) }</pre> <p>mit:</p> <p>number := CNC-Fehlernummer</p> <p>text := " fehlerspezifischer Text "</p> <p>Beispiel:</p> <pre>error_filter[0].reason "D012:", 123000, 123001</pre> <p>Wird ein Fehler gemeldet, so wird in den definierten Plattform-/ Kanal-/ Achsfiltern nachgesehen, ob hierfür eine benutzerspezifische Filterregel definiert ist.</p>
Parameter	error_filter[i].reason mit i = 0 ... 3 (Maximale Anzahl der Filter, applikationsspezifisch)
Datentyp	STRING
Datenbereich	Maximal 96 Zeichen
Dimension	----
Standardwert	*
Anmerkungen	* Hinweis: Der Standardwert der Variablen ist ein Leerstring.

P-STUP-00187	Fehleraktion
Beschreibung	Aktion, die bei Auftreten des entsprechenden Fehlers durchgeführt werden soll.
Parameter	error_filter[i].action mit i = 0 ... 3 (Maximale Anzahl der Filter, applikationsspezifisch)
Datentyp	STRING
Datenbereich	ACTION = NONE DRIVE_STATE_REQ PRE_RUN_STATE_REQ RUN_STATE_REQ NONE: Keine Aktion DRIVE_STATE_REQ: Auslesen des Antriebsstatus PRE_RUN_STATE_REQ: Fehler während Hochlauf des Steuerungsbusses in PRE-Run-Zustand RUN_STATE_REQ: Fehler während Hochlauf des Steuerungsbusses in Run-Zustand
Dimension	----
Standardwert	*
Anmerkungen	Für SERCOS-Antriebs-Profile: DRIVE_STATE_REQ: S-0-0095 diagnostic PRE_RUN_STATE_REQ: S-0-0021: list of unknown operation data in CP2 -> CP3, command 127 RUN_STATE_REQ: S-0-0022: list of unknown operation data in CP3 -> CP4, command 128 Für ProfiDrive-Profile: <all actions> Parameter 945 Für CANopen-Profile <all actions> Parameter ID603F * Hinweis: Der Standardwert der Variablen ist ein Leerstring.

P-STUP-00188	Bedingte Aktivierung
Beschreibung	Beim Setzen des entsprechenden Bits über die Oberfläche oder die SPS (HLI::ControlUnit-Aktivieren der Fehlerfilterregeln - Plattform [► 52]) wird diese Filterregel aktiviert.
Parameter	error_filter[i].conditional_activation mit i = 0 ... 3 (Maximale Anzahl der Filter, applikationsspezifisch)
Datentyp	UNS32
Datenbereich	32 Bit
Dimension	----
Standardwert	0
Anmerkungen	Parametrierbeispiel: <i>error_filter[0].conditional_activation 0x2</i> Ein Aktivierungsbit = 0 bedeutet, dass die Aktion immer durchgeführt wird.

P-STUP-00189	Bedingte Aktion
Beschreibung	Aktion, die bei Auftreten des entsprechenden Fehlers und beim Zutreffen der Bedingung durchgeführt werden soll.
Parameter	error_filter[i].conditional_action mit i = 0 ... 3 (Maximale Anzahl der Filter, applikationsspezifisch)
Datentyp	STRING
Datenbereich	<p>ACTION = NONE ([HIDE] [FORCE])</p> <p>FORCE = F_WARNING F_SYNTAX F_ERROR F_SEVERE F_FATAL</p> <p>HIDE = [HIDE] [HIDE_LOG] [HIDE_PRINT] [HIDE_REPORT]</p> <p>NONE: keine Aktion</p> <p>HIDE: Jede Fehlerausgabe unterdrücken</p> <p>HIDE_LOG: Fehlerausgabe in Error-Log-Datei wird unterdrückt</p> <p>HIDE_DISPLAY: Fehlerausgabe wird unterdrückt</p> <p>HIDE_REPORT: Applikationsspezifische Fehlerausgabe wird unterdrückt</p> <p>F_WARNING: Fehler wird als WARNING ausgegeben (Behebungsklasse = 0)</p> <p>F_SYNTAX: Fehler wird als Syntaxfehler ausgegeben (Behebungsklasse = 2)</p> <p>F_ERROR: Fehler durch NC-Programm oder andere Bedienaktion (Fehlerbehebungsklasse = 5)</p> <p>F_SEVERE: Schwerer Fehler, erfordert Warmstart, (Behebungsklasse = 6)</p> <p>F_FATAL: Schwerer Fehler, erfordert kompletten Kaltstart (Behebungsklasse = 7)</p>
Dimension	----
Standardwert	*
Anmerkungen	* Hinweis: Der Standardwert der Variablen ist ein Leerstring.

P-STUP-00190	Bedingter Filteraktivierung
Beschreibung	<p>Die einzelnen Fehlerkennungen können als Nummern oder Texte aufgelistet werden, wobei die gesamte Zeile folgender Syntax entsprechen muss:</p> <p>(number text) { , (number text) }</p> <p>mit:</p> <p>number := CNC-Fehlernummer</p> <p>text := " fehlerspezifischer Text "</p>
Parameter	error_filter[i].conditional_param mit i = 0 ... 3 (Maximale Anzahl der Filter, applikationsspezifisch)
Datentyp	STRING
Datenbereich	Maximal 96 Zeichen
Dimension	----
Standardwert	*
Anmerkungen	<p>Parametrierbeispiel:</p> <p><i>error_filter[0].conditional_param "D012:", 123, 1001</i></p> <p>Individuelle Filtertexte werden nur beim Auslesen des SERCOS-Antriebsfehlers S95 geprüft. Fehlernummer werden nur bei SERCOS-Antriebsfehler (S21 und S22) und bei ProfiDrive-Antriebsfehler (Parameter 945) geprüft.</p> <p>* Hinweis: Der Standardwert der Variablen ist ein Leerstring.</p>

P-STUP-00191	Ausgabe einer zusätzlichen Fehlerinformation
Beschreibung	<p>Dieser Text wird im Falle des Zutreffens der Filterbedingung transparent über die CNC_ERROR_INFO-Datenstruktur weitergereicht. D.h. hierüber hat der Anwender die Möglichkeit, bedingt eine zusätzliche Fehlerinformation mit auszugeben.</p>
Parameter	error_filter[i].conditional_output mit i = 0 ... 3 (Maximale Anzahl der Filter, applikationsspezifisch)
Datentyp	STRING
Datenbereich	Maximal 32 Zeichen
Dimension	----
Standardwert	*
Anmerkungen	* Hinweis: Der Standardwert der Variablen ist ein Leerstring.

P-STUP-00200	Name der Datei für Fehlermeldungstexte der CNC-Zyklen
Beschreibung	Name der Datei, welche die zur ID (Fehlernummer) gehörigen Fehlermeldungstexte der CNC-Zyklen enthält. Diese können zur Ausgabe in die Log-Datei verwendet werden. Anhand dieser Datei erfolgt die Zuordnung einer Fehlernummer zum zugehörigen Fehlermeldungstext. Die Datei enthält für jede Fehler-ID eine Zeile nach folgendem Format: <Error-ID> TABULATOR <Error-Text>
Parameter	error_text_cycles_of_id
Datentyp	STRING
Datenbereich	Maximal 256 Zeichen
Dimension	----
Standardwert	err_text_cycles_eng.txt
Anmerkungen	

3.2.2 Kanalparameter

P-CHAN-00338	Name der Handsatz-Logdatei
Beschreibung	Wird der Name angegeben, so wird jedes Handsatzkommando des NC-Kanals in diese Datei protokolliert. Neben späterer Diagnosefähigkeit wird diese Datei auch für die Fehleranzeige genutzt. D.h. kommt es innerhalb des Handsatzes zu einem CNC-Fehler, so wird in der Fehlermeldung der kommandierte Handsatz gleich angezeigt.
Parameter	mdi_log_file
Datentyp	STRING
Datenbereich	Maximal 256 Zeichen
Dimension	----
Standardwert	*
Anmerkungen	* Hinweis: Der Standardwert der Variablen ist ein Leerstring.

P-CHAN-00339	Maximale Größe der Handsatz-Logdatei
Beschreibung	Da die Protokolldatei bei jedem neuen Handsatz anwächst, kann diese über diese Angabe in der Größe begrenzt werden. Wird die Größe der Protokolldatei überschritten, so wird diese vor dem Protokollieren des aktuellen Handsatzes zunächst automatisch geleert.
Parameter	mdi_log_file_max_size
Datentyp	UNS32
Datenbereich	0 ... MAX(UNS32)
Dimension	----
Standardwert	0 *
Anmerkungen	* keine Größenbegrenzung

P-CHAN-00378	Fehlerursache (Filterung von Fehlermeldungen im Kanal)
Beschreibung	<p>Die einzelnen Fehlerkennungen können als Nummern oder Texte aufgelistet werden, wobei die gesamte Zeile folgender Syntax entsprechen muss: (number text) { , (number text) }</p> <p>mit: number := CNC-Fehlernummer text := " fehlerspezifischer Text "</p> <p>Beispiel: error_filter[0].reason "D012:", 123000, 123001</p> <p>Wird ein Fehler gemeldet, so wird in den definierten Plattform-/ Kanal-/Achsfilttern nachgesehen, ob hierfür eine benutzerspezifische Filterregel definiert ist.</p>
Parameter	error_filter[i].reason mit i = 0 ... 3 (Maximale Anzahl der Filter, applikationsspezifisch)
Datentyp	STRING
Datenbereich	Maximal 96 Zeichen
Dimension	----
Standardwert	*
Anmerkungen	* Hinweis: Der Standardwert der Variablen ist ein Leerstring.

P-CHAN-00379	Fehleraktion (Filterung von Fehlermeldungen im Kanal)	
Beschreibung	Aktion, die bei Auftreten des entsprechenden Fehlers durchgeführt werden soll.	
Parameter	error_filter[i].action mit i = 0 ... 3 (Maximale Anzahl der Filter, applikationsspezifisch)	
Datentyp	STRING	
Datenbereich	ACTION = NONE DRIVE_STATE_REQ PRE_RUN_STATE_REQ RUN_STATE_REQ	
	Schlüsselwort	Bedeutung
	NONE	Keine Aktion
	DRIVE_STATE_REQ	Auslesen des Antriebsstatus
	PRE_RUN_STATE_REQ	Fehler während Hochlauf des Steuerungsbusses in PRE-Run-Zustand
	RUN_STATE_REQ	Fehler während Hochlauf des Steuerungsbusses in Run-Zustand
Dimension	----	
Standardwert	*	
Anmerkungen	<p>Für SERCOS-Antriebs-Profile:</p> <ul style="list-style-type: none"> • DRIVE_STATE_REQ S-0-0095 diagnostic • PRE_RUN_STATE_REQ S-0-0021: list of unknown operation data in CP2 -> CP3, command 127 • RUN_STATE_REQ S-0-0022: list of unknown operation data in CP3 -> CP4, command 128 <p>Für ProfiDrive-Profile:</p> <ul style="list-style-type: none"> • <all actions> Parameter 945 <p>Für CANopen-Profile</p> <ul style="list-style-type: none"> • <all actions> Parameter ID603F <p>* Hinweis: Der Standardwert der Variablen ist ein Leerstring.</p>	

P-CHAN-00380	Bedingte Aktivierung (Filterung von Fehlermeldungen im Kanal)	
Beschreibung	Beim Setzen des entsprechenden Bits über die Oberfläche oder die SPS (HLI::Control Unit-Aktivieren der Fehlerfilterregeln - Kanal [▶ 53]) wird diese Filterregel aktiviert.	
Parameter	error_filter[i].conditional_activation mit i = 0 ... 3 (Maximale. Anzahl der Filter, applikationsspezifisch)	
Datentyp	UNS32	
Datenbereich	32 Bit	
Dimension	----	
Standardwert	0	
Anmerkungen	<p>Parametrierbeispiel:</p> <p><i>error_filter[0].conditional_activation 0x2</i></p> <p>Ein Aktivierungsbit = 0 bedeutet, dass die Aktion immer durchgeführt wird.</p>	

P-CHAN-00381	Bedingte Aktion (Filterung von Fehlermeldungen im Kanal)
Beschreibung	Aktion, die bei Auftreten des entsprechenden Fehlers und beim Zutreffen der Bedingung durchgeführt werden soll.
Parameter	error_filter[i].conditional_action mit i = 0 ... 3 (Maximale Anzahl der Filter, applikationsspezifisch)
Datentyp	STRING
Datenbereich	<p>ACTION = NONE ([HIDE] [FORCE])</p> <p>FORCE = F_WARNING F_SYNTAX F_ERROR F_SEVERE F_FATAL</p> <p>HIDE = [HIDE] [HIDE_LOG] [HIDE_PRINT] [HIDE_REPORT]</p> <p>NONE: keine Aktion</p> <p>HIDE: Jede Fehlerausgabe unterdrücken</p> <p>HIDE_LOG: Fehlerausgabe in Error-Log-Datei wird unterdrückt</p> <p>HIDE_DISPLAY: Fehlerausgabe wird unterdrückt</p> <p>HIDE_REPORT: Applikationsspezifische Fehlerausgabe wird unterdrückt</p> <p>F_WARNING: Fehler wird als WARNING ausgegeben (Behebungsstufe = 0)</p> <p>F_SYNTAX: Fehler wird als Syntaxfehler ausgegeben (Behebungsstufe = 2)</p> <p>F_ERROR: Fehler durch NC-Programm oder andere Bedienaktion (Fehlerbehebungsstufe = 5)</p> <p>F_SEVERE: Schwerer Fehler, erfordert Warmstart, (Behebungsstufe = 6)</p> <p>F_FATAL: Schwerer Fehler, erfordert kompletten Kaltstart (Behebungsstufe = 7)</p>
Dimension	----
Standardwert	*
Anmerkungen	* Hinweis: Der Standardwert der Variablen ist ein Leerstring.

P-CHAN-00382	Bedingter Filteraktivierung (Filterung von Fehlermeldungen im Kanal)
Beschreibung	<p>Die einzelnen Fehlerkennungen können als Nummern oder Texte aufgelistet werden, wobei die gesamte Zeile folgender Syntax entsprechen muss:</p> <p>(number text) { , (number text) }</p> <p>mit:</p> <p>number := CNC-Fehlernummer</p> <p>text := " fehlerspezifischer Text "</p>
Parameter	error_filter[i].conditional_param mit i = 0 ... 3 (Maximale Anzahl der Filter, applikationsspezifisch)
Datentyp	STRING
Datenbereich	Maximal 96 Zeichen
Dimension	----
Standardwert	*
Anmerkungen	<p>Parametrierbeispiel:</p> <p><i>error_filter[0].conditional_param "D012:", 123, 1001</i></p> <p>Individuelle Filtertexte werden aktuell nur beim Auslesen des SERCOS-Antriebsfehlers S95 geprüft.</p> <p>Fehlernummer werden nur bei SERCOS-Antriebsfehler (S21 und S22) und bei ProfiDrive-Antriebsfehler (Parameter 945) geprüft.</p> <p>* Hinweis: Der Standardwert der Variablen ist ein Leerstring.</p>

P-CHAN-00383	Ausgabe einer zusätzlichen Fehlerinformation (Filterung von Fehlermeldungen im Kanal)
Beschreibung	Dieser Text wird bei zutreffender Filterbedingung transparent über die CNC_ERROR_INFO-Datenstruktur weitergereicht. D.h. hierüber hat der Anwender die Möglichkeit, bedingt eine zusätzliche Fehlerinformation mit auszugeben.
Parameter	error_filter[i].conditional_output mit i = 0 ... 3 (Maximale Anzahl der Filter, applikationsspezifisch)
Datentyp	STRING
Datenbereich	Maximal 32 Zeichen
Dimension	----
Standardwert	*
Anmerkungen	* Hinweis: Der Standardwert der Variablen ist ein Leerstring.

3.2.3 Achsparameter

P-AXIS-00627	Fehlerursache (Filterung von Achsfehlermeldungen)	
Beschreibung	<p>Die einzelnen Fehlerkennungen können als Nummern oder Texte aufgelistet werden, wobei die gesamte Zeile folgender Syntax entsprechen muss:</p> <p>(number text) { , (number text) }</p> <p>mit:</p> <p>number := CNC-Fehlernummer</p> <p>text := "fehlerspezifischer Text"</p> <p>Beispiel:</p> <p>error_filter[0].reason "D012:", 123000, 123001</p> <p>Wird ein Fehler gemeldet, so wird in den definierten Plattform-/ Kanal-/Achsfilttern nachgesehen, ob hierfür eine benutzerspezifische Filterregel definiert ist.</p>	
Parameter	error_filter[i].reason mit i = 0 ... 3 (Maximale Anzahl der Filter, applikationsspezifisch)	
Datentyp	STRING	
Datenbereich	Maximal 96 Zeichen	
Achstypen	T, R, S	
Dimension	T: ----	R,S: ----
Standardwert	*	
Anmerkungen	* Hinweis: Der Standardwert der Variablen ist ein Leerstring.	

P-AXIS-00628	Fehleraktion (Filterung von Achsfehlermeldungen)	
Beschreibung	Aktion, die bei Auftreten des entsprechenden Fehlers durchgeführt werden soll.	
Parameter	error_filter[i].action mit i = 0 ... 3 (Maximale Anzahl der Filter, applikationsspezifisch)	
Datentyp	STRING	
Datenbereich	ACTION = NONE DRIVE_STATE_REQ PRE_RUN_STATE_REQ RUN_STATE_REQ NONE: Keine Aktion DRIVE_STATE_REQ: Auslesen des Antriebsstatus PRE_RUN_STATE_REQ: Fehler während Hochlauf des Steuerungsbusses in PRE-Run-Zustand RUN_STATE_REQ: Fehler während Hochlauf des Steuerungsbusses in Run-Zustand	
Achstypen	T, R, S	
Dimension	T: ----	R,S: ----
Standardwert	*	
Anmerkungen	Für SERCOS-Antriebs-Profile: DRIVE_STATE_REQ S-0-0095 diagnostic PRE_RUN_STATE_REQ S-0-0021: list of unknown operation data in CP2 -> CP3, command 127 RUN_STATE_REQ S-0-0022: list of unknown operation data in CP3 -> CP4, command 128 Für Profidrive-Profile: <all actions> Parameter 945 Für CANopen-Profile <all actions> Parameter ID603F * Hinweis: Der Standardwert der Variablen ist ein Leerstring.	

P-AXIS-00629	Bedingte Aktivierung (Filterung von Achsfehlermeldungen)	
Beschreibung	Beim Setzen des entsprechenden Bits über die Oberfläche oder die SPS (HLI:: ControlUnit-Aktivieren der Fehlerfilterregeln - Achse [► 54]) wird diese Filterregel aktiviert.	
Parameter	error_filter[i].conditional_activation mit i = 0 ... 3 (Maximale. Anzahl der Filter, applikationsspezifisch)	
Datentyp	UNS32	
Datenbereich	32 Bit	
Achstypen	T, R, S	
Dimension	T: ----	R,S: ----
Standardwert	0	
Anmerkungen	Parametrierbeispiel: <i>error_filter[0].conditional_activation 0x2</i> Ein Aktivierungsbit = 0 bedeutet, dass die Aktion immer durchgeführt wird.	

P-AXIS-00630	Bedingte Aktion (Filterung von Achsfehlermeldungen)	
Beschreibung	Aktion, die bei Auftreten des entsprechenden Fehlers und beim Zutreffen der Bedingung durchgeführt werden soll.	
Parameter	error_filter[i].conditional_action mit i = 0 ... 3 (Maximale Anzahl der Filter, applikationsspezifisch)	
Datentyp	STRING	
Datenbereich	ACTION = NONE ([HIDE] [FORCE]) FORCE = F_WARNING F_SYNTAX F_ERROR F_SEVERE F_FATAL HIDE = [HIDE] [HIDE_LOG] [HIDE_PRINT] [HIDE_REPORT]	
	NONE:	keine Aktion
	HIDE:	Jede Fehlerausgabe unterdrücken
	HIDE_LOG:	Fehlerausgabe in Error-Log-Datei wird unterdrückt
	HIDE_DISPLAY:	Fehlerausgabe wird unterdrückt
	HIDE_REPORT:	Applikationsspezifische Fehlerausgabe wird unterdrückt
	F_WARNING:	Fehler wird als WARNING ausgegeben (Behungsklasse = 0)
	F_SYNTAX:	Fehler wird als Syntaxfehler ausgegeben (Behebungs-klasse = 2)
	F_ERROR:	Fehler durch NC-Programm oder andere Bedienaktion (Fehlerbehebungs-klasse = 5)
	F_SEVERE:	Schwerer Fehler, erfordert Warmstart, (Behebungs-klasse = 6)
	F_FATAL:	Schwerer Fehler, erfordert kompletten Kaltstart (Behebungs-klasse = 7)
Achstypen	T, R, S	
Dimension	T: ----	R,S: ----
Standardwert	*	
Anmerkungen	* Hinweis: Der Standardwert der Variablen ist ein Leerstring.	

P-AXIS-00631	Bedingte Filteraktivierung (Filterung von Achsfehlermeldungen)	
Beschreibung	<p>Die einzelnen Fehlerkennungen können als Nummern oder Texte aufgelistet werden, wobei die gesamte Zeile folgender Syntax entsprechen muss: (number text) { , (number text) }</p> <p>mit: number := CNC-Fehlernummer text := " fehlerspezifischer Text "</p>	
Parameter	error_filter[i].conditional_param mit i = 0 ... 3 (Maximale Anzahl der Filter, applikationsspezifisch)	
Datentyp	STRING	
Datenbereich	Maximal 96 Zeichen	
Achstypen	T, R, S	
Dimension	T: ----	R,S: ----
Standardwert	*	
Anmerkungen	<p>Parametrierbeispiel: <i>error_filter[0].conditional_param "D012:", 123, 1001</i></p> <p>Individuelle Filtertexte werden aktuell nur beim Auslesen des SERCOS-Antriebsfehlers S95 geprüft.</p> <p>Fehlernummer werden nur bei SERCOS-Antriebsfehler (S21 und S22) und bei Profidrive-Antriebsfehler (Parameter 945) geprüft.</p> <p>* Hinweis: Der Standardwert der Variablen ist ein Leerstring.</p>	

P-AXIS-00632	Ausgabe einer zusätzlichen Fehlerinformation (Filterung von Achsfehlermeldungen)	
Beschreibung	<p>Dieser Text wird im Falle des Zutreffens der Filterbedingung transparent über die CNC_ERROR_INFO-Datenstruktur weitergereicht. D.h. hierüber hat der Anwender die Möglichkeit, bedingt eine zusätzliche Fehlerinformation mit auszugeben.</p>	
Parameter	error_filter[i].conditional_output mit i = 0 ... 3 (Maximale Anzahl der Filter, applikationsspezifisch)	
Datentyp	STRING	
Datenbereich	Maximal 32 Zeichen	
Achstypen	T, R, S	
Dimension	T: ----	R,S: ----
Standardwert	*	
Anmerkungen	* Hinweis: Der Standardwert der Variablen ist ein Leerstring.	

3.3 CNC-Objekte

Informationen zu Adressierung von CNC-Objekten finden Sie unter [FCT-C13//Beschreibung].

Aktivieren von Filterregeln

Name	cnc_error_filter_w		
Beschreibung	<p>Die einzelnen Filterregeln der Plattform lassen sich über die Oberfläche durch Setzen der entsprechenden Bitmaske ein- und ausschalten. Voraussetzung hierfür ist die Parametrierung von P-STUP-00188.</p> <p>Es lassen sich somit maximal 32 verschiedene Filterregeln schalten.</p> <p>Beispiel der zugeordneten Filterdefinition:</p> <pre>error_filter[0].activation_bit 0x1</pre>		
Task	COM (Port 553)		
Indexgruppe	0x120100	Indexoffset	0x26C
Datentyp	UNS32	Länge	4
Attribute	write	Einheit	-
Anmerkungen	Siehe Fehlerfilterdefinition [▶ 16] in Hochlaufliste		

Name	mc_error_filter_w		
Beschreibung	<p>Die einzelnen Filterregeln eines Kanals lassen sich über die Oberfläche durch Setzen der entsprechenden Bitmaske ein- und ausschalten. Voraussetzung hierfür ist die Parametrierung von P-CHAN-00380.</p> <p>Es lassen sich somit maximal 32 verschiedene Filterregeln schalten.</p> <p>Beispiel der zugeordneten Filterdefinition:</p> <pre>error_filter[0].activation_bit 0x1</pre>		
Task	COM (Port 553)		
Indexgruppe	0x12010<C _{ID} >	Indexoffset	0x419
Datentyp	UNS32	Länge	4
Attribute	write	Einheit	-
Anmerkungen			

Name	ac_<A _{ID} >_error_filter_w		
Beschreibung	<p>Die einzelnen Filterregeln einer Achse lassen sich über die Oberfläche durch Setzen der entsprechenden Bitmaske ein- und ausschalten. Voraussetzung dafür ist Parametrierung von P-AXIS-00629.</p> <p>Es lassen sich somit maximal 32 verschiedene Filterregeln schalten.</p> <p>Beispiel der zugeordneten Filterdefinition:</p> <pre>error_filter[0].activation_bit 0x1</pre>		
Task	COM (Port 553)		
Indexgruppe	0x120200	Indexoffset	0x<A _{ID} >0028
Datentyp	UNS32	Länge	4
Attribute	write	Einheit	-
Anmerkungen			

Zugriff auf Protokollierungsmodus

Name	cnc_error_manager_mode_r		
Beschreibung	<p>CNC Objekt zum Lesen des Modus des Fehlermanagements.</p> <p>Siehe P-STUP-00167.</p>		
Task	COM (Port 553)		
Indexgruppe	0x120100	Indexoffset	0x293
Datentyp	UN32	Länge/Byte	4
Attribute	read	Einheit	-
Anmerkungen	Die Zuordnung und Bedeutung der Bits sind in der Tabelle [▶ 50] beschrieben.		

Name	cnc_error_manager_mode_w		
Beschreibung	<p>CNC Objekt zum Schreiben des Mode des Fehlermanagements.</p> <p>Siehe P-STUP-00167.</p>		
Task	COM (Port 553)		
Indexgruppe	0x120100	Indexoffset	0x26D
Datentyp	UN32	Länge/Byte	4
Attribute	write	Einheit	-
Anmerkungen	Die Zuordnung und Bedeutung der Bits sind in der Tabelle [▶ 50] beschrieben.		

Kennung	Bitbelegung	Bedeutung
FILTER_OFF	0x00004	Es werden keine Filter ausgewertet
VERBOSE	0x00008	Erweiterte interne Diagnose
WITHOUT_ERROR_MANAGER	0x00001	Direkte Ausgabe ohne Fehlermanagement

Kennung	Bitbelegung	Bedeutung
PRINT	0x00010	Ausgabe als Print ausführen
LOG	0x00020	Ausgabe auf Log-Datei protokollieren
REPORT	0x00040	Ausgabe auf Log-Datei protokollieren
SEND_TO_PLC_NONE	0x20000	Ausgabe an die SPS unterdrücken
PRINT_EXTENDED	0x00100	Erweiterte Print-Ausgabe
LOG_EXTENDED	0x00200	Erweiterte Log-Ausgabe
REPORT_EXTENDED	0x00400	Erweiterte applikationsspezifische Ausgabe
PRINT_NO_WARNINGS	0x01000	Warnungen werden in der Print-Ausgabe unterdrückt
LOG_NO_WARNINGS	0x02000	Warnungen werden in der Log-Ausgabe unterdrückt
REPORT_NO_WARNINGS	0x04000	Warnungen werden in der Report-Ausgabe unterdrückt
SEND_TO_PLC_NO_WARNINGS	0x08000	Warnungen an SPS unterdrücken
STARTUP_NO_WARNINGS	0x10000	Warnungen während Steuerungshochlauf unterdrücken
NO_WARNINGS	0x1f000	Unterdrücken aller Warnungen
TC3_EVENT_LOGGER	0x80000	Ausgabe an TC3-Event-Logger

Name	log errors		
Beschreibung	Mit diesem Objekt kann festgelegt werden ob Fehlermeldungen der CNC über die SPS ausgegeben werden. (TRUE: Standard)		
Task	GEO (Port 551)		
Indexgruppe	0x120300	Indexoffset	0x4
Datentyp	BOOLEAN	Länge	1
Attribute	read/ write	Einheit	[-]
Anmerkungen	Werte : TRUE/FALSE		

3.4 HLI-Parameter

Aktivieren der Fehlerfilterregeln - Plattform	
Beschreibung	<p>Die einzelnen Fehlerfilter können hier entsprechend ihres Aktivierungsbits durch die 32-Bit-Maske ein-/ausgeschaltet werden.</p> <p>Z.B. Nachfolgende Regel wird durch Setzen des ersten Bits (command_w = 0x00000001) wirksam:</p> <pre>error_filter[0].activation_bit 0x1</pre>
Datentyp	MC_CONTROL_UN32_UNIT, s. Beschreibung Control Unit
Einheit	[-]
Zugriff	PLC liest request_r + state_r und schreibt command_w + enable_w
ST-Pfad	gpPform^.error_filter
Kommandierter und angeforderter Wert	
ST-Element	.command_w .request_r
Datentyp	UDINT
Rückgabewert	
ST-Element	.state_r
Datentyp	UDINT
Umleitung	
ST-Element	.enable_w

Aktivieren der Fehlerfilterregeln - Kanal	
Beschreibung	<p>Die einzelnen Fehlerfilter können hier entsprechend ihres Aktivierungsbits durch die 32-Bit-Maske ein-/ausgeschaltet werden.</p> <p>Z.B. Nachfolgende Regel wird durch Setzen des ersten Bits (<code>command_w = 0x00000001</code>) wirksam:</p> <pre>error_filter[0].activation_bit 0x1</pre>
Datentyp	MC_CONTROL_UN32_UNIT, s. Beschreibung Control Unit
Einheit	[-]
Zugriff	PLC liest <code>request_r + state_r</code> und schreibt <code>command_w + enable_w</code>
ST-Pfad	<code>gpCh[channel_idx]^head.error_filter</code>
Kommandierter und angeforderter Wert	
ST-Element	.command_w .request_r
Datentyp	UDINT
Rückgabewert	
ST-Element	.state_r
Datentyp	UDINT
Umleitung	
ST-Element	.enable_w

Aktivieren der Fehlerfilterregeln - Achse	
Beschreibung	<p>Die einzelnen Fehlerfilter können hier entsprechend ihres Aktivierungsbits durch die 32-Bit-Maske ein-/ausgeschaltet werden.</p> <p>Z.B. Nachfolgende Regel wird durch Setzen des ersten Bits (<code>command_w = 0x00000001</code>) wirksam:</p> <pre>error_filter[0].activation_bit 0x1</pre>
Datentyp	MC_CONTROL_UN32_UNIT, s. Beschreibung Control Unit
Einheit	[-]
Zugriff	PLC liest <code>request_r + state_r</code> und schreibt <code>command_w + enable_w</code>
ST-Pfad	<code>gpAx[axis_idx]^head.error_filter</code>
Kommandierter und angeforderter Wert	
ST-Element	.command_w .request_r
Datentyp	UDINT
Rückgabewert	
ST-Element	.state_r
Datentyp	UDINT
Umleitung	
ST-Element	.enable_w

4 Anhang

4.1 Anregungen, Korrekturen und neueste Dokumentation

Sie finden Fehler, haben Anregungen oder konstruktive Kritik? Gerne können Sie uns unter documentation@isg-stuttgart.de kontaktieren. Die aktuellste Dokumentation finden Sie in unserer Onlinehilfe (DE/EN):



QR-Code Link: <https://www.isg-stuttgart.de/documentation-kernel/>

Der o.g. Link ist eine Weiterleitung zu:

<https://www.isg-stuttgart.de/fileadmin/kernel/kernel-html/index.html>



Hinweis

Mögliche Änderung von Favoritenlinks im Browser:

Technische Änderungen der Webseitenstruktur betreffend der Ordnerpfade oder ein Wechsel des HTML-Frameworks und damit der Linkstruktur können nie ausgeschlossen werden.

Wir empfehlen, den o.g. „QR-Code Link“ als primären Favoritenlink zu speichern.

PDFs zum Download:

DE:

<https://www.isg-stuttgart.de/produkte/softwareprodukte/isg-kernel/dokumente-und-downloads>

EN:

<https://www.isg-stuttgart.de/en/products/softwareproducts/isg-kernel/documents-and-downloads>

E-Mail: documentation@isg-stuttgart.de

Stichwortverzeichnis

F

Fehlerfilter

aktivieren-Achse	54
aktivieren-Kanal.....	53
aktivieren-Plattform	52

P

P-AXIS-00627	45
P-AXIS-00628	46
P-AXIS-00629	46
P-AXIS-00630	47
P-AXIS-00631	48
P-AXIS-00632	48
P-CHAN-00338	40
P-CHAN-00339	40
P-CHAN-00378	41
P-CHAN-00379	42
P-CHAN-00380	42
P-CHAN-00381	43
P-CHAN-00382	44
P-CHAN-00383	44
P-STUP-00166	31
P-STUP-00167	32
P-STUP-00168	33
P-STUP-00169	33
P-STUP-00170	34
P-STUP-00171	34
P-STUP-00172	34
P-STUP-00173	35
P-STUP-00186	35
P-STUP-00187	36
P-STUP-00188	36
P-STUP-00189	37
P-STUP-00190	38
P-STUP-00191	38
P-STUP-00200	39



© Copyright
ISG Industrielle Steuerungstechnik GmbH
STEP, Gropiusplatz 10
D-70563 Stuttgart
Alle Rechte vorbehalten
www.isg-stuttgart.de
support@isg-stuttgart.de

