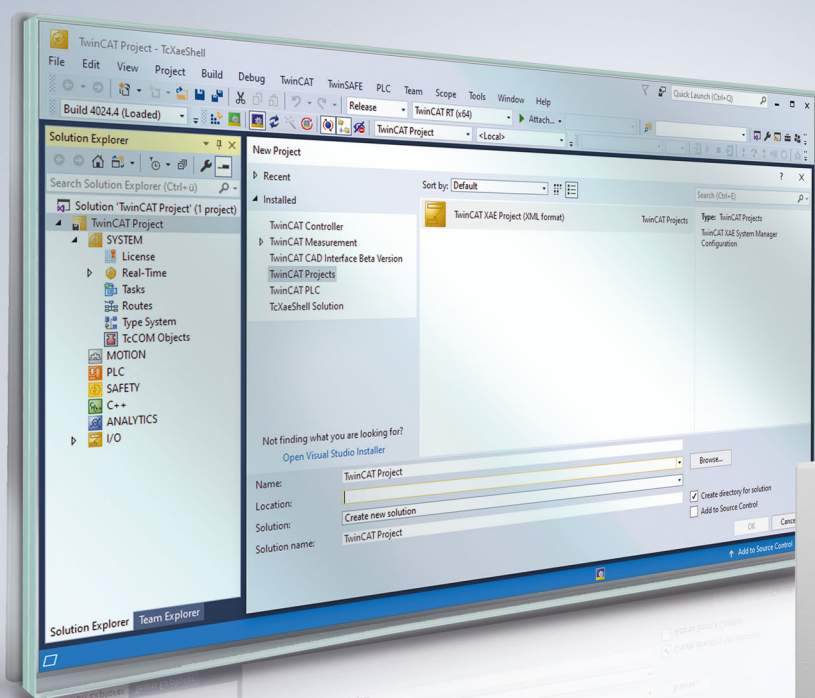


BECKHOFF New Automation Technology

Handbuch | DE

TF5065

TwinCAT 3 | Motion Control XFC/XFC NC I



Inhaltsverzeichnis

| | | |
|----------|---|-----------|
| 1 | Vorwort | 5 |
| 1.1 | Hinweise zur Dokumentation | 5 |
| 1.2 | Zu Ihrer Sicherheit..... | 6 |
| 1.3 | Hinweise zur Informationssicherheit | 7 |
| 2 | Einleitung | 8 |
| 3 | Totzeitkompensation | 9 |
| 4 | SPS-Bibliotheken | 12 |
| 4.1 | Tc2_MC2_XFC | 12 |
| 4.1.1 | Funktionsbausteine | 12 |
| 4.1.2 | Funktionen | 40 |
| 4.1.3 | Datentypen..... | 41 |
| 4.2 | Tc2_NciXFC | 52 |
| 4.2.1 | Funktionsbausteine | 52 |
| 4.2.2 | Datentypen..... | 58 |
| 4.3 | Tc3_MC2_AdvancedHoming_XFC | 59 |
| 4.3.1 | Übersicht | 59 |
| 4.3.2 | Funktionsbausteine | 60 |
| 4.3.3 | Datentypen..... | 78 |
| 5 | Beispiele | 79 |
| 6 | Support und Service | 80 |

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, stets die aktuell gültige 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 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.

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®

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 Zu Ihrer Sicherheit

Sicherheitsbestimmungen

Lesen Sie die folgenden Erklärungen zu Ihrer Sicherheit. Beachten und befolgen Sie stets produktspezifische Sicherheitshinweise, die Sie gegebenenfalls an den entsprechenden Stellen in diesem Dokument vorfinden.

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.

Signalwörter

Im Folgenden werden die Signalwörter eingeordnet, die in der Dokumentation verwendet werden. Um Personen- und Sachschäden zu vermeiden, lesen und befolgen Sie die Sicherheits- und Warnhinweise.

Warnungen vor Personenschäden

GEFAHR

Es besteht eine Gefährdung mit hohem Risikograd, die den Tod oder eine schwere Verletzung zur Folge hat.

WARNUNG

Es besteht eine Gefährdung mit mittlerem Risikograd, die den Tod oder eine schwere Verletzung zur Folge haben kann.

VORSICHT

Es besteht eine Gefährdung mit geringem Risikograd, die eine mittelschwere oder leichte Verletzung zur Folge haben kann.

Warnung vor Umwelt- oder Sachschäden

HINWEIS

Es besteht eine mögliche Schädigung für Umwelt, Geräte oder Daten.

Information zum Umgang mit dem Produkt



Diese Information beinhaltet z. B.:
Handlungsempfehlungen, Hilfestellungen oder weiterführende Informationen zum Produkt.

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 [hier](#).

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](#).

2 Einleitung

Mit eXtreme Fast Control (XFC) wird eine Technik bezeichnet, die mit EtherCAT, speziellen I/O-Klemmen und TwinCAT auf dem PC eine sehr schnelle, zeitlich hoch präzise Reaktion ermöglicht. Mit den Distributed-Clocks (DC) von EtherCAT und entsprechenden Klemmen können so einfach verteiltes Latchen oder Nockenschaltwerke realisiert werden.

- Bausteine zum hochgenauen Erfassen und Schalten von digitalen Signalen bezogen auf Achspositionen
- EtherCAT-Distributed-Clocks mit den zeitstempelbasierten Ein- und Ausgangs-EtherCAT-Klemmen, wie zum Beispiel EL1252, EL2252 oder EL2262
- Konvertierungsbausteine von DC-Zeit in Position und umgekehrt
- Komfortabler PLCopen-konformer TouchProbe-Baustein
- Digitales Nockenschaltwerk als PLCopen-konformer Baustein

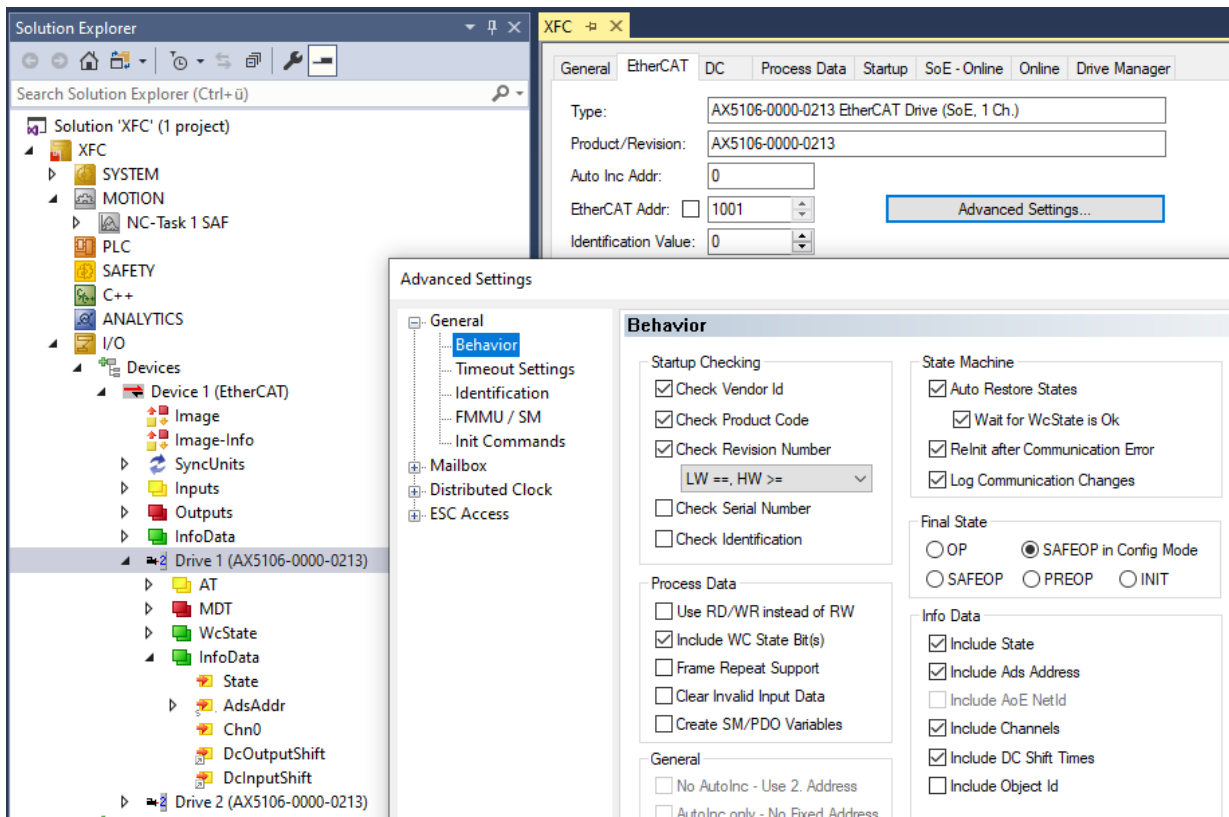
3 Totzeitkompensation

Voraussetzung für die hochgenaue Umrechnung von Positionen in Zeiten und umgekehrt ist eine exakte Totzeitkompensation der Achsen. In TwinCAT 3 steht eine solche Totzeitkompensation für EtherCAT- und Sercos-Achsen zur Verfügung, die weitestgehend automatisch arbeitet. Dennoch kann eine manuelle Konfiguration notwendig sein, um beispielsweise antriebsinterne Totzeiten auszugleichen.

Unterstützung der Distributed Clocks

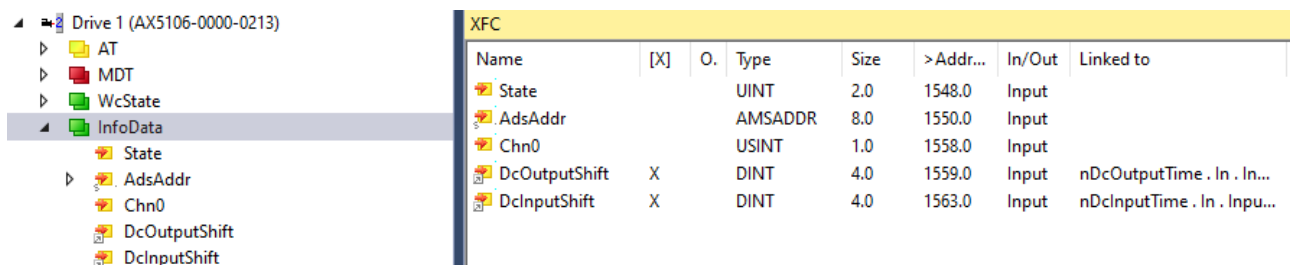
Die Unterstützung der Distributed Clocks muss bei EtherCAT-Antrieben zunächst wie folgt aktiviert werden:

1. Rufen Sie den Dialog EtherCAT **Advanced Settings** des Antriebs auf.



2. Aktivieren Sie den Schalter **Include DC Shift Times**.

⇒ Die Zeitinformationen werden in den Info-Daten (**InfoData**) des Antriebs zur Verfügung gestellt und später mit der NC-Achse verknüpft.

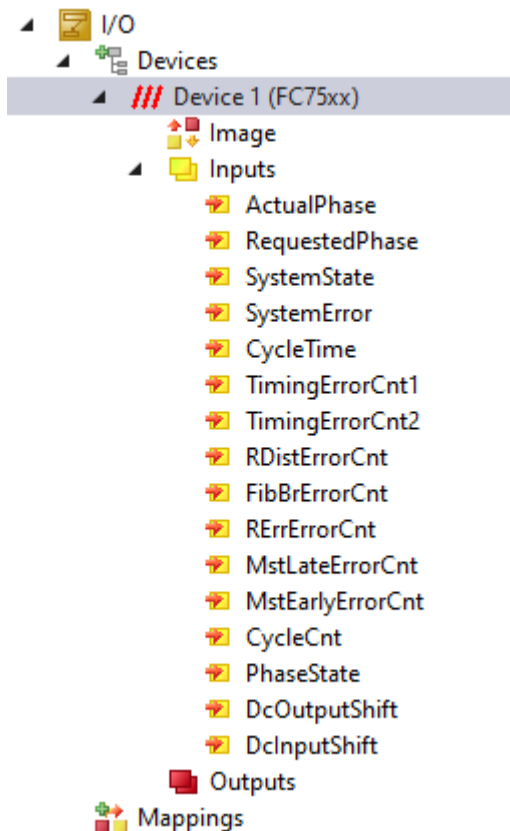


DcInputShift ist die Zeit, die benötigt wird, um Statusinformationen, wie z. B. die Istposition eines Antriebs, in die Steuerung zu übertragen. Also die Zeit zwischen der Erfassung und der Auswertung dieser Daten.

DcOutputShift ist die Zeit für die Ausgabe der Prozessdaten an den Antrieb, d.h. für den Zeitverzug zwischen Berechnung und Wirkung dieser Daten.

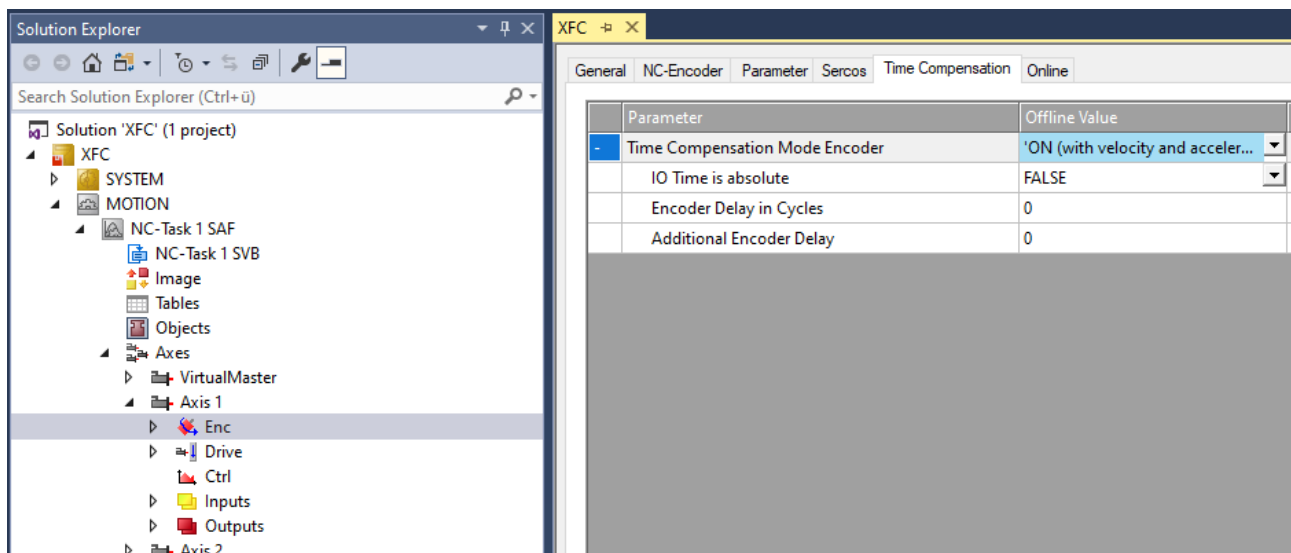
Die Zeitinformationen werden dynamisch vom System bereitgestellt und werden von der NC zur Totzeitkompensation einer Achse verwendet.

Für Sercos-Achsen werden die Zeiten **DcInputShift** und **DcOutputShift** von der Sercos-Karte zur Verfügung gestellt und müssen nicht konfiguriert werden. Beim Verknüpfen eines Antriebs mit einer NC-Achse werden diese Zeiten ebenfalls verknüpft.



Kompensation der Encoder-Totzeit

Die Totzeitkompensation für die Datenerfassungsseite aktivieren Sie auf dem Reiter **Time Compensation** des Achs-Encoders. Dadurch wird die vom System bereit gestellte Totzeit aus **DcInputShift** zur Kompensationsberechnung verwendet.



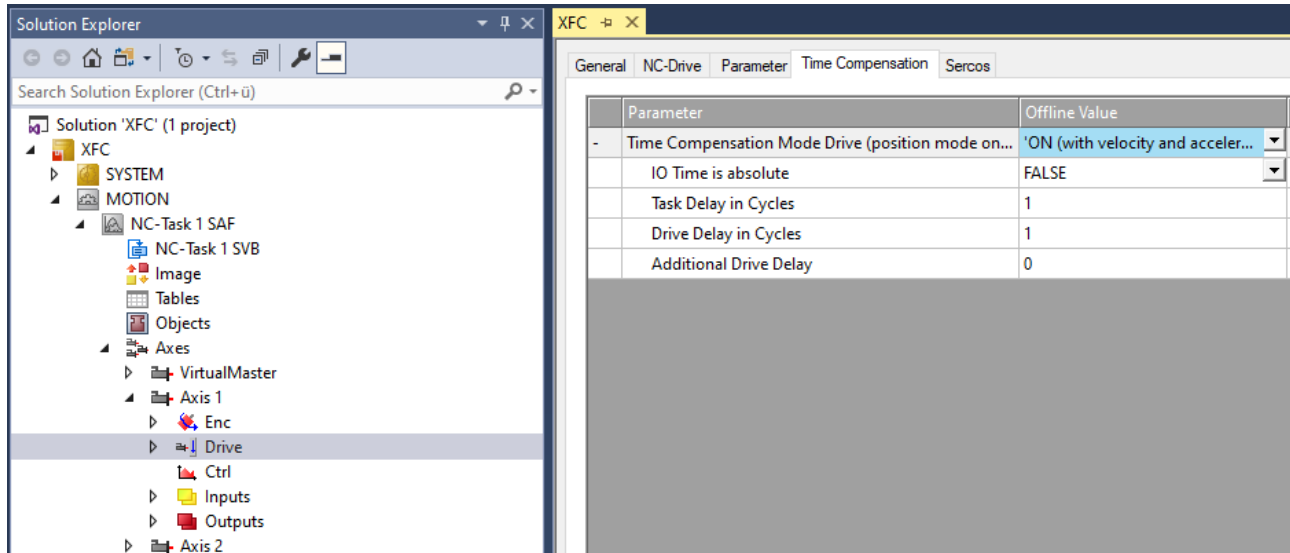
In besonderen Fällen, wenn es beispielsweise durch die verwendete Hardware zu zusätzlichen Totzeiten kommt, kann es notwendig werden, weitere Zeiten zu konfigurieren.

Der Wert **Encoder Delay in Cycles** gibt zusätzliche Verzögerungen um ganze I/O-Zyklen an. Diese Zeit ist demnach keine feste Größe, sondern ändert sich mit der Zykluszeit.

Der Wert **Additional Encoder Delay** ist ein fester Zeitwert in μs , der durch die verwendete Hardware verursacht wird.

Kompensation der Antriebs-Totzeit

Die Totzeitkompensation in Ausgaberrichtung aktivieren Sie auf dem Reiter **Time Compensation** des Drives der NC-Achse. Dadurch wird die vom System bereitgestellte Zeit DcOutputShift zur Kompensationsberechnung verwendet.



Auch hier können in besonderen Fällen weitere Zeiten konfiguriert werden.

Der Wert **Task Delay in Cycles** wird durch die Einstellung der Taskkonfiguration verursacht. Abhängig vom eingestellten Timing der Tasks kann sich die Totzeit um einen Zyklus verlängern.

Der Wert **Drive Delay in Cycles** gibt zusätzliche Verzögerungen um ganze I/O-Zyklen an, die durch den Antrieb verursacht werden (z.B. in Abhängigkeit von dem Interpolationstyp des Antriebs).

Der Wert **Additional Drive Delay** ist ein fester Zeitwert in μs , der durch die verwendete Hardware verursacht wird.

Auswirkung der Totzeitkompensation

Durch die Totzeitkompensation werden alle zyklisch mit der SPS ausgetauschten Daten der NC (NcToPlc) auf den aktuellen Zeitpunkt umgerechnet. Insbesondere Istposition, Sollposition und Schleppfehler der Achse beziehen sich auf den gerade aktuellen Zeitpunkt und spiegeln die physikalische Position der Achse zu diesem Zeitpunkt wider. Sie können damit durch die SPS für weitere hochgenaue Zeit- und Positionsberechnungen herangezogen werden. (Siehe insbesondere die Basisfunktionen [XFC_GetCurDcTaskTime \[► 40\]](#) , [XFC_TimeOfPosition \[► 13\]](#) und [XFC_PositionAtTime \[► 12\]](#).)

4 SPS-Bibliotheken

4.1 Tc2_MC2_XFC

Die Bibliothek Tc2_Mc2_XFC erleichtert die zeitgenaue Erfassung von Achspositionen und die positionsgenaue Ausgabe von Digitalsignalen in Verbindung mit EtherCAT-XFC-Klemmen (Zeitstempel-Klemmen und Oversampling-Klemmen).

So kann jederzeit ein genauer Zusammenhang zwischen Zeit und Position hergestellt werden.

Wichtige Anwendungsfälle sind die Erfassung von Latch-Positionen (Touch Probe bzw. Messtaster-Funktion) und die Realisierung digitaler Nockenschaltwerke. Hierfür stellt die Bibliothek verschiedene Funktionsbausteine bereit.

4.1.1 Funktionsbausteine

4.1.1.1 Basisbausteine

4.1.1.1.1 XFC_PositionAtTime



Der Funktionsbaustein XFC_PositionAtTime berechnet eine Achsposition, welche zu einer gegebenen Zeit gültig sein wird, bzw. gültig gewesen ist.

Die Funktion extrapoliert die Position bezogen auf die aktuelle Position und Dynamik. Eine genaue Extrapolation ist nur über einen kurzen Zeitraum möglich, da sich die Dynamik der Achse ändern kann.

Der Funktionsbaustein benötigt genau einen Aufruf, um das Ergebnis bereitzustellen. Er kann also ähnlich wie eine Funktion verwendet werden, gibt aber neben der Position eventuell auch einen Fehler zurück. Dieser Fehler muss ausgewertet werden, um sicherzustellen, dass die berechnete Position gültig ist.

Eingänge

```
VAR_INPUT
  DcTime   : T_DCTIME32;
  Options  : ST_NcTimeConversionOptions;
END_VAR
```

| Name | Typ | Beschreibung |
|---------|---|--|
| DcTime | T_DCTIME32 | Distributed Clock System Time. Enthält die unteren 32 Bit der vollständigen DcTime und deckt damit einen Zeitbereich von +/- 2 Sekunden um den aktuellen Zeitpunkt ab. Um einen möglichst genauen Positionswert zu berechnen, sollte die Zeit sehr nahe an der aktuellen Zeit liegen, also möglichst nur wenige SPS- bzw. NC-Zyklen in der Zukunft oder in der Vergangenheit. |
| Options | ST_NcTimeConversionOptions [► 48] | Datenstruktur mit Optionen für die Extrapolation der Position. |

Ein-/Ausgänge

```
VAR_IN_OUT
  Axis     : AXIS_REF;
END_VAR
```

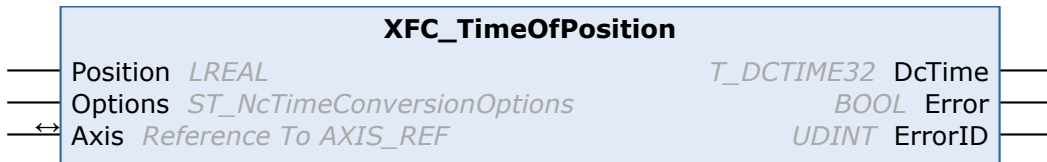
| Name | Typ | Beschreibung |
|------|----------|---|
| Axis | AXIS_REF | Die Achsdatenstruktur vom Typ AXIS_REF adressiert eine Achse eindeutig im System. Sie enthält unter anderem den aktuellen Status der Achse, wie Position, Geschwindigkeit oder Fehlerzustand. |

 **Ausgänge**

```
VAR_OUTPUT
    Position : LREAL;
    Error    : BOOL;
    ErrorID  : UDINT;
END_VAR
```

| Name | Typ | Beschreibung |
|----------|-------|---|
| Position | LREAL | Extrapolierte Position, die zum vorgegebenen Zeitpunkt DcTime erreicht sein wird bzw. erreicht wurde. |
| Error | BOOL | Wird im Fehlerfall TRUE. |
| ErrorID | UDINT | Liefert bei einem gesetzten Error-Ausgang die Fehlernummer. |

4.1.1.1.2 XFC_TimeOfPosition



Der Funktionsbaustein XFC_TimeOfPosition berechnet den Zeitpunkt, zu dem eine Achse an einer vorgegebenen Position sein wird, bzw. gewesen ist.

Der Funktionsbaustein extrapoliert den Zeitpunkt bezogen auf die aktuelle Position und Dynamik. Eine genaue Extrapolation ist nur über einen kurzen Zeitraum möglich, da sich die Dynamik der Achse ändern kann.

Er benötigt genau einen Aufruf, um das Ergebnis bereitzustellen. Er kann also ähnlich wie eine Funktion verwendet werden, gibt aber neben der Zeit eventuell auch einen Fehler zurück. Dieser Fehler muss ausgewertet werden, um sicherzustellen, dass die berechnete Zeit DcTime gültig ist.

 **Eingänge**

```
VAR_INPUT
    Position : LREAL;
    Options  : ST_NcTimeConversionOptions;
END_VAR
```

| Name | Typ | Beschreibung |
|----------|---|--|
| Position | LREAL | Absolute Achsposition |
| Options | ST_NcTimeConversionOptions [▶ 48] | Datenstruktur mit Optionen für die Extrapolation der Position. |

 **Ein-/Ausgänge**

```
VAR_IN_OUT
    Axis : AXIS_REF;
END_VAR
```

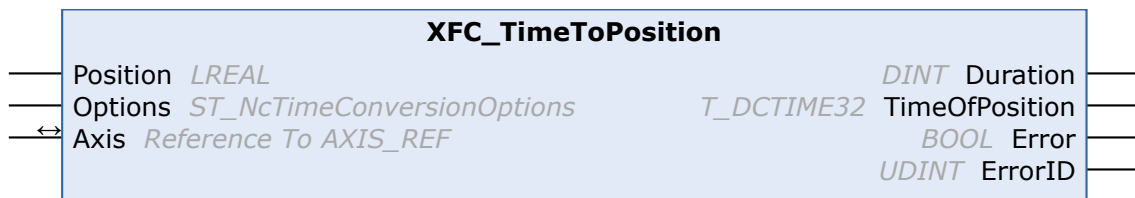
| Name | Typ | Beschreibung |
|------|----------|---|
| Axis | AXIS_REF | Die Achsdatenstruktur vom Typ AXIS_REF adressiert eine Achse eindeutig im System. Sie enthält unter anderem den aktuellen Status der Achse, wie Position, Geschwindigkeit oder Fehlerzustand. |

Ausgänge

```
VAR_OUTPUT
  DcTime   : T_DCTIME32;
  Error    : BOOL;
  ErrorID  : UDINT;
END_VAR
```

| Name | Typ | Beschreibung |
|---------|------------|--|
| DcTime | T_DCTIME32 | Distributed Clock System Time zu der die Position erreicht sein wird bzw. zu der sie erreicht wurde. DcTime enthält die unteren 32 Bit der vollständigen DcTime und deckt damit einen Zeitbereich von +/- 2 Sekunden um den aktuellen Zeitpunkt ab. |
| Error | BOOL | Wird im Fehlerfall TRUE. |
| ErrorID | UDINT | Liefert bei einem gesetzten Error-Ausgang die Fehlernummer. |

4.1.1.1.3 XFC_TimeToPosition




Der Funktionsbaustein XFC_TimeToPosition berechnet die Zeitdauer, innerhalb derer eine Achse eine Position erreichen wird, bzw. die Zeitdauer, die vergangen ist, seit die Achse diese Position passiert hat.

Der Funktionsbaustein extrapoliert den Zeitpunkt bezogen auf die aktuelle Position und Dynamik. Eine genaue Extrapolation ist nur über einen kurzen Zeitraum möglich, da sich die Dynamik der Achse ändern kann.

Er benötigt genau einen Aufruf, um das Ergebnis bereitzustellen. Er kann also ähnlich wie eine Funktion verwendet werden, gibt aber neben der Zeit eventuell auch einen Fehler zurück. Dieser Fehler muss ausgewertet werden, um sicherzustellen, dass die berechnete Zeitdauer gültig ist.

Eingänge

```
VAR_INPUT
  Position   : LREAL;
  Options    : ST_NcTimeConversionOptions;
END_VAR
```

| Name | Typ | Beschreibung |
|----------|---|--|
| Position | LREAL | Absolute Achsposition |
| Options | ST_NcTimeConversionOptions  48 | Datenstruktur mit Optionen für die Extrapolation der Position. |

Ein-/Ausgänge

```
VAR_IN_OUT
  Axis      : AXIS_REF;
END_VAR
```

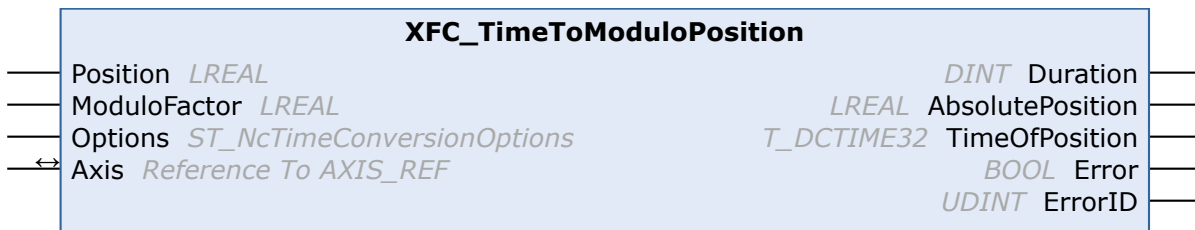
| Name | Typ | Beschreibung |
|------|----------|---|
| Axis | AXIS_REF | Die Achsdatenstruktur vom Typ AXIS_REF adressiert eine Achse eindeutig im System. Sie enthält unter anderem den aktuellen Status der Achse, wie Position, Geschwindigkeit oder Fehlerzustand. |

Ausgänge

```
VAR_OUTPUT
  Duration      : DINT;
  TimeOfPosition : T_DCTIME32;
  Error         : BOOL;
  ErrorID      : UDINT;
END_VAR
```

| Name | Typ | Beschreibung |
|----------------|------------|--|
| Duration | DINT | Zeitdauer in Nanosekunden, nach der die Position erreicht sein wird (> 0) bzw. seit dem die Position passiert wurde (< 0). Duration ist ein Differenzwert aus zwei Variablen des Typs T_DCTIME32 Distributed Clock System Time. |
| TimeOfPosition | T_DCTIME32 | Distributed Clock System Time zu der die Position erreicht sein wird bzw. zu der sie erreicht wurde. DcTime enthält die unteren 32 Bit der vollständigen DcTime und deckt damit einen Zeitbereich von +/- 2 Sekunden um den aktuellen Zeitpunkt ab. |
| Error | BOOL | Wird im Fehlerfall TRUE. |
| ErrorID | UDINT | Liefert bei einem gesetzten Error-Ausgang die Fehlernummer. |

4.1.1.1.4 XFC_TimeToModuloPosition



Der Funktionsbaustein XFC_TimeToModuloPosition berechnet die Zeitdauer, innerhalb derer eine Achse eine Position erreichen wird, bzw. die Zeitdauer, die vergangen ist, seit die Achse diese Position passiert hat. Die Position ist in diesem Fall die in Fahrtrichtung nächstliegende Modulo-Position.

Die Funktion extrapoliert den Zeitpunkt bezogen auf die aktuelle Position und Dynamik. Eine genaue Extrapolation ist nur über einen kurzen Zeitraum möglich, da sich die Dynamik der Achse ändern kann.

Der Funktionsbaustein benötigt genau einen Aufruf, um das Ergebnis bereitzustellen. Er kann also ähnlich wie eine Funktion verwendet werden, gibt aber neben der Zeit eventuell auch einen Fehler zurück. Dieser Fehler muss ausgewertet werden, um sicherzustellen, dass die berechnete Zeitdauer gültig ist.

Eingänge

```
VAR_INPUT
  Position      : LREAL;
  ModuloFactor  : LREAL;
  Options       : ST_NcTimeConversionOptions;
END_VAR
```

| Name | Typ | Beschreibung |
|--------------|---|---|
| Position | LREAL | Absolute Achsposition |
| ModuloFactor | LREAL | Modulo-Teiler, der für die Berechnung verwendet werden soll. ModuloFactor kann mit dem Modulo-Faktor der Achse identisch sein, z. B. 360. Es kann aber auch ein davon abweichender Faktor verwendet werden. |
| Options | ST_NcTimeConversionOptions ▶ 48 | Datenstruktur mit Optionen für die Extrapolation der Position. |

Ein-/Ausgänge

```
VAR_IN_OUT
  Axis      : AXIS_REF;
END_VAR
```

| Name | Typ | Beschreibung |
|------|-----------------|---|
| Axis | <u>AXIS_REF</u> | Die Achsdatenstruktur vom Typ AXIS_REF adressiert eine Achse eindeutig im System. Sie enthält unter anderem den aktuellen Status der Achse, wie Position, Geschwindigkeit oder Fehlerzustand. |

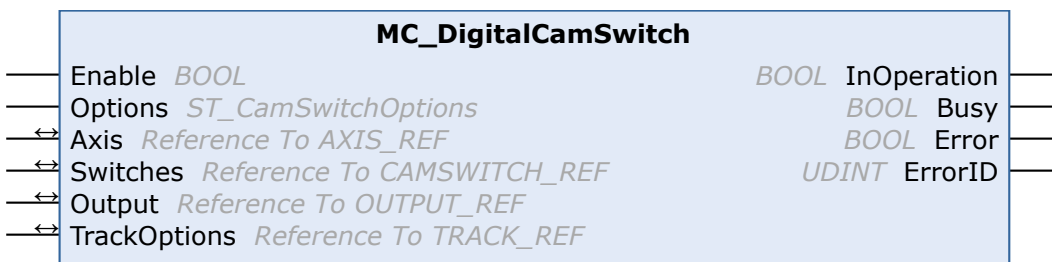
Ausgänge

```
VAR_OUTPUT
  Duration      : DINT;
  AbsolutePosition : LREAL;
  TimeOfPosition : T_DCTIME32;
  Error         : BOOL;
  ErrorID       : UDINT;
END_VAR
```

| Name | Typ | Beschreibung |
|------------------|------------|---|
| Duration | DINT | Zeitdauer in Nanosekunden, nach der die Position erreicht sein wird. Duration ist ein Differenzwert aus zwei Variablen des Typs T_DCTIME32 Distributed Clock System Time. |
| AbsolutePosition | LREAL | Absolute Position (nicht modulo), die mit der Modulo-Position und der ermittelten Zeit korrespondiert. |
| TimeOfPosition | T_DCTIME32 | Distributed Clock System Time, zu der die Position erreicht sein wird bzw. zu der sie erreicht wurde. DcTime enthält die unteren 32 Bit der vollständigen DcTime und deckt damit einen Zeitbereich von +/- 2 Sekunden um den aktuellen Zeitpunkt ab. |
| Error | BOOL | Wird im Fehlerfall TRUE. |
| ErrorID | UDINT | Liefert bei einem gesetzten Error-Ausgang die Fehlernummer. |

4.1.1.2 Nockenschaltwerk

4.1.1.2.1 MC_DigitalCamSwitch



Der Funktionsbaustein MC_DigitalCamSwitch ist ein digitales Nockenschaltwerk mit einer oder mehreren Nocken auf einer digitalen Ausgabespur.

Durch entsprechende Parametrierung können Positions-, Zeit- und Bremsnocken realisiert werden. Weitere Ausgabespuren werden mit unabhängigen Instanzen des Funktionsbausteins realisiert.

Die Output-Datenstruktur enthält neben dem Schaltzustand des digitalen Ausgangs genaue Zeitinformationen über die nächsten Schaltvorgänge. Mit dieser Information kann die eigentliche Ausgabe an einer XFC-Ausgangsklemme mit einem nachgeschalteten Funktionsbaustein erfolgen ([XFC_EL2252_V2](#) [[▶ 34](#)] oder [XFC_EL2262](#) [[▶ 37](#)]).

 **Eingänge**

```
VAR_INPUT
  Enable : BOOL;
  Options : ST_CamSwitchOptions;
END_VAR
```

| Name | Typ | Beschreibung |
|---------|--|--|
| Enable | BOOL | Über den Eingang Enable wird das Nockenschaltwerk aktiviert. Der Ausgangszustand bleibt unverändert, solange Enable FALSE ist. |
| Options | ST_CamSwitchOptions [► 49] | Optionaler Parameter |

 **Ein-/Ausgänge**

```
VAR_IN_OUT
  Axis : AXIS_REF;
  Switches : CAMSWITCH_REF;
  Output : OUTPUT_REF;
  TrackOptions : TRACK_REF;
END_VAR
```

| Name | Typ | Beschreibung |
|--------------|--------------------------------------|---|
| Axis | AXIS_REF | Die Achsdatenstruktur vom Typ AXIS_REF adressiert eine Achse eindeutig im System. Sie enthält unter anderem den aktuellen Status der Achse, wie Position, Geschwindigkeit oder Fehlerzustand. |
| Switches | CAMSWITCH_REF [► 42] | Die Datenstruktur Switches enthält einen Verweis auf die Parametrierung aller Nocken auf der Nockenspur. |
| Output | OUTPUT_REF [► 43] | Die Datenstruktur Output enthält den berechneten Zustand des digitalen Ausgangs und die zugehörigen Zeitstempel für die Ausgabe an einer digitalen XFC Ausgangsklemme. |
| TrackOptions | TRACK_REF [► 44] | Die Datenstruktur TrackOptions enthält die Parametrierung der Nockenspur. |

 **Ausgänge**

```
VAR_OUTPUT
  InOperation : BOOL;
  Busy : BOOL;
  Error : BOOL;
  ErrorID : UDINT;
END_VAR
```

| Name | Typ | Beschreibung |
|-------------|-------|--|
| InOperation | BOOL | InOperation ist TRUE, solange das Nockenschaltwerk aktiv ist und die Nockenspur gemäß der Nockenparametrierung berechnet wird. |
| Busy | BOOL | Busy ist TRUE solange die Funktion des Bausteins nicht beendet ist. |
| Error | BOOL | Wird im Fehlerfall TRUE. |
| ErrorID | UDINT | Liefert bei einem gesetzten Error-Ausgang die Fehlernummer. |

Beispiel für zwei digitale Nockenspuren

| CamSwitchRefTrack1 : CAMSWITCH_REF | |
|------------------------------------|------------------------------|
| | Value |
| NumberOfSwitches | 3 |
| pSwitches | ADR(CamSwitchArrayTrack1) |
| SizeOfSwitches | SIZEOF(CamSwitchArrayTrack1) |

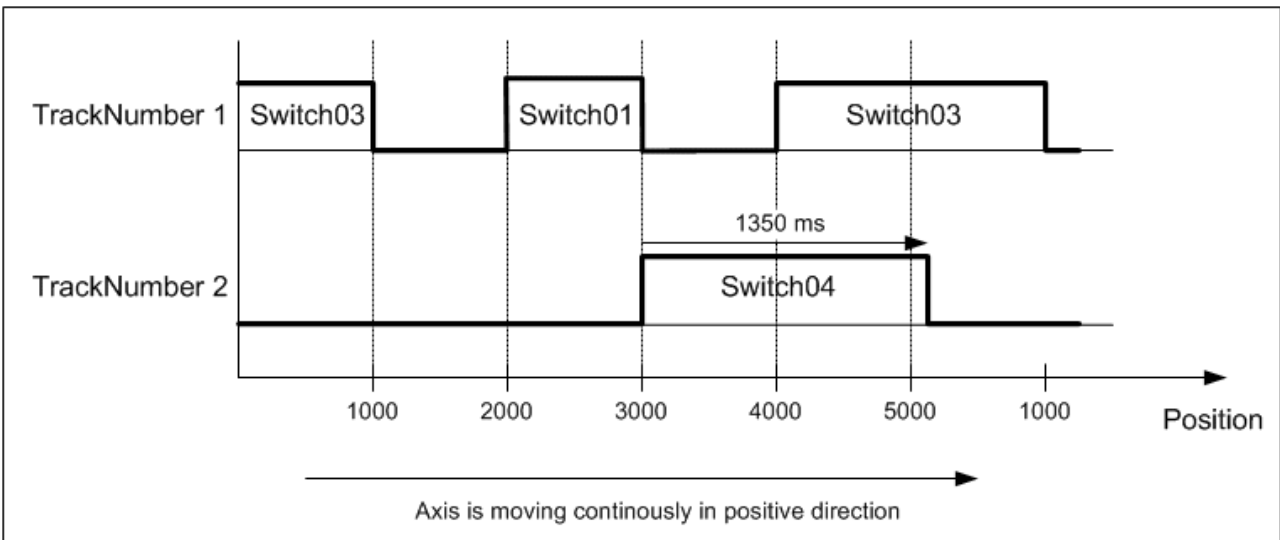
| CamSwitchArrayTrack1 : Array [1..n] OF MC_CamSwitch | | | | | |
|---|----------|----------|----------|-----|----------|
| | Switch 1 | Switch 2 | Switch 3 | ... | Switch n |
| FirstOnPosition | 2000 | 2500 | 4000 | | |
| LastOnPosition | 3000 | 3000 | 1000 | | |
| AxisDirection | POSITIVE | NEGATIVE | BOTH | | |
| CamSwitchMode | POSITION | POSITION | POSITION | | |
| Duration [s] | — | — | — | | |

| CamSwitchRefTrack2 : CAMSWITCH_REF | |
|------------------------------------|------------------------------|
| | Value |
| NumberOfSwitches | 1 |
| pSwitches | ADR(CamSwitchArrayTrack2) |
| SizeOfSwitches | SIZEOF(CamSwitchArrayTrack2) |

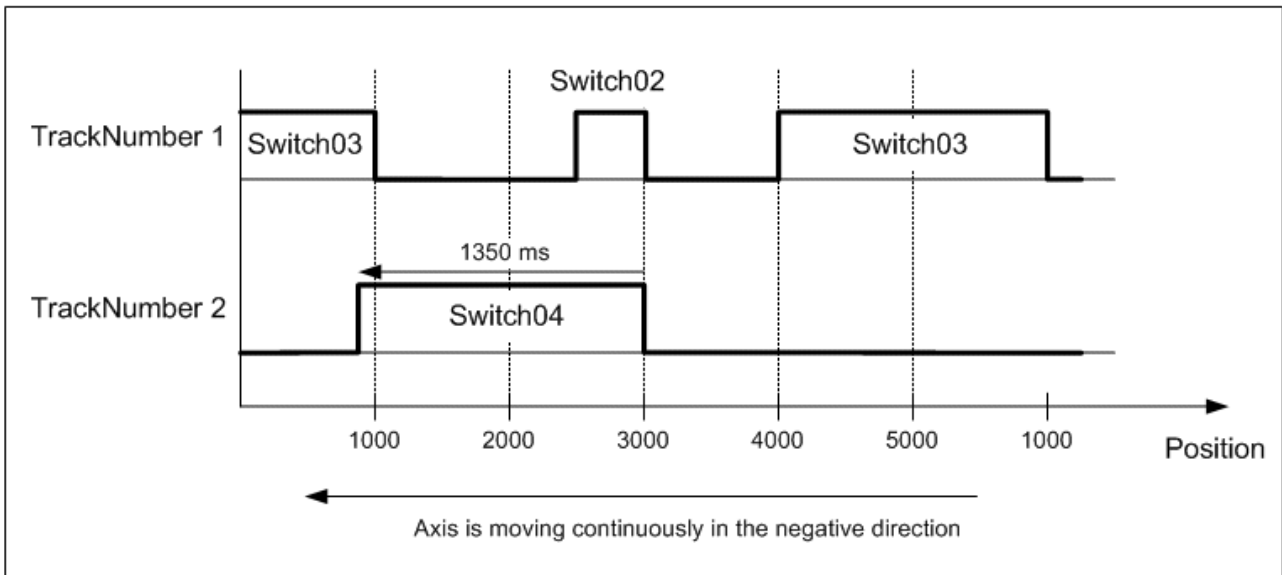
| CamSwitchArrayTrack2 : Array [1..m] OF MC_CamSwitch | | | |
|---|----------|-----|----------|
| | Switch 1 | ... | Switch m |
| FirstOnPosition | 3000 | | |
| LastOnPosition | — | | |
| AxisDirection | BOTH | | |
| CamSwitchMode | TIME | | |
| Duration [s] | 1,350 | | |

Mit den Nockendaten ergeben sich die folgenden Schaltdiagramme. Die Schaltfolge wird ohne jegliche Zeitkompensation und Hysterese dargestellt und ist aufgrund der Nockendaten für beide Fahrrichtungen unterschiedlich.

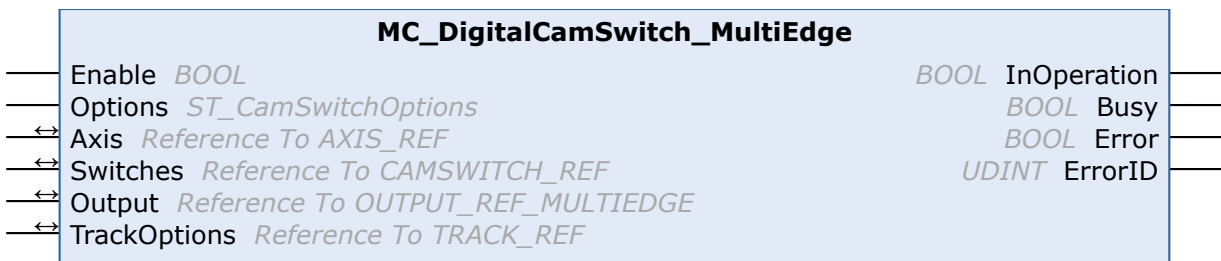
Schaltfolge für positive Fahrrichtung



Schaltfolge für negative Fahrrichtung



4.1.1.2 MC_DigitalCamSwitch_MultiEdge



Der Funktionsbaustein MC_DigitalCamSwitch_MultiEdge ist ein digitales Nockenschaltwerk mit einer oder mehreren Nocken auf einer digitalen Ausgabespur. Der Funktionsbaustein ergänzt den Baustein MC_DigitalCamSwitch um die Eigenschaft, mehrere Schaltvorgänge während eines SPS-Zyklus durchführen zu können. Die Schaltvorgänge werden durch Positionsnocken definiert. Weitere Ausgabespuren werden mit unabhängigen Instanzen des Funktionsbausteins realisiert.

Die Output-Datenstruktur enthält neben dem Schaltzustand des digitalen Ausgangs genaue Zeitinformationen über die nächsten Schaltvorgänge. Mit dieser Information kann die eigentliche Ausgabe an einer XFC-Multitimestamp-Ausgangsklemme mit einem nachgeschalteten Funktionsbaustein erfolgen (XFC_EL1259_MultiEdge, XFC_EL2212_MultiEdge [▶ 32], XFC_EL2258_MultiEdge [▶ 36] oder XFC_EL2262_MultiEdge [▶ 38]).

i Zeitnocken und Bremsnocken können mit dem Baustein MC_DigitalCamSwitch_MultiEdge nicht verwendet werden. Klemmen ohne Multiti-mestamp-Funktionalität sind in Verbindung mit diesem Baustein nicht geeignet.

Eingänge

```
VAR_INPUT
  Enable : BOOL;
  Options : ST_CamSwitchOptions;
END_VAR
```

| Name | Typ | Beschreibung |
|---------|----------------------------|--|
| Enable | BOOL | Über den Eingang Enable wird das Nockenschaltwerk aktiviert. Der Ausgangszustand bleibt unverändert, solange Enable FALSE ist. |
| Options | ST_CamSwitchOptions [▶ 49] | Optionale Parameter |

Ein-/Ausgänge

```
VAR_IN_OUT
  Axis      : AXIS_REF;
  Switches  : CAMSWITCH_REF;
  Output    : OUTPUT_REF_MULTIEDGE;
  TrackOptions : TRACK_REF;
END_VAR
```

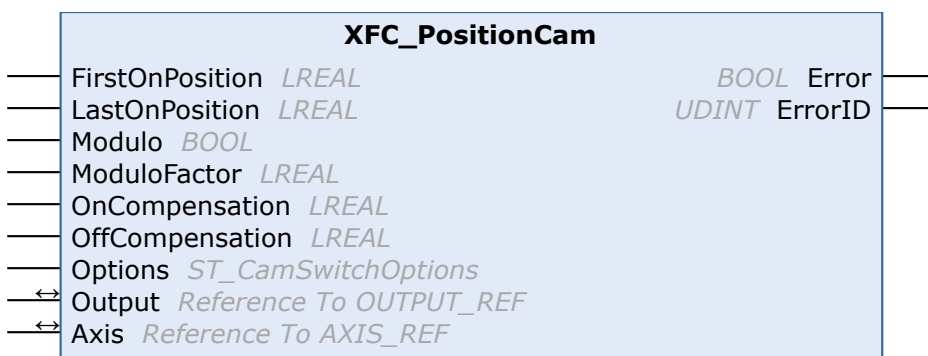
| Name | Typ | Beschreibung |
|--------------|---|--|
| Axis | AXIS_REF | Die Achsdatenstruktur vom Typ <code>AXIS_REF</code> adressiert eine Achse eindeutig im System. Sie enthält unter anderem den aktuellen Status der Achse, wie Position, Geschwindigkeit oder Fehlerzustand. |
| Switches | CAMSWITCH_REF [► 42] | Die Datenstruktur <code>Switches</code> enthält einen Verweis auf die Parametrierung aller Nocken auf der Nockenspur. |
| Output | OUTPUT_REF_MULTIEDGE [► 43] | Die Datenstruktur <code>Output</code> enthält den berechneten Zustand des digitalen Ausgangs und die zugehörigen Zeitstempel für die Ausgabe an einer digitalen XFC Ausgangsklemme. |
| TrackOptions | TRACK_REF [► 44] | Die Datenstruktur <code>TrackOptions</code> enthält die Parametrierung der Nockenspur. |

Ausgänge

```
VAR_OUTPUT
  InOperation : BOOL;
  Busy        : BOOL;
  Error       : BOOL;
  ErrorID     : UDINT;
END_VAR
```

| Name | Typ | Beschreibung |
|-------------|-------|--|
| InOperation | BOOL | InOperation ist TRUE, solange das Nockenschaltwerk aktiv ist und die Nockenspur gemäß der Nockenparametrierung berechnet wird. |
| Busy | BOOL | Busy ist TRUE solange die Funktion des Bausteins nicht beendet ist. |
| Error | BOOL | Wird im Fehlerfall TRUE. |
| ErrorID | UDINT | Liefert bei einem gesetzten Error-Ausgang die Fehlernummer. |

4.1.1.2.3 XFC_PositionCam



Der Funktionsbaustein `XFC_PositionCam` realisiert eine Positionsnocke, die einen digitalen Ausgang positionabhängig ein- und ausschaltet.

Im Gegensatz zum digitalen Nockenschaltwerk `MC_DigitalCamSwitch` [► 16] schaltet der Funktionsbaustein genau eine Nocke auf einer digitalen Ausgangsspur. Der Baustein ist damit einfacher parametrierbar, kann aber nicht eingesetzt werden, wenn mehrere Nocken auf einer Ausgangsspur notwendig sind.

Die Output-Datenstruktur enthält neben dem Schaltzustand des digitalen Ausgangs genaue Zeitinformationen über die nächsten Schaltvorgänge. Mit dieser Information kann die eigentliche Ausgabe an einer XFC-Ausgangsklemme mit einem nachgeschalteten Funktionsbaustein erfolgen (XFC_EL2252_V2 [[▶ 34](#)] oder XFC_EL2262 [[▶ 37](#)]).

 **Eingänge**

```
VAR_INPUT
  FirstOnPosition : LREAL;
  LastOnPosition : LREAL;
  Modulo          : BOOL;
  ModuloFactor   : LREAL := 360;
  OnCompensation : LREAL;
  OffCompensation : LREAL;
  Options        : ST_CamSwitchOptions;
END_VAR
```

| Name | Typ | Beschreibung |
|-----------------|--|---|
| FirstOnPosition | LREAL | Erste Position ab der die Nocke eingeschaltet ist. |
| LastOnPosition | LREAL | Letzte Position bis zu der die Nocke eingeschaltet ist. Die Nockenfunktion ist invertiert, wenn LastOnPosition < FirstOnPosition |
| Modulo | BOOL | Wenn Modulo TRUE ist, werden alle Positionen modulo interpretiert. Die Nockenfunktion wird dadurch zyklisch wiederholt. Zur Berechnung des Modulo-Zyklus wird der Parameter ModuloFactor verwendet. |
| ModuloFactor | LREAL | ModuloFactor gibt die Länge eines Modulo-Zyklus in der Positionseinheit der Achse an und wird nur verwendet, wenn Modulo TRUE ist. |
| OnCompensation | LREAL | Kompensationszeit für die steigende Flanke der Nocke in [s]. Der Schaltzeitpunkt wird bei negativen Werten vonOnCompensation vorgezogen und sonst verzögert. Der hier parametrisierte Wert OnCompensation hat Vorrang vor <u>TRACK_REF</u> [▶ 44]. |
| OffCompensation | LREAL | Kompensationszeit für die fallende Flanke der Nocke in [s]. Der Schaltzeitpunkt wird bei negativen Werten vonOffCompensation vorgezogen und sonst verzögert. Der hier parametrisierte Wert OffCompensation hat Vorrang vor <u>TRACK_REF</u> [▶ 44]. |
| Options | ST_CamSwitchOptions [▶ 49] | Optionale Parameter |

 /  **Ein-/Ausgänge**

```
VAR_IN_OUT
  Output : OUTPUT_REF;
  Axis   : AXIS_REF;
END_VAR
```

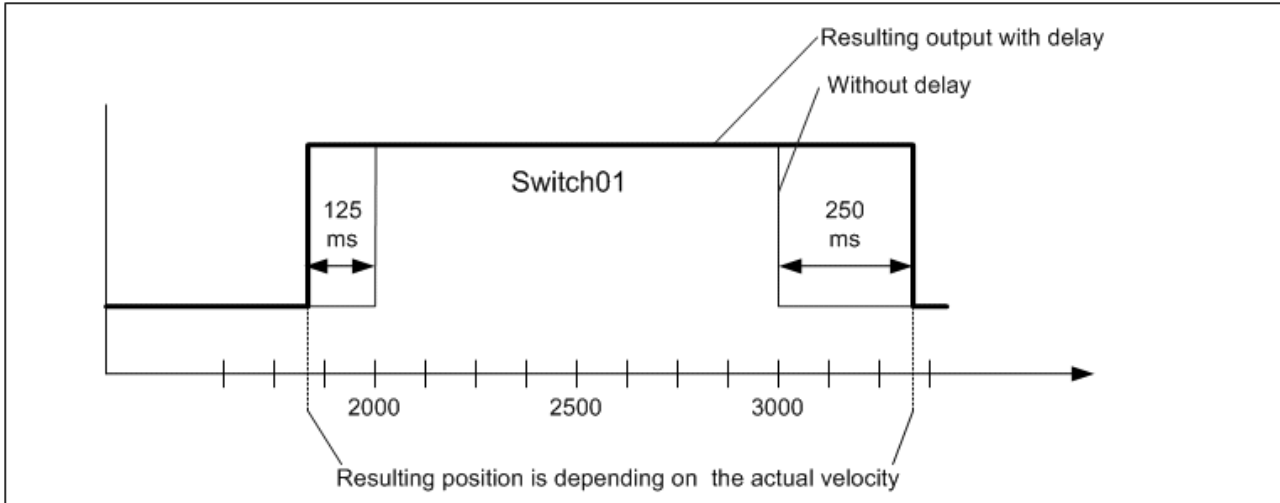
| Name | Typ | Beschreibung |
|--------|--|--|
| Output | <u>OUTPUT_REF</u> [▶ 43] | Die Datenstruktur Output enthält den berechneten Zustand des digitalen Ausgangs und die zugehörigen Zeitstempel für die Ausgabe an einer digitalen XFC Ausgangsklemme |
| Axis | <u>AXIS_REF</u> | Die Achsdatenstruktur vom Typ <u>AXIS_REF</u> adressiert eine Achse eindeutig im System. Sie enthält unter anderem den aktuellen Status der Achse, wie Position, Geschwindigkeit oder Fehlerzustand. |

Ausgänge

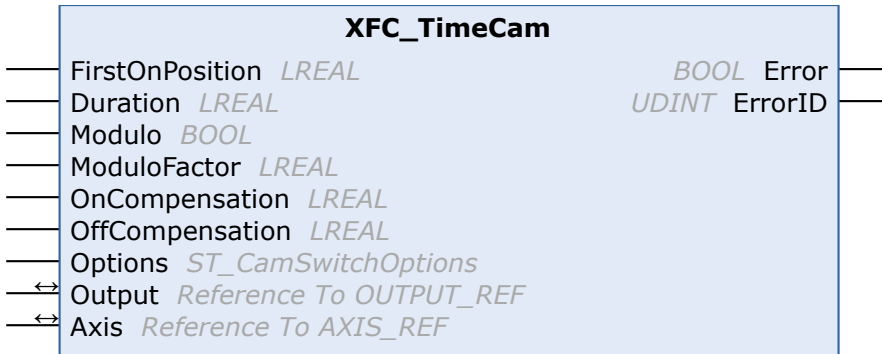
```
VAR_OUTPUT
  Error      : BOOL;
  ErrorID    : UDINT;
END_VAR
```

| Name | Typ | Beschreibung |
|---------|-------|--|
| Error | BOOL | Wird im Fehlerfall TRUE. |
| ErrorID | UDINT | Liefert bei einem gesetzten Error-Ausgang eine Fehlernummer. |

Wirkungsweise der Zeitkompensation



4.1.1.2.4 XFC_TimeCam



Der Funktionsbaustein XFC_TimeCam realisiert eine Zeitnocke, die einen digitalen Ausgang positionsabhängig einschaltet und nach einer Zeit ausschaltet.

Im Gegensatz zum digitalen Nockenschaltwerk [MC_DigitalCamSwitch](#) [► 16] schaltet der Funktionsbaustein genau eine Nocke auf einer digitalen Ausgangsspur. Der Baustein ist damit einfacher parametrierbar, kann aber nicht eingesetzt werden, wenn mehrere Nocken auf einer Ausgangsspur notwendig sind.

Die Output-Datenstruktur enthält neben dem Schaltzustand des digitalen Ausgangs genaue Zeitinformationen über die nächsten Schaltvorgänge. Mit dieser Information kann die eigentliche Ausgabe an einer XFC-Ausgangsklemme mit einem nachgeschalteten Funktionsbaustein erfolgen ([XFC_EL2252_V2](#) [► 34] oder [XFC_EL2262](#) [► 37]).

Eingänge

```
VAR_INPUT
  FirstOnPosition : LREAL;
  Duration        : LREAL;
  Modulo          : BOOL;
```

```

ModuloFactor      : LREAL := 360;
OnCompensation   : LREAL;
OffCompensation   : LREAL;
Options           : ST_CamSwitchOptions;
END_VAR
    
```

| Name | Typ | Beschreibung |
|-----------------|--|--|
| FirstOnPosition | LREAL | Erste Position, ab der die Nocke eingeschaltet ist. |
| Duration | LREAL | Einschaltdauer der Nocke in [s]. |
| Modulo | BOOL | Wenn Modulo TRUE ist, werden alle Positionen modulo interpretiert. Die Nockenfunktion wird dadurch zyklisch wiederholt. Zur Berechnung des Modulo-Zyklus wird der Parameter ModuloFactor verwendet. |
| ModuloFactor | LREAL | ModuloFactor gibt die Länge eines Modulo-Zyklus in der Positionseinheit der Achse an und wird nur verwendet, wenn Modulo TRUE ist. |
| OnCompensation | LREAL | Kompensationszeit für die steigende Flanke der Nocke in [s]. Der Schaltzeitpunkt wird bei negativen Werten von OnCompensation vorgezogen und sonst verzögert. Der hier parametrisierte Wert OnCompensation hat Vorrang vor TRACK_REF [▶ 44] . |
| OffCompensation | LREAL | Kompensationszeit für die fallende Flanke der Nocke in [s]. Der Schaltzeitpunkt wird bei negativen Werten von OffCompensation vorgezogen und sonst verzögert. Der hier parametrisierte Wert OffCompensation hat Vorrang vor TRACK_REF [▶ 44] . |
| Options | ST_CamSwitchOptions [▶ 49] | Optionale Parameter |

 **Ein-/Ausgänge**

```

VAR_IN_OUT
    Output : OUTPUT_REF;
    Axis   : AXIS_REF;
END_VAR
    
```

| Name | Typ | Beschreibung |
|--------|-----------------------------------|---|
| Output | OUTPUT_REF [▶ 43] | Die Datenstruktur Output enthält den berechneten Zustand des digitalen Ausgangs und die zugehörigen Zeitstempel für die Ausgabe an einer digitalen XFC-Ausgangsklemme |
| Axis | AXIS_REF | Die Achsdatenstruktur vom Typ AXIS_REF adressiert eine Achse eindeutig im System. Sie enthält unter anderem den aktuellen Status der Achse, wie Position, Geschwindigkeit oder Fehlerzustand. |

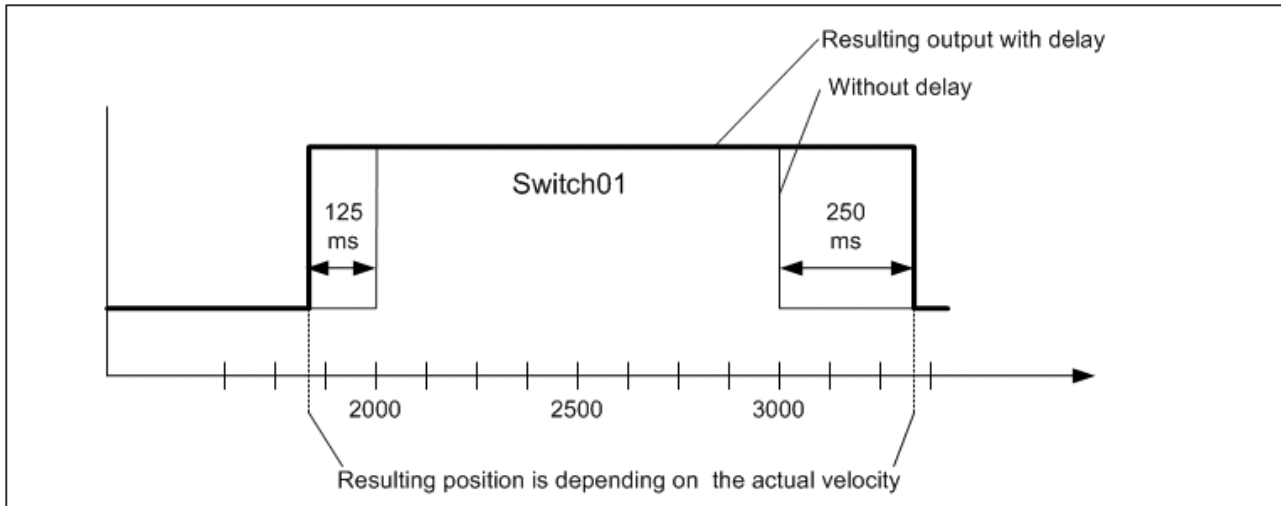
 **Ausgänge**

```

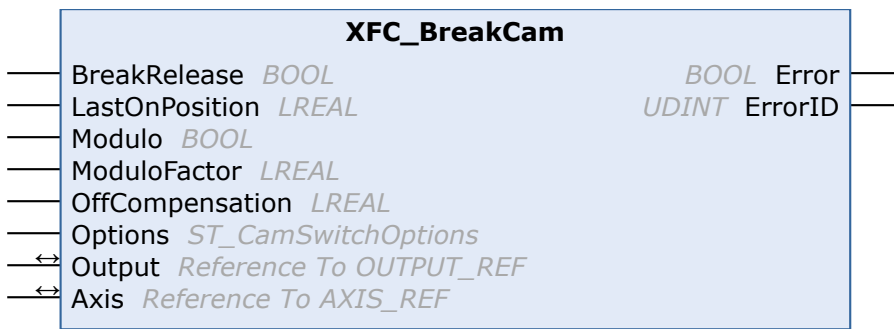
VAR_OUTPUT
    Error      : BOOL;
    ErrorID    : UDINT;
END_VAR
    
```

| Name | Typ | Beschreibung |
|---------|-------|--|
| Error | BOOL | Wird im Fehlerfall TRUE. |
| ErrorID | UDINT | Liefert bei einem gesetzten Error-Ausgang eine Fehlernummer. |

Wirkungsweise der Zeitkompensation



4.1.1.2.5 XFC_BreakCam



Der Funktionsbaustein XFC_BreakCam realisiert eine Bremsnocke, die einen digitalen Ausgang positionsabhängig ausschaltet, sobald die Bremsfreigabe BreakRelease entzogen wird.

Die Output-Datenstruktur enthält neben dem Schaltzustand des digitalen Ausgangs genaue Zeitinformationen über die nächsten Schaltvorgänge. Mit dieser Information kann die eigentliche Ausgabe an einer XFC-Ausgangsklemme mit einem nachgeschalteten Funktionsbaustein erfolgen ([XFC_EL2252 V2](#) [► 34] oder [XFC_EL2262](#) [► 37]).

🔌 Eingänge

```
VAR_INPUT
  BreakRelease      : LREAL;
  LastOnPosition    : LREAL;
  Modulo            : BOOL;
  ModuloFactor      : LREAL := 360;
  OffCompensation   : LREAL;
  Options           : ST_CamSwitchOptions;
END_VAR
```

| Name | Typ | Beschreibung |
|----------------|-------|---|
| BreakRelease | LREAL | Bremsfreigabe. Solange BreakRelease TRUE ist, bleibt die Nocke aktiv. Nachdem BreakRelease FALSE wird, wird die Nocke an der Position LastOnPosition ausgeschaltet. |
| LastOnPosition | LREAL | Letzte Position bis zu der die Nocke eingeschaltet ist. |
| Modulo | BOOL | Wenn Modulo TRUE ist, werden alle Positionen modulo interpretiert. Die Nockenfunktion wird dadurch zyklisch wiederholt. Zur Berechnung des Modulo-Zyklus wird der Parameter ModuloFactor verwendet. |

| Name | Typ | Beschreibung |
|-----------------|----------------------------|--|
| ModuloFactor | LREAL | ModuloFactor gibt die Länge eines Modulo-Zyklus in der Positionseinheit der Achse an und wird nur verwendet, wenn Modulo=TRUE ist. |
| OffCompensation | LREAL | Kompensationszeit für die fallende Flanke der Nocke in [s]. Der Schaltzeitpunkt wird bei negativen Werten vonOffCompensation vorgezogen und sonst verzögert. Der hier parametrisierte Wert OffCompensation hat Vorrang vor TRACK_REF [▶ 44]. |
| Options | ST_CamSwitchOptions [▶ 49] | Optionale Parameter |

Ein-/Ausgänge

```
VAR_IN_OUT
  Output : OUTPUT_REF;
  Axis   : AXIS_REF;
END_VAR
```

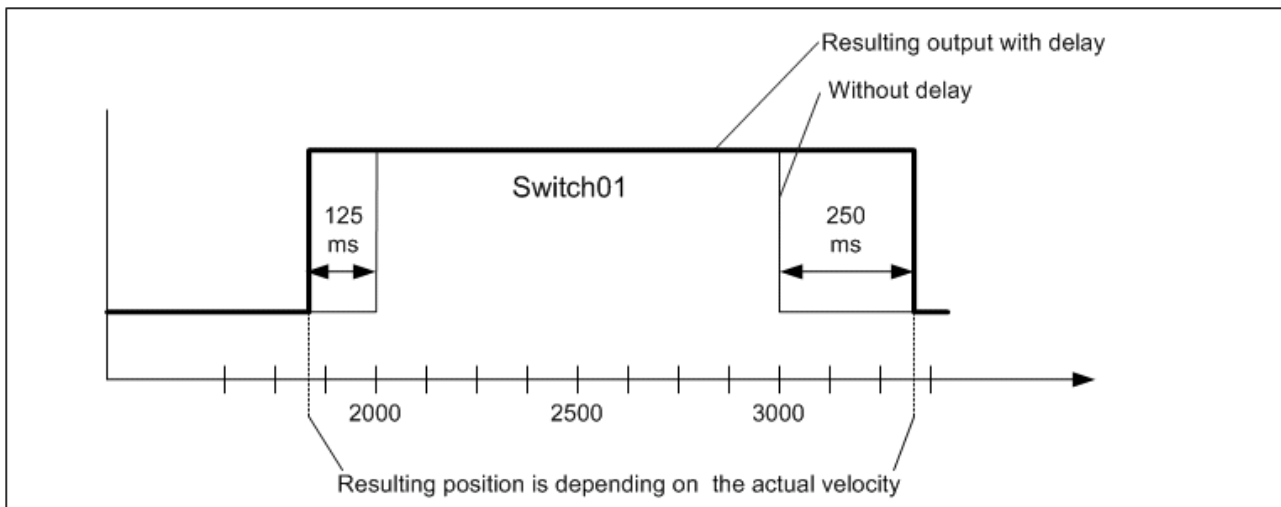
| Name | Typ | Beschreibung |
|--------|-------------------|---|
| Output | OUTPUT_REF [▶ 43] | Die Datenstruktur Output enthält den berechneten Zustand des digitalen Ausgangs und die zugehörigen Zeitstempel für die Ausgabe an einer digitalen XFC-Ausgangsklemme. |
| Axis | AXIS_REF | Die Achsdatenstruktur vom Typ AXIS_REF adressiert eine Achse eindeutig im System. Sie enthält unter anderem den aktuellen Status der Achse, wie Position, Geschwindigkeit oder Fehlerzustand. |

Ausgänge

```
VAR_OUTPUT
  Error      : BOOL;
  ErrorID    : UDINT;
END_VAR
```

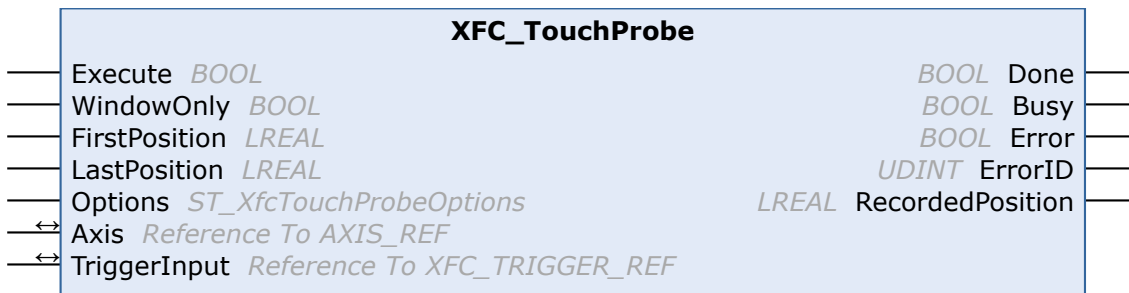
| Name | Typ | Beschreibung |
|---------|-------|--|
| Error | BOOL | Wird im Fehlerfall TRUE. |
| ErrorID | UDINT | Liefert bei einem gesetzten Error-Ausgang eine Fehlernummer. |

Wirkungsweise der Zeitkompensation



4.1.1.3 TouchProbe

4.1.1.3.1 XFC_TouchProbe



Der Funktionsbaustein XFC_TouchProbe erfasst eine Achsposition zum Zeitpunkt der Flanke eines digitalen Eingangssignals (Messtasterfunktion).

Das digitale Eingangssignal wird mit einer XFC-Eingangsklemme (z. B. EL1252) mit Zeitstempeln für die fallende und steigende Signalflanke erfasst. Der Funktionsbaustein ermittelt die Achsposition, bei der der Flankenwechsel stattgefunden hat und gibt sie als RecordedPosition aus.

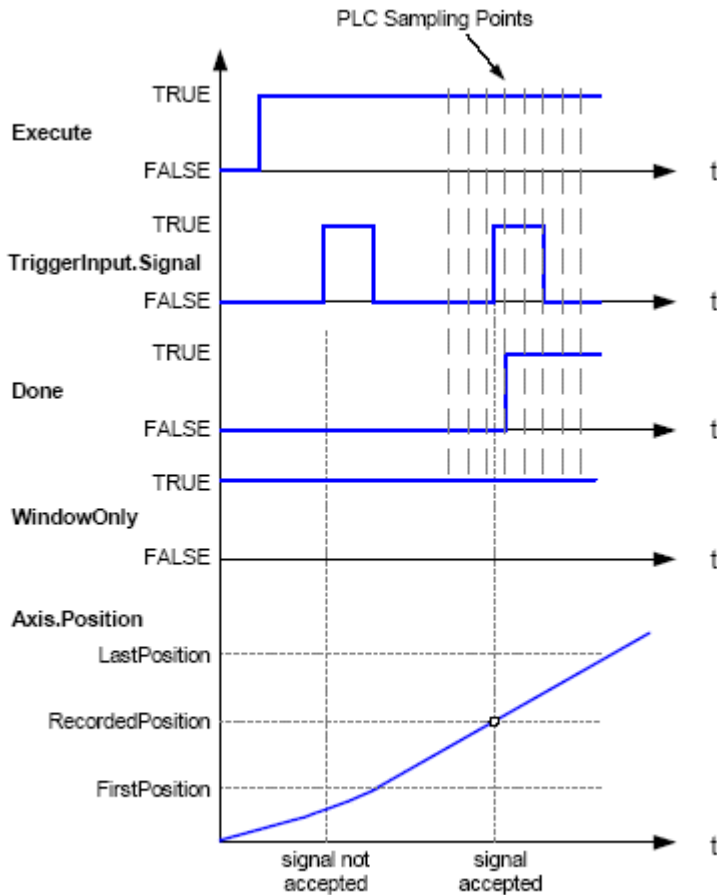
Im Gegensatz zur klassischen TouchProbe-Funktion MC_TouchProbe ist der digitale Eingang nicht direkt an die Antriebshardware gebunden. Über den Zeitstempel des Eingangs kann die Position jeder EtherCAT- oder Sercos-Achse im System aufgezeichnet werden, die mittels Totzeitkompensation [▶ 9] exakt synchronisiert ist.

Der Funktionsbaustein kann freilaufend oder im Single-Shot-Betrieb verwendet werden. Im freilaufenden Modus wird jede Flanke des Eingangssignals erfasst (maximal eine Flanke pro SPS-Zyklus). Im Single-Shot-Betrieb wird nur einmalig die nächste Flanke erfasst bis der Funktionsbaustein neu getriggert wird.

Über die optionale Fensterfunktion können Signalflanken außerhalb des definierten Positionsfilters ignoriert werden.

Im Beispiel [▶ 79] wird gezeigt, wie der Funktionsbaustein freilaufend (`FreeRun = TRUE`) oder im Single-Shot-Mode (`FreeRun = FALSE`) verwendet werden kann.

Signalverlauf



Timing example TouchProbe

Eingänge

```

VAR_INPUT
  Execute      : BOOL;
  WindowOnly  : BOOL;
  FirstPosition : LREAL;
  LastPosition : LREAL;
  Options     : ST_XfcTouchProbeOptions;
END_VAR
    
```

| Name | Typ | Beschreibung |
|------------|------|--|
| Execute | BOOL | Wenn Execute aktiv ist, wird die Achsposition an der definierten Signalfanke des Eingangssignals erfasst. Eine fallende Flanke an Execute beendet den Vorgang sofort. Abhängig von der Konfiguration in TriggerInput.FreeRun wird einmalig die nächste Signalfanke erfasst und ausgewertet. Wenn FreeRun TRUE ist, wird fortlaufend mit jeder definierten Flanke des Eingangssignals ein neuer Positionswert erfasst, während Execute TRUE bleibt. |
| WindowOnly | BOOL | Wenn diese Option aktiv ist, wird nur eine Position innerhalb des Fensters zwischen FirstPosition und LastPosition erfasst. Positionen außerhalb des Fensters werden verworfen. Erst wenn die erfasste Position innerhalb des Fensters liegt, wird Done TRUE. Das Erfassungsfenster kann absolut oder modulo interpretiert werden. Dazu ist das Flag ModuloPositions [▶ 45] in der Struktur TriggerInput [▶ 45] entsprechend zu setzen. Bei absoluten Positionen gibt es exakt ein |

| Name | Typ | Beschreibung |
|---------------|-----------------------------------|--|
| | | Fenster. Bei Modulo-Positionen wiederholt sich das Fenster innerhalb des in den Achsparametern festgelegten Modulozyklus (z. B. 0 bis 360 Grad). |
| FirstPosition | LREAL | Anfangsposition des Erfassungsfensters, wenn WindowOnly TRUE ist. Diese Position kann absolut oder modulo interpretiert werden. Dazu ist in der Struktur TriggerInput (siehe unten) das Flag <u>ModuloPositions</u> [► 45] entsprechend zu setzen. |
| LastPosition | LREAL | Endposition des Erfassungsfensters, wenn WindowOnly TRUE ist. Diese Position kann absolut oder modulo interpretiert werden. Dazu ist in der Struktur TriggerInput (siehe unten) das Flag <u>ModuloPositions</u> [► 45] entsprechend zu setzen. |
| Options | ST_XfcTouchProbeOptions [► 49] | Optionale Parameter |

Ein-/Ausgänge

```
VAR_IN_OUT
  Axis      : AXIS_REF;
  TriggerInput : XFC_TRIGGER_REF;
END_VAR
```

| Name | Typ | Beschreibung |
|--------------|------------------------|---|
| Axis | AXIS_REF | Die Achsdatenstruktur vom Typ AXIS_REF adressiert eine Achse eindeutig im System. Sie enthält unter anderem den aktuellen Status der Achse, wie Position, Geschwindigkeit oder Fehlerzustand. |
| TriggerInput | XFC_TRIGGER_REF [► 45] | TriggerInput ist eine Datenstruktur zur Beschreibung der Trigger-Quelle und zur Einspeisung von Zustand und Zeitstempel eines digitalen Eingangssignals. Diese Datenstruktur wird vom Anwender gefüllt. |

Ausgänge

```
VAR_OUTPUT
  Done      : BOOL;
  Busy      : BOOL;
  Error     : BOOL;
  ErrorID   : UDINT;
  RecordedPosition : LREAL;
END_VAR
```

| Name | Typ | Beschreibung |
|------------------|-------|---|
| Done | BOOL | Der Wert RecordedPosition ist gültig. Falls TriggerInput.FreeRun TRUE ist, bleibt Done nur für einen SPS-Zyklus TRUE und wird selbstständig zurückgesetzt, da TouchProbe automatisch neu aktiviert wird. |
| Busy | BOOL | Wird TRUE sobald der Baustein aktiv ist und wird FALSE nachdem er sich wieder im Grundzustand befindet. Falls TriggerInput.FreeRun TRUE ist, bleibt Busy permanent TRUE, auch wenn Done oder Error TRUE werden, da TouchProbe automatisch neu aktiviert wird. |
| Error | BOOL | Wird im Fehlerfall TRUE. |
| ErrorID | UDINT | Liefert bei einem gesetzten Error-Ausgang die Fehlernummer |
| RecordedPosition | LREAL | Erfasste Achsposition zum Zeitpunkt des Trigger-Signals. Wenn TriggerInput.FreeRun TRUE ist, läuft der Funktionsbaustein freilaufend, so dass jede gültige |

| Name | Typ | Beschreibung |
|------|-----|--|
| | | Änderung des Eingangssignals zu einer neuen RecordedPosition führt. Die Position kann jeweils ausgewertet werden, wenn Done TRUE wird. |

4.1.1.3.2 XFC_EL1258_TouchProbe



Der Funktionsbaustein XFC_EL1258_TouchProbe erfasst die Achspositionen zum Zeitpunkt der Flanken eines digitalen Eingangssignals (Messtasterfunktion).

Das digitale Eingangssignal wird mit einer XFC-Eingangsklemme EL1258 mit Zeitstempeln für fallende und steigende Signalfanken erfasst. Der Funktionsbaustein ermittelt die Achspositionen, bei der ein Flankenwechsel stattgefunden hat und gibt sie als RecordedPositions aus.

Im Gegensatz zur klassischen TouchProbe-Funktion MC_TouchProbe ist der digitale Eingang nicht direkt an die Antriebshardware gebunden. Über die Zeitstempel des Eingangs kann die Position jeder EtherCAT oder Sercos-Achse im System aufgezeichnet werden, die mittels Totzeitkompensation exakt synchronisiert ist.

Der Funktionsbaustein kann freilaufend oder im Single-Shot-Betrieb verwendet werden. Im freilaufenden Modus wird jede Flanke des Eingangssignals erfasst (auch mehrere Flanken pro SPS-Zyklus). Im Single-Shot-Betrieb werden nur einmalig die nächsten Flanken erfasst bis der Funktionsbaustein neu getriggert wird.

Über die optionale Fensterfunktion können Signalfanken außerhalb des definierten Positionsfilters ignoriert werden.

Eingänge

```

VAR_INPUT
  Execute      : BOOL;
  WindowOnly  : BOOL;
  FirstPosition : LREAL;
  LastPosition : LREAL;
  Options      : ST_XfcTouchProbeOptions;
  EL1258      : EL1258_MT_IoInterface;
END_VAR
    
```

| Name | Typ | Beschreibung |
|------------|------|--|
| Execute | BOOL | Wenn Execute aktiv ist, wird die Achsposition an der definierten Signalfanke des Eingangssignals erfasst. Eine fallende Flanke an Execute beendet den Vorgang sofort. Abhängig von der Konfiguration in TriggerInput.FreeRun wird einmalig die nächste Signalfanke erfasst und ausgewertet. Wenn FreeRun TRUE ist, wird fortlaufend mit jeder definierten Flanke des Eingangssignals ein neuer Positionswert erfasst, während Execute TRUE bleibt. |
| WindowOnly | BOOL | Wenn diese Option aktiv ist, wird nur eine Position innerhalb des Fensters zwischen FirstPosition und LastPosition erfasst. Positionen außerhalb des Fensters werden verworfen. Erst wenn die erfasste Position innerhalb des Fensters liegt, wird Done TRUE. Das Erfassungsfenster kann absolut oder modulo interpretiert werden. Dazu ist das Flag <u>ModuloPositions</u> |

| Name | Typ | Beschreibung |
|---------------|-----------------------------------|--|
| | | [▶ 45] in der Struktur TriggerInput [▶ 45] entsprechend zu setzen. Bei absoluten Positionen gibt es exakt ein Fenster. Bei Modulo-Positionen wiederholt sich das Fenster innerhalb des in den Achsparametern festgelegten Modulozyklus (z. B. 0 bis 360 Grad). |
| FirstPosition | LREAL | Anfangsposition des Erfassungsfensters, wenn WindowOnly TRUE ist. Diese Position kann absolut oder modulo interpretiert werden. Dazu ist in der Struktur TriggerInput (siehe unten) das Flag <u>ModuloPositions</u> [▶ 45] entsprechend zu setzen. |
| LastPosition | LREAL | Endposition des Erfassungsfensters, wenn WindowOnly TRUE ist. Diese Position kann absolut oder modulo interpretiert werden. Dazu ist in der Struktur TriggerInput (siehe unten) das Flag <u>ModuloPositions</u> [▶ 45] entsprechend zu setzen. |
| Options | ST_XfcTouchProbeOptions [▶ 49] | Optionale Parameter |
| EL1258 | EL1258_MT_IoInterface [▶ 49] | Prozessabbild der Klemme |

Ein-/Ausgänge

```
VAR_IN_OUT
  Axis          : AXIS_REF;
  TriggerInput : XFC_TRIGGER_REF;
END_VAR
```

| Name | Typ | Beschreibung |
|--------------|--------------------|---|
| Axis | AXIS_REF | Die Achsdatenstruktur vom Typ AXIS_REF adressiert eine Achse eindeutig im System. Sie enthält unter anderem den aktuellen Status der Achse, wie Position, Geschwindigkeit oder Fehlerzustand. |
| TriggerInput | XFC_MT_TRIGGER_REF | TriggerInput ist eine Datenstruktur zur Beschreibung der Trigger-Quelle und zur Einspeisung von Zustand und Zeitstempel eines digitalen Eingangssignals. Diese Datenstruktur wird vom Anwender gefüllt. |

Ausgänge

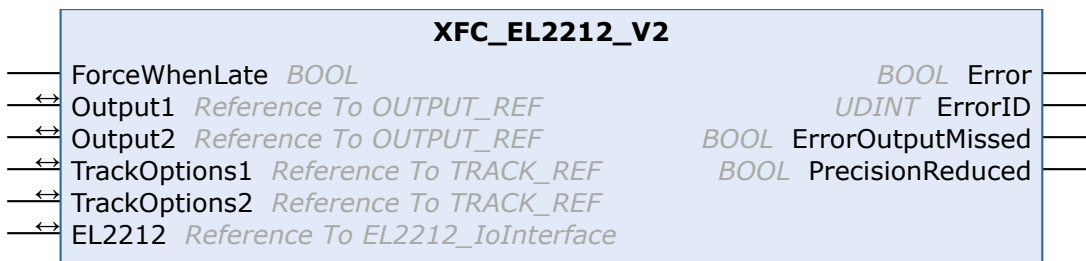
```
VAR_OUTPUT
  Done          : BOOL;
  Busy          : BOOL;
  Error         : BOOL;
  ErrorID       : UDINT;
  RecordedPositions : ARRAY[1..10];
  NoOfRecordedPositions : UDINT;
  Diagnostic    : ST_EL1258_Diagnostics;
END_VAR
```

| Name | Typ | Beschreibung |
|-------|------|---|
| Done | BOOL | Der Wert RecordedPosition ist gültig. Falls TriggerInput.FreeRun TRUE ist, bleibt Done nur für einen SPS-Zyklus TRUE und wird selbstständig zurückgesetzt, da TouchProbe automatisch neu aktiviert wird. |
| Busy | BOOL | Wird TRUE sobald der Baustein aktiv ist und wird FALSE nachdem er sich wieder im Grundzustand befindet. Falls TriggerInput.FreeRun TRUE ist, bleibt Busy permanent TRUE, auch wenn Done oder Error TRUE werden, da TouchProbe automatisch neu aktiviert wird. |
| Error | BOOL | Wird im Fehlerfall TRUE. |

| Name | Typ | Beschreibung |
|-----------------------|---|--|
| ErrorID | UDINT | Liefert bei einem gesetzten Error-Ausgang die Fehlernummer. |
| RecordedPositions | ARRAY[1..10] | Ein oder mehrere erfasste Achspositionen zum Zeitpunkt des Trigger-Ereignisses. Wenn TriggerInput.FreeRun TRUE ist, läuft der Funktionsbaustein freilaufend, so dass jede gültige Änderung des Eingangssignals zu einer neuen RecordedPosition führt. Die Positionen können jeweils ausgewertet werden, wenn Done TRUE ist. |
| NoOfRecordedPositions | UDINT | Anzahl der erfassten Positionen. |
| Diagnostic | ST_EL1258_Diagnostics [► 46] | Datenstruktur, die Diagnosedaten enthält, die zur Fehleranalyse herangezogen werden können. |

4.1.1.4 Ausgaberroutinen für digitale Klemmen

4.1.1.4.1 XFC_EL2212_V2



Der Funktionsbaustein XFC_EL2212_V2 wickelt die Ausgabe einer digitalen Nocke mit der XFC-Zeitstempelklemme EL2212 ab.

Die Ausgabe der Daten an die Klemme erfolgt erst kurz vor Erreichen des Zeitstempels einer der Outputs. Dafür werden bis zur Aktivierung und Quittierung der Ausgänge vier SPS-Zyklen benötigt. Erst danach kann ein weiterer Flankenwechsel erfolgen. Die minimale Zeit zwischen zwei Flankenwechseln des Ausgangssignals beträgt also vier SPS-Zyklen, um Fehler oder Verlust der Präzision zu vermeiden. Falls das Ausgangssignal durch ein Nockenschaltwerk generiert wird, kann aus der maximalen Geschwindigkeit und der SPS-Zykluszeit eine Mindest-Nockenbreite berechnet werden.

Die Ausgänge Output1 und Output2 können nicht vollständig unabhängig voneinander verwendet werden, da die Aktivierung mit nur einem Zeitstempel erfolgt. Als Voraussetzung müssen die Schaltflanken beider Kanäle genügend weit auseinander liegen. In diesem Fall wird der jeweils nächstliegende Zeitstempel an den Block angelegt.

Eingänge

```
VAR_INPUT
    ForceWhenLate : BOOL;
END_VAR
```

| Name | Typ | Beschreibung |
|---------------|------|---|
| ForceWhenLate | BOOL | Wenn ForceWhenLate TRUE ist, wird der Ausgang auch dann geschaltet, wenn der Zeitstempel bereits überschritten ist. Es wird empfohlen, ForceWhenLate zu setzen, um den Ausfall einer Schaltflanke bei Schwankungen des Zeitsignals zu vermeiden. |

Ein-/Ausgänge

```
VAR_IN_OUT
  Output1      : OUTPUT_REF;
  Output2      : OUTPUT_REF;
  TrackOptions1 : TRACK_REF;
  TrackOptions2 : TRACK_REF;
  EL2212       : EL2212_IoInterface;
END_VAR
```

| Name | Typ | Beschreibung |
|---------------|---|--|
| Output1 | OUTPUT_REF [► 43] | Ausgangszustand für den Kanal 1 der Klemme. Die Datenstruktur Output enthält den berechneten Zustand des digitalen Ausgangs und die zugehörigen Zeitstempel für die Ausgabe an einer digitalen XFC-Ausgangsklemme |
| Output2 | OUTPUT_REF [► 43] | Ausgangszustand für den Kanal 2 der Klemme. Die Datenstruktur Output enthält den berechneten Zustand des digitalen Ausgangs und die zugehörigen Zeitstempel für die Ausgabe an einer digitalen XFC-Ausgangsklemme. |
| TrackOptions1 | TRACK_REF [► 44] | Die Datenstruktur TrackOptions1 enthält die Parametrierung der Nockenspur 1 für Output1. |
| TrackOptions2 | TRACK_REF [► 44] | Die Datenstruktur TrackOptions2 enthält die Parametrierung der Nockenspur 2 für Output2. |
| EL2212 | EL2212_IoInterface [► 50] | Prozessabbild der Klemme |

Ausgänge

```
VAR_OUTPUT
  Error          : BOOL;
  ErrorID        : UDINT;
  ErrorOutputMissed : BOOL;
  PrecisionReduced : BOOL;
END_VAR
```

| Name | Typ | Beschreibung |
|-------------------|-------|---|
| Error | BOOL | Wird im Fehlerfall TRUE. |
| ErrorID | UDINT | Liefert bei einem gesetzten Error-Ausgang eine Fehlernummer |
| ErrorOutputMissed | BOOL | Der exakte in den Strukturen Output1 oder Output2 übergebene Schaltungspunkt konnte nicht eingehalten werden und der Ausgangszustand bleibt unverändert. Falls ForceWhenLate TRUE ist, wird der Ausgangszustand immer ausgegeben und ErrorOutputMissed wird nicht TRUE. |
| PrecisionReduced | BOOL | Der exakte in den Strukturen Output1 oder Output2 übergebene Schaltungspunkt konnte nicht eingehalten werden, der Ausgangszustand wurde aber verspätet ausgegeben. PrecisionReduced ist als Warnung zu verstehen und kann nur TRUE werden, wenn ForceWhenLate TRUE ist. |

4.1.1.4.2 XFC_EL2212_MultiEdge



Der Funktionsbaustein XFC_EL2212_MultiEdge wickelt die Ausgabe eines Multi-Edge-Nockenschaltwerkes MC_DigitalCamSwitch_MultiEdge [► 19] über die XFC-Zeitstempelklemme EL2212 ab.

 **Eingänge**

```
VAR_INPUT
    ForceWhenLate : BOOL;
    Reset         : BOOL;
END_VAR
```

| Name | Typ | Beschreibung |
|---------------|------|---|
| ForceWhenLate | BOOL | Wenn ForceWhenLate TRUE ist, wird der Ausgang auch dann geschaltet, wenn der Zeitstempel bereits überschritten ist. Es wird empfohlen, ForceWhenLate zu setzen, um den Ausfall einer Schaltflanke bei Schwankungen des Zeitsignals zu vermeiden. |
| Reset | BOOL | Reset initiates a reset of the terminal |

 **Ein-/Ausgänge**

```
VAR_IN_OUT
    Output          : OUTPUT_REF_MULTIEDGE;
    TrackOptions    : TRACK_REF;
    EL2212          : EL2212_IoInterface;
END_VAR
```

| Name | Typ | Beschreibung |
|--------------|---------------------------------------|--|
| Output | <u>OUTPUT_REF_MULTIEDGE</u> [▶ 43] | Ausgangszustand für einen Kanal der Klemme. Die Datenstruktur Output enthält die nächsten berechneten Zustände des digitalen Ausgangs und die zugehörigen Zeitstempel für die Ausgabe an einer digitalen XFC Ausgangsklemme. |
| TrackOptions | <u>TRACK_REF</u> [▶ 44] | Die Datenstruktur TrackOptions enthält die Parametrierung der Nockenspur. |
| EL2212 | <u>EL2212_IoInterface</u> [▶ 50] | Prozessabbild der Klemme |

 **Ausgänge**

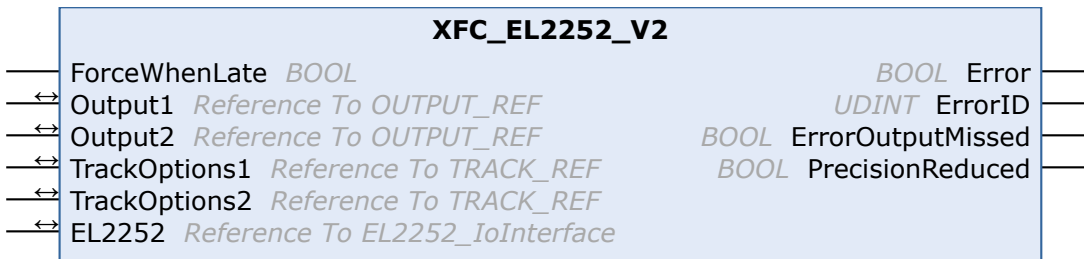
```
VAR_OUTPUT
    Error           : BOOL;
    ErrorID         : UDINT;
    ErrorOutputMissed : BOOL;
    PrecisionReduced : BOOL;
    Diagnostics     : ST_EL2258_Diagnostics;
END_VAR
```

| Name | Typ | Beschreibung |
|-------------------|--|--|
| Error | BOOL | Wird im Fehlerfall TRUE. |
| ErrorID | UDINT | Liefert bei einem gesetzten Error-Ausgang eine Fehlernummer |
| ErrorOutputMissed | BOOL | Einer der in der Struktur Output übergebenen Schaltepunkte konnte nicht eingehalten werden und der Ausgangszustand bleibt unverändert. Falls ForceWhenLate TRUE ist, wird der Ausgangszustand immer ausgegeben und ErrorOutputMissed wird nicht TRUE. |
| PrecisionReduced | BOOL | Einer der in der Struktur Output übergebenen Schaltepunkte konnte nicht eingehalten werden, der Ausgangszustand wurde aber verspätet ausgegeben. PrecisionReduced ist als Warnung zu verstehen und kann nur TRUE werden, wenn ForceWhenLate TRUE ist. |
| Diagnostics | <u>ST_EL2258_Diagnostics</u> [▶ 47] | Datenstruktur, die Diagnosedaten enthält, die zur Fehleranalyse herangezogen werden können. |

Prozessabbild

Die Klemme wird für den Betrieb mit diesem Funktionsbaustein im Multi-Timestamp-Modus mit bis zu 10 Zeitstempeln in das Prozessabbild eingefügt und der DC-Mode muss aktiviert werden.

4.1.1.4.3 XFC_EL2252_V2



Der Funktionsbaustein XFC_EL2252_V2 wickelt die Ausgabe einer digitalen Nocke mit der XFC-Zeitstempelklemme EL2252 ab.

Die Ausgabe der Daten an die Klemme erfolgt erst kurz vor Erreichen des Zeitstempels einer der Outputs. Dafür werden bis zur Aktivierung und Quittierung der Ausgänge vier SPS-Zyklen benötigt. Erst danach kann ein weiterer Flankenwechsel erfolgen. Die minimale Zeit zwischen zwei Flankenwechseln des Ausgangssignals beträgt also vier SPS-Zyklen um Fehler oder Verlust der Präzision zu vermeiden. Falls das Ausgangssignal durch ein Nockenschaltwerk generiert wird, kann aus der maximalen Geschwindigkeit und der SPS-Zykluszeit eine Mindest-Nockenbreite berechnet werden.

Die Ausgänge Output1 und Output2 können nicht vollständig unabhängig voneinander verwendet werden, da die Aktivierung mit nur einem Zeitstempel erfolgt. Als Voraussetzung müssen die Schaltflanken beider Kanäle genügend weit auseinander liegen. In diesem Fall wird der jeweils nächstliegende Zeitstempel an den Block angelegt.

Eingänge

```
VAR_INPUT
  ForceWhenLate : BOOL;
END_VAR
```

| Name | Typ | Beschreibung |
|---------------|------|--|
| ForceWhenLate | BOOL | Wenn ForceWhenLate TRUE ist, so wird der Ausgang auch dann geschaltet, wenn der Zeitstempel bereits überschritten ist. Es wird empfohlen, ForceWhenLate zu setzen, um den Ausfall einer Schaltflanke bei Schwankungen des Zeitsignals zu vermeiden. |

Ein-/Ausgänge

```
VAR_IN_OUT
  Output1      : OUTPUT_REF;
  Output2      : OUTPUT_REF;
  TrackOptions1 : TRACK_REF;
  TrackOptions2 : TRACK_REF;
  EL2252       : EL2252_IoInterface;
END_VAR
```

| Name | Typ | Beschreibung |
|---------|-----------------------------------|--|
| Output1 | OUTPUT_REF [▶ 43] | Ausgangszustand für den Kanal 1 der Klemme. Die Datenstruktur Output enthält den berechneten Zustand des digitalen Ausgangs und die zugehörigen Zeitstempel für die Ausgabe an einer digitalen XFC-Ausgangsklemme. |
| Output2 | OUTPUT_REF [▶ 43] | Ausgangszustand für den Kanal 2 der Klemme. Die Datenstruktur Output enthält den berechneten Zustand des digitalen Ausgangs und die zugehörigen Zeitstempel für die Ausgabe an einer digitalen XFC-Ausgangsklemme. |

| Name | Typ | Beschreibung |
|---------------|---------------------------|--|
| TrackOptions1 | TRACK_REF [▶ 44] | Die Datenstruktur TrackOptions1 enthält die Parametrierung der Nockenspur 1 für Output1. |
| TrackOptions2 | TRACK_REF [▶ 44] | Die Datenstruktur TrackOptions2 enthält die Parametrierung der Nockenspur 2 für Output2. |
| EL2252 | EL2252_IoInterface [▶ 50] | Prozessabbild der Klemme |

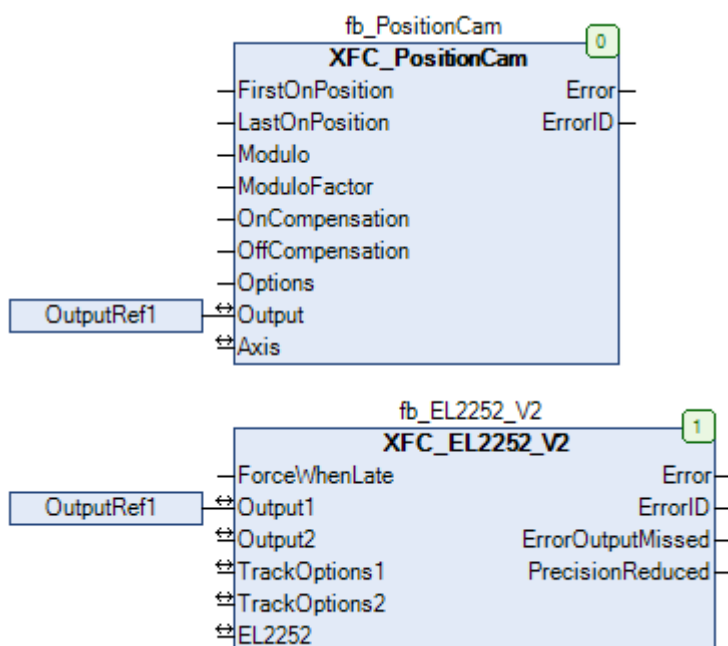
Ausgänge

```
VAR_OUTPUT
    Error          : BOOL;
    ErrorID        : UDINT;
    ErrorOutputMissed : BOOL;
    PrecisionReduced : BOOL;
END_VAR
```

| Name | Typ | Beschreibung |
|-------------------|-------|---|
| Error | BOOL | Wird im Fehlerfall TRUE. |
| ErrorID | UDINT | Liefert bei einem gesetzten Error-Ausgang eine Fehlernummer |
| ErrorOutputMissed | BOOL | Der exakte in den Strukturen Output1 oder Output2 übergebene Schaltungspunkt konnte nicht eingehalten werden und der Ausgangszustand bleibt unverändert. Falls ForceWhenLate TRUE ist, wird der Ausgangszustand immer ausgegeben und ErrorOutputMissed wird nicht TRUE. |
| PrecisionReduced | BOOL | Der exakte in den Strukturen Output1 oder Output2 übergebene Schaltungspunkt konnte nicht eingehalten werden, der Ausgangszustand wurde aber verspätet ausgegeben. PrecisionReduced ist als Warnung zu verstehen und kann nur TRUE werden, wenn ForceWhenLate TRUE ist. |

Beispiel für die Ausgabe einer Positionsnocke

Der Funktionsbaustein XFC_PositionCam [▶ 20] ist hardwareunabhängig und übergibt die notwendigen Schaltungsinformationen zur Ausgabe an den Funktionsbaustein XFC_EL2252_V2 in der Datenstruktur OutputRef1. Die Datenstruktur wird zur Datenübergabe an beiden Bausteinen verwendet.



Ausgabe zweier Positionsnocken an einer Ausgabeklemme

Die Ausgänge Output1 und Output2 können mit der EL2252 nicht vollständig unabhängig voneinander verwendet werden, da die Aktivierung mit nur einem Zeitstempel erfolgt. Als Voraussetzung müssen die Schaltflanken beider Kanäle genügend weit auseinander liegen (4 SPS-Zyklen). Sobald die Schaltflanken näher zusammenliegen, werden sie ungenau ausgegeben.

4.1.1.4.4 XFC_EL2258_MultiEdge



Der Funktionsbaustein XFC_EL2258_MultiEdge wickelt die Ausgabe eines Multi-Edge-Nockenschaltwerkes MC_DigitalCamSwitch_MultiEdge [► 19] über die XFC-Zeitstempelklemme EL2258 ab.

Eingänge

```
VAR_INPUT
  ForceWhenLate : BOOL;
  Reset         : BOOL;
END_VAR
```

| Name | Typ | Beschreibung |
|---------------|------|--|
| ForceWhenLate | BOOL | Wenn ForceWhenLate TRUE ist, wird der Ausgang auch dann geschaltet, wenn der Zeitstempel bereits überschritten ist. |
| Reset | BOOL | Es wird empfohlen, ForceWhenLate zu setzen, um den Ausfall einer Schaltflanke bei Schwankungen des Zeitsignals zu vermeiden. |

Ein-/Ausgänge

```
VAR_IN_OUT
  Output       : OUTPUT_REF_MULTIEDGE;
  TrackOptions : TRACK_REF;
  EL2258       : EL2258_IoInterface;
END_VAR
```

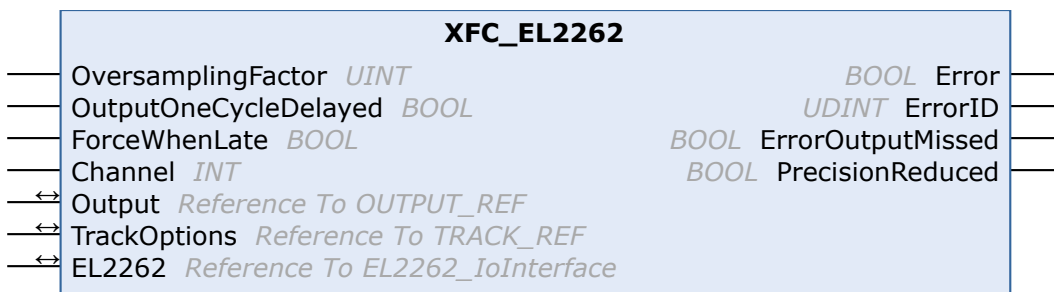
| Name | Typ | Beschreibung |
|--------------|-----------------------------|--|
| Output | OUTPUT_REF_MULTIEDGE [► 43] | Ausgangszustand für einen Kanal der Klemme. Die Datenstruktur Output enthält die nächsten berechneten Zustände des digitalen Ausganges und die zugehörigen Zeitstempel für die Ausgabe an einer digitalen XFC Ausgabeklemme. |
| TrackOptions | TRACK_REF [► 44] | Die Datenstruktur TrackOptions enthält die Parametrierung der Nockenspur. |
| EL2258 | EL2258_IoInterface [► 50] | Prozessabbild der Klemme |

Ausgänge

```
VAR_OUTPUT
  Error           : BOOL;
  ErrorID        : UDINT;
  ErrorOutputMissed : BOOL;
  PrecisionReduced : BOOL;
  Diagnostics    : ST_EL2258_Diagnostics;
END_VAR
```

| Name | Typ | Beschreibung |
|-------------------|---|---|
| Error | BOOL | Wird im Fehlerfall TRUE. |
| ErrorID | UDINT | Liefert bei einem gesetzten Error-Ausgang eine Fehlernummer |
| ErrorOutputMissed | BOOL | Einer der in der Struktur Output übergebenen Schaltepunkte konnte nicht eingehalten werden und der Ausgangszustand bleibt unverändert. Falls ForceWhenLate TRUE ist, wird der Ausgangszustand immer ausgegeben und ErrorOutputMissed wird nicht TRUE. |
| PrecisionReduced | BOOL | Einer der in der Struktur Output übergebenen Schaltepunkte konnte nicht eingehalten werden, der Ausgangszustand wurde aber verspätet ausgegeben. PrecisionReduced ist als Warnung zu verstehen und kann nur TRUE werden, wenn ForceWhenLate TRUE ist. |
| Diagnostics | ST_EL2258_Diagnostics ▶ 47 | Datenstruktur, die Diagnosedaten enthält, die zur Fehleranalyse herangezogen werden können. |

4.1.1.4.5 XFC_EL2262



Der Funktionsbaustein XFC_EL2262 wickelt die Ausgabe einer digitalen Nocke mit der XFC-Oversampling-Klemme EL2262 ab.

Die maximale Frequenz ist von der Zykluszeit abhängig. Der minimale Abstand zwischen zwei steigenden Signalfanken beträgt zwei SPS-Zyklen. Der minimale Abstand zwischen der steigenden und fallenden Flanke kann kleiner als ein SPS-Zyklus sein. Die Schaltungsgenauigkeit wird durch den eingestellten Oversampling-Faktor der Klemme bestimmt.

Die beiden Kanäle der Klemme sind voneinander unabhängig und werden durch zwei Instanzen des XFC_EL2262 Bausteins bedient.

Eingänge

```

VAR_INPUT
    OversamplingFactor      : UINT;
    OutputOneCycleDelayed  : BOOL; (* TRUE if EL2262 is updated with the NC SAF task at the beginning
of the next cycle *)
    ForceWhenLate          : BOOL;
    Channel                 : INT;
END_VAR
    
```

| Name | Typ | Beschreibung |
|-----------------------|------|--|
| OversamplingFactor | UINT | Oversampling Faktor der EL2262 Klemme |
| OutputOneCycleDelayed | BOOL | OutputOneCycleDelayed ist TRUE, falls die Ausgabe des Prozessabbildes aufgrund des eingestellten Timings um einen Zyklus verzögert ist. OutputOneCycleDelayed ist abhängig vom Timing der Ausgabetausk, an welche die EL2262 gebunden ist. |
| ForceWhenLate | BOOL | Wenn sich die Zeitinformation von Zyklus zu Zyklus leicht ändert, kann es passieren, dass eine Schaltflanke nicht ausgegeben werden kann. ForceWhenLate erzwingt in |

| Name | Typ | Beschreibung |
|---------|-----|--|
| | | einer solchen Situation ein bestmögliches Schalten. Der Ausgang PrecisionReduced wird in diesem Fall TRUE und kann zur Diagnose verwendet werden. (Zum Beispiel kann im Falle eines erhöhten Jitters in einer Achsposition die Ausgabe einer Schaltflanke nicht exakt auf einen Ausgabezyklus bestimmt werden.) |
| Channel | INT | Kanalnummer 0 oder 1 der EL2262 Klemme |

 **Ein-/Ausgänge**

```
VAR_IN_OUT
  Output      : OUTPUT_REF;
  TrackOptions : TRACK_REF;
  EL2262      : EL2262_IoInterface;
END_VAR
```

| Name | Typ | Beschreibung |
|--------------|---|---|
| Output | OUTPUT_REF [▶ 43] | Die Datenstruktur Output enthält den berechneten Zustand des digitalen Ausgangs und die zugehörigen Zeitstempel für die Ausgabe an einer digitalen XFC Ausgangsklemme |
| TrackOptions | TRACK_REF [▶ 44] | Die Datenstruktur TrackOptions enthält die Parametrierung der Nockenspur. |
| EL2262 | EL2262_IoInterface [▶ 50] | Prozessabbild der Klemme |

 **Ausgänge**

```
VAR_OUTPUT
  Error          : BOOL;
  ErrorID        : UDINT;
  ErrorOutputMissed : BOOL;
  PrecisionReduced : BOOL;
END_VAR
```

| Name | Typ | Beschreibung |
|-------------------|-------|--|
| Error | BOOL | Wird im Fehlerfall TRUE. |
| ErrorID | UDINT | Liefert bei einem gesetzten Error-Ausgang eine Fehlernummer |
| ErrorOutputMissed | BOOL | Zeigt an, dass eine Schaltflanke nicht auf einen Zyklus genau bestimmt werden kann und daher nicht ausgegeben werden kann. Um die Schaltflanke bestmöglich auszugeben kann der Eingang ForceWhenLate gesetzt werden. |
| PrecisionReduced | BOOL | Zeigt an, dass eine Schaltflanke nicht auf einen Zyklus genau bestimmt werden kann. Die Schaltflanke wurde aber bestmöglich ausgegeben. |

4.1.1.4.6 XFC_EL2262_MultiEdge



Der Funktionsbaustein XFC_EL2262_Multiedge wickelt die Ausgabe von digitalen Nocken mit der XFC-Oversampling-Klemme EL2262 ab.

Die maximale Frequenz ist von der Zykluszeit abhängig. Der minimale Abstand zwischen zwei steigenden Signalfanken beträgt $(2 \cdot \text{SPS-Zykluszeit}) / \text{Oversampling-Faktor}$. Der minimale Abstand zwischen einer steigenden und fallenden Flanke beträgt $\text{SPS-Zykluszeit} / \text{Oversampling-Faktor}$. Die Schaltgenauigkeit wird durch den eingestellten Oversampling-Faktor der Klemme bestimmt.

 **Eingänge**

```
VAR_INPUT
  Reset           : BOOL;
  Preset          : BOOL;
  OversamplingFactor : UINT;
  OutputOneCycleDelayed : BOOL; (* TRUE if EL2262 is updated with the NC SAF task at the beginning of the next cycle *)
  ForceWhenLate   : BOOL; (* forces the output even when the timestamp is missed *)
  Channel         : INT; (* select 0 or 1 for Output0 or Output1 *)
END_VAR
```

| Name | Typ | Beschreibung |
|-----------------------|------|--|
| Reset | BOOL | Der Klemmenausgang wird deaktiviert. |
| Preset | BOOL | Der Klemmenausgang wird aktiviert. |
| OversamplingFactor | UINT | Oversampling Faktor der EL2262 Klemme |
| OutputOneCycleDelayed | BOOL | OutputOneCycleDelayed ist TRUE, falls die Ausgabe des Prozessabbildes aufgrund des eingestellten Timings um einen Zyklus verzögert ist. OutputOneCycleDelayed ist abhängig vom Timing der Ausgabetausk, an welche die EL2262 gebunden ist. |
| ForceWhenLate | BOOL | Wenn sich die Zeitinformation von Zyklus zu Zyklus leicht ändert, kann es passieren, dass eine Schaltflanke nicht ausgegeben werden kann. ForceWhenLate erzwingt in einer solchen Situation ein bestmögliches Schalten. Der Ausgang PrecisionReduced wird in diesem Fall TRUE und kann zur Diagnose verwendet werden. (Zum Beispiel kann im Falle eines erhöhten Jitters in einer Achsposition die Ausgabe einer Schaltflanke nicht exakt auf einen Ausgabezyklus bestimmt werden.) |
| Channel | INT | Legt den Ausgabekanal der EL2262 fest, wobei 0 = Output0 und 1 = Output1 entspricht. |

 **Ein-/Ausgänge**

```
VAR_IN_OUT
  Output           : OUTPUT_REF_MULTIEDGE;
  TrackOptions     : TRACK_REF;
  EL2262           : EL2262_IoInterface;
END_VAR
```

| Name | Typ | Beschreibung |
|--------------|---|---|
| Output | OUTPUT_REF_MULTIEDGE [43] | Die Datenstruktur Output enthält ein Array von berechneten Zuständen des digitalen Ausgangs und die zugehörigen Zeitstempel für die Ausgabe an einer digitalen XFC Ausgangsklemme |
| TrackOptions | TRACK_REF [44] | Die Datenstruktur TrackOptions enthält die Parametrierung der Nockenspur. |
| EL2262 | EL2262_IoInterface [50] | Prozessabbild der Klemme |

Ausgänge

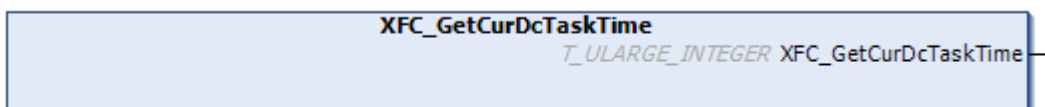
```
VAR_OUTPUT
  Error          : BOOL;
  ErrorID        : UDINT;
  ErrorOutputMissed : BOOL;
  PrecisionReduced : BOOL;
  Diagnostics    : ST_EL2262_Diagnostics;
END_VAR
```

| Name | Typ | Beschreibung |
|-------------------|--|--|
| Error | BOOL | Wird im Fehlerfall TRUE. |
| ErrorID | UDINT | Liefert bei einem gesetzten Error-Ausgang eine Fehlernummer |
| ErrorOutputMissed | BOOL | Einer der in der Struktur Output übergebenen Schaltepunkte konnte nicht eingehalten werden und der Ausgangszustand bleibt unverändert. Falls ForceWhenLate TRUE ist, wird der Ausgangszustand immer ausgegeben und ErrorOutputMissed wird nicht TRUE. |
| PrecisionReduced | BOOL | Einer der in der Struktur Output übergebenen Schaltepunkte konnte nicht eingehalten werden, der Ausgangszustand wurde aber verspätet ausgegeben. PrecisionReduced ist als Warnung zu verstehen und kann nur TRUE werden, wenn ForceWhenLate TRUE ist. |
| Diagnostics | ST_EL2262_Diagnostics > 471 | Datenstruktur die Diagnosedaten enthält, die zur Fehleranalyse herangezogen werden können. |

4.1.2 Funktionen

4.1.2.1 Basisfunktionen

4.1.2.1.1 XFC_GetCurDcTaskTime



Die Funktion XFC_GetCurDcTaskTime ermittelt die Zeit des Beginns des aktuellen SPS-Zyklus.

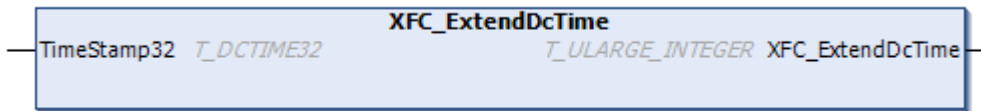
Die Funktion optimiert die Aufrufe der Systemfunktion F_GetCurDcTaskTime, indem mehrere Anfragen innerhalb eines SPS-Task-Zyklus mit derselben Zeit beantwortet werden, ohne die Systemfunktion jeweils erneut aufzurufen.

Rückgabewert

```
FUNCTION XFCF_GetCurDcTaskTime : T_DCTIME
```

| Name | Typ | Beschreibung |
|-----------------------|----------|---|
| XFCF_GetCurDcTaskTime | T_DCTIME | Rückgabewert der Funktion. Vollständige 64 Bit Distributed Clock System Time. Zeitpunkt des Anfangs des aktuellen SPS-Zyklus (Zyklus der Task, die diese Funktion aufruft.) |

4.1.2.1.2 XFC_ExtendDcTime



Die Funktion `XFC_ExtendDcTime` erweitert einen 32 Bit Zeitstempel auf 64 Bit.

Die Erweiterung auf einen vollständigen Zeitstempel setzt voraus, dass der 32 Bit Zeitstempel im aktuellen Zeitbereich gültig ist. Es ist nicht möglich, einen Zeitstempel der mehr als etwa +/- 2 Sekunden vor oder nach dem aktuellen Zeitpunkt gültig ist, fehlerfrei zu erweitern.

Eingänge

```
VAR_INPUT
    TimeStamp32 : T_DCTIME32;
END_VAR
```

| Name | Typ | Beschreibung |
|-------------|------------|---|
| TimeStamp32 | T_DCTIME32 | Distributed Clock System Time. Enthält die unteren 32 Bit der vollständigen DcTime und deckt damit einen Zeitbereich von +/- 2 Sekunden um den aktuellen Zeitpunkt ab. |

Rückgabewert

```
FUNCTION XFCF_ExtendDcTime : T_DCTIME
```

| Name | Typ | Beschreibung |
|-------------------|----------|---|
| XFCF_ExtendDcTime | T_DCTIME | Rückgabewert der Funktion. Vollständige 64 Bit Distributed Clock System Time. |

4.1.3 Datentypen

4.1.3.1 MC_CamSwitch

Der Datentyp `MC_CamSwitch` enthält alle Parameter einer digitalen Nocke für ein digitales Nockenschaltwerk `MC_DigitalCamSwitch` [▶ 16].

```
TYPE MC_CamSwitch :
STRUCT
    FirstOnPosition : LREAL;
    LastOnPosition : LREAL;
    AxisDirection : E_CamSwitchDirection;
    CamSwitchMode : E_CamSwitchMode;
    Duration : LREAL;
END_STRUCT
END_TYPE
```

Die Datenstruktur zur Parametrierung eines digitalen Nockenschaltwerks ist üblicherweise ein ARRAY OF `MC_CamSwitch`. Eine weitere Struktur `CAMSWITCH_REF` [▶ 42] verweist auf diese Struktur.

| Name | Typ | Beschreibung |
|-----------------|---------------------------------------|--|
| FirstOnPosition | LREAL | Erste Position, ab der die Nocke eingeschaltet ist. |
| LastOnPosition | LREAL | Letzte Position, bis zu der die Nocke eingeschaltet ist. Die Nockenfunktion ist invertiert, wenn <code>LastOnPosition < FirstOnPosition</code> . <code>LastOnPosition</code> wird bei Zeitnocken nicht verwendet. |
| AxisDirection | <u>E_CamSwitchDirection</u> [▶ 51] | <code>AxisDirection</code> legt fest, in welcher Fahrtrichtung der Achse die digitale Nocke aktiv ist (positiv, negativ oder beide Richtungen). |

| Name | Typ | Beschreibung |
|---------------|--------------------------|--|
| CamSwitchMode | E_CamSwitchMode ▶ 511 | Typ der digitalen Nocke (Positionsnocke, Zeitnocke oder Bremsnocke). |
| Duration | LREAL | Duration definiert die Einschaltdauer der Nocke in [s] und wird ausschließlich bei Zeitnocken verwendet. |

```

TYPE E_CamSwitchDirection :
(
  CAMSWITCHDIRECTION_BOTH,      (* digital cam will work in both directions *)
  CAMSWITCHDIRECTION_POSITIVE, (* digital cam is just working in positive direction *)
  CAMSWITCHDIRECTION_NEGATIVE  (* digital cam is just working in negative direction *)
);
END_TYPE

```

```

TYPE E_CamSwitchMode :
(
  CAMSWITCHMODE_POSITION, (* position cam *)
  CAMSWITCHMODE_TIME,     (* time cam *)
  CAMSWITCHMODE_BREAK     (* break cam *)
);
END_TYPE

```

4.1.3.2 CAMSWITCH_REF

Der Datentyp CAMSWITCH_REF verweist auf eine Datenstruktur mit Nockenparametern für ein digitales Nockenschaltwerk [MC_DigitalCamSwitch](#) [▶ 16].

```

TYPE CAMSWITCH_REF :
STRUCT
  NumberOfSwitches : UDINT;
  pSwitches        : POINTER TO MC_CamSwitch;
  SizeOfSwitches  : UDINT;
END_STRUCT
END_TYPE

```

| Name | Typ | Beschreibung |
|------------------|----------------------------|--|
| NumberOfSwitches | UDINT | Anzahl der genutzten Array-Elemente. Kann kleiner sein, als die maximale Anzahl. |
| pSwitches | POINTER TO MC_CamSwitch | Pointer auf das digital cam array. |
| SizeOfSwitches | UDINT | Maximale Größe des digital cam arrays. |

Die eigentliche Datenstruktur zur Parametrierung eines digitalen Nockenschaltwerks ist üblicherweise ein `ARRAY OF MC_CamSwitch` [▶ 41]. `CAMSWITCH_REF` [▶ 42] verweist mit einem `POINTER` auf diese Struktur und definiert eindeutig die Größe der Struktur und die Anzahl der tatsächlich verwendeten Nocken.

Eine Variable des Typs `CAMSWITCH_REF` wird, wie in dem folgenden Beispiel gezeigt, initialisiert:

```

VAR
  CamSwitchArray : ARRAY[1..3] OF MC_CamSwitch;
  CamSwitchRef   : CAMSWITCH_REF;
END_VAR

(* real number of defined digital cams *)
CamSwitchRef.NumberOfSwitches := 1; (* 1..3 *)
(* pointer to the digital cam data array *)
CamSwitchRef.pSwitches       := ADR(CamSwitchArray);
(* maximum size of the digital cam data array *)
CamSwitchRef.SizeOfSwitches  := SIZEOF(CamSwitchArray);

```

Beispiel mit zwei Nockenspuren

| CamSwitchRefTrack1 : CAMSWITCH_REF | |
|------------------------------------|------------------------------|
| | Value |
| NumberOfSwitches | 3 |
| pSwitches | ADR(CamSwitchArrayTrack1) |
| SizeOfSwitches | SIZEOF(CamSwitchArrayTrack1) |

| CamSwitchArrayTrack1 : Array [1..n] OF MC_CamSwitch | | | | | |
|---|----------|----------|----------|-----|----------|
| | Switch 1 | Switch 2 | Switch 3 | ... | Switch n |
| FirstOnPosition | 2000 | 2500 | 4000 | | |
| LastOnPosition | 3000 | 3000 | 1000 | | |
| AxisDirection | POSITIVE | NEGATIVE | BOTH | | |
| CamSwitchMode | POSITION | POSITION | POSITION | | |
| Duration [s] | — | — | — | | |

| CamSwitchRefTrack2 : CAMSWITCH_REF | |
|------------------------------------|------------------------------|
| | Value |
| NumberOfSwitches | 1 |
| pSwitches | ADR(CamSwitchArrayTrack2) |
| SizeOfSwitches | SIZEOF(CamSwitchArrayTrack2) |

| CamSwitchArrayTrack2 : Array [1..m] OF MC_CamSwitch | | | |
|---|----------|-----|----------|
| | Switch 1 | ... | Switch m |
| FirstOnPosition | 3000 | | |
| LastOnPosition | — | | |
| AxisDirection | BOTH | | |
| CamSwitchMode | TIME | | |
| Duration [s] | 1,350 | | |

4.1.3.3 OUTPUT_REF

Der Datentyp OUTPUT_REF enthält Daten zur Beschreibung des Zustandes eines digitalen Ausgangs. Neben dem Schaltzustand werden Zeitstempel für Zustandswechsel mitgeführt.

```

TYPE OUTPUT_REF :
STRUCT
  Level          : BOOL;          (* current level of the digital output *)

  NextStateChangeValid : BOOL;    (* time value NextStateChange is valid *)
  NextStateChange   : T_DCTIME32; (* time of next state change -
current level will be inverted *)

  NextOnTimeValid   : BOOL;    (* time value NextOnTime is valid *)
  NextOnTime        : T_DCTIME32; (* time when the digital output is turned ON next time *)

  NextOffTimeValid  : BOOL;    (* time value NextOffTime is valid *)
  NextOffTime       : T_DCTIME32; (* time when the digital output is turned OFF next time *)
END_STRUCT
END_TYPE
    
```

| Name | Typ | Beschreibung |
|----------------------|------------|---|
| Level | BOOL | Aktueller Schaltzustand des digitalen Ausgangs |
| NextStateChangeValid | BOOL | NextStateChangeValid ist TRUE, wenn der Zeitstempel NextStateChange gültig ist. |
| NextStateChange | T_DCTIME32 | Zeitpunkt des nächsten Zustandswechsels (Distributed Clock TimeStamp) |
| NextOnTimeValid | BOOL | NextOnTimeValid ist TRUE, wenn der Zeitstempel NextOnTime gültig ist. |
| NextOnTime | T_DCTIME32 | Zeitpunkt der nächsten positiven Schaltflanke (Distributed Clock TimeStamp) |
| NextOffTimeValid | BOOL | NextOffTimeValid ist TRUE, wenn der Zeitstempel NextOffTime gültig ist. |
| NextOffTime | T_DCTIME32 | Zeitpunkt der nächsten negativen Schaltflanke (Distributed Clock TimeStamp) |

4.1.3.4 OUTPUT_REF_MULTIEDGE

Der Datentyp OUTPUT_REF_MULTIEDGE enthält Daten zur Beschreibung des Zustandes eines digitalen Ausgangs. Neben dem Schaltzustand werden Zeitstempel für Zustandswechsel mitgeführt. Der Datentyp wird im Zusammenhang mit Klemmen verwendet, die mittels Multi-Timestamp mehrere Schaltvorgänge pro SPS-Zyklus ermöglichen.

```

TYPE OUTPUT_REF_MULTIEDGE :
STRUCT
  SwitchEvent : ARRAY [0..TCMC2_XFC_MAXINDEXOFMULTIEDGEOUTPUTEVENTS] OF ST_SwitchEvent;
END_STRUCT
END_TYPE

TYPE ST_SwitchEvent :
STRUCT
  ID      : UDINT;
  Valid   : BOOL;      (* time value is valid *)
  Level   : BOOL;      (* next level of the digital signal *)
  Position : LREAL;
  DcTime  : T_DCTIME32; (* time when the digital output changes *)
  Duration : DINT;
END_STRUCT
END_TYPE

```

| Name | Typ | Beschreibung |
|----------|------------|---|
| ID | UDINT | Interne Kennung der Schaltflanke |
| Valid | BOOL | Valid ist TRUE, wenn der Zeitstempel DcTime gültig ist. |
| Level | BOOL | Aktueller Schaltzustand des digitalen Ausgangs |
| Position | LREAL | Schaltposition des Schaltvorgangs |
| DcTime | T_DCTIME32 | Zeitpunkt des nächsten Zustandswechsels (Distributed Clock TimeStamp) |
| Duration | DINT | Nicht genutzt |

4.1.3.5 TRACK_REF

Der Datentyp TRACK_REF enthält die Parameter einer digitalen Nockenspur für ein digitales Nockenschaltwerk [MC_DigitalCamSwitch](#) [► 16].

```

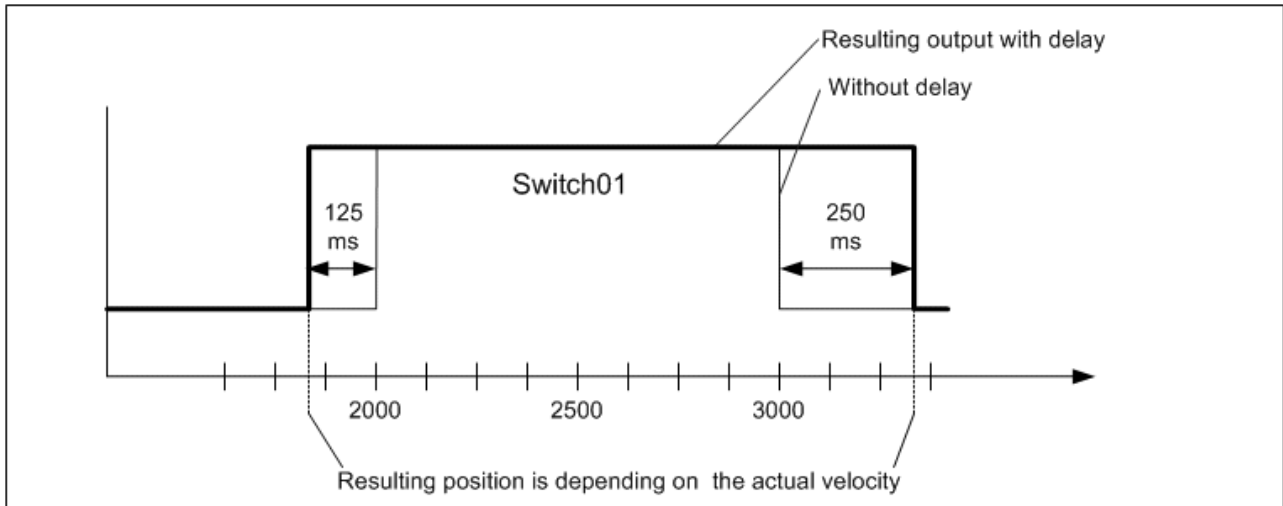
TYPE TRACK_REF :
STRUCT
  ModuloPositions : BOOL := TRUE; (* all cam positions are interpreted as modulo positions when TRUE *)
  ModuloFactor    : LREAL := 360; (* e. g. 360 degrees *)
  OnCompensation  : LREAL;      (* compensation time [s] *)
  OffCompensation : LREAL;      (* compensation time [s] *)
  Hysteresis      : LREAL;      (* distance from last switch position (+ or -) *)
  BreakRelease    : BOOL;      (* allow break to be released when TRUE, break cams will be activated when FALSE *)
  Force           : BOOL;      (* override all digital cams and set track ON *)
  Disable         : BOOL;      (* override all digital cams and set track OFF - overrides Force as well *)
END_STRUCT
END_TYPE

```

| Name | Typ | Beschreibung |
|-----------------|-------|---|
| ModuloPositions | BOOL | Wenn Modulo TRUE ist, werden alle Positionen modulo interpretiert. Die Nockenfunktion wird dadurch zyklisch wiederholt. Zur Berechnung des Modulo-Zyklus wird der Parameter ModuloFactor verwendet. |
| ModuloFactor | LREAL | ModuloFactor gibt die Länge eines Modulo-Zyklus in der Positionseinheit der Achse an und wird nur verwendet, wenn Modulo TRUE ist. |
| OnCompensation | LREAL | Kompensationszeit für die steigende Flanke der Nocke in [s]. Der Schaltzeitpunkt wird bei negativen Werten von OnCompensation vorgezogen und sonst verzögert. |
| OffCompensation | LREAL | Kompensationszeit für die fallende Flanke der Nocke in [s]. Der Schaltzeitpunkt wird bei negativen Werten von OffCompensation vorgezogen und sonst verzögert. |
| Hysteresis | LREAL | nicht implementiert. Hysterese der Schaltvorgänge bei Drehrichtungsumkehr. Die Hysterese wird in der Positionseinheit der Achse angegeben. |
| BreakRelease | BOOL | Bremsfreigabe für Bremsnocken auf diese Nockenspur |

| Name | Typ | Beschreibung |
|---------|------|--|
| Force | BOOL | Der digitale Ausgang wird unabhängig von den Nocken auf dieser Spur aktiv geschaltet. Disable hat Vorrang vor Force. |
| Disable | BOOL | Der digitale Ausgang wird unabhängig von den Nocken auf dieser Spur inaktiv geschaltet. Disable hat Vorrang vor Force. |

Wirkungsweise der Zeitkompensation



4.1.3.6 XFC_TRIGGER_REF

Der Datentyp XFC_TRIGGER_REF enthält Status und Parameter eines digitalen Eingangs, der für den Funktionsbaustein XFC_TouchProbe [► 26] verwendet wird.

```

TYPE XFC_TRIGGER_REF :
STRUCT
    Signal           : BOOL;
    TimestampRisingEdge : T_DCTIME32;
    TimestampFallingEdge : T_DCTIME32;
    Edge             : E_SignalEdge;
    FreeRun          : BOOL;
    EncoderIndex     : UINT;
    ModuloPositions  : BOOL;
    ModuloFactor     : LREAL := 360.0;
END_STRUCT
END_TYPE
    
```

| Name | Typ | Beschreibung |
|----------------------|------------|---|
| Signal | BOOL | Aktueller Zustand des digitalen Eingangssignals. Der aktuelle Zustand muss an dieser Stelle eingespeist werden. |
| TimestampRisingEdge | T_DCTIME32 | Zeitstempel der letzten steigenden Flanke des digitalen Eingangssignals. Es ist nur notwendig, den Zeitstempel der über Edge definierten Flanke einzuspeisen. Falls das Eingangssignal einen 64 Bit Zeitstempel T_DCTIME liefert, werden hier nur die unteren 32 Bit eingespeist. Es muss daher sichergestellt sein, dass der Wert zum Zeitpunkt der Auswertung nicht älter als 2 Sekunden ist. |
| TimestampFallingEdge | T_DCTIME32 | Zeitstempel der letzten fallenden Flanke des digitalen Eingangssignals. Es ist nur notwendig, den Zeitstempel der über Edge definierten Flanke einzuspeisen. Falls das Eingangssignal einen 64 Bit Zeitstempel |

| Name | Typ | Beschreibung |
|-----------------|--------------|---|
| | | T_DCTIME liefert, werden hier nur die unteren 32 Bit eingespeist. Es muss daher sichergestellt sein, dass der Wert zum Zeitpunkt der Auswertung nicht älter als 2 Sekunden ist. |
| Edge | E_SignalEdge | Edge definiert die Signalflanke, die zur Auswertung der Achsposition verwendet werden soll. TYPE E_SignalEdge : (RisingEdge, FallingEdge); END_TYPE |
| FreeRun | BOOL | Wenn FreeRun TRUE ist, wird der Eingang fortlaufend gelatcht. Beim Funktionsbaustein XFC_TouchProbe [► 26] muss dann der Eingang Execute TRUE bleiben. Es ist also keine Flanke an Execute notwendig, um den nächsten neuen Positionswert aufzuzeichnen. |
| EncoderIndex | UINT | Wenn mehr als ein Encoder mit der Achse verbunden ist, kann der Encoder-Index [0..9] hier festgelegt werden. Der erste Encoder hat den Index 0. |
| ModuloPositions | BOOL | Wenn ModuloPositions TRUE ist, werden alle Positionen modulo interpretiert. Zur Berechnung des Modulo-Zyklus wird der Parameter ModuloFactor verwendet. |
| ModuloFactor | LREAL | ModuloFactor gibt die Länge eines Modulo-Zyklus in der Positionseinheit der Achse an und wird nur verwendet, wenn ModuloPositions TRUE ist. |

4.1.3.7 ST_EL1258_Diagnostics

Der Datentyp ST_EL1258_Diagnostics enthält Diagnosedaten die zur Fehleranalyse herangezogen werden können.

```

TYPE ST_EL1258_Diagnostics :
STRUCT
  EventsInInputBuffer:      UDINT;
  NoOfReceiveEvents:       UDINT;
  NoOfReceiveEventsRising: UDINT;
  NoOfReceiveEventsFalling: UDINT;
  NoOfRecordedEventsRising: UDINT;
  NoOfRecordedEventsFalling: UDINT;
  ErrorBufferOverflow:     BOOL;
  ErrorModuloInput:       BOOL;
END_STRUCT
END_TYPE

```

| Name | Typ | Beschreibung |
|---------------------------|-------|--|
| EventsInInputBuffer | UDINT | Zählt die Events, solange ErrorBufferOverflow FALSE ist. |
| NoOfReceiveEvents | UDINT | Anzahl der Input-Events von EL1258. |
| NoOfReceiveEventsRising | UDINT | Anzahl der steigenden Flanken von EL1258. |
| NoOfReceiveEventsFalling | UDINT | Anzahl der fallenden Flanken von EL1258. |
| NoOfRecordedEventsRising | UDINT | Anzahl der steigenden Flanken in den aufgenommenen Events, solange WindowOnly FALSE ist und NoOfReceiveEventsRising = NoOfRecordedEventsRising. |
| NoOfRecordedEventsFalling | UDINT | Anzahl der fallenden Flanken in den aufgenommenen Events, solange WindowOnly FALSE ist und NoOfReceiveEventsFalling = NoOfRecordedEventsFalling. |
| ErrorBufferOverflow | BOOL | Zeigt an, dass der Ausgabepuffer der EL1258 voll ist. |
| ErrorModuloInput | BOOL | NoOfRecordedEventsRising + NoOfRecordedEventsFalling = NoOfRecordedEvents. |

4.1.3.8 ST_EL2258_Diagnostics

Der Datentyp ST_EL2258_Diagnostics enthält Diagnosedaten die zur Fehleranalyse herangezogen werden können.

```

TYPE ST_EL2258_Diagnostics :
STRUCT
    ErrorOnOutputMissed      : BOOL;
    ErrorOffOutputMissed     : BOOL;
    ErrorNoOfEventsExceeded  : BOOL;
    ErrorBufferOverflow      : BOOL;
    ErrorEventDistance       : BOOL;
    OnPrecisionReduced       : BOOL;
    OffPrecisionReduced      : BOOL;
    LastOutputLevel          : BOOL;
    ActivatedOnValues        : INT;
    ActivatedOffValues       : INT;
END_STRUCT
END_TYPE
    
```

| Name | Typ | Beschreibung |
|-------------------------|------|--|
| ErrorOnOutputMissed | BOOL | Zeigt an, dass eine steigende Schaltflanke nicht auf einen Zyklus genau bestimmt werden konnte und daher nicht ausgegeben werden kann. Um die Schaltflanke bestmöglich auszugeben kann der Eingang ForceWhenLate gesetzt werden. |
| ErrorOffOutputMissed | BOOL | Zeigt an, dass eine fallende Schaltflanke nicht auf einen Zyklus genau bestimmt werden konnte und daher nicht ausgegeben werden kann. Um die Schaltflanke bestmöglich auszugeben, kann der Eingang ForceWhenLate gesetzt werden. |
| ErrorNoOfEventsExceeded | BOOL | Zeigt an, dass zu viele Flanken für einen Zyklus geliefert wurden und somit nicht alle ausgegeben werden können. |
| ErrorBufferOverflow | BOOL | Zeigt an, dass der Ausgabepuffer der EL2258 voll ist. |
| ErrorEventDistance | BOOL | Zeigt an, dass der Abstand zweier aufeinanderfolgender Flanken zu gering ist. |
| OnPrecisionReduced | BOOL | Zeigt an, dass eine steigende Schaltflanke nicht auf einen Zyklus genau bestimmt werden konnte. Die Schaltflanke wurde aber bestmöglich ausgegeben. |
| OffPrecisionReduced | BOOL | Zeigt an, dass eine fallende Schaltflanke nicht auf einen Zyklus genau bestimmt werden konnte. Die Schaltflanke wurde aber bestmöglich ausgegeben. |
| LastOutputLevel | BOOL | Zeigt an welchen Signalzustand der Kanal der EL2262 nach dem folgenden Update haben wird. |
| ActivatedOnValues | INT | Anzahl der in diesem Zyklus aktivierten steigenden Flanken |
| ActivatedOffValues | INT | Anzahl der in diesem Zyklus aktivierten fallenden Flanken |

4.1.3.9 ST_EL2262_Diagnostics

Der Datentyp ST_EL2262_Diagnostics enthält Diagnosedaten die zur Fehleranalyse herangezogen werden können.

```

TYPE ST_EL2262_Diagnostics :
STRUCT
    ErrorOnOutputMissed      : BOOL;
    ErrorOffOutputMissed     : BOOL;
    OnPrecisionReduced       : BOOL;
    OffPrecisionReduced      : BOOL;
    LastOutputLevel          : BOOL;
    ActivatedOnValues        : INT;
    ActivatedOffValues       : INT;
END_STRUCT
END_TYPE
    
```

| Name | Typ | Beschreibung |
|----------------------|------|---|
| ErrorOnOutputMissed | BOOL | Zeigt an, dass eine steigende Schaltflanke nicht auf einen Zyklus genau bestimmt werden konnte und daher nicht ausgegeben werden kann. Um die Schaltflanke bestmöglich auszugeben, kann der Eingang ForceWhenLate gesetzt werden. |
| ErrorOffOutputMissed | BOOL | Zeigt an, dass eine fallende Schaltflanke nicht auf einen Zyklus genau bestimmt werden konnte und daher nicht ausgegeben werden kann. Um die Schaltflanke bestmöglich auszugeben, kann der Eingang ForceWhenLate gesetzt werden. |
| OnPrecisionReduced | BOOL | Zeigt an, dass eine steigende Schaltflanke nicht auf einen Zyklus genau bestimmt werden konnte. Die Schaltflanke wurde aber bestmöglich ausgegeben. |
| OffPrecisionReduced | BOOL | Zeigt an, dass eine fallende Schaltflanke nicht auf einen Zyklus genau bestimmt werden konnte. Die Schaltflanke wurde aber bestmöglich ausgegeben. |
| LastOutputLevel | BOOL | Zeigt an, welchen Signalzustand der Kanal der EL2262 nach dem folgenden Update haben wird. |
| ActivatedOnValues | INT | Anzahl der in diesem Zyklus aktivierten steigenden Flanken |
| ActivatedOffValues | INT | Anzahl der in diesem Zyklus aktivierten fallenden Flanken |

4.1.3.10 ST_NcTimeConversionOptions

```

TYPE ST_NcTimeConversionOptions :
STRUCT
  SubIndex :          UINT;
  InterpolationOptions : UINT;
  CompensationTime :  DINT;
END_STRUCT
END_TYPE

```

| Name | Typ | Beschreibung |
|----------------------|------|--|
| SubIndex | UINT | Für Achsen mit mehr als einem Encoder kann in SubIndex der Index (0..9) des Encoders angegeben werden, auf den sich die Position bezieht. Bei Path-Bausteinen, die sich auf mehrere interpolierte Achsen beziehen, kann kein SubIndex spezifiziert werden. |
| InterpolationOptions | UINT | Bitmaske für spezielle Optionen: InterpolationOptions.0 = FALSE: Die Extrapolation der Position wird mit der aktuellen Geschwindigkeit durchgeführt, ohne die aktuelle Beschleunigung zu berücksichtigen. InterpolationOptions.0 = TRUE: Die Beschleunigung der Achse wird in die Extrapolation der Position einbezogen. InterpolationOptions.12 = TRUE: Der Positionspuffer wird ignoriert. InterpolationOptions.15 = TRUE: eine Logger-Meldung wird erzwungen. |
| CompensationTime | DINT | Zusätzliche Kompensationszeit |

4.1.3.11 XFC_MT_TRIGGER_REF

```

TYPE XFC_MT_TRIGGER_REF :
STRUCT
  Signal :          BOOL;
  FreeRun :         BOOL;
  EncoderIndex :    UINT;
  ModuloPositions : BOOL;

```



```

ModuloFactor      : LREAL := 360.0;
END_STRUCT
END_TYPE
    
```

| Name | Typ | Beschreibung |
|-----------------|-------|--|
| Signal | BOOL | Aktueller Zustand des digitalen Eingangssignals. Der aktuelle Zustand muss an dieser Stelle eingespeist werden. |
| FreeRun | BOOL | Wenn FreeRun TRUE ist, wird der Eingang fortlaufend gelatcht. Beim Funktionsbaustein XFC_EL1258_TouchProbe muss dann der Eingang Execute TRUE bleiben. Es ist also keine Flanke an Execute notwendig, um den nächsten neuen Positionswert aufzuzeichnen. |
| EncoderIndex | UINT | Wenn mehr als ein Encoder mit der Achse verbunden ist, kann der Encoder-Index [0..9] hier festgelegt werden. Der erste Encoder hat den Index 0. |
| ModuloPositions | BOOL | Wenn ModuloPositions TRUE ist, werden alle Positionen modulo interpretiert. Zur Berechnung des Modulo-Zyklus wird der Parameter ModuloFactor verwendet. |
| ModuloFactor | LREAL | ModuloFactor gibt die Länge eines Modulo-Zyklus in der Positionseinheit der Achse an und wird nur verwendet, wenn ModuloPositions TRUE ist. |

4.1.3.12 ST_CamSwitchOptions

| Name | Typ | Beschreibung |
|-----------------|------|--|
| EncoderIndex | UINT | Wenn mehr als ein Encoder mit der Achse verbunden ist, kann der Encoder-Index [0..9] hier festgelegt werden. Der erste Encoder hat den Index 0. Bei Path-Bausteinen, die sich auf mehrere interpolierte Achsen beziehen, kann kein SubIndex spezifiziert werden. |
| UseAcceleration | BOOL | UseAcceleration kann TRUE gesetzt werden, um die Beschleunigung der Achse in die Positionsberechnungen mit einzubeziehen. UseAcceleration kann von Vorteil sein, wenn die Sollwerte der Beschleunigung verwendet werden können. Bei Encoder-Achsen, die ein verrauschtes Positionssignal liefern, ist UseAcceleration eventuell von Nachteil, da auch die Beschleunigung fehlerbehaftet ist. |

4.1.3.13 ST_XFCTouchProbeOptions

| Name | Typ | Beschreibung |
|-----------------|------|---|
| UseAcceleration | BOOL | UseAcceleration kann TRUE gesetzt werden, um die Beschleunigung der Achse in die Positions-Berechnungen mit einzubeziehen. UseAcceleration kann von Vorteil sein, wenn die Sollwerte der Beschleunigung verwendet werden können. Bei Encoder-Achsen, die ein verrauschtes Positionssignal liefern, ist UseAcceleration eventuell von Nachteil, da auch die Beschleunigung fehlerhaft ist. |

4.1.3.14 EL1258_MT_IoInterface

| Name | Typ | Beschreibung |
|-----------------|-----------------------------|----------------------------------|
| Status | _EL1258_Status | |
| InputEventState | _EL1258_InputEvents | MTI Eingänge |
| InputEventTime | ARRAY [1..10] OF T_DCTIME32 | MTI Eingang der Event-Zeit 1..10 |
| Ctrl | _EL1258_Ctrl | |

4.1.3.15 EL2252_IoInterface

| Name | Typ | Beschreibung |
|------------|------------------|-----------------------------|
| SysTime | T_ULARGE_INTEGER | Nicht genutzt |
| Feedback | BYTE | Nicht genutzt |
| WcStateOut | BOOL | Nicht genutzt |
| WcStateIn | BOOL | Nicht genutzt |
| State | UINT | Nicht genutzt |
| Activate | BYTE | Aktiviert den Ausgang |
| StartTime | T_ULARGE_INTEGER | Aktiviert den Ausgang |
| Output1 | BOOL | Signal des Ausgangskanals 1 |
| TriState1 | BOOL | Kanal 1 |
| Output1 | BOOL | Signal des Ausgangskanals 2 |
| TriState1 | BOOL | Kanal 2 |

4.1.3.16 EL2212_IoInterface

| Name | Typ | Beschreibung |
|-----------|------------------|-----------------------------------|
| Status1 | WORD | Statuskanal 1 |
| Status1 | WORD | Statuskanal 2 |
| Feedback | BYTE | Nicht genutzt |
| WcState | BOOL | Nicht genutzt |
| State | UINT | Nicht genutzt |
| Output1 | BOOL | Signal Ausgangskanal 1 |
| TriState1 | BOOL | Kanal 1 |
| Reset1 | BOOL | Kanal 1 |
| Output2 | BOOL | Signal Ausgangskanal 2 |
| TriState2 | BOOL | Kanal 2 |
| Reset2 | BOOL | Kanal 2 |
| Activate | BYTE | PDO 0x1602, aktiviert den Ausgang |
| StartTime | T_ULARGE_INTEGER | PDO 0x1602, aktiviert den Ausgang |

4.1.3.17 EL2262_IoInterface

| Name | Typ | Beschreibung |
|---------------------|------------------------|---|
| StartTimeNextOutput | T_DCTIME32 | Zeitstempel der nächsten Signalausgabe der Klemme EL2262. |
| WcState | BOOL | |
| State | UINT | |
| CycleCounter0 | WORD | |
| Output0 | ARRAY [0..31] OF DWORD | Array mit 1024 bis 10000fachem Oversampling – Kanal 1 |
| CycleCounter1 | WORD | |
| Output1 | ARRAY [0..31] OF DWORD | Array mit 1024 bis 10000fachem Oversampling – Kanal 1 |

4.1.3.18 EL2258_MT_IoInterface

| Name | Typ | Beschreibung |
|--------|----------------|--------------|
| Status | _EL2258_Status | MTO Status |
| Ctrl | _EL2258_Ctrl | MTO Ctrl |

| Name | Typ | Beschreibung |
|-----------------|-----------------------------|----------------------|
| Outputs | DWORD | MTO Outputs |
| OutputEventTime | ARRAY [1..10] OF T_DCTIME32 | MTO Ouput event time |

4.1.3.19 E_CamSwitchDirection

| Name | Beschreibung |
|-----------------------------|--|
| CAMSWITCHDIRECTION_BOTH | Die digitale Nocke arbeitet in beide Richtungen. |
| CAMSWITCHDIRECTION_POSITIVE | Die digitale Nocke arbeitet in positiver Richtung. |
| CAMSWITCHDIRECTION_NEGATIVE | Die digitale Nocke arbeitet in negativer Richtung. |

4.1.3.20 E_CamSwitchMode

| Name | Beschreibung |
|------------------------|----------------|
| CAMSWITCHMODE_POSITION | Positionsnocke |
| CAMSWITCHMODE_TIME | Zeitnocke |
| CAMSWITCHMODE_BREAK | Bremsnocke |

4.2 Tc2_NciXFC

Die Bibliothek Tc2_NciXFC erleichtert die zeitgenaue Erfassung von relativen Bahnwegen und die bahngenaue Ausgabe von Digitalsignalen in Verbindung mit EtherCAT-XFC-Klemmen. Die benötigten Ausgabe-Bausteine sind Bestandteil der Tc2_MC2_XFC Bibliothek.

So kann jederzeit ein genauer Zusammenhang zwischen Zeit und relativer Bahnposition hergestellt werden.

Zur Berechnung der Zeitstempel bzw. Positionen stellt die Bibliothek verschiedene Funktionsbausteine bereit.

4.2.1 Funktionsbausteine

4.2.1.1 Basisbausteine

4.2.1.1.1 XFC_PathPositionAtTime



Der Funktionsbaustein XFC_PathPositionAtTime berechnet einen relativen Bahnweg, zu einem gegebenen Zeitpunkt bezogen auf die aktuelle Bahnposition.

Die Funktion extrapoliert den Bahnweg bezogen auf die aktuelle Dynamik. Eine genaue Extrapolation ist nur über einen kurzen Zeitraum möglich, da sich die Dynamik der Gruppe ändern kann.

Der Funktionsbaustein benötigt genau einen Aufruf, um das Ergebnis bereitzustellen. Er kann also ähnlich wie eine Funktion verwendet werden, gibt aber neben dem relativen Bahnweg eventuell auch einen Fehler zurück. Dieser Fehler muss ausgewertet werden, um sicherzustellen, dass der berechnete Bahnweg gültig ist.

Eingänge

```
VAR_INPUT
  GrpId   : UDINT;
  DcTime  : T_DCTIME32;
  Options : ST_NcTimeConversionOptions;
END_VAR
```

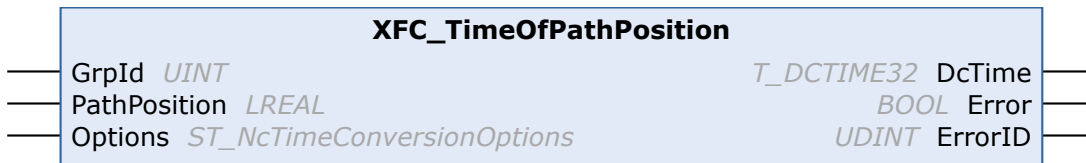
| Name | Typ | Beschreibung |
|---------|---|---|
| GrpId | UDINT | Gruppen-Id der Nci-Gruppe. Hierüber wird die angefragte Nci-Gruppe eindeutig im System identifiziert. |
| DcTime | T_DCTIME32 | Distributed Clock System Time. DcTime enthält die unteren 32 Bit der vollständigen DcTime und deckt damit einen Zeitbereich von +/- 2 Sekunden um den aktuellen Zeitpunkt ab. Um einen möglichst genauen Bahnweg zu berechnen, sollte die Zeit sehr nahe an der aktuellen Zeit liegen, also möglichst nur wenige SPS- bzw. NC-Zyklen in der Zukunft oder in der Vergangenheit. |
| Options | ST_NcTimeConversionOptions [► 48] | Datenstruktur mit Optionen für die Extrapolation des relativen Bahnweges. |

Ausgänge

```
VAR_OUTPUT
  PathPosition : LREAL;
  Error        : BOOL;
  ErrorID      : UDINT;
END_VAR
```

| Name | Typ | Beschreibung |
|--------------|-------|--|
| PathPosition | LREAL | Extrapolierter relativer Bahnweg, bis zum vorgegebenen Zeitpunkt DcTime. |
| Error | BOOL | Wird im Fehlerfall TRUE |
| ErrorID | UDINT | Liefert bei einem gesetzten Error-Ausgang die Fehlernummer. |

4.2.1.1.2 XFC_TimeOfPathPosition



Der Funktionsbaustein XFC_TimeOfPathPosition berechnet den Zeitpunkt, zu dem eine Nci-Gruppe einen vorgegebenen relativen Bahnweg zurückgelegt hat bzw. haben wird.

Die Funktion extrapoliert den Zeitpunkt bezogen auf die aktuelle Bahnposition und Dynamik. Eine genaue Extrapolation ist nur über einen kurzen Zeitraum möglich, da sich die Dynamik der Nci-Gruppe ändern kann.

Der Funktionsbaustein benötigt genau einen Aufruf, um das Ergebnis bereitzustellen. Er kann also ähnlich wie eine Funktion verwendet werden, gibt aber neben der Zeit eventuell auch einen Fehler zurück. Dieser Fehler muss ausgewertet werden, um sicherzustellen, dass die berechnete Zeit DcTime gültig ist.

Eingänge

```
VAR_INPUT
  GrpId      : UDINT;
  PathPosition : LREAL;
  Options    : ST_NcTimeConversionOptions;
END_VAR
```

| Name | Typ | Beschreibung |
|--------------|-----------------------------------|---|
| GrpId | UDINT | Gruppen-Id der Nci-Gruppe. Hierüber wird die angefragte Nci-Gruppe eindeutig im System identifiziert. |
| PathPosition | LREAL | Relativer Bahnweg |
| Options | ST_NcTimeConversionOptions [► 48] | Datenstruktur mit Optionen für die Extrapolation des relativen Bahnweges. |

Ausgänge

```
VAR_OUTPUT
  DcTime : T_DCTIME32;
  Error   : BOOL;
  ErrorID : UDINT;
END_VAR
```

| Name | Typ | Beschreibung |
|--------|------------|--|
| DcTime | T_DCTIME32 | Distributed Clock System Time, zu der der relative Bahnweg PathPosition zurückgelegt sein wird bzw. zu der dieser durchfahren wurde. DcTime enthält die unteren 32 Bit der vollständigen DcTime und deckt damit einen Zeitbereich von +/- 2 Sekunden um den aktuellen Zeitpunkt ab. |

| Name | Typ | Beschreibung |
|---------|-------|---|
| Error | BOOL | Wird im Fehlerfall TRUE. |
| ErrorID | UDINT | Liefert bei einem gesetzten Error-Ausgang die Fehlernummer. |

4.2.1.1.3 XFC_TimeToPathPosition



Der Funktionsbaustein XFC_TimeToPathPosition berechnet die Zeitdauer, innerhalb der eine Nci-Gruppe einen relativen Bahnweg zurückgelegt hat bzw. die seit diesem vergangen ist.

Die Funktion extrapoliert den Zeitpunkt bezogen auf die aktuelle Bahnposition und Dynamik. Eine genaue Extrapolation ist nur über einen kurzen Zeitraum möglich, da sich die Dynamik der Gruppe ändern kann.

Der Funktionsbaustein benötigt genau einen Aufruf, um das Ergebnis bereitzustellen. Er kann also ähnlich wie eine Funktion verwendet werden, gibt aber neben der Zeit eventuell auch einen Fehler zurück. Dieser Fehler muss ausgewertet werden, um sicherzustellen, dass die berechnete Zeitdauer gültig ist.

Eingänge

```
VAR_INPUT
  GrpId      : UDINT;
  PathId     : UDINT;
  PathPosition : LREAL;
  Options    : ST_NcTimeConversionOptions;
END_VAR
```

| Name | Typ | Beschreibung |
|--------------|---|---|
| GrpId | UDINT | Gruppen-Id der Nci-Gruppe. Hierüber wird die angefragte Nci-Gruppe eindeutig im System identifiziert. |
| PathId | UDINT | Eindeutige, kontinuierlich über den Bahnverlauf steigende Id die zum relativen Bahnweg gehört. |
| PathPosition | LREAL | Relativer Bahnweg |
| Options | ST_NcTimeConversionOptions [▶ 48] | Datenstruktur mit Optionen für die Extrapolation des relativen Bahnweges. |

Ausgänge

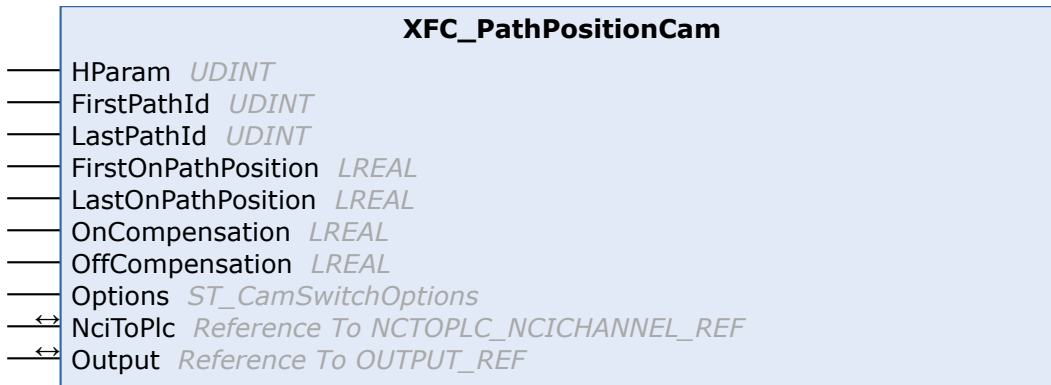
```
VAR_OUTPUT
  Duration      : DINT;
  TimeOfPosition : T_DCTIME32;
  Error         : BOOL;
  ErrorID      : UDINT;
END_VAR
```

| Name | Typ | Beschreibung |
|----------------|------------|--|
| Duration | DINT | Zeitdauer in Nanosekunden, nach der der relative Bahnweg zurückgelegt sein wird (> 0) bzw. die seit diesem vergangen ist (< 0). Duration ist ein Differenzwert aus zwei Variablen des Typs T_DCTIME32 |
| TimeOfPosition | T_DCTIME32 | Distributed Clock System Time zu der der relative Bahnweg zurückgelegt sein wird bzw. wurde. |

| Name | Typ | Beschreibung |
|---------|-------|--|
| | | DcTime enthält die unteren 32 Bit der vollständigen DcTime und deckt damit einen Zeitbereich von +/- 2 Sekunden um den aktuellen Zeitpunkt ab. |
| Error | BOOL | Wird im Fehlerfall TRUE. |
| ErrorID | UDINT | Liefert bei einem gesetzten Error-Ausgang die Fehlernummer. |

4.2.1.2 Nockenschaltwerk

4.2.1.2.1 XFC_PathPositionCam



Der Funktionsbaustein XFC_PathPositionCam realisiert eine Bahnnocke, die einen digitalen Ausgang bahnabhängig ein- und ausschaltet.

Im Gegensatz zum digitalen Nockenschaltwerk MC_PathDigitalCamSwitch_MultiEdge [► 56] schaltet der Funktionsbaustein genau eine Nocke auf einer digitalen Ausgangsspur. Der Baustein ist damit einfacher parametrierbar, kann aber nicht eingesetzt werden, wenn mehrere Nocken auf einer Ausgangsspur notwendig sind.

Die Output-Datenstruktur enthält neben dem Schaltzustand des digitalen Ausgangs genaue Zeitinformationen über die nächsten Schaltvorgänge.

Mit dieser Information kann die eigentliche Ausgabe an einer XFC-Ausgangsklemme mit einem nachgeschalteten Funktionsbaustein erfolgen (XFC_EL2252_V2 [► 34] oder XFC_EL2262 [► 37]).

Eingänge

```

VAR_INPUT
  HParam          : UDINT;
  FirstPathId     : UDINT;
  LastPathId      : UDINT;
  FirstOnPathPosition : LREAL;
  LastOnPathPosition : LREAL;
  OnCompensation  : LREAL;
  OffCompensation  : LREAL;
  Options         : ST_CamSwitchOptions;
END_VAR
    
```

| Name | Typ | Beschreibung |
|-------------|-------|--|
| HParam | UDINT | H-Parameterwert der mit dem Schaltzustand korrespondiert.. |
| FirstPathId | UDINT | Eindeutige, kontinuierlich über den Bahnverlauf steigende Id die zum relativen Bahnweg bis zum Einschalten der Nocke gehört. |
| LastPathId | UDINT | Eindeutige, kontinuierlich über den Bahnverlauf steigende Id die zum relativen Bahnweg bis zum Ausschalten der Nocke gehört. |

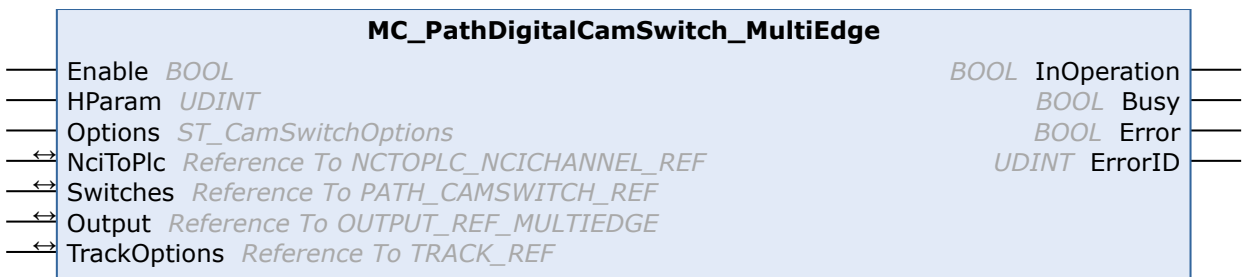
| Name | Typ | Beschreibung |
|---------------------|--------------------------------------|---|
| FirstOnPathPosition | LREAL | Relativer Bahnweg bis zum Einschalten der Nocke. |
| LastOnPathPosition | LREAL | Relativer Bahnweg bis zum Ausschalten der Nocke. |
| OnCompensation | LREAL | Kompensationszeit für die steigende Flanke der Nocke in [s]. Der Schaltzeitpunkt wird bei negativen Werten von OnCompensation vorgezogen und sonst verzögert. Der hier parametrisierte Wert OnCompensation hat Vorrang vor TRACK_REF |
| OffCompensation | LREAL | Kompensationszeit für die fallende Flanke der Nocke in [s]. Der Schaltzeitpunkt wird bei negativen Werten von OffCompensation vorgezogen und sonst verzögert. Der hier parametrisierte Wert OffCompensation hat Vorrang vor TRACK_REF |
| Options | <u>ST_CamSwitchOptions</u> [▶ 49] | Optionale Parameter |

Ein-/Ausgänge

```
VAR_IN_OUT
  NciToPlc : NciChannelToPlc;
  Output   : OUTPUT_REF;
END_VAR
```

| Name | Typ | Beschreibung |
|----------|--------------------------|---|
| NciToPlc | NciChannelToPlc | Struktur des zyklischen Kanalinterfaces von der NCI zur SPS. Auf diese Struktur wird nur lesend zugegriffen |
| Output | <u>OUTPUT_REF</u> [▶ 43] | Die Datenstruktur Output enthält den berechneten Zustand des digitalen Ausgangs und die zugehörigen Zeitstempel für die Ausgabe an einer digitalen XFC Ausgangsklemme |

4.2.1.2.2 MC_PathDigitalCamSwitch_MultiEdge



Der Funktionsbaustein MC_PathDigitalCamSwitch_MultiEdge ist ein digitales Nockenschaltwerk mit einer oder mehreren Nocken auf einer digitalen Ausgabespur. Der Funktionsbaustein ist in der Lage mehrere Schaltvorgänge während eines SPS-Zyklus durchzuführen. Die Schaltvorgänge werden durch Positionsnocken definiert. Weitere Ausgabespuren werden mit unabhängigen Instanzen des Funktionsbausteins realisiert.

Die Output-Datenstruktur enthält neben den Schaltzuständen des digitalen Ausgangs genaue Zeitinformationen über die nächsten Schaltvorgänge. Mit dieser Information kann die eigentliche Ausgabe an einer XFC-Multitimestamp-Ausgangsklemme mit einem nachgeschalteten Funktionsbaustein erfolgen (XFC_EL1259_MultiEdge, XFC_EL2212_MultiEdge [▶ 32], XFC_EL2262_MultiEdge [▶ 38] oder XFC_EL2258_MultiEdge [▶ 36]).



Zeitnocken und Bremsnocken können mit dem Baustein MC_PathDigitalCamSwitch_MultiEdge nicht verwendet werden. Klemmen ohne Multitimestamp-Funktionalität sind in Verbindung mit diesem Baustein nicht geeignet.

 **Eingänge**

```
VAR_INPUT
  Enable : BOOL;
  HParam : DINT;
  Options : ST_CamSwitchOptions;
END_VAR
```

| Name | Typ | Beschreibung |
|---------|---|--|
| Enable | BOOL | Über den Eingang Enable wird das Nockenschaltwerk aktiviert. Der Ausgangszustand bleibt unverändert, solange Enable=FALSE ist. |
| HParam | DINT | H-Parameterwert, der mit dem Schaltzustand TRUE korrespondiert. |
| Options | ST_CamSwitchOptions > 49 | Optionale Parameter |

 **Ein-/Ausgänge**

```
VAR_IN_OUT
  NciToPlc : NciChannelToPlc
  Switches : PATH_CAMSWITCH_REF;
  Output : OUTPUT_REF_MULTIEDGE;
  TrackOptions : TRACK_REF;
END_VAR
```

| Name | Typ | Beschreibung |
|--------------|--|---|
| NciToPlc | NciChannelToPlc | Struktur des zyklischen Kanalinterfaces von der NCI zur SPS. Auf diese Struktur wird nur lesend zugegriffen. |
| Switches | PATH_CAMSWITCH_REF > 58 | Die Datenstruktur Switches enthält einen Verweis auf die Parametrierung aller Nocken auf der Nockenspur. |
| Output | OUTPUT_REF_MULTIEDGE > 43 | Die Datenstruktur Output enthält die berechneten Zustände des digitalen Ausgangs und die zugehörigen Zeitstempel für die Ausgabe an einer digitalen XFC Ausgangsklemme. |
| TrackOptions | TRACK_REF > 44 | Die Datenstruktur TrackOptions enthält die Parametrierung der Nockenspur. |

 **Ausgänge**

```
VAR_OUTPUT
  InOperation : BOOL;
  Busy : BOOL;
  Error : BOOL;
  ErrorID : UDINT;
END_VAR
```

| Name | Typ | Beschreibung |
|-------------|-------|--|
| InOperation | BOOL | InOperation ist TRUE, solange das Nockenschaltwerk aktiv ist und die Nockenspur gemäß der Nockenparametrierung berechnet wird. |
| Busy | BOOL | Busy ist TRUE solange die Funktion des Bausteins nicht beendet ist. |
| Error | BOOL | Wird im Fehlerfall TRUE. |
| ErrorID | UDINT | Liefert bei einem gesetzten Error-Ausgang die Fehlernummer |

4.2.2 Datentypen

4.2.2.1 MC_PathCamSwitch

Der Datentyp MC_PathCamSwitch enthält alle Parameter einer digitalen Nocke für ein digitales Nockenschaltwerk [MC_PathDigitalCamSwitch_MultiEdge](#) [► 56].

```
TYPE MC_PathCamSwitch :
STRUCT
  FirstOnPosition : LREAL;
  LastOnPosition  : LREAL;
  FirstPathId     : UDINT;
  LastPathId      : UDINT;
  CamSwitchMode   : E_CamSwitchMode;
  Duration        : LREAL;
END_STRUCT
END_TYPE
```

Die Datenstruktur zur Parametrierung eines digitalen Nockenschaltwerks ist üblicherweise ein ARRAY OF MC_PathCamSwitch. Eine weitere Struktur [PATH_CAMSWITCH_REF](#) [► 58] verweist auf diese Struktur.

| Name | Typ | Beschreibung |
|-----------------|---|--|
| FirstOnPosition | LREAL | Erste Position, ab der die Nocke eingeschaltet ist. |
| LastOnPosition | LREAL | Letzte Position, bis zu der die Nocke eingeschaltet ist. Die Nockenfunktion ist invertiert, wenn LastOnPosition < FirstOnPosition. LastOnPosition wird bei Zeitnocken nicht verwendet. |
| FirstPathId | UDINT | Eindeutige, kontinuierlich über den Bahnverlauf steigende Id die zum relativen Bahnweg bis zum Einschalten der Nocke gehört. |
| LastPathId | UDINT | Eindeutige, kontinuierlich über den Bahnverlauf steigende Id die zum relativen Bahnweg bis zum Ausschalten der Nocke gehört. |
| CamSwitchMode | E_CamSwitchMode [► 51] | Typ der digitalen Nocke (Positionsnocke, Zeitnocke oder Bremsnocke). |
| Duration | LREAL | Duration definiert die Einschaltdauer der Nocke in [s] und wird ausschließlich bei Zeitnocken verwendet. |

```
TYPE E_CamSwitchMode :
(
  CAMSWITCHMODE_POSITION, (* position cam *)
  CAMSWITCHMODE_TIME,    (* time cam *)
  CAMSWITCHMODE_BREAK   (* break cam *)
);
END_TYPE
```

4.2.2.2 PATH_CAMSWITCH_REF

Der Datentyp PATH_CAMSWITCH_REF verweist auf eine Datenstruktur mit Nockenparametern für ein digitales Nockenschaltwerk [MC_PathDigitalCamSwitch_MultiEdge](#) .

```
TYPE PATH_CAMSWITCH_REF :
STRUCT
  NumberOfSwitches : UDINT;
  pSwitches        : POINTER TO MC_PathCamSwitch;
  SizeOfSwitches   : UDINT;
END_STRUCT
END_TYPE
```

| Name | Typ | Beschreibung |
|------------------|--|---|
| NumberOfSwitches | UDINT | Anzahl der genutzten Array-Elemente, kann kleiner sein als die maximale Anzahl. |
| pSwitches | POINTER TO MC_PathCamSwitch [► 58] | Pointer auf ein Array von Switchen, die mit ADR übergeben werden. |

| Name | Typ | Beschreibung |
|----------------|-------|----------------------------|
| SizeOfSwitches | UDINT | Maximale Größe des Arrays. |

Die eigentliche Datenstruktur zur Parametrierung eines digitalen Nockenschaltwerkes ist üblicherweise ein ARRAY OF `MC_PathCamSwitch` [► 58]. `PATH_CAMSWITCH_REF` verweist mit einem POINTER auf diese Struktur und definiert eindeutig die Größe der Struktur und die Anzahl der tatsächlich verwendeten Nocken.

Eine Variable des Typs `PATH_CAMSWITCH_REF` wird, wie in dem folgenden Beispiel gezeigt, initialisiert:

```
VAR
    CamSwitchArray    : ARRAY[1..3] OF MC_PathCamSwitch;
    CamSwitchRef      : PATH_CAMSWITCH_REF;
END_VAR

(* real number of defined digital cams *)
CamSwitchRef.NumberOfSwitches := 1; (* 1..3 *)
(* pointer to the digital cam data array *)
CamSwitchRef.pSwitches      := ADR(CamSwitchArray);
(* maximum size of the digital cam data array *)
CamSwitchRef.SizeOfSwitches := SIZEOF(CamSwitchArray);
```

4.3 Tc3_MC2_AdvancedHoming_XFC

4.3.1 Übersicht

Die TwinCAT Motion Control SPS-Bibliothek `Tc3_MC2_AdvancedHoming_XFC` enthält Funktionsbausteine zur Programmierung von Maschinenapplikationen. Die `Tc3_MC2_AdvancedHoming_XFC` richtet sich nach der ebenfalls überarbeiteten PLCopen Spezifikation für Motion Control Funktionsbausteine V2.0. Diese Bibliothek enthält Bausteine des Part 5–Homing Procedures (www.PLCopen.org).



TwinCAT Version

Die Bibliothek `Tc3_MC2_AdvancedHoming_XFC` ist ab der TwinCAT Version 3.1 Build 4020 einsetzbar. Bei remote programmierten Steuerungen ist darauf zu achten, dass sowohl auf dem Programmier-PC als auch auf dem Steuerungs-PC eine entsprechende Version installiert ist. Bei Steuerungen mit dem Betriebssystem Windows CE ist die Version des installierten Images ausschlaggebend.

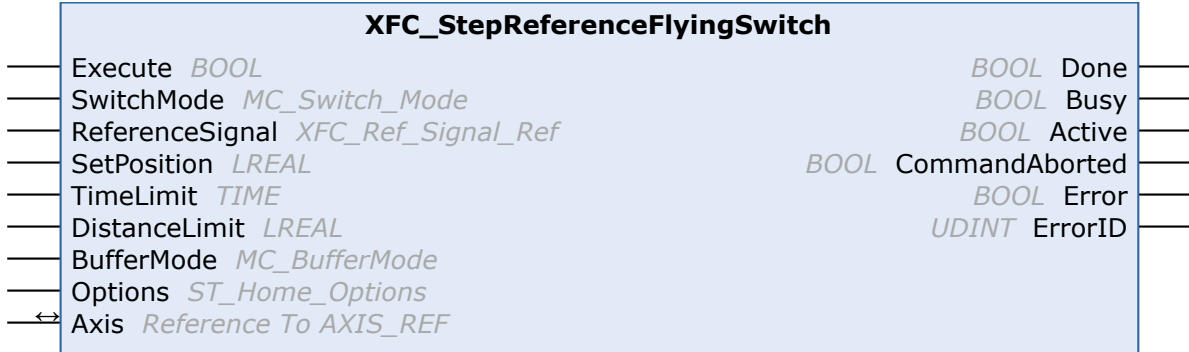


Weitere Informationen, u.a. zu Referenziersequenzen, finden Sie in der Motion-Bibliothek [Tc3_MC2_AdvancedHoming](#).

4.3.2 Funktionsbausteine

4.3.2.1 Referenzierfunktionen (passiv)

4.3.2.1.1 XFC_StepReferenceFlyingSwitch



Der Funktionsbaustein MC_StepReferenceFlyingSwitch führt eine Referenzierung während einer laufenden Bewegung über einen absolut positionierten externen physikalischen Schalter durch.

Die Ausführung startet oder modifiziert selbst keine Bewegung.

Eingänge

```
VAR_INPUT
  Execute      : BOOL;
  SwitchMode   : BOOL;
  ReferenceSignal : XFC_Ref_Signal_Ref;
  SetPosition  : LREAL;
  TimeLimit    : TIME;
  DistanceLimit : LREAL;
  BufferMode    : MC_BufferMode;
  Options      : ST_Home_Options;
END_VAR
```

| Name | Typ | Beschreibung |
|-----------------|---|--|
| Execute | BOOL | Mit einer steigenden Flanke am Eingang Execute wird das Kommando ausgeführt. |
| SwitchMode | BOOL | Enumeration legt die Endbedingung für den Suchvorgang fest. |
| ReferenceSignal | XFC_Ref_Signal_Ref [► 78] | Konfiguration der Referenzsignalquelle. |
| SetPosition | LREAL | Positionswert, auf den die Achsposition gesetzt werden soll. |
| TimeLimit | TIME | Ein Überschreiten der Zeit führt zum Abbruch des Suchvorgangs. |
| DistanceLimit | LREAL | Ein Überschreiten dieser Distanz, bezogen auf die Startposition, führt zum Abbruch des Suchvorgangs. |
| BufferMode | MC_BufferMode | Zurzeit nicht implementiert. |
| Options | ST_Home_Options | Wird momentan nicht verwendet. |

Ein/Ausgänge

```
VAR_IN_OUT
  Axis : AXIS_REF;
END_VAR
```

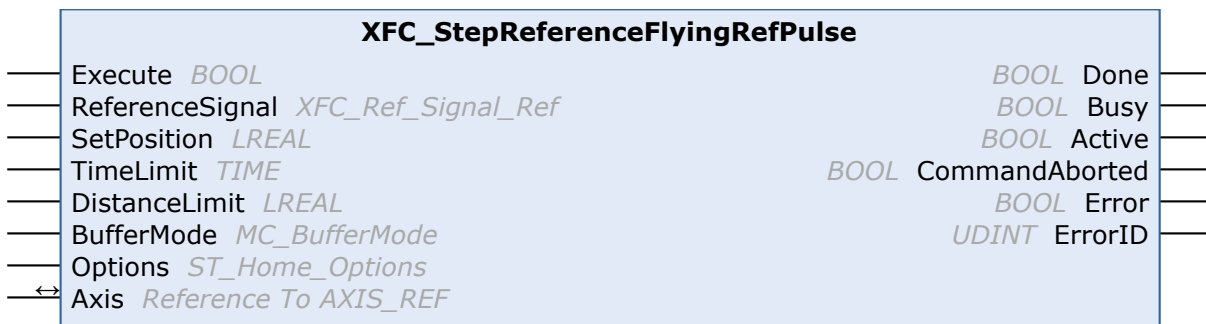
| Name | Typ | Beschreibung |
|------|--------------------------|--|
| Axis | AXIS_REF | Achsdatenstruktur vom Typ AXIS_REF, welche eine Achse eindeutig im System adressiert. Sie enthält unter anderem den aktuellen Status der Achse wie Position, Geschwindigkeit oder Fehlerzustand. |

Ausgänge

```
VAR_OUTPUT
  Done          : BOOL;
  Busy          : BOOL;
  Active        : BOOL;
  CommandAborted : BOOL;
  Error         : BOOL;
  ErrorID       : UDINT;
END_VAR
```

| Name | Typ | Beschreibung |
|----------------|-------|--|
| Done | BOOL | Wird TRUE, wenn das Kommando erfolgreich beendet wurde. |
| Busy | BOOL | Der Busy-Ausgang wird TRUE, sobald das Kommando mit Execute gestartet wird und bleibt TRUE, solange der Befehl abgearbeitet wird. Wenn Busy wieder FALSE wird, so ist der Funktionsbaustein bereit für einen neuen Auftrag. Gleichzeitig ist einer der Ausgänge Done, CommandAborted oder Error gesetzt. |
| Active | BOOL | Zeigt an, dass das Kommando ausgeführt wird. |
| CommandAborted | BOOL | Wird TRUE, wenn das Kommando nicht vollständig ausgeführt werden konnte. |
| Error | BOOL | Wird TRUE, sobald ein Fehler eintritt. |
| ErrorID | UDINT | Liefert bei einem gesetzten Fehler-Ausgang die Fehlernummer. |

4.3.2.1.2 XFC_StepReferenceFlyingRefPulse



Der Funktionsbaustein XFC_StepReferenceFlyingRefPulse führt eine Referenzierung während einer laufenden Bewegung auf den Nullimpuls eines Gebers durch.

Die Ausführung startet oder modifiziert selbst keine Bewegung.

Eingänge

```
VAR_INPUT
  Execute          : BOOL;
  ReferenceSignal  : XFC_Ref_Signal_Ref;
  SetPosition      : LREAL;
  TimeLimit        : TIME;
  DistanceLimit    : LREAL;
  BufferMode        : MC_BufferMode;
  Options          : ST_Home_Options;
END_VAR
```

| Name | Typ | Beschreibung |
|-----------------|------------------------------|---|
| Execute | BOOL | Mit einer steigenden Flanke am Eingang Execute wird das Kommando ausgeführt. |
| ReferenceSignal | XFC_Ref_Signal_Ref [▶ 78] | Konfiguration der Referenzsignalquelle. |
| SetPosition | LREAL | Positionswert, auf den die Achsposition gesetzt werden soll. |
| TimeLimit | TIME | Ein Überschreiten der Zeit führt zum Abbruch des Suchvorganges. |
| DistanceLimit | LREAL | Ein Überschreiten dieser Distanz bezogen auf die Startposition führt zum Abbruch des Suchvorganges. |
| BufferMode | MC_BufferMode | Zurzeit nicht implementiert. |
| Options | ST_Home_Options | Wird momentan nicht verwendet. |

🔌 Ein/Ausgänge

```
VAR_IN_OUT
  Axis : AXIS_REF;
END_VAR
```

| Name | Typ | Beschreibung |
|------|-----------------|--|
| Axis | <u>AXIS_REF</u> | Achsdatenstruktur vom Typ AXIS_REF, welche eine Achse eindeutig im System adressiert. Sie enthält unter anderem den aktuellen Status der Achse wie Position, Geschwindigkeit oder Fehlerzustand. |

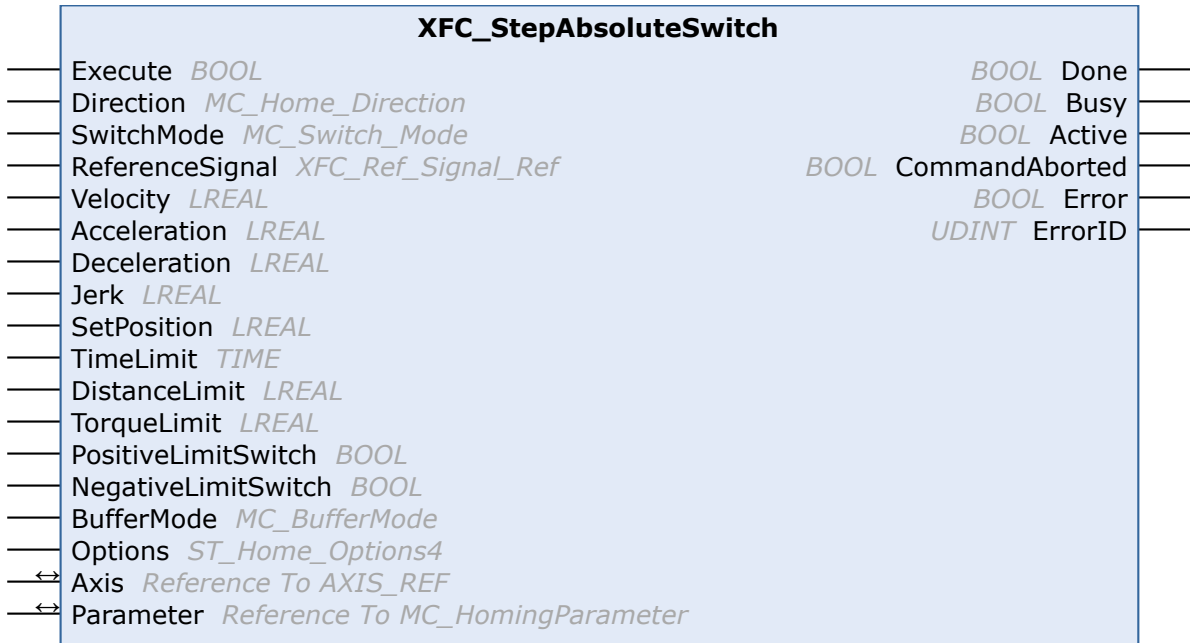
🔌 Ausgänge

```
VAR_OUTPUT
  Done      : BOOL;
  Busy      : BOOL;
  Active    : BOOL;
  CommandAborted : BOOL;
  Error     : BOOL;
  ErrorID   : UDINT;
END_VAR
```

| Name | Typ | Beschreibung |
|----------------|-------|--|
| Done | BOOL | Wird TRUE, wenn das Kommando erfolgreich beendet wurde. |
| Busy | BOOL | Der Busy-Ausgang wird TRUE, sobald das Kommando mit Execute gestartet wird und bleibt TRUE, solange der Befehl abgearbeitet wird. Wenn Busy wieder FALSE wird, so ist der Funktionsbaustein bereit für einen neuen Auftrag. Gleichzeitig ist einer der Ausgänge Done, CommandAborted oder Error gesetzt. |
| Active | BOOL | Zeigt an, dass das Kommando ausgeführt wird. |
| CommandAborted | BOOL | Wird TRUE, wenn das Kommando nicht vollständig ausgeführt werden konnte. |
| Error | BOOL | Wird TRUE, sobald ein Fehler eintritt. |
| ErrorID | UDINT | Liefert bei einem gesetzten Error-Ausgang die Fehlernummer. |

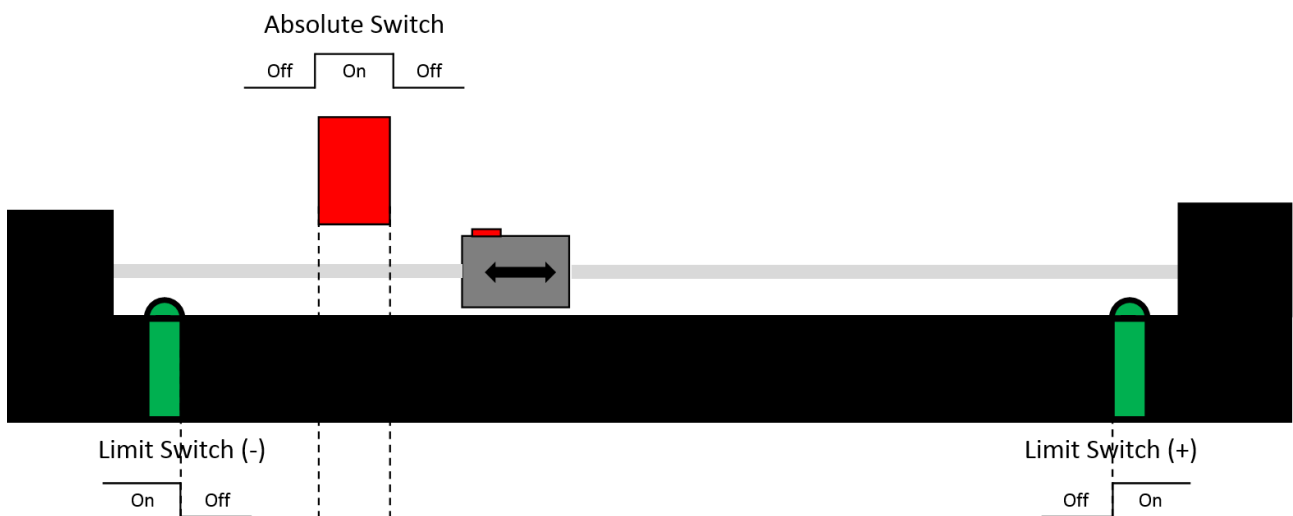
4.3.2.2 Schritt-Funktionen

4.3.2.2.1 XFC_StepAbsoluteSwitch

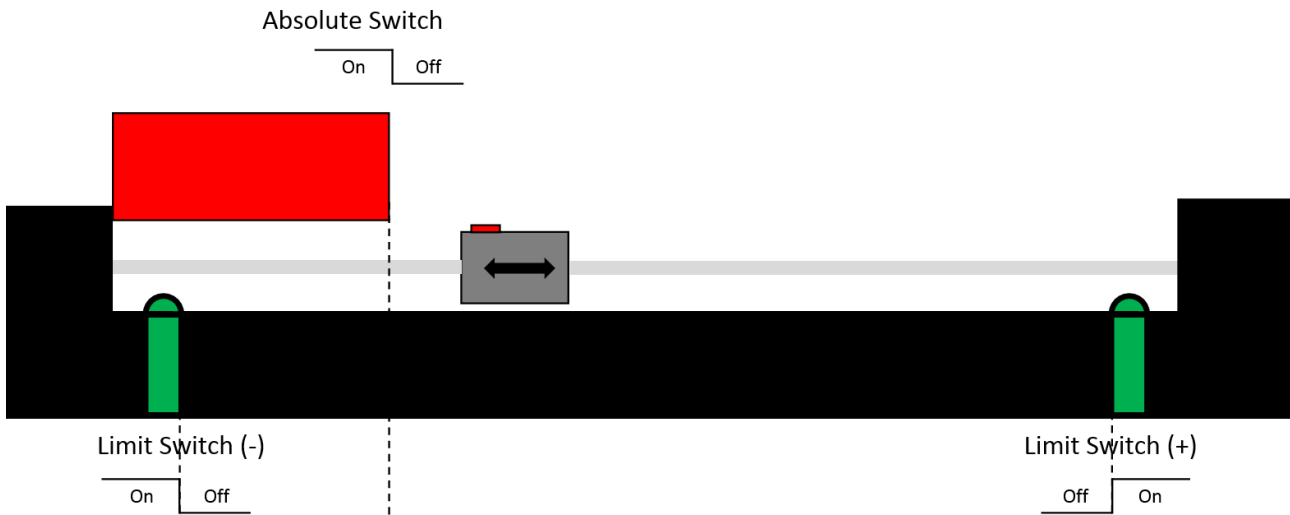


Dieser Funktionsbaustein führt eine Suche nach einem absolut positionierten externen physischen Schalter durch.

Generell hat ein absoluter Schalter zwei "Off"- und einen "On"-Bereich.



Kann der absolute Schalter nicht überfahren werden, hat dieser nur einen "Off"- und einen "On"-Bereich.



Eingänge

```

VAR_INPUT
  Execute           : BOOL;
  Direction         : MC_Home_Direction;
  SwitchMode       : MC_Switch_Mode;
  ReferenceSignal   : XFC_Ref_Signal_Ref;
  Velocity         : LREAL;
  Acceleration     : LREAL;
  Deceleration     : LREAL;
  Jerk             : LREAL;
  SetPosition      : LREAL;
  TimeLimit        : TIME;
  DistanceLimit    : LREAL;
  TorqueLimit      : LREAL;
  PositiveLimitSwitch : BOOL;
  NegativeLimitSwitch : BOOL;
  BufferMode       : MC_BufferMode;
  Options         : ST_Home_Options4;
END_VAR

```

| Name | Typ | Beschreibung |
|-----------------|---|--|
| Execute | BOOL | Mit einer steigenden Flanke am Eingang Execute wird das Kommando ausgeführt. |
| Direction | MC_Home_Direction | Enumeration legt die Startbewegungsrichtung für den Suchvorgang fest. |
| SwitchMode | MC_Switch_Mode | Enumeration legt die Endbedingung für den Suchvorgang fest. |
| ReferenceSignal | XFC_Ref_Signal_Ref [▶ 78] | Diese Struktur legt die Quelle des Referenznockensignals fest. |
| Velocity | LREAL | Maximale Geschwindigkeit, mit der gefahren werden soll (>0). |
| Acceleration | LREAL | Beschleunigung (≥0). Bei einem Wert von 0 wirkt die Standardbeschleunigung aus der Achs-konfiguration im System Manager. |
| Deceleration | LREAL | Verzögerung (≥0). Bei einem Wert von 0 wirkt die Standardverzögerung aus der Achskonfiguration im System Manager. |
| Jerk | LREAL | Ruck (≥0). Bei einem Wert von 0 wirkt der Standardruck aus der Achskonfiguration im System Manager. |
| SetPosition | LREAL | Positionswert, auf den die Achsposition gesetzt werden soll. |
| TimeLimit | TIME | Ein Überschreiten dieser Zeit führt zum Abbruch des Suchvorgangs. |

| Name | Typ | Beschreibung |
|---------------------|------------------|--|
| DistanceLimit | LREAL | Ein Überschreiten dieser Distanz bezogen auf die Startposition führt zum Abbruch des Suchvorgangs. |
| TorqueLimit | LREAL | Das Drehmoment des Motors wird auf diesen Wert begrenzt, um mechanischen Beschädigungen vorzubeugen. |
| PositiveLimitSwitch | BOOL | Signal des Hardwareendschalters in logisch positiver Bewegungsrichtung (PositiveLimitSwitch = FALSE im zulässigen Verfahrbereich). |
| NegativeLimitSwitch | BOOL | Signal des Hardwareendschalters in logisch negativer Bewegungsrichtung (NegativeLimitSwitch = FALSE im zulässigen Verfahrbereich). |
| BufferMode | MC_BufferMode | Zurzeit nicht implementiert |
| Options | ST_Home_Options4 | DisableDriveAccess: Bei Beckhoff-Antrieben auf FALSE, für Fremdantriebe gewöhnlich auf TRUE zu setzen (siehe Info). EnableLagErrorDetection: Bei den Step-Funktionen wird die Schleppfehlerüberwachung abgeschaltet, um einen reibungslosen Referenziervorgang zu gewährleisten. Sollte es bei einer Anwendung sinnvoll sein, die Schleppfehlerüberwachung aktiv zu behalten, kann dies über das Setzen dieses Flags erreicht werden. |

i Ist DisableDriveAccess = TRUE, ist der Anwender selbst dafür verantwortlich, benötigte Antriebsparameter zu modifizieren und zu rekonstruieren. Die für die angestrebte Homing-Sequenz benötigten Parameter sind mit dem Hersteller des Fremdantriebes abzustimmen.

 **Ein-/Ausgänge**

```
VAR_IN_OUT
  Axis      : AXIS_REF;
  Parameter : MC_HomingParameter;
END_VAR
```

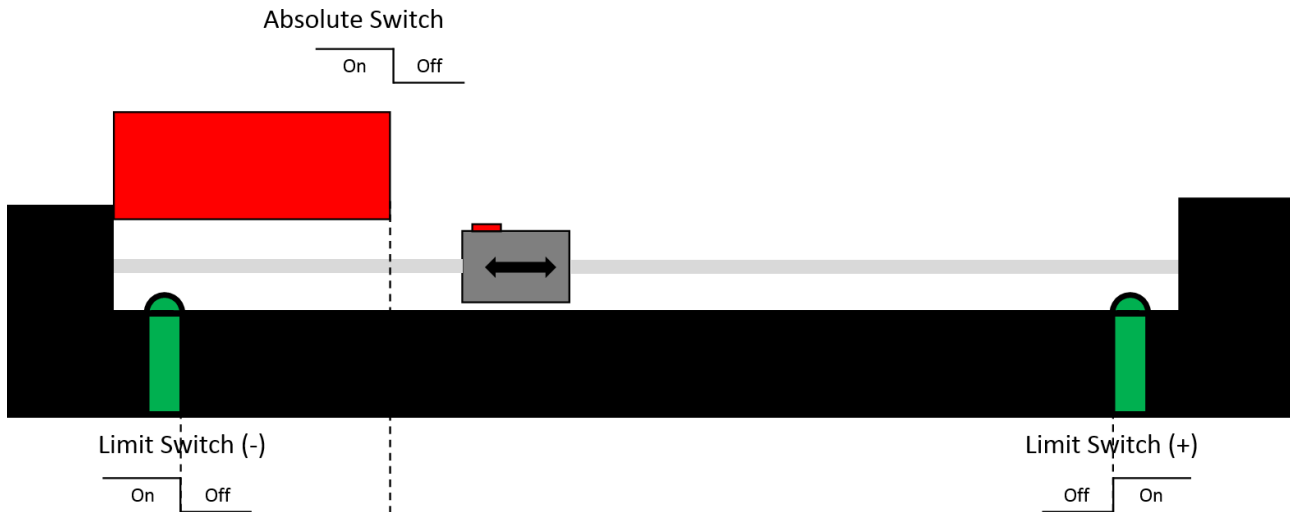
| Name | Typ | Beschreibung |
|-----------|--------------------|--|
| Axis | AXIS_REF | Achsdatenstruktur vom Typ AXIS_REF, welche eine Achse eindeutig im System adressiert. Sie enthält unter anderem den aktuellen Status der Achse wie Position, Geschwindigkeit oder Fehlerzustand. |
| Parameter | MC_HomingParameter | Datenstruktur vom Typ MC_HomingParameter, die über die gesamte Homing-Sequenz von Baustein zu Baustein übergeben werden muss. |

 **Ausgänge**

```
VAR_OUTPUT
  Done      : BOOL;
  Busy      : BOOL;
  Active    : BOOL;
  CommandAborted : BOOL;
  Error     : BOOL;
  ErrorID   : UDINT;
END_VAR
```

| Name | Typ | Beschreibung |
|------|------|---|
| Done | BOOL | Wird TRUE, wenn das Kommando erfolgreich beendet wurde. |
| Busy | BOOL | Der Busy-Ausgang wird TRUE, sobald das Kommando mit Execute gestartet wird und bleibt TRUE, |

Kann der absolute Schalter nicht überfahren werden, hat dieser nur einen "Off"- und einen "On"-Bereich.



Die „...Detection“-Version dieses Funktionsbausteins manipuliert die aktuelle Position der Achse am Ende der Sequenz nicht, sondern liefert die erkannte Position als „RecordedPosition“ an den Anwender zurück.

Eingänge

```

VAR_INPUT
    Execute           : BOOL;
    Direction         : MC_Home_Direction;
    SwitchMode        : MC_Switch_Mode;
    ReferenceSignal    : XFC_Ref_Signal_Ref;
    Velocity          : LREAL;
    Acceleration       : LREAL;
    Deceleration       : LREAL;
    Jerk              : LREAL;
    TimeLimit         : TIME;
    DistanceLimit     : LREAL;
    TorqueLimit       : LREAL;
    PositiveLimitSwitch : BOOL;
    NegativeLimitSwitch : BOOL;
    BufferMode         : MC_BufferMode;
    Options           : ST_Home_Options4;
END_VAR
    
```

| Name | Typ | Beschreibung |
|-----------------|----------------------------------|---|
| Execute | BOOL | Mit einer steigenden Flanke am Eingang Execute wird das Kommando ausgeführt. |
| Direction | <u>MC_Home_Direction</u> | Enumeration legt die Startbewegungsrichtung für den Suchvorgang fest. |
| SwitchMode | <u>MC_Switch_Mode</u> | Enumeration legt die Endbedingung für den Suchvorgang fest. |
| ReferenceSignal | <u>XFC_Ref_Signal_Ref [▶ 78]</u> | Diese Struktur legt die Quelle des Referenznockensignals fest. |
| Velocity | LREAL | Maximale Geschwindigkeit, mit der gefahren werden soll (>0). |
| Acceleration | LREAL | Beschleunigung (≥0). Bei einem Wert von 0 wirkt die Standardbeschleunigung aus der Achskonfiguration im System Manager. |
| Deceleration | LREAL | Verzögerung (≥0). Bei einem Wert von 0 wirkt die Standardverzögerung aus der Achskonfiguration im System Manager. |
| Jerk | LREAL | Ruck (≥0). Bei einem Wert von 0 wirkt der Standardruck aus der Achskonfiguration im System Manager. |
| TimeLimit | TIME | Ein Überschreiten dieser Zeit führt zum Abbruch des Suchvorgangs. |

| Name | Typ | Beschreibung |
|---------------------|------------------|---|
| DistanceLimit | LREAL | Ein Überschreiten dieser Distanz bezogen auf die Startposition führt zum Abbruch des Suchvorgangs. |
| TorqueLimit | LREAL | Das Drehmoment des Motors wird auf diesen Wert begrenzt, um mechanischen Beschädigungen vorzubeugen. |
| PositiveLimitSwitch | BOOL | Signal des Hardwareendschalters in logisch positiver Bewegungsrichtung (PositiveLimitSwitch = FALSE im zulässigen Verfahrbereich). |
| NegativeLimitSwitch | BOOL | Signal des Hardwareendschalters in logisch negativer Bewegungsrichtung (NegativeLimitSwitch = FALSE im zulässigen Verfahrbereich). |
| BufferMode | MC_BufferMode | Zurzeit nicht implementiert |
| Options | ST_Home_Options4 | DisableDriveAccess: Bei Beckhoff-Antrieben auf FALSE, für Fremdantriebe gewöhnlich auf TRUE zu setzen (siehe Info). EnableLagErrorDetection: Bei den Step-Funktionen wird die Schleppfehlerüberwachung abgeschaltet, um einen reibungslosen Referenziervorgang zu gewährleisten. Sollte es bei einer Anwendung sinnvoll sein die Schleppfehlerüberwachung aktiv zu behalten, kann dies über das Setzen dieses Flags erreicht werden. |

i Ist DisableDriveAccess = TRUE, ist der Anwender selbst dafür verantwortlich, benötigte Antriebsparameter zu modifizieren und zu rekonstruieren. Die für die angestrebte Homing-Sequenz benötigten Parameter sind mit dem Hersteller des Fremdantriebes abzustimmen.

Ein-/Ausgänge

```
VAR_IN_OUT
  Axis      : AXIS_REF;
  Parameter : MC_HomingParameter;
END_VAR
```

| Name | Typ | Beschreibung |
|-----------|--------------------|--|
| Axis | AXIS_REF | Achsdatenstruktur vom Typ AXIS_REF, welche eine Achse eindeutig im System adressiert. Sie enthält unter anderem den aktuellen Status der Achse wie Position, Geschwindigkeit oder Fehlerzustand. |
| Parameter | MC_HomingParameter | Datenstruktur vom Typ MC_HomingParameter, die über die gesamte Homing-Sequenz von Baustein zu Baustein übergeben werden muss. |

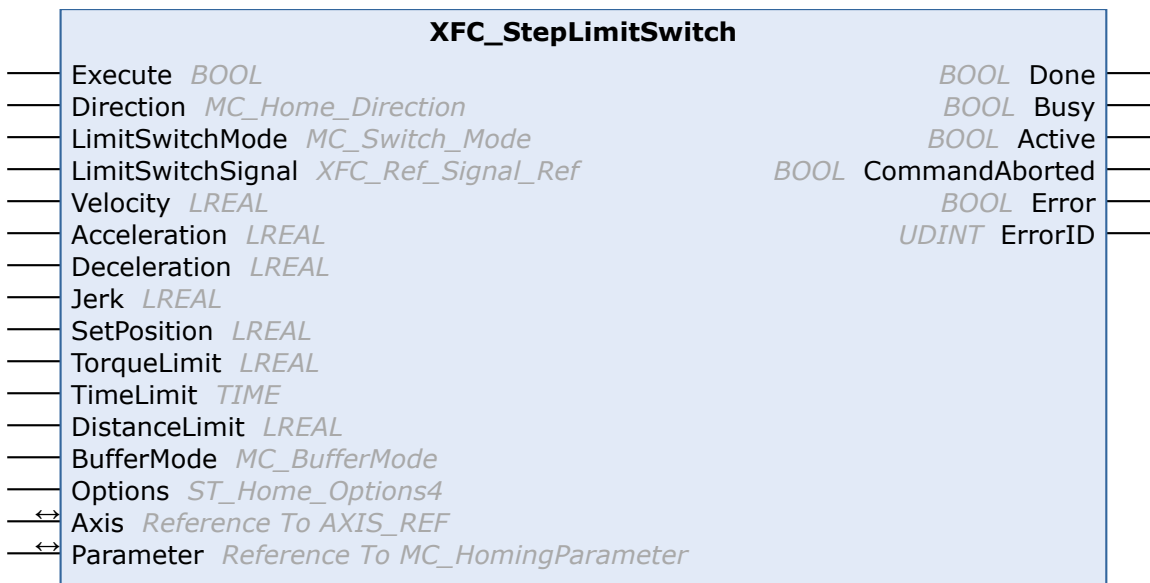
Ausgänge

```
VAR_OUTPUT
  Done           : BOOL;
  Busy           : BOOL;
  Active         : BOOL;
  CommandAborted : BOOL;
  Error          : BOOL;
  ErrorID        : UDINT;
  RecordedPosition : LREAL;
END_VAR
```

| Name | Typ | Beschreibung |
|------|------|---|
| Done | BOOL | Wird TRUE, wenn das Kommando erfolgreich beendet wurde. |

| Name | Typ | Beschreibung |
|------------------|-------|--|
| Busy | BOOL | Der Busy-Ausgang wird TRUE, sobald das Kommando mit Execute gestartet wird und bleibt TRUE, solange der Befehl abgearbeitet wird. Wenn Busy wieder FALSE wird, so ist der Funktionsbaustein bereit für einen neuen Auftrag. Gleichzeitig ist einer der Ausgänge Done, CommandAborted oder Error gesetzt. |
| Active | BOOL | Zeigt an, dass das Kommando ausgeführt wird. |
| CommandAborted | BOOL | Wird TRUE, wenn das Kommando nicht vollständig ausgeführt werden konnte. |
| Error | BOOL | Wird TRUE, sobald ein Fehler eintritt. |
| ErrorID | UDINT | Liefert bei einem gesetzten Error-Ausgang die Fehlernummer. |
| RecordedPosition | LREAL | Achsposition, an der das Ereignis registriert wurde. |

4.3.2.2.3 XFC_StepLimitSwitch



Dieser Funktionsbaustein führt die Suche nach einem Hardwareendschalter durch.

Eingänge

```

VAR_INPUT
  Execute          : BOOL;
  Direction        : MC_Home_Direction;
  LimitSwitchMode  : MC_Switch_Mode;
  LimitSwitchSignal : XFC_Ref_Signal_Ref;
  Velocity         : LREAL;
  Acceleration     : LREAL;
  Deceleration     : LREAL;
  Jerk             : LREAL;
  SetPosition      : LREAL;
  TimeLimit        : TIME;
  DistanceLimit    : LREAL;
  TorqueLimit      : LREAL;
  BufferMode        : MC_BufferMode;
  Options          : ST_Home_Options4;
END_VAR
    
```

| Name | Typ | Beschreibung |
|-----------|--------------------------|--|
| Execute | BOOL | Mit einer steigenden Flanke am Eingang Execute wird das Kommando ausgeführt. |
| Direction | <u>MC_Home_Direction</u> | Enumeration legt die Startbewegungsrichtung für den Suchvorgang fest. |

| Name | Typ | Beschreibung |
|-------------------|---------------------------|--|
| LimitSwitchMode | MC Switch Mode | Enumeration legt die Endbedingung für den Suchvorgang fest. |
| LimitSwitchSignal | XFC Ref Signal Ref [► 78] | Diese Struktur legt die Quelle des Referenznockensignals fest. |
| Velocity | LREAL | Maximale Geschwindigkeit, mit der gefahren werden soll (>0). |
| Acceleration | LREAL | Beschleunigung (≥ 0). Bei einem Wert von 0 wirkt die Standardbeschleunigung aus der Achs-konfiguration im System Manager. |
| Deceleration | LREAL | Verzögerung (≥ 0). Bei einem Wert von 0 wirkt die Standardverzögerung aus der Achskonfiguration im System Manager. |
| Jerk | LREAL | Ruck (≥ 0). Bei einem Wert von 0 wirkt der Standardruck aus der Achskonfiguration im System Manager. |
| SetPosition | LREAL | Positionswert, auf den die Achsposition gesetzt werden soll. |
| TimeLimit | TIME | Ein Überschreiten dieser Zeit führt zum Abbruch des Suchvorgangs. |
| DistanceLimit | LREAL | Ein Überschreiten dieser Distanz bezogen auf die Startposition führt zum Abbruch des Suchvorgangs. |
| TorqueLimit | LREAL | Das Drehmoment des Motors wird auf diesen Wert begrenzt, um mechanischen Beschädigungen vorzubeugen. |
| BufferMode | MC_BufferMode | Zurzeit nicht implementiert. |
| Options | ST Home Options4 | <p>DisableDriveAccess: Bei Beckhoff-Antrieben auf FALSE, für Fremdantriebe gewöhnlich auf TRUE zu setzen (siehe Info).</p> <p>EnableLagErrorDetection: Bei den Step-Funktionen wird die Schleppfehlerüberwachung abgeschaltet, um einen reibungslosen Referenziervorgang zu gewährleisten. Sollte es bei einer Anwendung sinnvoll sein die Schleppfehlerüberwachung aktiv zu behalten, kann dies über das Setzen dieses Flags erreicht werden.</p> |



Ist DisableDriveAccess = TRUE, ist der Anwender selbst dafür verantwortlich, benötigte Antriebsparameter zu modifizieren und zu rekonstruieren. Die für die angestrebte Homing-Sequenz benötigten Parameter sind mit dem Hersteller des Fremdantriebes abzustimmen.

 /  **Ein-/Ausgänge**

```
VAR_IN_OUT
  Axis      : AXIS_REF;
  Parameter : MC_HomingParameter;
END_VAR
```

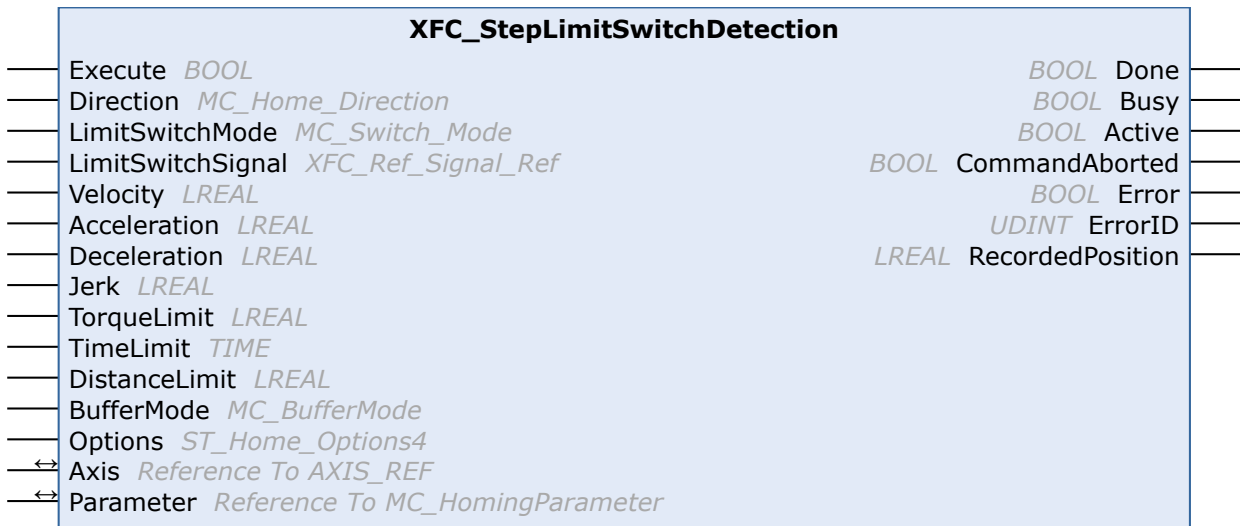
| Name | Typ | Beschreibung |
|-----------|--------------------|--|
| Axis | <u>AXIS_REF</u> | Achsdatenstruktur vom Typ <u>AXIS_REF</u> , welche eine Achse eindeutig im System adressiert. Sie enthält unter anderem den aktuellen Status der Achse wie Position, Geschwindigkeit oder Fehlerzustand. |
| Parameter | MC_HomingParameter | Datenstruktur vom Typ <u>MC_HomingParameter</u> , die über die gesamte Homing-Sequenz von Baustein zu Baustein übergeben werden muss. |

 **Ausgänge**

```
VAR_OUTPUT
  Done      : BOOL;
  Busy      : BOOL;
  Active    : BOOL;
  CommandAborted : BOOL;
  Error     : BOOL;
  ErrorID   : UDINT;
END_VAR
```

| Name | Typ | Beschreibung |
|----------------|-------|--|
| Done | BOOL | Wird TRUE, wenn das Kommando erfolgreich beendet wurde. |
| Busy | BOOL | Der Busy-Ausgang wird TRUE, sobald das Kommando mit Execute gestartet wird und bleibt TRUE, solange der Befehl abgearbeitet wird. Wenn Busy wieder FALSE wird, so ist der Funktionsbaustein bereit für einen neuen Auftrag. Gleichzeitig ist einer der Ausgänge Done, CommandAborted oder Error gesetzt. |
| Active | BOOL | Zeigt an, dass das Kommando ausgeführt wird. |
| CommandAborted | BOOL | Wird TRUE, wenn das Kommando nicht vollständig ausgeführt werden konnte. |
| Error | BOOL | Wird TRUE, sobald ein Fehler eintritt. |
| ErrorID | UDINT | Liefert bei einem gesetzten Error-Ausgang die <u>Fehlernummer</u> . |

4.3.2.2.4 XFC_StepLimitSwitchDetection



Dieser Funktionsbaustein führt die Suche nach einem Hardwareendschalter durch.

Die „...Detection“-Version dieses Funktionsbausteins manipuliert die aktuelle Position der Achse am Ende der Sequenz nicht, sondern liefert die erkannte Position als „RecordedPosition“ an den Anwender zurück.

Eingänge

```

VAR_INPUT
Execute          : BOOL;
Direction        : MC_Home_Direction;
LimitSwitchMode  : MC_Switch_Mode;
LimitSwitchSignal : XFC_Ref_Signal_Ref;
Velocity         : LREAL;
Acceleration     : LREAL;
Deceleration     : LREAL;
Jerk            : LREAL;
TimeLimit       : TIME;
DistanceLimit   : LREAL;
TorqueLimit     : LREAL;
BufferMode      : MC_BufferMode;
Options         : ST_Home_Options4;
END_VAR

```

| Name | Typ | Beschreibung |
|-------------------|----------------------------------|--|
| Execute | BOOL | Mit einer steigenden Flanke am Eingang Execute wird das Kommando ausgeführt. |
| Direction | <u>MC_Home_Direction</u> | Enumeration legt die Startbewegungsrichtung für den Suchvorgang fest. |
| LimitSwitchMode | <u>MC_Switch_Mode</u> | Enumeration legt die Endbedingung für den Suchvorgang fest. |
| LimitSwitchSignal | <u>XFC_Ref_Signal_Ref</u> [▶ 78] | Diese Struktur legt die Quelle des Referenznockensignals fest. |
| Velocity | LREAL | Maximale Geschwindigkeit, mit der gefahren werden soll (>0). |
| Acceleration | LREAL | Beschleunigung (≥0). Bei einem Wert von 0 wirkt die Standardbeschleunigung aus der Achs-konfiguration im System Manager. |
| Deceleration | LREAL | Verzögerung (≥0). Bei einem Wert von 0 wirkt die Standardverzögerung aus der Achskonfiguration im System Manager. |
| Jerk | LREAL | Ruck (≥0). Bei einem Wert von 0 wirkt der Standardruck aus der Achskonfiguration im System Manager. |
| TimeLimit | TIME | Ein Überschreiten dieser Zeit führt zum Abbruch des Suchvorgangs. |

| Name | Typ | Beschreibung |
|---------------|------------------|--|
| DistanceLimit | LREAL | Ein Überschreiten dieser Distanz bezogen auf die Startposition führt zum Abbruch des Suchvorgangs. |
| TorqueLimit | LREAL | Das Drehmoment des Motors wird auf diesen Wert begrenzt, um mechanischen Beschädigungen vorzubeugen. |
| BufferMode | MC_BufferMode | Zurzeit nicht implementiert. |
| Options | ST_Home_Options4 | DisableDriveAccess: Bei Beckhoff-Antrieben auf FALSE, für Fremdantriebe gewöhnlich auf TRUE zu setzen (siehe Info). EnableLagErrorDetection: Bei den Step-Funktionen wird die Schleppfehlerüberwachung abgeschaltet, um einen reibungslosen Referenziervorgang zu gewährleisten. Sollte es bei einer Anwendung sinnvoll sein, die Schleppfehlerüberwachung aktiv zu behalten, kann dies über das Setzen dieses Flags erreicht werden. |

i Ist DisableDriveAccess = TRUE, ist der Anwender selbst dafür verantwortlich, benötigte Antriebsparameter zu modifizieren und zu rekonstruieren. Die für die angestrebte Homing-Sequenz benötigten Parameter sind mit dem Hersteller des Fremdantriebes abzustimmen.

 **Ein-/Ausgänge**

```
VAR_IN_OUT
  Axis      : AXIS_REF;
  Parameter : MC_HomingParameter;
END_VAR
```

| Name | Typ | Beschreibung |
|-----------|--------------------|--|
| Axis | AXIS_REF | Achsdatenstruktur vom Typ AXIS_REF, welche eine Achse eindeutig im System adressiert. Sie enthält unter anderem den aktuellen Status der Achse wie Position, Geschwindigkeit oder Fehlerzustand. |
| Parameter | MC_HomingParameter | Datenstruktur vom Typ MC_HomingParameter, die über die gesamte Homing-Sequenz von Baustein zu Baustein übergeben werden muss. |

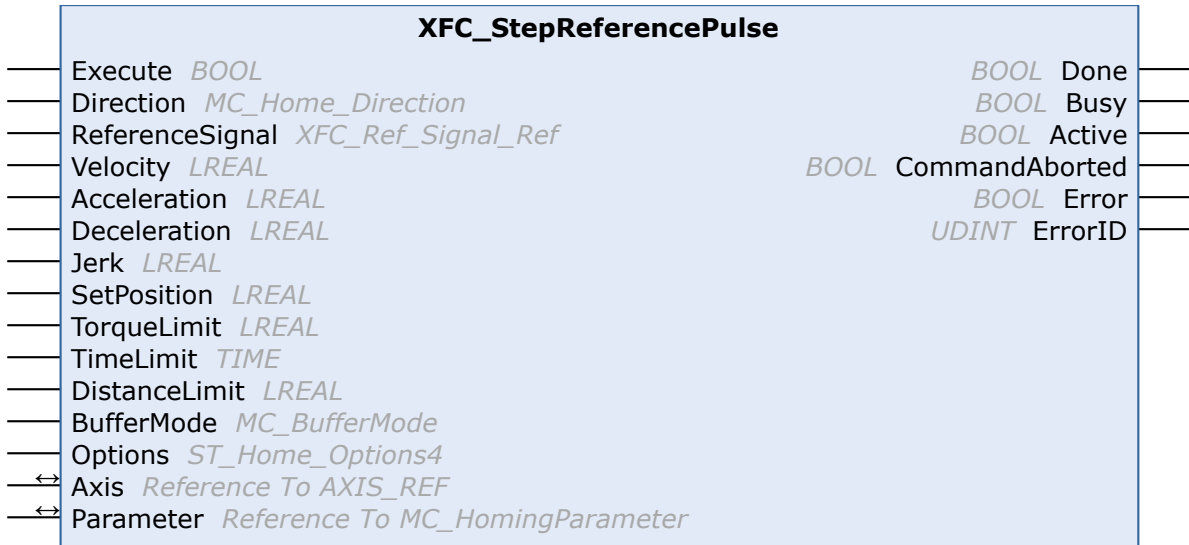
 **Ausgänge**

```
VAR_OUTPUT
  Done           : BOOL;
  Busy           : BOOL;
  Active         : BOOL;
  CommandAborted : BOOL;
  Error          : BOOL;
  ErrorID        : UDINT;
  RecordedPosition : LREAL;
END_VAR
```

| Name | Typ | Beschreibung |
|----------------|------|--|
| Done | BOOL | Wird TRUE, wenn das Kommando erfolgreich beendet wurde. |
| Busy | BOOL | Der Busy-Ausgang wird TRUE, sobald das Kommando mit Execute gestartet wird und bleibt TRUE, solange der Befehl abgearbeitet wird. Wenn Busy wieder FALSE wird, so ist der Funktionsbaustein bereit für einen neuen Auftrag. Gleichzeitig ist einer der Ausgänge Done, CommandAborted oder Error gesetzt. |
| Active | BOOL | Zeigt an, dass das Kommando ausgeführt wird. |
| CommandAborted | BOOL | Wird TRUE, wenn das Kommando nicht vollständig ausgeführt werden konnte. |
| Error | BOOL | Wird TRUE, sobald ein Fehler eintritt. |

| Name | Typ | Beschreibung |
|------------------|-------|---|
| ErrorID | UDINT | Liefert bei einem gesetzten Error-Ausgang die Fehlernummer. |
| RecordedPosition | LREAL | Achsposition, an der das Ereignis registriert wurde. |

4.3.2.2.5 XFC_StepReferencePulse



Dieser Funktionsbaustein führt die Suche nach einem Nullimpuls eines Gebers durch.

Ein Nullimpuls ist nicht in allen Gebern vorhanden und tritt nur einmal pro Geberumdrehung auf. Der Vorteil der Verwendung eines Nullimpulses zum Referenzieren liegt in der hohen Genauigkeit dieses Signals gegenüber dem eines Standardsensors.

Eingänge

```

VAR_INPUT
  Execute      : BOOL;
  Direction    : MC_Home_Direction;
  SwitchMode   : MC_Switch_Mode;
  ReferenceSignal : XFC_Ref_Signal_Ref;
  Velocity     : LREAL;
  Acceleration : LREAL;
  Deceleration : LREAL;
  Jerk        : LREAL;
  SetPosition  : LREAL;
  TimeLimit   : TIME;
  DistanceLimit : LREAL;
  TorqueLimit  : LREAL;
  BufferMode   : MC_BufferMode;
  Options     : ST_Home_Options4;
END_VAR

```

| Name | Typ | Beschreibung |
|-----------------|----------------------------------|--|
| Execute | BOOL | Mit einer steigenden Flanke am Eingang Execute wird das Kommando ausgeführt. |
| Direction | <u>MC_Home_Direction</u> | Enumeration legt die Startbewegungsrichtung für den Suchvorgang fest. |
| SwitchMode | <u>MC_Switch_Mode</u> | Enumeration legt die Endbedingung für den Suchvorgang fest. |
| ReferenceSignal | <u>XFC Ref Signal Ref [► 78]</u> | Diese Struktur legt die Quelle des Referenznockensignals fest. |
| Velocity | LREAL | Maximale Geschwindigkeit, mit der gefahren werden soll (>0). |
| Acceleration | LREAL | Beschleunigung (≥0). Bei einem Wert von 0 wirkt die Standardbeschleunigung aus der Achs-konfiguration im System Manager. |

| Name | Typ | Beschreibung |
|---------------|------------------|---|
| Deceleration | LREAL | Verzögerung (≥ 0). Bei einem Wert von 0 wirkt die Standardverzögerung aus der Achskonfiguration im System Manager. |
| Jerk | LREAL | Ruck (≥ 0). Bei einem Wert von 0 wirkt der Standardruck aus der Achskonfiguration im System Manager. |
| SetPosition | LREAL | Positionswert, auf den die Achsposition gesetzt werden soll. |
| TimeLimit | TIME | Ein Überschreiten dieser Zeit führt zum Abbruch des Suchvorgangs. |
| DistanceLimit | LREAL | Ein Überschreiten dieser Distanz bezogen auf die Startposition führt zum Abbruch des Suchvorgangs. |
| TorqueLimit | LREAL | Das Drehmoment des Motors wird auf diesen Wert begrenzt, um mechanischen Beschädigungen vorzubeugen. |
| BufferMode | MC_BufferMode | Zurzeit nicht implementiert. |
| Options | ST_Home_Options4 | DisableDriveAccess: Bei Beckhoff-Antrieben auf FALSE, für Fremdantriebe gewöhnlich auf TRUE zu setzen (siehe Info). EnableLagErrorDetection: Bei den Step-Funktionen wird die Schleppfehlerüberwachung abgeschaltet, um einen reibungslosen Referenzvorgang zu gewährleisten. Sollte es bei einer Anwendung sinnvoll sein, die Schleppfehlerüberwachung aktiv zu behalten, kann dies über das Setzen dieses Flags erreicht werden. |



Ist DisableDriveAccess = TRUE, ist der Anwender selbst dafür verantwortlich, benötigte Antriebsparameter zu modifizieren und zu rekonstruieren. Die für die angestrebte Homing-Sequenz benötigten Parameter sind mit dem Hersteller des Fremdantriebes abzustimmen.

Ein-/Ausgänge

```
VAR_IN_OUT
  Axis      : AXIS_REF;
  Parameter : MC_HomingParameter;
END_VAR
```

| Name | Typ | Beschreibung |
|-----------|--------------------|--|
| Axis | AXIS_REF | Achsdatenstruktur vom Typ AXIS_REF, welche eine Achse eindeutig im System adressiert. Sie enthält unter anderem den aktuellen Status der Achse wie Position, Geschwindigkeit oder Fehlerzustand. |
| Parameter | MC_HomingParameter | Datenstruktur vom Typ MC_HomingParameter, die über die gesamte Homing-Sequenz von Baustein zu Baustein übergeben werden muss. |

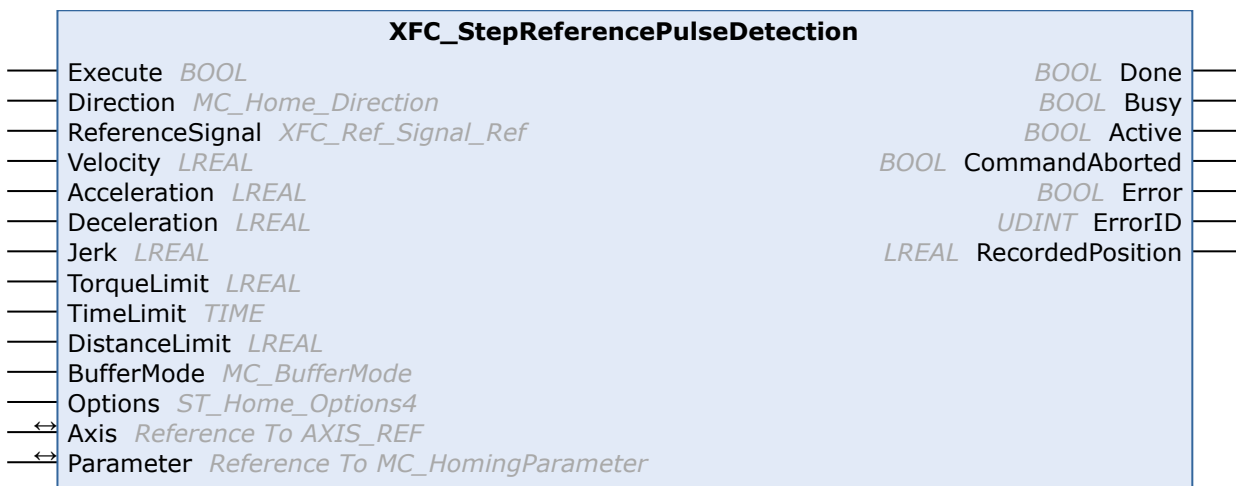
Ausgänge

```
VAR_OUTPUT
  Done      : BOOL;
  Busy      : BOOL;
  Active    : BOOL;
  CommandAborted : BOOL;
  Error     : BOOL;
  ErrorID   : UDINT;
END_VAR
```

| Name | Typ | Beschreibung |
|------|------|---|
| Done | BOOL | Wird TRUE, wenn das Kommando erfolgreich beendet wurde. |

| Name | Typ | Beschreibung |
|----------------|-------|--|
| Busy | BOOL | Der Busy-Ausgang wird TRUE, sobald das Kommando mit Execute gestartet wird und bleibt TRUE, solange der Befehl abgearbeitet wird. Wenn Busy wieder FALSE wird, so ist der Funktionsbaustein bereit für einen neuen Auftrag. Gleichzeitig ist einer der Ausgänge Done, CommandAborted oder Error gesetzt. |
| Active | BOOL | Zeigt an, dass das Kommando ausgeführt wird. |
| CommandAborted | BOOL | Wird TRUE, wenn das Kommando nicht vollständig ausgeführt werden konnte. |
| Error | BOOL | Wird TRUE, sobald ein Fehler eintritt. |
| ErrorID | UDINT | Liefert bei einem gesetzten Error-Ausgang die Fehlernummer. |

4.3.2.2.6 XFC_StepReferencePulseDetection



Dieser Funktionsbaustein führt die Suche nach einem Nullimpuls eines Gebers durch.

Ein Nullimpuls ist nicht in allen Gebern vorhanden und tritt nur einmal pro Geberumdrehung auf. Der Vorteil der Verwendung eines Nullimpulses zum Referenzieren liegt in der hohen Genauigkeit dieses Signals gegenüber dem eines Standardsensors.

Die „...Detection“-Version dieses Funktionsbausteins manipuliert die aktuelle Position der Achse am Ende der Sequenz nicht, sondern liefert die erkannte Position als „RecordedPosition“ an den Anwender zurück.

Eingänge

```

VAR_INPUT
  Execute          : BOOL;
  Direction        : MC_Home_Direction;
  SwitchMode       : MC_Switch_Mode;
  ReferenceSignal  : XFC_Ref_Signal_Ref;
  Velocity         : LREAL;
  Acceleration     : LREAL;
  Deceleration     : LREAL;
  Jerk             : LREAL;
  TimeLimit        : TIME;
  DistanceLimit    : LREAL;
  TorqueLimit      : LREAL;

```

```

BufferMode      : MC_BufferMode;
Options         : ST_Home_Options4;
END_VAR
    
```

| Name | Typ | Beschreibung |
|-----------------|----------------------------------|---|
| Execute | BOOL | Mit einer steigenden Flanke am Eingang Execute wird das Kommando ausgeführt. |
| Direction | <u>MC_Home_Direction</u> | Enumeration legt die Startbewegungsrichtung für den Suchvorgang fest. |
| SwitchMode | <u>MC_Switch_Mode</u> | Enumeration legt die Endbedingung für den Suchvorgang fest. |
| ReferenceSignal | <u>XFC_Ref_Signal_Ref [▶ 78]</u> | Diese Struktur legt die Quelle des Referenznockensignals fest. |
| Velocity | LREAL | Maximale Geschwindigkeit, mit der gefahren werden soll (>0). |
| Acceleration | LREAL | Beschleunigung (≥0). Bei einem Wert von 0 wirkt die Standardbeschleunigung aus der Achs-konfiguration im System Manager. |
| Deceleration | LREAL | Verzögerung (≥0). Bei einem Wert von 0 wirkt die Standardverzögerung aus der Achskonfiguration im System Manager. |
| Jerk | LREAL | Ruck (≥0). Bei einem Wert von 0 wirkt der Standardruck aus der Achskonfiguration im System Manager. |
| TimeLimit | TIME | Ein Überschreiten dieser Zeit führt zum Abbruch des Suchvorgangs. |
| DistanceLimit | LREAL | Ein Überschreiten dieser Distanz bezogen auf die Startposition führt zum Abbruch des Suchvorgangs. |
| TorqueLimit | LREAL | Das Drehmoment des Motors wird auf diesen Wert begrenzt, um mechanischen Beschädigungen vorzubeugen. |
| BufferMode | MC_BufferMode | Zurzeit nicht implementiert. |
| Options | <u>ST_Home_Options4</u> | <p>DisableDriveAccess: Bei Beckhoff-Antrieben auf FALSE, für Fremdantriebe gewöhnlich auf TRUE zu setzen (siehe Info).</p> <p>EnableLagErrorDetection: Bei den Step-Funktionen wird die Schleppfehlerüberwachung abgeschaltet, um einen reibungslosen Referenziervorgang zu gewährleisten. Sollte es bei einer Anwendung sinnvoll sein, die Schleppfehlerüberwachung aktiv zu behalten, kann dies über das Setzen dieses Flags erreicht werden.</p> |



Ist DisableDriveAccess = TRUE, ist der Anwender selbst dafür verantwortlich, benötigte Antriebsparameter zu modifizieren und zu rekonstruieren. Die für die angestrebte Homing-Sequenz benötigten Parameter sind mit dem Hersteller des Fremdantriebes abzustimmen.

Ein-/Ausgänge

```

VAR_IN_OUT
    Axis      : AXIS_REF;
    Parameter : MC_HomingParameter;
END_VAR
    
```

| Name | Typ | Beschreibung |
|------|-----------------|--|
| Axis | <u>AXIS_REF</u> | Achsdatenstruktur vom Typ AXIS_REF, welche eine Achse eindeutig im System adressiert. Sie enthält unter anderem den aktuellen Status der Achse wie Position, Geschwindigkeit oder Fehlerzustand. |

| Name | Typ | Beschreibung |
|-----------|--------------------|---|
| Parameter | MC_HomingParameter | Datenstruktur vom Typ MC_HomingParameter, die über die gesamte Homing-Sequenz von Baustein zu Baustein übergeben werden muss. |

Ausgänge

```
VAR_OUTPUT
  Done          : BOOL;
  Busy          : BOOL;
  Active        : BOOL;
  CommandAborted : BOOL;
  Error         : BOOL;
  ErrorID       : UDINT;
  RecordedPosition : LREAL;
END_VAR
```

| Name | Typ | Beschreibung |
|------------------|-------|--|
| Done | BOOL | Wird TRUE, wenn das Kommando erfolgreich beendet wurde. |
| Busy | BOOL | Der Busy-Ausgang wird TRUE, sobald das Kommando mit Execute gestartet wird und bleibt TRUE, solange der Befehl abgearbeitet wird. Wenn Busy wieder FALSE wird, so ist der Funktionsbaustein bereit für einen neuen Auftrag. Gleichzeitig ist einer der Ausgänge Done, CommandAborted oder Error gesetzt. |
| Active | BOOL | Zeigt an, dass das Kommando ausgeführt wird. |
| CommandAborted | BOOL | Wird TRUE, wenn das Kommando nicht vollständig ausgeführt werden konnte. |
| Error | BOOL | Wird TRUE, sobald ein Fehler eintritt. |
| ErrorID | UDINT | Liefert bei einem gesetzten Error-Ausgang die <u>Fehlernummer</u> . |
| RecordedPosition | LREAL | Achsposition, an der das Ereignis registriert wurde. |

4.3.3 Datentypen

4.3.3.1 Allgemein

4.3.3.1.1 XFC_Ref_Signal_Ref

```
TYPE XFC_Ref_Signal_Ref :
STRUCT
  Level          : BOOL;
  TimeStampRisingEdge : UDINT;
  TimeStampFallingEdge : UDINT;
END_STRUCT
END_TYPE
```

| Name | Typ | Beschreibung |
|----------------------|-------|---|
| Level | BOOL | Hier muss der aktuelle Signalzustand des Sensors übergeben werden. |
| TimeStampRisingEdge | UDINT | Zeitstempel der letzten steigenden Flanke des digitalen Eingangssignals. Falls das Eingangssignal ein 64 Bit Zeitstempel T_DCTIME liefert, werden hier nur die unteren 32 Bit eingespeist. Es muss daher sichergestellt sein, dass der Wert zum Zeitpunkt der Auswertung nicht älter als 2 Sekunden ist. |
| TimeStampFallingEdge | UDINT | Zeitstempel der letzten fallenden Flanke des digitalen Eingangssignals. Falls das Eingangssignal ein 64 Bit Zeitstempel T_DCTIME liefert, werden hier nur die unteren 32 Bit eingespeist. Es muss daher sichergestellt sein, dass der Wert zum Zeitpunkt der Auswertung nicht älter als 2 Sekunden ist. |

5 Beispiele

XFC TouchProbe

Der Funktionsbaustein [XFC_TouchProbe](#) [► 26] erfasst eine Achsposition zum Zeitpunkt der Flanke eines digitalen Eingangssignals (Messtasterfunktion). Das folgende Beispiel zeigt, wie der Funktionsbaustein freilaufend (`FreeRun = TRUE`) oder im Single-Shot-Mode (`FreeRun = FALSE`) verwendet werden kann.

Download: https://infosys.beckhoff.com/content/1031/TF5065_TC3_PLC_Motion_Control_XFC/XFC_NCI/Resources/12595575307.zip

NCI XFC Fast output switching

Dieses Beispiel zeigt, wie ein Ausgang mit Hilfe der XFC Funktionalitäten im Zusammenhang mit der NCI geschaltet werden kann.

Download: https://infosys.beckhoff.com/content/1031/TF5065_TC3_PLC_Motion_Control_XFC/XFC_NCI/Resources/13632374795.zip

6 Support und Service

Beckhoff und seine weltweiten Partnerfirmen bieten einen umfassenden Support und Service, der eine schnelle und kompetente Unterstützung bei allen Fragen zu Beckhoff Produkten und Systemlösungen zur Verfügung stellt.

Downloadfinder

Unser Downloadfinder beinhaltet alle Dateien, die wir Ihnen zum Herunterladen anbieten. Sie finden dort Applikationsberichte, technische Dokumentationen, technische Zeichnungen, Konfigurationsdateien und vieles mehr.

Die Downloads sind in verschiedenen Formaten erhältlich.

Beckhoff Niederlassungen und Vertretungen

Wenden Sie sich bitte an Ihre Beckhoff Niederlassung oder Ihre Vertretung für den lokalen Support und Service zu Beckhoff Produkten!

Die Adressen der weltweiten Beckhoff Niederlassungen und Vertretungen entnehmen Sie bitte unserer Internetseite: www.beckhoff.com

Dort finden Sie auch weitere Dokumentationen zu Beckhoff Komponenten.

Beckhoff Support

Der Support bietet Ihnen einen umfangreichen technischen Support, der Sie nicht nur bei dem Einsatz einzelner Beckhoff Produkte, sondern auch bei weiteren umfassenden Dienstleistungen unterstützt:

- Support
- Planung, Programmierung und Inbetriebnahme komplexer Automatisierungssysteme
- umfangreiches Schulungsprogramm für Beckhoff Systemkomponenten

Hotline: +49 5246 963-157

E-Mail: support@beckhoff.com

Beckhoff Service

Das Beckhoff Service-Center unterstützt Sie rund um den After-Sales-Service:

- Vor-Ort-Service
- Reparaturservice
- Ersatzteilservice
- Hotline-Service

Hotline: +49 5246 963-460

E-Mail: service@beckhoff.com

Beckhoff Unternehmenszentrale

Beckhoff Automation GmbH & Co. KG

Hülshorstweg 20
33415 Verl
Deutschland

Telefon: +49 5246 963-0

E-Mail: info@beckhoff.com

Internet: www.beckhoff.com

Mehr Informationen:
www.beckhoff.com/TF5065

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

