

Handbuch | DE

TS5065

TwinCAT 2 | Motion Control XFC/XFC NC I



Inhaltsverzeichnis

1	Vorwort	5
1.1	Hinweise zur Dokumentation.....	5
1.2	Sicherheitshinweise.....	6
2	Übersicht	7
3	Totzeitkompensation	8
4	PLC-API	13
4.1	TcMC2_XFC.....	13
4.1.1	CAMSWITCH_REF.....	13
4.1.2	MC_CamSwitch.....	13
4.1.3	MC_DigitalCamSwitch.....	14
4.1.4	MC_DigitalCamSwitch_MultiEdge.....	17
4.1.5	OUTPUT_REF.....	18
4.1.6	OUTPUT_REF_MULTIEDGE.....	19
4.1.7	ST_EL2258_Diagnostics.....	19
4.1.8	ST_EL2262_Diagnostics.....	20
4.1.9	TRACK_REF.....	21
4.1.10	XFC_BreakCam.....	22
4.1.11	XFC_EL1259_MultiEdge.....	24
4.1.12	XFC_EL2212_MultiEdge.....	25
4.1.13	XFC_EL2212_V2.....	26
4.1.14	XFC_EL2252_V2.....	27
4.1.15	XFC_EL2258_Multiedge.....	28
4.1.16	XFC_EL2262.....	29
4.1.17	XFC_EL2262_MultiEdge.....	31
4.1.18	XFC_ExtendDcTime.....	32
4.1.19	XFC_GetCurDcTaskTime.....	33
4.1.20	XFC_PositionAtTime.....	33
4.1.21	XFC_PositionCam.....	34
4.1.22	XFC_TimeCam.....	36
4.1.23	XFC_TimeOfPosition.....	38
4.1.24	XFC_TimeToModuloPosition.....	39
4.1.25	XFC_TimeToPosition.....	40
4.1.26	XFC_TouchProbe.....	42
4.1.27	XFC_TRIGGER_REF.....	45
4.2	TcNci_XFC.....	45
4.2.1	MC_PathCamSwitch.....	45
4.2.2	MC_PathDigitalCamSwitch_MultiEdge.....	46
4.2.3	PATH_CAMSWITCH_REF.....	47
4.2.4	XFC_PathPositionAtTime.....	48
4.2.5	XFC_PathPositionCam.....	49
4.2.6	XFC_TimeOfPathPosition.....	50
4.2.7	XFC_TimeToPathPosition.....	51

1 Vorwort

1.1 Hinweise zur Dokumentation

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

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

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

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

Disclaimer

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

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

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

Marken

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

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

Patente

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

EP1590927, EP1789857, EP1456722, EP2137893, DE102015105702

mit den entsprechenden Anmeldungen und Eintragungen in verschiedenen anderen Ländern.

EtherCAT®

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.

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

1.2 Sicherheitshinweise

Sicherheitsbestimmungen

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

Haftungsausschluss

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

Qualifikation des Personals

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

Erklärung der Symbole

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

GEFAHR

Akute Verletzungsgefahr!

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

WARNUNG

Verletzungsgefahr!

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

VORSICHT

Schädigung von Personen!

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

HINWEIS

Schädigung von Umwelt oder Geräten

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

Tipp oder Fingerzeig

i Dieses Symbol kennzeichnet Informationen, die zum besseren Verständnis beitragen.

2 Übersicht

Voraussetzung ist eine hochgenaue Totzeitkompensation der Achsen, die ab TwinCAT 2.11 für EtherCAT- und Sercos-Antriebe zur Verfügung steht.

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

Die Bibliothek TcMC2_XFC erleichtert die zeitgenaue Erfassung von Achspositionen und die positionsgenaue Ausgabe von Digitalsignalen in Verbindung mit EtherCAT-XFC-Klemmen (Zeitstempel-Klemmen und Oversampling-Klemmen).

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.

Die Bibliothek TcNci_XFC 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 TcMC2_XFC Bibliothek.

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

Um die Bibliothek in einem Projekt zu verwenden, müssen Sie zusätzlich folgende Bibliotheken miteinschließen:

TcNci.lib (enthält die Struktur des zyklischen Kanal-Interface)

TcMc2_XFC.lib [► 13] (enthält die benötigten Ausgabebausteine für die Ansteuerung der XFC-Ausgangsklemmen, sowie weitere wiederverwendete Strukturen)

TcMC2.lib (Wiederverwendete Strukturen)

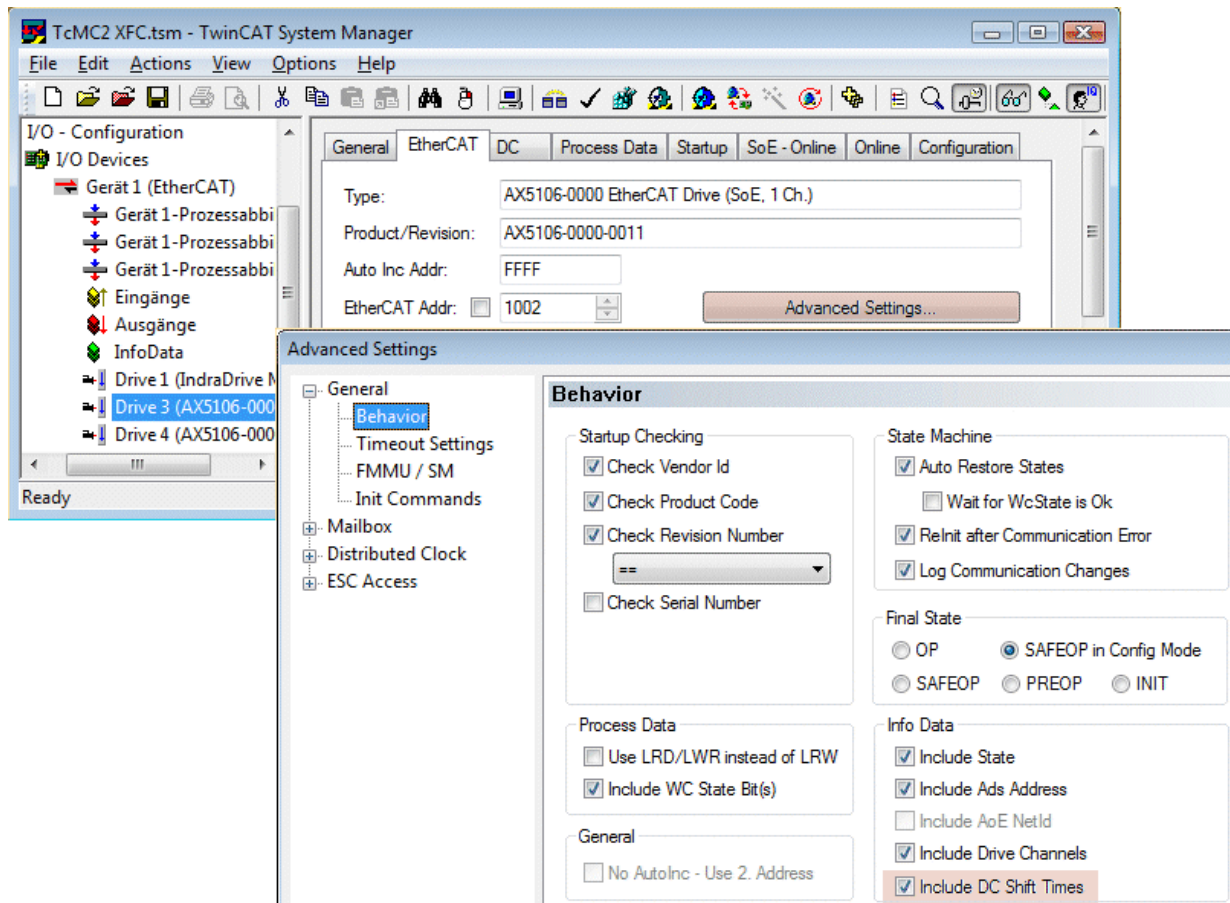
3 Totzeitkompensation

Voraussetzung für die hochgenaue Umrechnung von Positionen in Zeiten und umgekehrt ist eine exakte Totzeitkompensation der Achsen. Ab TwinCAT 2.11 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

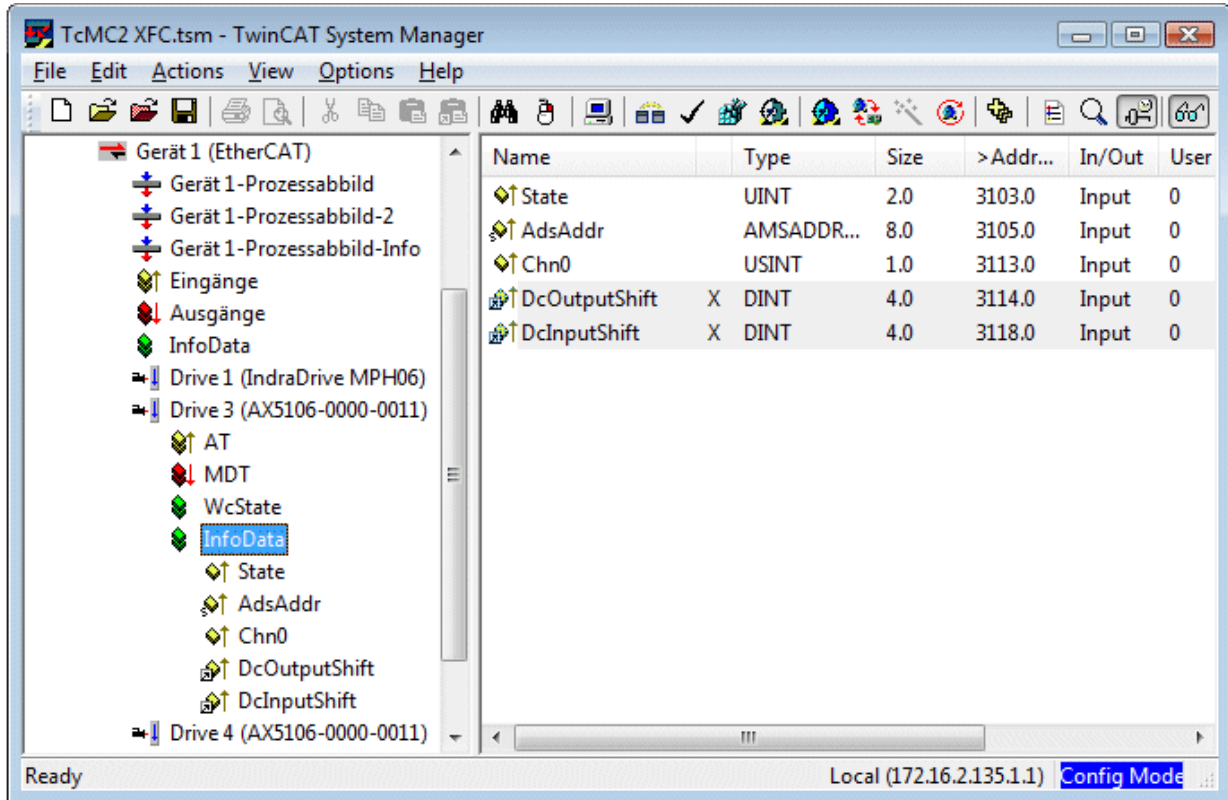
Die Unterstützung der Distributed Clocks muss bei EtherCAT-Antrieben zunächst wie folgt aktiviert werden:

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.

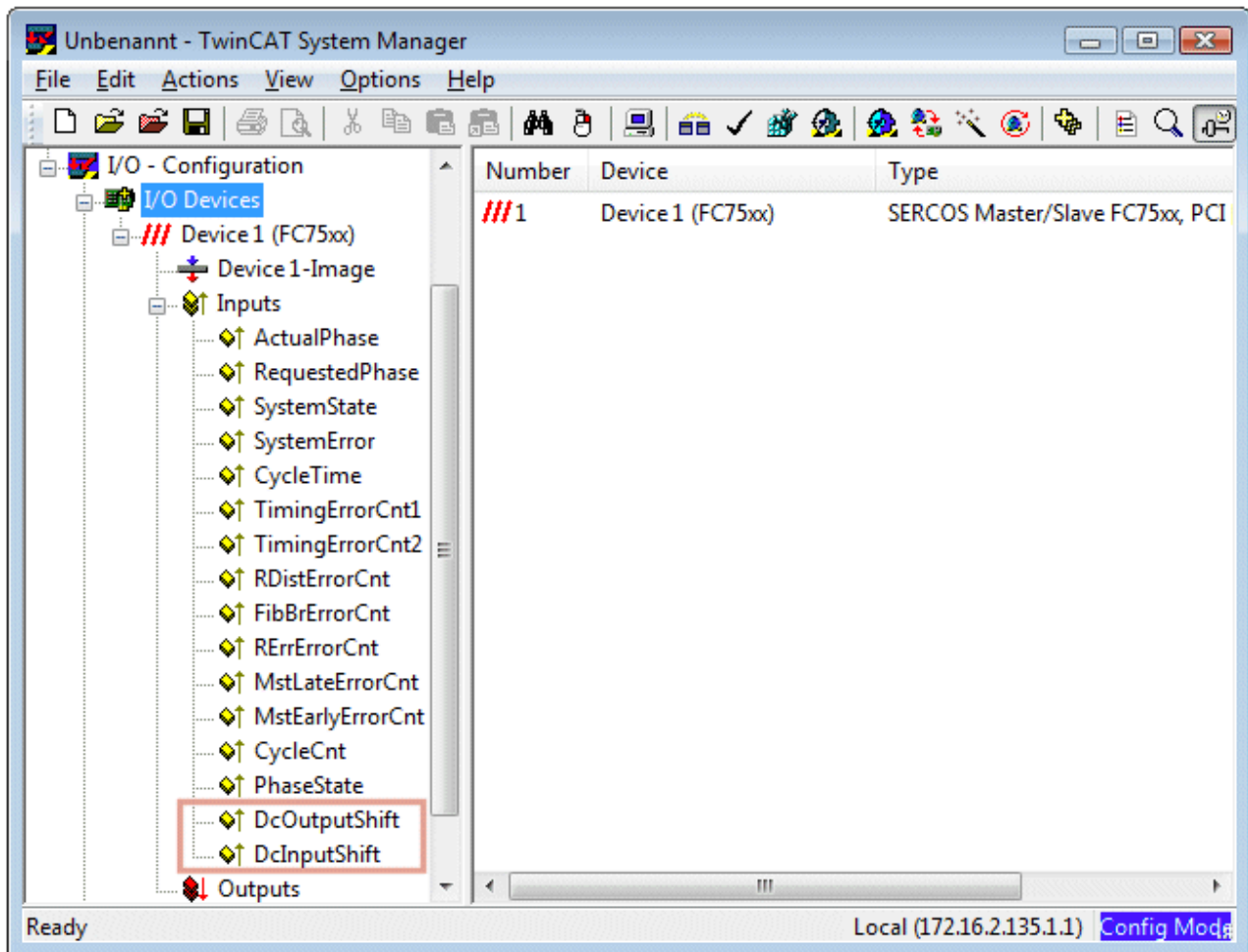


"DcInputShift" ist die Zeit, die benötigt wird, um Statusinformationen, wie z. B. die Istposition eines Antriebs, in die Steuerung zu übertragen. Also die Zeit zwischen der Erfassung und der Auswertung dieser Daten.

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

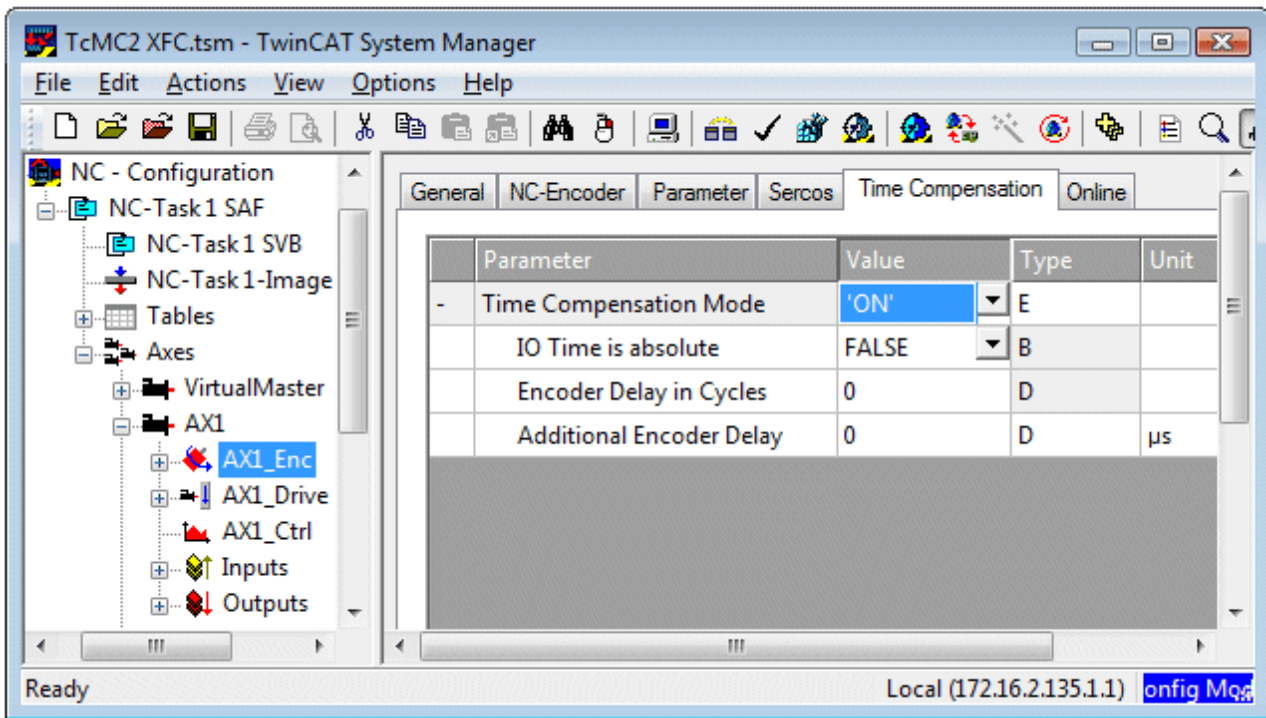
Die Zeitinformationen werden dynamisch vom System bereitgestellt und werden von der NC zur Totzeitkompensation einer Achse verwendet.

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.



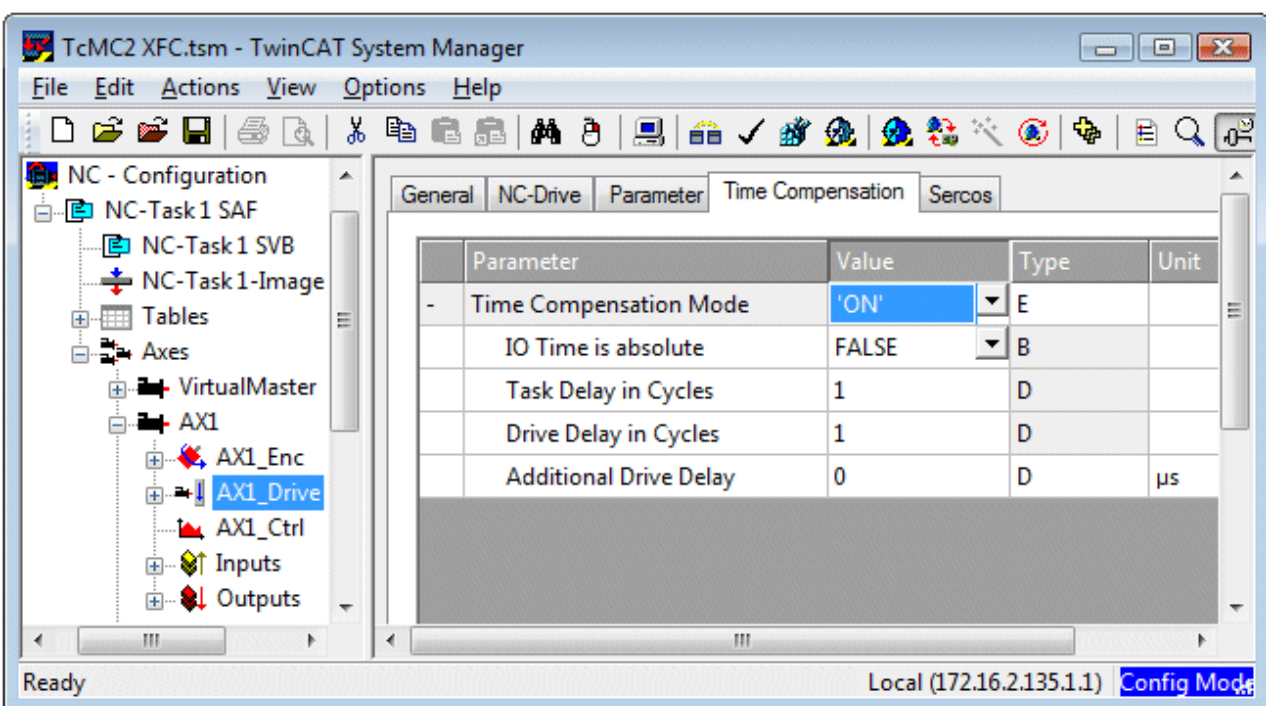
In besonderen Fällen, wenn es beispielsweise durch die verwendete Hardware zu zusätzlichen Totzeiten kommt, kann es notwendig sein, weitere Zeiten zu konfigurieren.

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 Drive 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 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.

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 \[► 33\]](#) , [XFC_TimeOfPosition \[► 38\]](#) und [XFC_PositionAtTime \[► 33\]](#).)

4 PLC-API

4.1 TcMC2_XFC

4.1.1 CAMSWITCH_REF

Der Datentyp *CAMSWITCH_REF* verweist auf eine Datenstruktur mit Nockenparametern für ein digitales Nockenschaltwerk *MC_DigitalCamSwitch*.

```

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

Die eigentliche Datenstruktur zur Parametrierung eines digitalen Nockenschaltwerks ist üblicherweise ein *ARRAY OF MC_CamSwitch* [► 13]. *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 *CAMSWITCH_REF* wird, wie in dem folgenden Beispiel gezeigt, initialisiert:

```

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

MC_CamSwitch [► 13]

```

(* 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		

4.1.2 MC_CamSwitch

Der Datentyp *MC_CamSwitch* enthält alle Parameter einer digitalen Nocke für ein digitales Nockenschaltwerk *MC_DigitalCamSwitch*.

```

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](#) [► 13] verweist auf diese Struktur.

FirstOnPosition	Erste Position, ab der die Nocke eingeschaltet ist.
LastOnPosition	Letzte Position, bis zu der die Nocke eingeschaltet ist. Die Nockenfunktion ist invertiert, wenn <i>LastOnPosition</i> < <i>FirstOnPosition</i> . <i>LastOnPosition</i> wird bei Zeitnocken nicht verwendet.
AxisDirection	AxisDirection legt fest, in welcher Fahrtrichtung der Achse die digitale Nocke aktiv ist (positiv, negativ oder beide Richtungen).
CamSwitchMode	Typ der digitalen Nocke (Positionsnocke, Zeitnocke oder Bremsnocke).
Duration	Duration definiert die Einschaltdauer der Nocke in [s] und wird ausschließlich bei Zeitnocken verwendet.

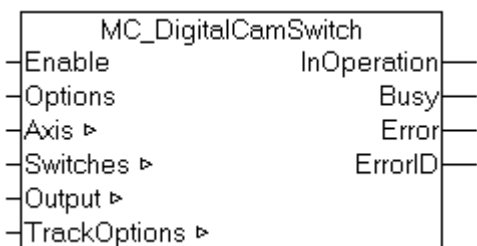
```

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 MC_DigitalCamSwitch



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 oder XFC_EL2262 [► 29]).

Eingänge

```

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

Enable	Über den Eingang <i>Enable</i> wird das Nockenschaltwerk aktiviert. Der Ausgangszustand bleibt unverändert, solange <i>Enable FALSE</i> ist.	
Options	Optionale Parameter	
Options.	EncoderIndex	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.
Options.	UseAcceleration	<i>UseAcceleration</i> kann TRUE gesetzt werden, um die Beschleunigung der Achse in die Positions-Berechnungen mit einzubeziehen. <i>UseAcceleration</i> kann von Vorteil sein, wenn die Sollwerte der Beschleunigung verwendet werden können. Bei Encoder-Achsen, die ein verrauschtes Positionssignal liefern, ist <i>UseAcceleration</i> eventuell von Nachteil, da auch die Beschleunigung fehlerhaft ist.

Ausgänge

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

InOperation	<i>InOperation</i> ist TRUE, solange das Nockenschaltwerk aktiv ist und die Nockenspur gemäß der Nockenparametrierung berechnet wird.
Busy	<i>Busy</i> ist TRUE solange die Funktion des Bausteins nicht beendet ist.
Error	Wird im Fehlerfall TRUE.
ErrorID	Liefert bei einem gesetzten Error-Ausgang die Fehlernummer

Ein/Ausgänge

```
VAR_IN_OUT
  Axis       : AXIS_REF;
  Switches   : CAMSWITCH_REF [▶ 13];
  Output     : OUTPUT_REF [▶ 18];
  TrackOptions : TRACK_REF [▶ 21];
END_VAR
```

Axis	Achsdatenstruktur
Switches	Die Datenstruktur <i>Switches</i> enthält einen Verweis auf die Parametrierung aller Nocken auf der Nockenspur.
Output	Die Datenstruktur <i>Output</i> enthält den berechneten Zustand des digitalen Ausgangs und die zugehörigen Zeitstempel für die Ausgabe an einer digitalen XFC Ausgangsklemme
TrackOptions	Die Datenstruktur <i>TrackOptions</i> enthält die Parametrierung der Nockenspur.

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.

Beispiel für zwei digitale Nockenspuren

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

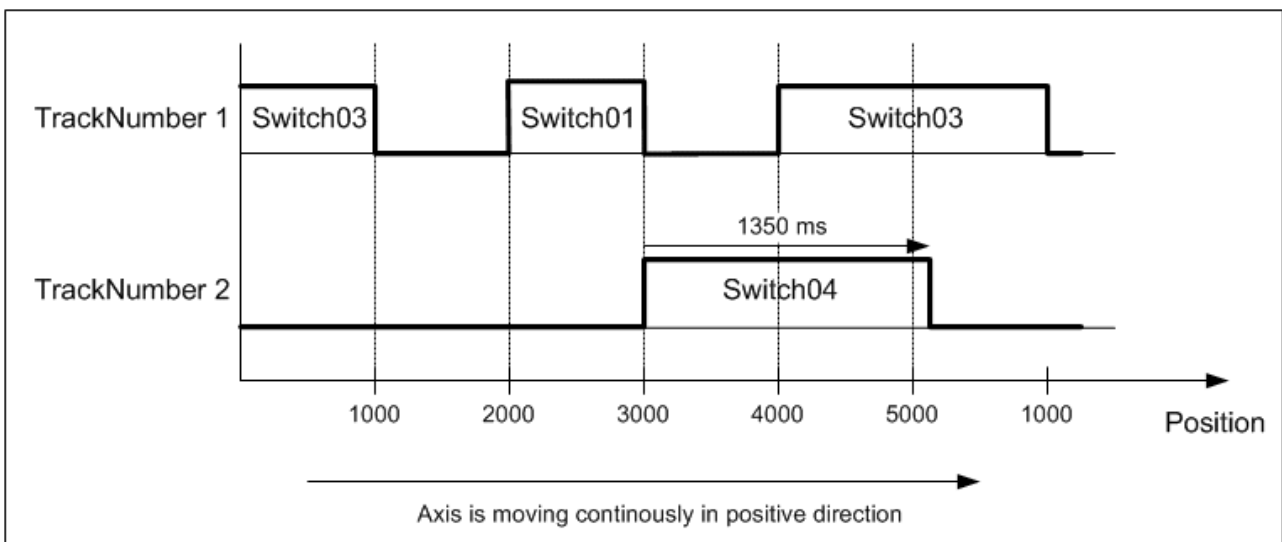
CamSwitchArrayTrack1 : Array [1..n] OF MC_CamSwitch					
	Switch 1	Switch 2	Switch 3	...	Switch n
FirstOnPosition	2000	2500	4000		
LastOnPosition	3000	3000	1000		
AxisDirection	POSITIVE	NEGATIVE	BOTH		
CamSwitchMode	POSITION	POSITION	POSITION		
Duration [s]	—	—	—		

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

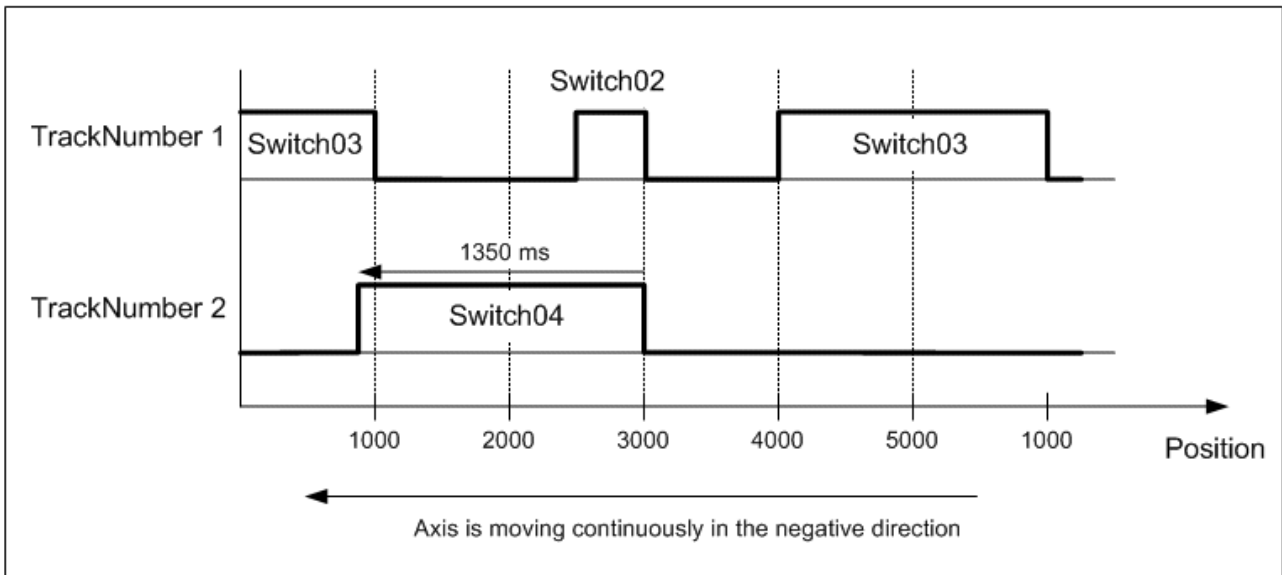
CamSwitchArrayTrack2 : Array [1..m] OF MC_CamSwitch			
	Switch 1	...	Switch m
FirstOnPosition	3000		
LastOnPosition	—		
AxisDirection	BOTH		
CamSwitchMode	TIME		
Duration [s]	1,350		

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.

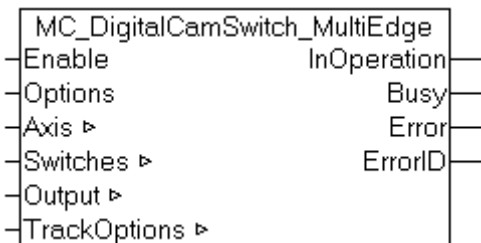
Schaltfolge für positive Fahrrichtung



Schaltfolge für negative Fahrrichtung



4.1.4 MC_DigitalCamSwitch_MultiEdge



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, XFC_EL2258_MultiEdge oder XFC_EL2262_MultiEdge).

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

Eingänge

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

Enable	Über den Eingang <i>Enable</i> wird das Nockenschaltwerk aktiviert. Der Ausgangszustand bleibt unverändert, solange <i>Enable</i> FALSE ist.
Options	Optionale Parameter

Options.	EncoderIndex	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.
Options.	UseAcceleration	UseAcceleration kann TRUE gesetzt werden, um die Beschleunigung der Achse in die Positions-Berechnungen mit einzubeziehen. <i>UseAcceleration</i> kann von Vorteil sein, wenn die Sollwerte der Beschleunigung verwendet werden können. Bei Encoder-Achsen, die ein verrauschtes Positionssignal liefern, ist <i>UseAcceleration</i> eventuell von Nachteil, da auch die Beschleunigung fehlerhaft ist.

Ausgänge

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

InOperation	<i>InOperation</i> ist TRUE, solange das Nockenschaltwerk aktiv ist und die Nockenspur gemäß der Nockenparametrierung berechnet wird.
Busy	<i>Busy</i> ist TRUE solange die Funktion des Bausteins nicht beendet ist.
Error	Wird im Fehlerfall TRUE.
ErrorID	Liefert bei einem gesetzten Error-Ausgang die Fehlernummer

Ein/Ausgänge

```
VAR_IN_OUT
  Axis       : AXIS_REF;
  Switches   : CAMSWITCH_REF [▶ 13];
  Output     : OUTPUT_REF_MULTIEDGE [▶ 19];
  TrackOptions : TRACK_REF [▶ 21];
END_VAR
```

Axis	Achsdatenstruktur
Switches	Die Datenstruktur <i>Switches</i> enthält einen Verweis auf die Parametrierung aller Nocken auf der Nockenspur.
Output	Die Datenstruktur <i>Output</i> enthält den berechneten Zustand des digitalen Ausganges und die zugehörigen Zeitstempel für die Ausgabe an einer digitalen XFC Ausgangsklemme
TrackOptions	Die Datenstruktur <i>TrackOptions</i> enthält die Parametrierung der Nockenspur.

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.

4.1.5 OUTPUT_REF

Der Datentyp `OUTPUT_REF` enthält Daten zur Beschreibung des Zustandes eines digitalen Ausganges. 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 *)
END_STRUCT
```

```

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
    
```

Level	Aktueller Schaltzustand des digitalen Ausgangs
NextStateChangeValid	<i>NextStateChangeValid</i> ist TRUE, wenn der Zeitstempel <i>NextStateChange</i> gültig ist.
NextStateChange	Zeitpunkt des nächsten Zustandswechsels (Distributed Clock TimeStamp)
NextOnTimeValid	<i>NextOnTimeValid</i> ist TRUE, wenn der Zeitstempel <i>NextOnTime</i> gültig ist.
NextOnTime	Zeitpunkt der nächsten positiven Schaltflanke (Distributed Clock TimeStamp)
NextOffTimeValid	<i>NextOffTimeValid</i> ist TRUE, wenn der Zeitstempel <i>NextOffTime</i> gültig ist.
NextOffTime	Zeitpunkt der nächsten negativen Schaltflanke (Distributed Clock TimeStamp)

4.1.6 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_MAXINDEXOFMULTIEDGEOUTPTEVENTS] 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
    
```

ID	Interne Kennung der Schaltflanke
Valid	Valid ist TRUE, wenn der Zeitstempel <i>DcTime</i> gültig ist.
Level	Aktueller Schaltzustand des digitalen Ausgangs
Position	Schaltposition des Schaltvorgangs
DcTime	Zeitpunkt des nächsten Zustandswechsels (Distributed Clock TimeStamp)
Duration	Nicht genutzt

4.1.7 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;
END_STRUCT
    
```

```

OnPrecisionReduced      : BOOL;
OffPrecisionReduced    : BOOL;
LastOutputLevel        : BOOL;
ActivatedOnValues       : INT;
ActivatedOffValues     : INT;
END_STRUCT
END_TYPE
    
```

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

4.1.8 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
    
```

ErrorOnOutputMissed	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 <i>ForceWhenLate</i> gesetzt werden.
ErrorOffOutputMissed	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 <i>ForceWhenLate</i> gesetzt werden.
OnPrecisionReduced	Zeigt an, dass eine steigende Schaltflanke nicht auf einen Zyklus genau bestimmt werden konnte. Die Schaltflanke wurde aber bestmöglich ausgegeben.
OffPrecisionReduced	Zeigt an, dass eine fallende Schaltflanke nicht auf einen Zyklus genau bestimmt werden konnte. Die Schaltflanke wurde aber bestmöglich ausgegeben.

LastOutputLevel	Zeigt an, welchen Signalzustand der Kanal der EL2262 nach dem folgenden Update haben wird.
ActivatedOnValues	Anzahl der in diesem Zyklus aktivierten steigenden Flanken
ActivatedOffValues	Anzahl der in diesem Zyklus aktivierten fallenden Flanken

4.1.9 TRACK_REF

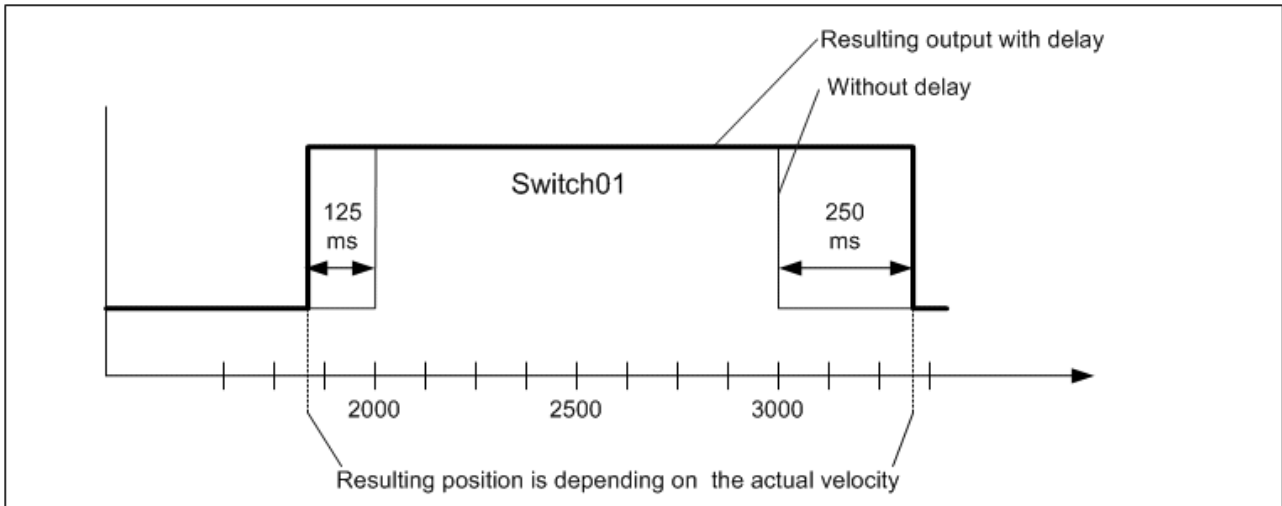
Der Datentyp *TRACK_REF* enthält die Parameter einer digitalen Nockenspur für ein digitales Nockenschaltwerk *MC_DigitalCamSwitch*.

```

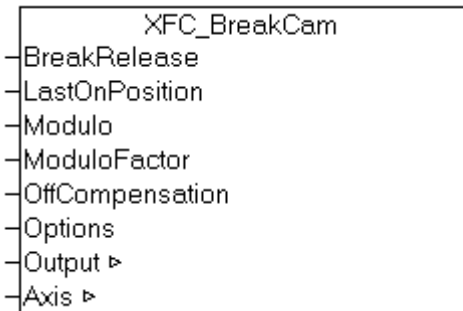
TYPE TRACK_REF :
STRUCT
  ModuloPositions : BOOL := TRUE; (* all cam positions are interpreted as modulo positions when TRUE *)
  ModuloFactor    : LREAL := 360; (* e. g. 360 degrees *)
  OnCompensation  : LREAL;      (* compensation time [s] *)
  OffCompensation : LREAL;      (* compensation time [s] *)
  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
    
```

ModuloPositions	Wenn <i>Modulo</i> TRUE ist, werden alle Positionen modulo interpretiert. Die Nockenfunktion wird dadurch zyklisch wiederholt. Zur Berechnung des Modulo-Zyklus wird der Parameter <i>ModuloFactor</i> verwendet.
ModuloFactor	<i>ModuloFactor</i> gibt die Länge eines Modulo-Zyklus in der Positionseinheit der Achse an und wird nur verwendet, wenn <i>Modulo</i> TRUE ist.
OnCompensation	Kompensationszeit für die steigende Flanke der Nocke in [s]. Der Schaltzeitpunkt wird bei negativen Werten von <i>OnCompensation</i> vorgezogen und sonst verzögert.
OffCompensation	Kompensationszeit für die fallende Flanke der Nocke in [s]. Der Schaltzeitpunkt wird bei negativen Werten von <i>OffCompensation</i> vorgezogen und sonst verzögert.
Hysteresis	nicht implementiert. Hysterese der Schaltvorgänge bei Drehrichtungsumkehr. Die Hysterese wird in der Positionseinheit der Achse angegeben.
BreakRelease	Bremsfreigabe für Bremsnocken auf diese Nockenspur
Force	Der digitale Ausgang wird unabhängig von den Nocken auf dieser Spur aktiv geschaltet. <i>Disable</i> hat Vorrang vor <i>Force</i> .
Disable	Der digitale Ausgang wird unabhängig von den Nocken auf dieser Spur inaktiv geschaltet. <i>Disable</i> hat Vorrang vor <i>Force</i> .

Wirkungsweise der Zeitkompensation



4.1.10 XFC_BreakCam



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 oder XFC_EL2262 [► 29]).

Eingänge

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

BreakRelease	Bremsfreigabe. Solange BreakRelease TRUE ist, bleibt die Nocke aktiv. Nachdem BreakRelease FALSE wird, wird die Nocke an der Position LastOnPosition ausgeschaltet.
LastOnPosition	Letzte Position bis zu der die Nocke eingeschaltet ist.
Modulo	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	ModuloFactor gibt die Länge eines Modulo-Zyklus in der Positionseinheit der Achse an und wird nur verwendet, wenn Modulo=TRUE ist.

OffCompensation	Kompensationszeit für die fallende Flanke der Nocke in [s]. Der Schaltzeitpunkt wird bei negativen Werten von <i>OffCompensation</i> vorgezogen und sonst verzögert. Der hier parametrisierte Wert <i>OffCompensation</i> hat Vorrang vor <i>TRACK_REF</i> [▶ 21].	
Options	Optionale Parameter	
Options.	EncoderIndex	Wenn mehr als ein Encoder mit der Achse verbunden ist, kann der Encoder-Index hier festgelegt werden.
Options.	UseAcceleration	<i>UseAcceleration</i> kann TRUE gesetzt werden, um die Beschleunigung der Achse in die Positions-Berechnungen mit einzubeziehen. <i>UseAcceleration</i> kann von Vorteil sein, wenn die Sollwerte der Beschleunigung verwendet werden können. Bei Encoder-Achsen, die ein verrauschtes Positionssignal liefern, ist <i>UseAcceleration</i> eventuell von Nachteil, da auch die Beschleunigung fehlerhaft ist.

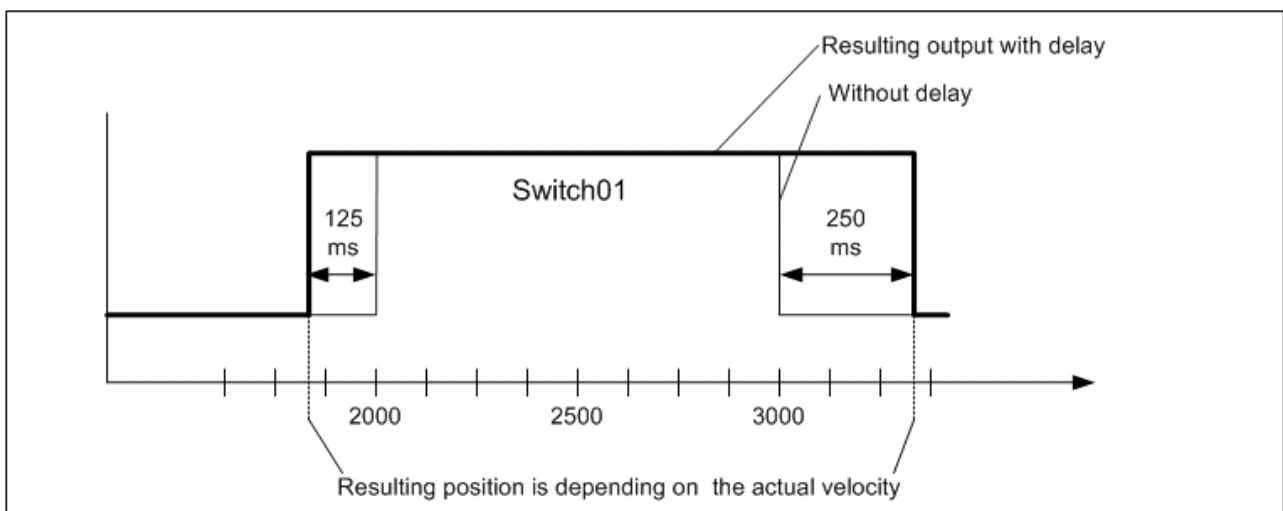
Ein/Ausgänge

```
VAR_IN_OUT
  Output : OUTPUT_REF [▶ 18];
  Axis   : AXIS_REF;
END_VAR
```

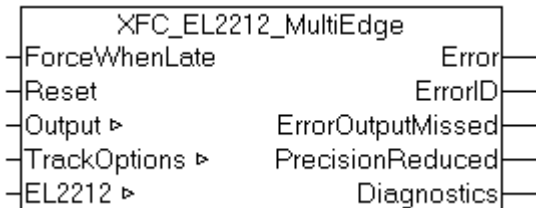
Output	Die Datenstruktur <i>Output</i> enthält den berechneten Zustand des digitalen Ausgangs und die zugehörigen Zeitstempel für die Ausgabe an einer digitalen XFC-Ausgangsklemme.
Axis	Achsdatenstruktur

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.

Wirkungsweise der Zeitkompensation



4.1.11 XFC_EL1259_MultiEdge



XFC_EL1259_MultiEdge wickelt die Ausgabe eines Multi-Edge-Nockenschaltwerkes MC_DigitalCamSwitch_MultiEdge XFC-Zeitstempelklemme EL1259 ab.

Eingänge

```

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

ForceWhenLate	Wenn <i>ForceWhenLate</i> TRUE ist, wird der Ausgang auch dann geschaltet, wenn der Zeitstempel bereits überschritten ist. Es wird empfohlen, <i>ForceWhenLate</i> zu setzen, um den Ausfall einer Schaltflanke bei Schwankungen des Zeitsignals zu vermeiden.
Reset	<i>Reset</i> initiates a reset of the terminal

Ausgänge

```

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

Error	Wird im Fehlerfall TRUE.
ErrorID	Liefert bei einem gesetzten Error-Ausgang eine Fehlernummer
ErrorOutputMissed	Der exakte durch <i>TimeStamp</i> definierte Schaltpunkt konnte nicht eingehalten werden und der Ausgangszustand bleibt unverändert. Falls <i>ForceWhenLate</i> TRUE ist, wird der Ausgangszustand immer ausgegeben und <i>ErrorOutputMissed</i> wird nicht TRUE.
PrecisionReduced	Der exakte durch <i>TimeStamp</i> definierte Schaltpunkt konnte nicht eingehalten werden und der Ausgangszustand wurde aber verspätet ausgegeben. <i>PrecisionReduced</i> ist als Warnung zu verstehen und kann nur TRUE werden, wenn <i>ForceWhenLate</i> TRUE ist.

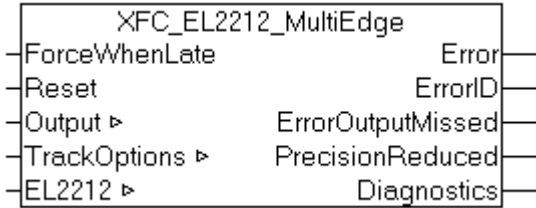
Ein/Ausgänge

```

VAR_IN_OUT
  Output      : OUTPUT_REF_MULTIEDGE;
  TrackOptions : TRACK_REF |> 21;
  EL1259     : EL1259_IoInterface;
END_VAR
  
```

Output	Ausgangszustand für einen Kanal der Klemme. Die Datenstruktur <i>Output</i> enthält die nächsten berechneten Zustände des digitalen Ausgangs und die zugehörigen Zeitstempel für die Ausgabe an einer digitalen XFC Ausgangsklemme.
TrackOptions	Die Datenstruktur <i>TrackOptions</i> enthält die Parametrierung der Nockenspur.
EL1259	Prozessabbild der Klemme

4.1.12 XFC_EL2212_MultiEdge



XFC_EL2212_MultiEdge wickelt die Ausgabe eines Multi-Edge-Nockenschaltwerkes MC_DigitalCamSwitch_MultiEdge XFC-Zeitstempelklemme EL2212 ab.

Eingänge

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

ForceWhenLate	Wenn <i>ForceWhenLate</i> TRUE ist, wird der Ausgang auch dann geschaltet, wenn der Zeitstempel bereits überschritten ist. Es wird empfohlen, <i>ForceWhenLate</i> zu setzen, um den Ausfall einer Schaltflanke bei Schwankungen des Zeitsignals zu vermeiden.
Reset	<i>Reset</i> initiates a reset of the terminal

Ausgänge

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

Error	Wird im Fehlerfall TRUE.
ErrorID	Liefert bei einem gesetzten Error-Ausgang eine Fehlernummer
ErrorOutputMissed	Einer der in der Struktur <i>Output</i> übergebenen Schaltpunkte konnte nicht eingehalten werden und der Ausgangszustand bleibt unverändert. Falls <i>ForceWhenLate</i> TRUE ist, wird der Ausgangszustand immer ausgegeben und <i>ErrorOutputMissed</i> wird nicht TRUE.
PrecisionReduced	Einer der in der Struktur <i>Output</i> übergebenen Schaltpunkte konnte nicht eingehalten werden, der Ausgangszustand wurde aber verspätet ausgegeben. <i>PrecisionReduced</i> ist als Warnung zu verstehen und kann nur TRUE werden, wenn <i>ForceWhenLate</i> TRUE ist.

Ein/Ausgänge

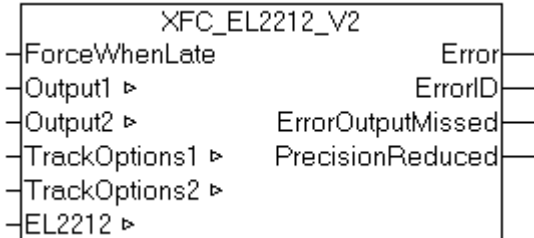
```
VAR_IN_OUT
    Output       : OUTPUT_REF_MULTIEDGE;
    TrackOptions : TRACK_REF [▸ 21];
    EL2212       : EL2212_IoInterface;
END_VAR
```

Output	Ausgangszustand für einen Kanal der Klemme. Die Datenstruktur <i>Output</i> enthält die nächsten berechneten Zustände des digitalen Ausgangs und die zugehörigen Zeitstempel für die Ausgabe an einer digitalen XFC Ausgangsklemme.
TrackOptions	Die Datenstruktur <i>TrackOptions</i> enthält die Parametrierung der Nockenspur.
EL2212	Prozessabbild der Klemme

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.

4.1.13 XFC_EL2212_V2



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
  
```

ForceWhenLate	Wenn <i>ForceWhenLate</i> TRUE ist, wird der Ausgang auch dann geschaltet, wenn der Zeitstempel bereits überschritten ist. Es wird empfohlen, <i>ForceWhenLate</i> zu setzen, um den Ausfall einer Schaltflanke bei Schwankungen des Zeitsignals zu vermeiden.
----------------------	---

Ausgänge

```

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

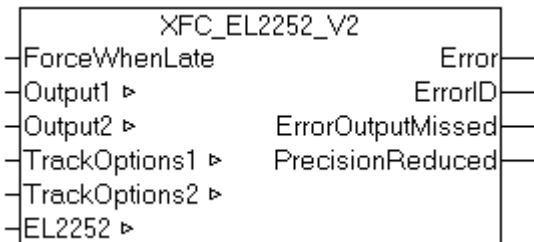
Error	Wird im Fehlerfall TRUE.
ErrorID	Liefert bei einem gesetzten Error-Ausgang eine Fehlernummer
ErrorOutputMissed	Der exakte in den Strukturen <i>Output1</i> oder <i>Output2</i> übergebene Schaltpunkt konnte nicht eingehalten werden und der Ausgangszustand bleibt unverändert. Falls <i>ForceWhenLate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

Ein/Ausgänge

```
VAR_IN_OUT
    Output1      : OUTPUT_REF [▷ 18];
    Output2      : OUTPUT_REF;
    TrackOptions1 : TRACK_REF [▷ 21];
    TrackOptions2 : TRACK_REF;
    EL2212       : EL2212_IoInterface;
END_VAR
```

Output1	Ausgangszustand für den Kanal 1 der Klemme. Die Datenstruktur <i>Output</i> enthält den berechneten Zustand des digitalen Ausgangs und die zugehörigen Zeitstempel für die Ausgabe an einer digitalen XFC-Ausgangsklemme
Output2	Ausgangszustand für den Kanal 2 der Klemme. Die Datenstruktur <i>Output</i> enthält den berechneten Zustand des digitalen Ausgangs und die zugehörigen Zeitstempel für die Ausgabe an einer digitalen XFC-Ausgangsklemme.
TrackOptions1	Die Datenstruktur <i>TrackOptions1</i> enthält die Parametrierung der Nockenspur 1 für Output1.
TrackOptions2	Die Datenstruktur <i>TrackOptions2</i> enthält die Parametrierung der Nockenspur 2 für Output2.
EL2212	Prozessabbild der Klemme

4.1.14 XFC_EL2252_V2



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
```

ForceWhenLate	Wenn <i>ForceWhenLate</i> TRUE ist, so wird der Ausgang auch dann geschaltet, wenn der Zeitstempel bereits überschritten ist. Es wird empfohlen, <i>ForceWhenLate</i> zu setzen, um den Ausfall einer Schaltflanke bei Schwankungen des Zeitsignals zu vermeiden.
----------------------	---

Ausgänge

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

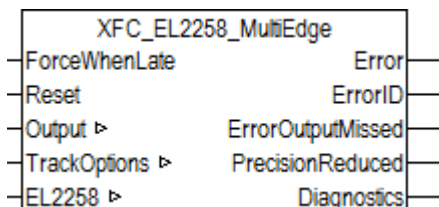
Error	Wird im Fehlerfall TRUE.
ErrorID	Liefert bei einem gesetzten Error-Ausgang eine Fehlernummer
ErrorOutputMissed	Der exakte in den Strukturen <i>Output1</i> oder <i>Output2</i> übergebene Schalterpunkt konnte nicht eingehalten werden und der Ausgangszustand bleibt unverändert. Falls <i>ForceWhenLate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

Ein/Ausgänge

```
VAR_IN_OUT
  Output1      : OUTPUT_REF [▷ 18];
  Output2      : OUTPUT_REF;
  TrackOptions1 : TRACK_REF [▷ 21];
  TrackOptions2 : TRACK_REF;
  EL2252       : EL2252_IoInterface;
END_VAR
```

Output1	Ausgangszustand für den Kanal 1 der Klemme. Die Datenstruktur <i>Output</i> enthält den berechneten Zustand des digitalen Ausgangs und die zugehörigen Zeitstempel für die Ausgabe an einer digitalen XFC-Ausgangsklemme.
Output2	Ausgangszustand für den Kanal 2 der Klemme. Die Datenstruktur <i>Output</i> enthält den berechneten Zustand des digitalen Ausgangs und die zugehörigen Zeitstempel für die Ausgabe an einer digitalen XFC-Ausgangsklemme.
TrackOptions1	Die Datenstruktur <i>TrackOptions1</i> enthält die Parametrierung der Nockenspur 1 für Output1.
TrackOptions2	Die Datenstruktur <i>TrackOptions2</i> enthält die Parametrierung der Nockenspur 2 für Output2.
EL2252	Prozessabbild der Klemme

4.1.15 XFC_EL2258_Multiedge



XFC_EL2258_Multiedge wickelt die Ausgabe eines Multi-Edge-Nockenschaltwerkes MC_DigitalCamSwitch_MultiEdge XFC-Zeitstempelklemme EL2258 ab.

Eingänge

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

ForceWhenLate	Wenn <i>ForceWhenLate</i> TRUE ist, wird der Ausgang auch dann geschaltet, wenn der Zeitstempel bereits überschritten ist. Es wird empfohlen, <i>ForceWhenLate</i> zu setzen, um den Ausfall einer Schaltflanke bei Schwankungen des Zeitsignals zu vermeiden.
Reset	<i>Reset</i> initiates a reset of the terminal

Ausgänge

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

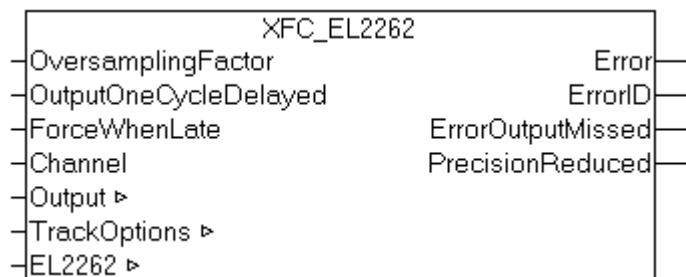
Error	Wird im Fehlerfall TRUE.
ErrorID	Liefert bei einem gesetzten Error-Ausgang eine Fehlernummer
ErrorOutputMissed	Einer der in der Struktur <i>Output</i> übergebenen Schaltepunkte konnte nicht eingehalten werden und der Ausgangszustand bleibt unverändert. Falls <i>ForceWhenLate</i> TRUE ist, wird der Ausgangszustand immer ausgegeben und <i>ErrorOutputMissed</i> wird nicht TRUE.
PrecisionReduced	Einer der in der Struktur <i>Output</i> übergebenen Schaltepunkte konnte nicht eingehalten werden, der Ausgangszustand wurde aber verspätet ausgegeben. <i>PrecisionReduced</i> ist als Warnung zu verstehen und kann nur TRUE werden, wenn <i>ForceWhenLate</i> TRUE ist.
Diagnostics	Datenstruktur [▶ 19] , die Diagnosedaten enthält, die zur Fehleranalyse heran gezogen werden können.

Ein/Ausgänge

```
VAR_IN_OUT
    Output           : OUTPUT_REF_MULTIEDGE;
    TrackOptions     : TRACK_REF [▶ 21];
    EL2258           : EL2258_IoInterface;
END_VAR
```

Output	Ausgangszustand für einen Kanal der Klemme. Die Datenstruktur <i>Output</i> enthält die nächsten berechneten Zustände des digitalen Ausgangs und die zugehörigen Zeitstempel für die Ausgabe an einer digitalen XFC Ausgangsklemme.
TrackOptions	Die Datenstruktur <i>TrackOptions</i> enthält die Parametrierung der Nockenspur.
EL2258	Prozessabbild der Klemme

4.1.16 XFC_EL2262



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 steigenden und fallenden Flanke kann kleiner als ein SPS-Zyklus sein. Die Schaltgenauigkeit 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
```

OversamplingFactor	Oversampling Faktor der EL2262 Klemme
OutputOneCycleDelayed	<i>OutputOneCycleDelayed</i> ist TRUE, falls die Ausgabe des Prozessabbildes aufgrund des eingestellten Timings um einen Zyklus verzögert ist. <i>OutputOneCycleDelayed</i> ist abhängig vom Timing der Ausgabetausk, an welche die EL2262 gebunden ist.
ForceWhenLate	Wenn sich die Zeitinformation von Zyklus zu Zyklus leicht ändert, kann es passieren, dass eine Schaltflanke nicht ausgegeben werden kann. <i>ForceWhenLate</i> erzwingt in einer solchen Situation ein bestmögliches Schalten. Der Ausgang <i>PrecisionReduced</i> 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	Kanalnummer 0 oder 1 der EL2262 Klemme

Ausgänge

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

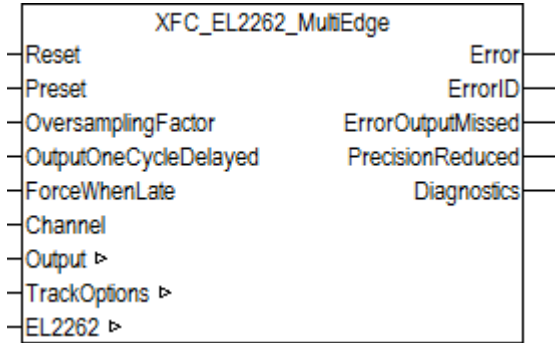
Error	Wird im Fehlerfall TRUE.
ErrorID	Liefert bei einem gesetzten Error-Ausgang eine Fehlernummer
ErrorOutputMissed	zeigt an, dass eine Schaltflanke nicht auf einen Zyklus genau bestimmt werden kann und daher nicht ausgegeben werden kann. Um die Schaltflanke bestmöglich auszugeben kann der Eingang <i>ForceWhenLate</i> gesetzt werden.
PrecisionReduced	zeigt an, dass eine Schaltflanke nicht auf einen Zyklus genau bestimmt werden kann. Die Schaltflanke wurde aber bestmöglich ausgegeben.

Ein/Ausgänge

```
VAR_IN_OUT
    Output           : OUTPUT_REF [▶ 18];
    TrackOptions     : TRACK_REF [▶ 21];
    EL2262           : EL2262_IoInterface;
END_VAR
```

Output	Die Datenstruktur <i>Output</i> enthält den berechneten Zustand des digitalen Ausgangs und die zugehörigen Zeitstempel für die Ausgabe an einer digitalen XFC Ausgangsklemme
TrackOptions	Die Datenstruktur <i>TrackOptions</i> enthält die Parametrierung der Nockenspur.
EL2262	Prozessabbild der Klemme

4.1.17 XFC_EL2262_MultiEdge



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 Signalflanken beträgt $(2 \cdot \text{SPS-Zykluszeit}) / \text{Oversampling-Faktor}$. Der minimale Abstand zwischen 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
    
```

Reset	Der Klemmenausgang wird deaktiviert.
Preset	Der Klemmenausgang wird aktiviert.
OversamplingFactor	Oversampling Faktor der EL2262 Klemme
OutputOneCycleDelayed	<i>OutputOneCycleDelayed</i> ist TRUE, falls die Ausgabe des Prozessabbildes aufgrund des eingestellten Timings um einen Zyklus verzögert ist. <i>OutputOneCycleDelayed</i> ist abhängig vom Timing der Ausgabetausk, an welche die EL2262 gebunden ist.
ForceWhenLate	Wenn sich die Zeitinformation von Zyklus zu Zyklus leicht ändert, kann es passieren, dass eine Schaltflanke nicht ausgegeben werden kann. <i>ForceWhenLate</i> erzwingt in einer solchen Situation ein bestmögliches Schalten. Der Ausgang <i>PrecisionReduced</i> 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	Legt den Ausgabekanal der EL2262 fest, wobei 0 = Output0 und 1 = Output1 entspricht.

Ausgänge

```

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

Error	Wird im Fehlerfall TRUE.
ErrorID	Liefert bei einem gesetzten Error-Ausgang eine Fehlernummer

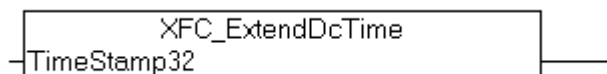
ErrorOutputMissed	Einer der in der Struktur <i>Output</i> übergebenen Schaltpunkte konnte nicht eingehalten werden und der Ausgangszustand bleibt unverändert. Falls <i>ForceWhenLate</i> TRUE ist, wird der Ausgangszustand immer ausgegeben und <i>ErrorOutputMissed</i> wird nicht TRUE.
PrecisionReduced	Einer der in der Struktur <i>Output</i> übergebenen Schaltpunkte konnte nicht eingehalten werden, der Ausgangszustand wurde aber verspätet ausgegeben. <i>PrecisionReduced</i> ist als Warnung zu verstehen und kann nur TRUE werden, wenn <i>ForceWhenLate</i> TRUE ist.
Diagnostics	Datenstruktur [▶ 20] die Diagnosedaten enthält die zur Fehleranalyse heran gezogen werden können

Ein/Ausgänge

```
VAR_IN_OUT
    Output      : OUTPUT_REF_MULTIEDGE;
    TrackOptions : TRACK_REF [▶ 21];
    EL2262      : EL2262_IoInterface;
END_VAR
```

Output	Die Datenstruktur [▶ 19] <i>Output</i> enthält ein Array von berechneten Zuständen des digitalen Ausgangs und die zugehörigen Zeitstempel für die Ausgabe an einer digitalen XFC Ausgangsklemme
TrackOptions	Die Datenstruktur <i>TrackOptions</i> enthält die Parametrierung der Nockenspur.
EL2262	Prozessabbild der Klemme

4.1.18 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
```

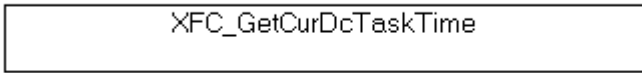
TimeStamp32	Distributed Clock System Time. <i>TimeStamp32</i> enthält die unteren 32 Bit der vollständigen <i>DcTime</i> und deckt damit einen Zeitbereich von +/- 2 Sekunden um den aktuellen Zeitpunkt ab.
--------------------	---

Rückgabewert

```
FUNCTION XFCF_ExtendDcTime : T_DCTIME
```

Rückgabewert der Funktion	Vollständige 64 Bit <i>Distributed Clock System Time</i> .
---------------------------	--

4.1.19 XFC_GetCurDcTaskTime



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

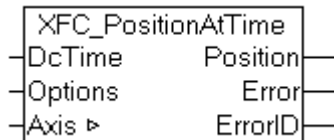
Die Funktion optimiert die Aufrufe der Systemfunktion *F_GetCurDcTaskTime*, indem mehrere Anfragen innerhalb eines SPS-Task-Zyklus mit derselben Zeit beantwortet werden, ohne die Systemfunktion jeweils erneut aufzurufen.

Rückgabewert

```
FUNCTION XFCF_GetCurDcTaskTime : T_DCTIME
```

Rückgabewert der Funktion	Vollständige 64 Bit <i>Distributed Clock System Time</i> . Zeitpunkt des Anfangs des aktuellen SPS-Zyklus (Zyklus der Task, die diese Funktion aufruft.)
---------------------------	--

4.1.20 XFC_PositionAtTime



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
```

DcTime	Distributed Clock System Time. <i>DcTime</i> enthält die unteren 32 Bit der vollständigen <i>DcTime</i> 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	Datenstruktur mit Optionen für die Extrapolation der Position.	
Options.	SubIndex	Für Achsen mit mehr als einem Encoder kann in <i>SubIndex</i> der Index (0..9) des Encoders angegeben werden, auf den sich die Position bezieht.
Options.	InterpolationOptions	0: Die Extrapolation der Position wird mit der aktuellen Geschwindigkeit durchgeführt, ohne die aktuelle Beschleunigung zu berücksichtigen.

		1: Die Beschleunigung der Achse wird in die Extrapolation der Position einbezogen.
Options.	CompensationTime	zusätzliche Kompensationszeit.

Ausgänge

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

Position	Extrapolierte Position, die zum vorgegebenen Zeitpunkt <i>DcTime</i> erreicht sein wird bzw. erreicht wurde.
Error	Wird im Fehlerfall TRUE.
ErrorID	Liefert bei einem gesetzten Error-Ausgang die Fehlernummer

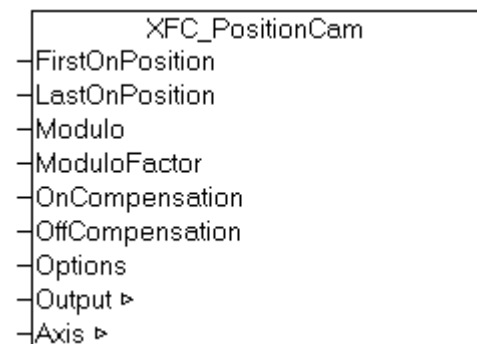
Ein/Ausgänge

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

Axis	Achsdatenstruktur
-------------	-------------------

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.

4.1.21 XFC_PositionCam



XFC_PositionCam realisiert eine Positionsnocke, die einen digitalen Ausgang positionsabhängig ein- und ausschaltet.

Im Gegensatz zum digitalen Nockenschaltwerk [MC_DigitalCamSwitch \[▶ 14\]](#) 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](#) oder [XFC_EL2262 \[▶ 29\]](#)).

Eingänge

```
VAR_INPUT
    FirstOnPosition : LREAL;
    LastOnPosition  : LREAL;
    Modulo           : BOOL;
    ModuloFactor     : LREAL := 360;
    OnCompensation  : LREAL;
```

```

OffCompensation : LREAL;
Options         : ST_CamSwitchOptions;
END_VAR
    
```

FirstOnPosition	Erste Position ab der die Nocke eingeschaltet ist.	
LastOnPosition	Letzte Position bis zu der die Nocke eingeschaltet ist. Die Nockenfunktion ist invertiert, wenn <i>LastOnPosition</i> < <i>FirstOnPosition</i>	
Modulo	Wenn <i>Modulo</i> TRUE ist, werden alle Positionen modulo interpretiert. Die Nockenfunktion wird dadurch zyklisch wiederholt. Zur Berechnung des Modulo-Zyklus wird der Parameter <i>ModuloFactor</i> verwendet.	
ModuloFactor	ModuloFactor gibt die Länge eines Modulo-Zyklus in der Positionseinheit der Achse an und wird nur verwendet, wenn <i>Modulo</i> TRUE ist.	
OnCompensation	Kompensationszeit für die steigende Flanke der Nocke in [s]. Der Schaltzeitpunkt wird bei negativen Werten von <i>OnCompensation</i> vorgezogen und sonst verzögert. Der hier parametrisierte Wert OnCompensation hat Vorrang vor TRACK_REF [▶ 21].	
OffCompensation	Kompensationszeit für die fallende Flanke der Nocke in [s]. Der Schaltzeitpunkt wird bei negativen Werten von <i>OffCompensation</i> vorgezogen und sonst verzögert. Der hier parametrisierte Wert OffCompensation hat Vorrang vor TRACK_REF [▶ 21].	
Options	Optionale Parameter	
Options.	EncoderIndex	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.
Options.	UseAcceleration	<i>UseAcceleration</i> kann TRUE gesetzt werden, um die Beschleunigung der Achse in die Positions-Berechnungen mit einzubeziehen. <i>UseAcceleration</i> kann von Vorteil sein, wenn die Sollwerte der Beschleunigung verwendet werden können. Bei Encoder-Achsen, die ein verrauschtes Positionssignal liefern, ist <i>UseAcceleration</i> eventuell von Nachteil, da auch die Beschleunigung fehlerhaft ist.

Ein/Ausgänge

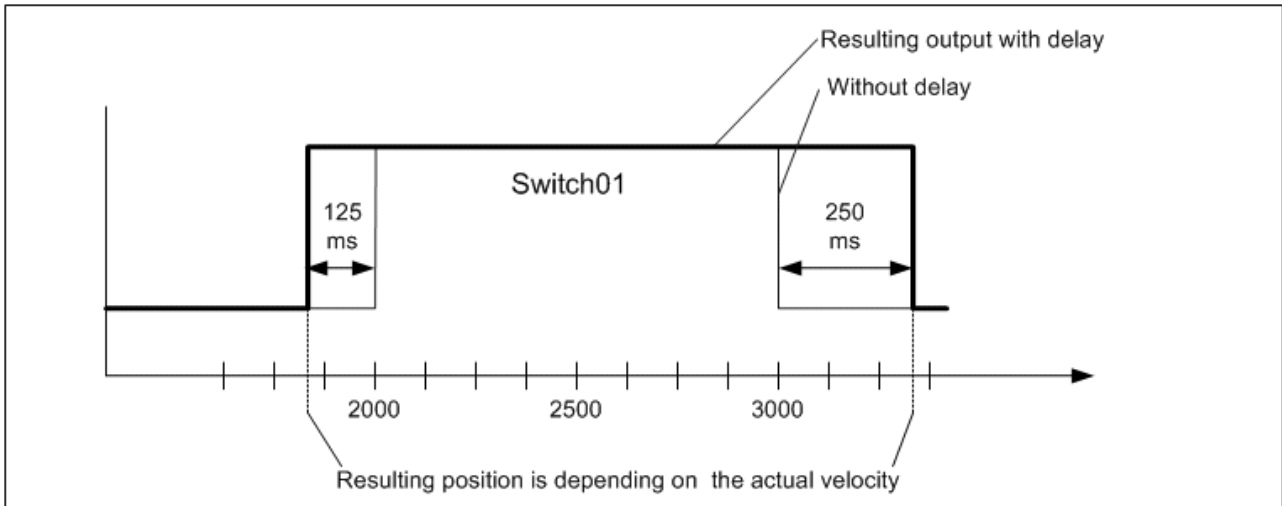
```

VAR_IN_OUT
    Output : OUTPUT_REF [▶ 18];
    Axis   : AXIS_REF;
END_VAR
    
```

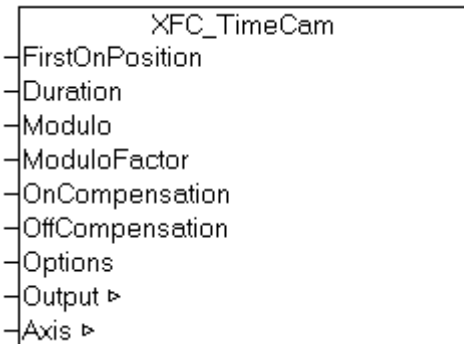
Output	Die Datenstruktur <i>Output</i> enthält den berechneten Zustand des digitalen Ausgangs und die zugehörigen Zeitstempel für die Ausgabe an einer digitalen XFC Ausgangsklemme
Axis	Achsdatenstruktur

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.

Wirkungsweise der Zeitkompensation



4.1.22 XFC_TimeCam



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

Im Gegensatz zum digitalen Nockenschaltwerk [MC_DigitalCamSwitch \[▶ 14\]](#) 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](#) oder [XFC_EL2262 \[▶ 29\]](#)).

Eingänge

```

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

FirstOnPosition	Erste Position, ab der die Nocke eingeschaltet ist.
Duration	Einschaltdauer der Nocke in [s].
Modulo	Wenn <i>Modulo</i> TRUE ist, werden alle Positionen modulo interpretiert. Die Nockenfunktion wird dadurch zyklisch wiederholt. Zur Berechnung des Modulo-Zyklus wird der Parameter <i>ModuloFactor</i> verwendet.

ModuloFactor	<i>ModuloFactor</i> gibt die Länge eines Modulo-Zyklus in der Positionseinheit der Achse an und wird nur verwendet, wenn <i>Modulo</i> TRUE ist.	
OnCompensation	Kompensationszeit für die steigende Flanke der Nocke in [s]. Der Schaltzeitpunkt wird bei negativen Werten von <i>OnCompensation</i> vorgezogen und sonst verzögert. Der hier parametrisierte Wert <i>OnCompensation</i> hat Vorrang vor <u>TRACK_REF</u> [▶ 21].	
OffCompensation	Kompensationszeit für die fallende Flanke der Nocke in [s]. Der Schaltzeitpunkt wird bei negativen Werten von <i>OffCompensation</i> vorgezogen und sonst verzögert. Der hier parametrisierte Wert <i>OffCompensation</i> hat Vorrang vor <u>TRACK_REF</u> [▶ 21].	
Options	Optionale Parameter	
Options.	EncoderIndex	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.
Options.	UseAcceleration	<i>UseAcceleration</i> kann TRUE gesetzt werden, um die Beschleunigung der Achse in die Positions-Berechnungen mit einzubeziehen. <i>UseAcceleration</i> kann von Vorteil sein, wenn die Sollwerte der Beschleunigung verwendet werden können. Bei Encoder-Achsen, die ein verrauschtes Positionssignal liefern, ist <i>UseAcceleration</i> eventuell von Nachteil, da auch die Beschleunigung fehlerhaft ist.

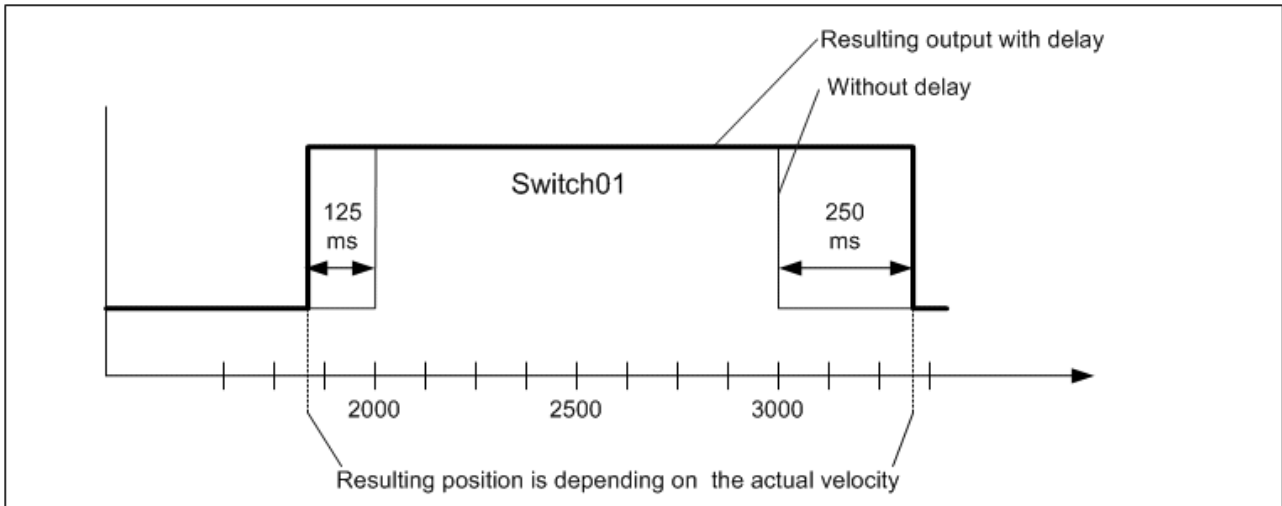
Ein/Ausgänge

```
VAR_IN_OUT
    Output : OUTPUT_REF [▶ 18];
    Axis   : AXIS_REF;
END_VAR
```

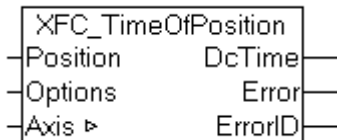
Output	Die Datenstruktur <i>Output</i> enthält den berechneten Zustand des digitalen Ausgangs und die zugehörigen Zeitstempel für die Ausgabe an einer digitalen XFC-Ausgangsklemme
Axis	Achsdatenstruktur

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.

Wirkungsweise der Zeitkompensation



4.1.23 XFC_TimeOfPosition



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

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 Zeit *DcTime* gültig ist.

Eingänge

```

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

Position	Absolute Achsposition	
Options	Datenstruktur mit Optionen für die Extrapolation der Position.	
Options.	SubIndex	Für Achsen mit mehr als einem Encoder kann in <i>SubIndex</i> der Index (0..9) des Encoders angegeben werden, auf den sich die Position bezieht.
Options.	InterpolationOptions	0: Die Extrapolation der Position wird mit der aktuellen Geschwindigkeit durchgeführt, ohne die aktuelle Beschleunigung zu berücksichtigen. 1: Die Beschleunigung der Achse wird in die Extrapolation der Position einbezogen.
Options.	CompensationTime	zusätzliche Kompensationszeit

Ausgänge

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

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

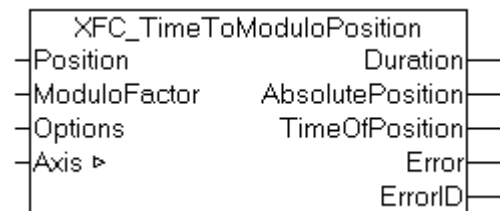
Ein/Ausgänge

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

Axis	Achsdatenstruktur
-------------	-------------------

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.

4.1.24 XFC_TimeToModuloPosition



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
```

Position	Absolute Achsposition
ModuloFactor	Modulo-Teiler, der für die Berechnung verwendet werden soll. <i>ModuloFactor</i> kann mit dem Modulo-Faktor der Achse identisch sein, z. B. 360. Es kann aber auch ein davon abweichender Faktor verwendet werden.
Options	Datenstruktur mit Optionen für die Extrapolation der Position.

Options.	SubIndex	Für Achsen mit mehr als einem Encoder kann in <i>SubIndex</i> der Index (0..9) des Encoders angegeben werden auf den sich die Position bezieht.
Options.	InterpolationOptions	0: Die Extrapolation der Position wird mit der aktuellen Geschwindigkeit durchgeführt, ohne die aktuelle Beschleunigung zu berücksichtigen. 1: Die Beschleunigung der Achse wird in die Extrapolation der Position einbezogen.
Options.	CompensationTime	zusätzliche Kompensationszeit

Ausgänge

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

Duration	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	Absolute Position (nicht modulo), die mit der Modulo-Position und der ermittelten Zeit korrespondiert.
TimeOfPosition	Distributed Clock System Time, zu der die <i>Position</i> erreicht sein wird bzw. zu der sie erreicht wurde. <i>DcTime</i> enthält die unteren 32 Bit der vollständigen <i>DcTime</i> und deckt damit einen Zeitbereich von +/- 2 Sekunden um den aktuellen Zeitpunkt ab.
Error	Wird im Fehlerfall TRUE.
ErrorID	Liefert bei einem gesetzten Error-Ausgang die Fehlernummer

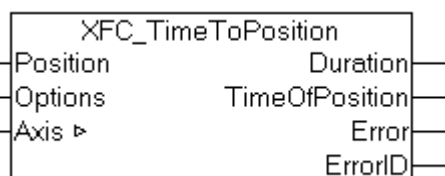
Ein/Ausgänge

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

Axis	Achsdatenstruktur
-------------	-------------------

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.

4.1.25 XFC_TimeToPosition



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.

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;
    Options       : ST_NcTimeConversionOptions;
END_VAR
```

Position	Absolute Achsposition	
Options	Datenstruktur mit Optionen für die Extrapolation der Position.	
Options.	SubIndex	Für Achsen mit mehr als einem Encoder kann in <i>SubIndex</i> der Index (0..9) des Encoders angegeben werden, auf den sich die Position bezieht.
Options.	InterpolationOptions	0: Die Extrapolation der Position wird mit der aktuellen Geschwindigkeit durchgeführt, ohne die aktuelle Beschleunigung zu berücksichtigen. 1: Die Beschleunigung der Achse wird in die Extrapolation der Position einbezogen.
Options.	CompensationTime	zusätzliche Kompensationszeit

Ausgänge

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

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

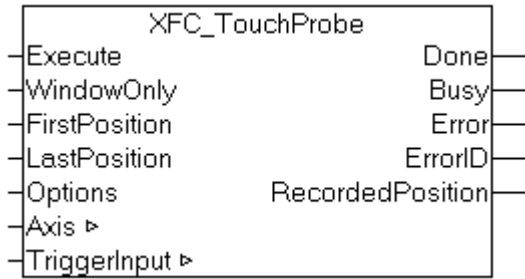
Ein/Ausgänge

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

Axis	Achsdatenstruktur
-------------	-------------------

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.

4.1.26 XFC_TouchProbe



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

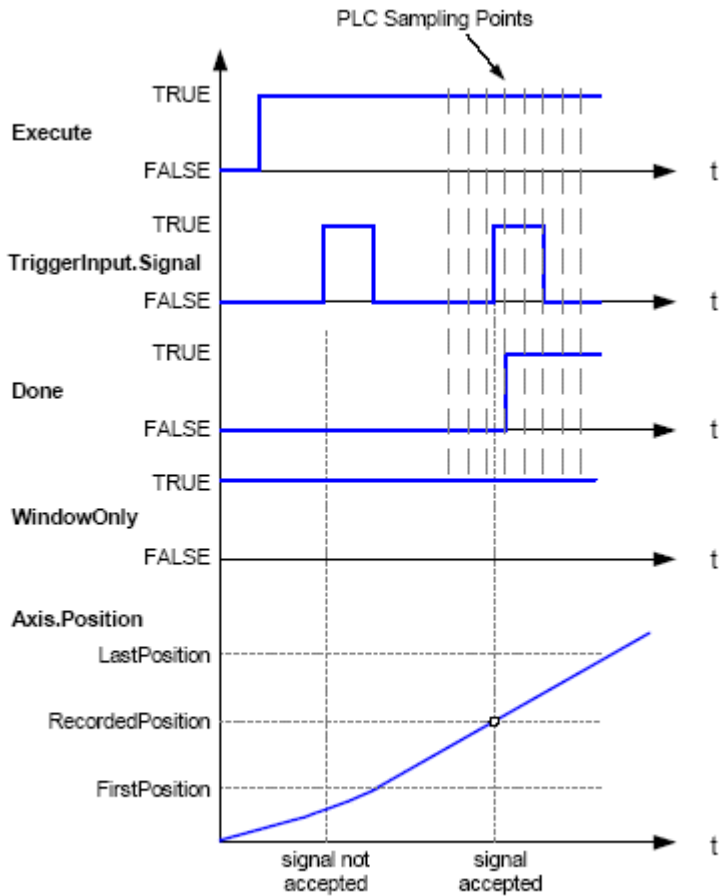
Das digitale Eingangssignal wird mit einer XFC-Eingangsklemme (z. B. EL1252) mit Zeitstempeln für die fallende und steigende Signalflanke erfasst. Der Funktionsbaustein ermittelt die Achsposition, bei der der Flankenwechsel stattgefunden hat und gibt sie als *RecordedPosition* aus.

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 [▸ 8] 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.

Signalverlauf



Timing example TouchProbe

Eingänge

```

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

<p>Execute</p>	<p>Wenn <i>Execute</i> aktiv ist, wird die Achsposition an der definierten Signalfanke des Eingangssignals erfasst. Eine fallende Flanke an <i>Execute</i> beendet den Vorgang sofort. Abhängig von der Konfiguration in <i>TriggerInput.FreeRun</i> wird einmalig die nächste Signalfanke erfasst und ausgewertet. Wenn <i>FreeRun</i> TRUE ist, wird fortlaufend mit jeder definierten Flanke des Eingangssignals ein neuer Positionswert erfasst, während <i>Execute</i> TRUE bleibt.</p>
<p>WindowOnly</p>	<p>Wenn diese Option aktiv ist, wird nur eine Position innerhalb des Fensters zwischen <i>FirstPosition</i> und <i>LastPosition</i> erfasst. Positionen außerhalb des Fensters werden verworfen. Erst wenn die erfasste Position innerhalb des Fensters liegt, wird <i>Done</i> TRUE. Das Erfassungsfenster kann absolut oder modulo interpretiert werden. Dazu ist das Flag <i>ModuloPositions</i> [▶ 45] in der Struktur <i>TriggerInput</i> [▶ 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).</p>

FirstPosition	Anfangsposition des Erfassungsfensters, wenn <i>WindowOnly</i> TRUE ist. Diese Position kann absolut oder modulo interpretiert werden. Dazu ist in der Struktur <i>TriggerInput</i> (siehe unten) das Flag <u>ModuloPositions</u> [▶ 45] entsprechend zu setzen.	
LastPosition	Endposition des Erfassungsfensters, wenn <i>WindowOnly</i> TRUE ist. Diese Position kann absolut oder modulo interpretiert werden. Dazu ist in der Struktur <i>TriggerInput</i> (siehe unten) das Flag <u>ModuloPositions</u> [▶ 45] entsprechend zu setzen.	
Options	Optionale Parameter	
Options.	UseAcceleration	<i>UseAcceleration</i> kann TRUE gesetzt werden, um die Beschleunigung der Achse in die Positions-Berechnungen mit einzubeziehen. <i>UseAcceleration</i> kann von Vorteil sein, wenn die Sollwerte der Beschleunigung verwendet werden können. Bei Encoder-Achsen, die ein verrauschtes Positionssignal liefern, ist <i>UseAcceleration</i> eventuell von Nachteil, da auch die Beschleunigung fehlerhaft ist.

Ausgänge

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

Done	Der Wert <i>RecordedPosition</i> ist gültig. Falls <i>TriggerInput.FreeRun</i> TRUE ist, bleibt <i>Done</i> nur für einen SPS-Zyklus TRUE und wird selbstständig zurückgesetzt, da <i>TouchProbe</i> automatisch neu aktiviert wird.
Busy	Wird TRUE sobald der Baustein aktiv ist und wird FALSE nachdem er sich wieder im Grundzustand befindet. Falls <i>TriggerInput.FreeRun</i> TRUE ist, bleibt <i>Busy</i> permanent TRUE, auch wenn <i>Done</i> oder <i>Error</i> TRUE werden, da <i>TouchProbe</i> automatisch neu aktiviert wird.
Error	Wird TRUE, sobald ein Fehler
ErrorID	Liefert bei einem gesetzten Error-Ausgang die Fehlernummer
RecordedPosition	Erfasste Achsposition zum Zeitpunkt des Trigger-Signals. Wenn <i>TriggerInput.FreeRun</i> TRUE ist, läuft der Funktionsbaustein freilaufend, so dass jede gültige Änderung des Eingangssignals zu einer neuen <i>RecordedPosition</i> führt. Die Position kann jeweils ausgewertet werden, wenn <i>Done</i> TRUE wird.

Ein/Ausgänge

```
VAR_IN_OUT
    Axis           : AXIS_REF;
    TriggerInput   : XFC_TRIGGER_REF [▶ 45];
END_VAR
```

Axis	Achsdatenstruktur
TriggerInput	<i>TriggerInput</i> [▶ 45] ist eine Datenstruktur zur Beschreibung der Trigger-Quelle und zur Einspeisung von Zustand und Zeitstempel eines digitalen Eingangssignals. Diese Datenstruktur wird vom Anwender gefüllt.

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.

4.1.27 XFC_TRIGGER_REF

Der Datentyp *XFC_TRIGGER_REF* enthält Status und Parameter eines digitalen Eingangs, der für die Funktion *XFC_TouchProbe* [► 42] 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
    
```

Signal	Aktueller Zustand des digitalen Eingangssignals. Der aktuelle Zustand muss an dieser Stelle eingespeist werden.
TimestampRisingEdge	Zeitstempel der letzten steigenden Flanke des digitalen Eingangssignals. Es ist nur notwendig, den Zeitstempel der über <i>Edge</i> definierten Flanke einzuspeisen. Falls das Eingangssignal einen 64 Bit Zeitstempel <i>T_DCTIME</i> 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	Zeitstempel der letzten fallenden Flanke des digitalen Eingangssignals. Es ist nur notwendig, den Zeitstempel der über <i>Edge</i> definierten Flanke einzuspeisen. Falls das Eingangssignal einen 64 Bit Zeitstempel <i>T_DCTIME</i> 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	<i>Edge</i> definiert die Signalfanke, die zur Auswertung der Achsposition verwendet werden soll. TYPE E_SignalEdge : (RisingEdge, FallingEdge); END_TYPE
FreeRun	Wenn <i>FreeRun</i> TRUE ist, wird der Eingang fortlaufend gelatcht. Beim Funktionsbaustein <i>XFC_TouchProbe</i> [► 42] muss dann der Eingang <i>Execute</i> TRUE bleiben. Es ist also keine Flanke an <i>Execute</i> notwendig, um den nächsten neuen Positionswert aufzuzeichnen.
EncoderIndex	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	Wenn <i>Modulo</i> TRUE ist, werden alle Positionen modulo interpretiert. Zur Berechnung des Modulo-Zyklus wird der Parameter <i>ModuloFactor</i> verwendet.
ModuloFactor	<i>ModuloFactor</i> gibt die Länge eines Modulo-Zyklus in der Positionseinheit der Achse an und wird nur verwendet, wenn <i>Modulo</i> TRUE ist.

4.2 TcNci_XFC

4.2.1 MC_PathCamSwitch

Der Datentyp *MC_PathCamSwitch* enthält alle Parameter einer digitalen Nocke für ein digitales Nockenschaltwerk *MC_PathDigitalCamSwitch_MultiEdge*.

```

TYPE MC_PathCamSwitch :
STRUCT
    FirstOnPosition : LREAL;
    LastOnPosition  : LREAL;
    FirstPathId     : UDINT;
    
```

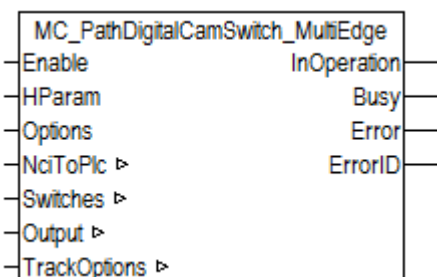
```
LastPathId      : UDINT;
CamSwitchMode   : E_CamSwitchMode;
Duration        : LREAL;
END_STRUCT
END_TYPE
```

Die Datenstruktur zur Parametrierung eines digitalen Nockenschaltwerks ist üblicherweise ein ARRAY OF MC_PathCamSwitch. Eine weitere Struktur [PATH CAMSWITCH REF \[► 47\]](#) verweist auf diese Struktur.

FirstOnPosition	Erste Position, ab der die Nocke eingeschaltet ist.
LastOnPosition	Letzte Position, bis zu der die Nocke eingeschaltet ist. Die Nockenfunktion ist invertiert, wenn <i>LastOnPosition</i> < <i>FirstOnPosition</i> . <i>LastOnPosition</i> wird bei Zeitnocken nicht verwendet.
FirstPathId	Eindeutige, kontinuierlich über den Bahnverlauf steigende Id die zum relativen Bahnweg bis zum Einschalten der Nocke gehört.
LastPathId	Eindeutige, kontinuierlich über den Bahnverlauf steigende Id die zum relativen Bahnweg bis zum Ausschalten der Nocke gehört.
CamSwitchMode	Typ der digitalen Nocke (Positionsnocke, Zeitnocke oder Bremsnocke).
Duration	<i>Duration</i> 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 MC_PathDigitalCamSwitch_MultiEdge



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 der Schaltzustände 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*, *XFC_EL2262_MultiEdge* oder *XFC_EL2258_MultiEdge*).

Hinweis 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
```

Enable	Über den Eingang <i>Enable</i> wird das Nockenschaltwerk aktiviert. Der Ausgangszustand bleibt unverändert, solange <i>Enable=FALSE</i> ist.	
HParam	H-Parameterwert, der mit dem Schaltzustand TRUE korrespondiert.	
Options	Optionale Parameter	
Options.	EncoderIndex	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.
Options.	UseAcceleration	UseAcceleration kann TRUE gesetzt werden, um die Beschleunigung der Achse in die Positions-Berechnungen mit einzubeziehen. <i>UseAcceleration</i> kann von Vorteil sein, wenn die Sollwerte der Beschleunigung verwendet werden können. Bei Encoder-Achsen, die ein verrauschtes Positionssignal liefern, ist <i>UseAcceleration</i> eventuell von Nachteil, da auch die Beschleunigung fehlerhaft ist.

Ausgänge

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

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

Ein/Ausgänge

```
VAR_IN_OUT
  NciToPlc      : NciChannelToPlc
  Switches      : PATH_CAMSWITCH_REF;
  Output        : OUTPUT_REF_MULTIEDGE [▶ 19];
  TrackOptions  : TRACK_REF [▶ 21];
END_VAR
```

NciToPlc	Struktur des zyklischen Kanalinterfaces von der NCI zur SPS. Auf diese Struktur wird nur lesend zugegriffen.
Switches	Die Datenstruktur <i>Switches</i> enthält einen Verweis auf die Parametrierung aller Nocken auf der Nockenspur.
Output	Die Datenstruktur <i>Output</i> enthält die berechneten Zustände des digitalen Ausgangs und die zugehörigen Zeitstempel für die Ausgabe an einer digitalen XFC Ausgangsklemme.
TrackOptions	Die Datenstruktur <i>TrackOptions</i> enthält die Parametrierung der Nockenspur.

4.2.3 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
    
```

Die eigentliche Datenstruktur zur Parametrierung eines digitalen Nockenschaltwerkes ist üblicherweise ein ARRAY OF MC_PathCamSwitch [▶ 45]. PATH_CAMSWITCH_REF verweist mit einem POINTER auf diese Struktur und definiert eindeutig die Größe der Struktur und die Anzahl der tatsächlich verwendeten Nocken.

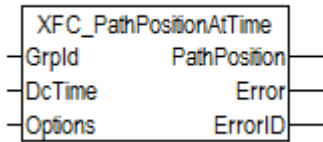
Eine Variable des Typs PATH_CAMSWITCH_REF wird, wie in dem folgenden Beispiel gezeigt, initialisiert:

```

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);
    
```

4.2.4 XFC_PathPositionAtTime



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
    
```

GrpId	Gruppen-Id der Nci-Gruppe. Hierüber wird die angefragte Nci-Gruppe eindeutig im System identifiziert.	
DcTime	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	Datenstruktur mit Optionen für die Extrapolation des relativen Bahnweges.	
Options.	SubIndex	nicht implementiert
Options.	InterpolationOptions	0: Die Extrapolation des relativen Bahnweges wird mit der aktuellen Bahngeschwindigkeit durchgeführt,

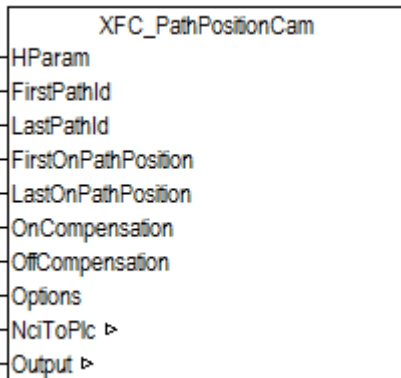
		ohne die aktuelle Bahnbeschleunigung zu berücksichtigen. 1: Die Bahnbeschleunigung der Achse wird in die Extrapolation des relativen Bahnweges einbezogen.
Options.	CompensationTime	zusätzliche Kompensationszeit.

Ausgänge

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

PathPosition	Extrapolierter relativer Bahnweg, bis zum vorgegebenen Zeitpunkt <i>DcTime</i> .
Error	Wird im Fehlerfall TRUE.
ErrorID	Liefert bei einem gesetzten Error-Ausgang die Fehlernummer

4.2.5 XFC_PathPositionCam



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

Im Gegensatz zum digitalen Nockenschaltwerk *MC_PathDigitalCamSwitch_MultiEdge* [▶ 46] 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* oder *XFC_EL2262* [▶ 29]).

Eingänge

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

HParam	H_parameterwert der mit dem Schaltzustand korrespondiert..
---------------	--

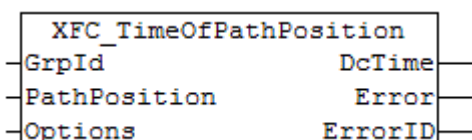
FirstPathId	Eindeutige, kontinuierlich über den Bahnverlauf steigende Id die zum relativen Bahnweg bis zum Einschalten der Nocke gehört.	
LastPathId	Eindeutige, kontinuierlich über den Bahnverlauf steigende Id die zum relativen Bahnweg bis zum Ausschalten der Nocke gehört.	
FirstOnPathPosition	Relativer Bahnweg bis zum Einschalten der Nocke.	
LastOnPathPosition	Relativer Bahnweg bis zum Ausschalten der Nocke.	
OnCompensation	Kompensationszeit für die steigende Flanke der Nocke in [s]. Der Schaltzeitpunkt wird bei negativen Werten von <i>OnCompensation</i> vorgezogen und sonst verzögert. Der hier parametrisierte Wert <i>OnCompensation</i> hat Vorrang vor <i>TRACK_REF</i>	
OffCompensation	Kompensationszeit für die fallende Flanke der Nocke in [s]. Der Schaltzeitpunkt wird bei negativen Werten von <i>OffCompensation</i> vorgezogen und sonst verzögert. Der hier parametrisierte Wert <i>OffCompensation</i> hat Vorrang vor <i>TRACK_REF</i>	
Options	Optionale Parameter	
Options.	EncoderIndex	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.
Options.	UseAcceleration	<i>UseAcceleration</i> kann TRUE gesetzt werden, um die Beschleunigung der Achse in die Positions-Berechnungen mit einzubeziehen. <i>UseAcceleration</i> kann von Vorteil sein, wenn die Sollwerte der Beschleunigung verwendet werden können. Bei Encoder-Achsen, die ein verrauschtes Positionssignal liefern, ist <i>UseAcceleration</i> eventuell von Nachteil, da auch die Beschleunigung fehlerhaft ist.

Ein/Ausgänge

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

NciToPlc	Struktur des zyklischen Kanalinterfaces von der NCI zur SPS. Auf diese Struktur wird nur lesend zugegriffen
Output	Die Datenstruktur [► 18] <i>Output</i> enthält den berechneten Zustand des digitalen Ausgangs und die zugehörigen Zeitstempel für die Ausgabe an einer digitalen XFC Ausgangsklemme

4.2.6 XFC_TimeOfPathPosition



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
```

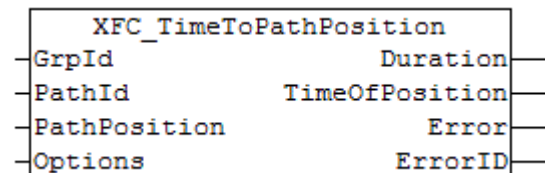
GrpId	Gruppen-Id der Nci-Gruppe. Hierüber wird die angefragte Nci-Gruppe eindeutig im System identifiziert.	
PathPosition	Relativer Bahnweg	
Options	Datenstruktur mit Optionen für die Extrapolation des relativen Bahnweges.	
Options.	SubIndex	nicht implementiert
Options.	InterpolationOptions	0: Die Extrapolation des relativen Bahnweges wird mit der aktuellen Bahngeschwindigkeit durchgeführt, ohne die aktuelle Bahnbeschleunigung zu berücksichtigen. 1: Die Bahnbeschleunigung der Achse wird in die Extrapolation des relativen Bahnweges einbezogen.
Options.	CompensationTime	zusätzliche Kompensationszeit

Ausgänge

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

DcTime	Distributed Clock System Time, zu der der relative Bahnweg <i>PathPosition</i> zurückgelegt sein wird bzw. zu der dieser durchfahren wurde. <i>DcTime</i> enthält die unteren 32 Bit der vollständigen <i>DcTime</i> und deckt damit einen Zeitbereich von +/- 2 Sekunden um den aktuellen Zeitpunkt ab.
Error	Wird im Fehlerfall TRUE.
ErrorID	Liefert bei einem gesetzten Error-Ausgang die Fehlernummer

4.2.7 XFC_TimeToPathPosition



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
```

GrpId	Gruppen-Id der Nci-Gruppe. Hierüber wird die angefragte Nci-Gruppe eindeutig im System identifiziert.	
PathId	Eindeutige, kontinuierlich über den Bahnverlauf steigende Id die zum relativen Bahnweg gehört.	
PathPosition	Relativer Bahnweg	
Options	Datenstruktur mit Optionen für die Extrapolation des relativen Bahnweges.	
Options.	SubIndex	nicht implementiert
Options.	InterpolationOptions	0: Die Extrapolation des relativen Bahnweges wird mit der aktuellen Bahngeschwindigkeit durchgeführt, ohne die aktuelle Bahnbeschleunigung zu berücksichtigen. 1: Die Bahnbeschleunigung der Achse wird in die Extrapolation des relativen Bahnweges einbezogen.
Options.	CompensationTime	zusätzliche Kompensationszeit.

Ausgänge

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

Duration	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	Distributed Clock System Time zu der der relative Bahnweg zurückgelegt sein wird bzw. wurde. <i>DcTime</i> enthält die unteren 32 Bit der vollständigen <i>DcTime</i> und deckt damit einen Zeitbereich von +/- 2 Sekunden um den aktuellen Zeitpunkt ab.	
Error	Wird im Fehlerfall TRUE.	
ErrorID	Liefert bei einem gesetzten Error-Ausgang die Fehlernummer	

Mehr Informationen:
www.beckhoff.de/ts5065

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

