

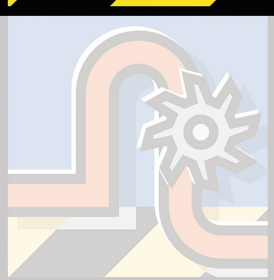
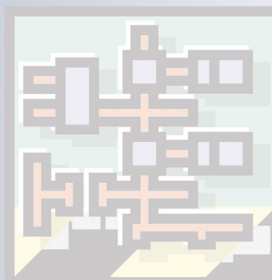
Handbuch | DE

# TS5800

TwinCAT 2 | Digital Cam Server



Supplement | Motion





# Inhaltsverzeichnis

<b>1</b>	<b>Vorwort</b> .....	<b>5</b>
1.1	Hinweise zur Dokumentation .....	5
1.2	Sicherheitshinweise .....	6
1.3	Hinweise zur Informationssicherheit .....	7
<b>2</b>	<b>Übersicht über das Nockenschaltwerk</b> .....	<b>8</b>
<b>3</b>	<b>Konfiguration des Nockenschaltwerks</b> .....	<b>9</b>
<b>4</b>	<b>Encoder-Konfiguration</b> .....	<b>12</b>
<b>5</b>	<b>Nocken-Konfiguration</b> .....	<b>15</b>
<b>6</b>	<b>Prozessabbild des Nockenschaltwerks</b> .....	<b>19</b>
<b>7</b>	<b>Programmierung des Nockenschaltwerks durch die SPS</b> .....	<b>23</b>
<b>8</b>	<b>Diagnose des Nockenschaltwerks</b> .....	<b>27</b>
<b>9</b>	<b>Arbeitsweise des Nockenschaltwerkes</b> .....	<b>29</b>



# 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.

## 1.3 Hinweise zur Informationssicherheit

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

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

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

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

## 2 Übersicht über das Nockenschaltwerk

Der TwinCAT Digital Cam Server realisiert ein schnelles digitales Nockenschaltwerk. Das Nockenschaltwerk wird als zusätzliche Option mit dem TwinCAT-System installiert und kann anschließend im TwinCAT System Manager konfiguriert werden. Neben der Grundparametrierung kann die Funktion des Nockenschaltwerks durch die SPS beeinflusst werden. So können einzelne Nocken aktiviert oder deaktiviert werden oder Nockenschaltpositionen können geändert werden.

### Konfigurationsmöglichkeiten

Das Nockenschaltwerk kann in einer eigenen schnellen Task mit einem separaten I/O-Bus betrieben werden und ist damit unabhängig von langsameren Zykluszeiten von SPS oder NC. Eine Konfiguration ohne separaten I/O-Bus ist ebenfalls möglich, falls extrem kurze Zykluszeiten nicht gefordert sind.

- Encoder-Quelle frei wählbar (z. B. NC-Antrieb)
- Zykluszeit abhängig von Encoder-Quelle (z. B. 250 µs)

### Optionale Konfiguration mit separatem I/O-Bus

- unabhängige schnelle Digital Cam Server-Task mit Zykluszeiten bis hinunter zu 80 µs
- Eigene Beckhoff Lightbus I/O-Karte mit angeschlossenem Encoder und Nockenausgängen

### Kennwerte

- bis zu 320 Ausgänge (Nockenspuren)
- bis zu 180 Nocken pro Ausgang
- bis zu 2048 Nocken insgesamt
- Weg-Nocken (Ein- und Ausschaltposition)
- Zeit-Nocken (Einschaltposition und Schaltdauer)
- Bremsnocken (Sonderfunktion für Maschinenstopp)

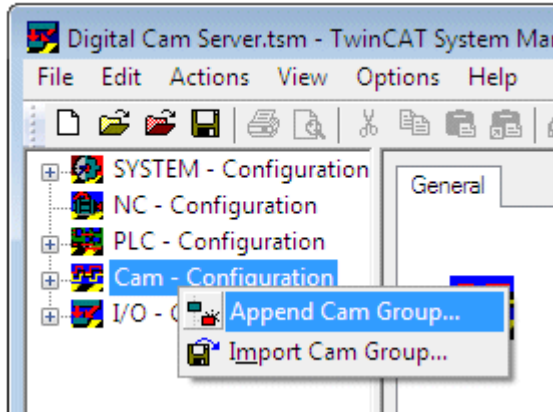


Alle genannten Variablen- und Parameter-Namen beziehen sich auf die englische Spracheinstellung des TwinCAT System Managers. Der System Manager benennt die Variablen und Parameter in einer anderen Spracheinstellung teilweise anders und wechselt einen Teil der Bezeichnungen nach einer Sprachumschaltung. Um einen eindeutigen Bezug zu dieser Dokumentation zu erreichen, empfiehlt es sich, die Spracheinstellung auf Englisch zu setzen, noch bevor ein neues Nockenschaltwerk angelegt wird. Für die Funktion des Nockenschaltwerkes hat die Spracheinstellung allerdings keine Bedeutung.

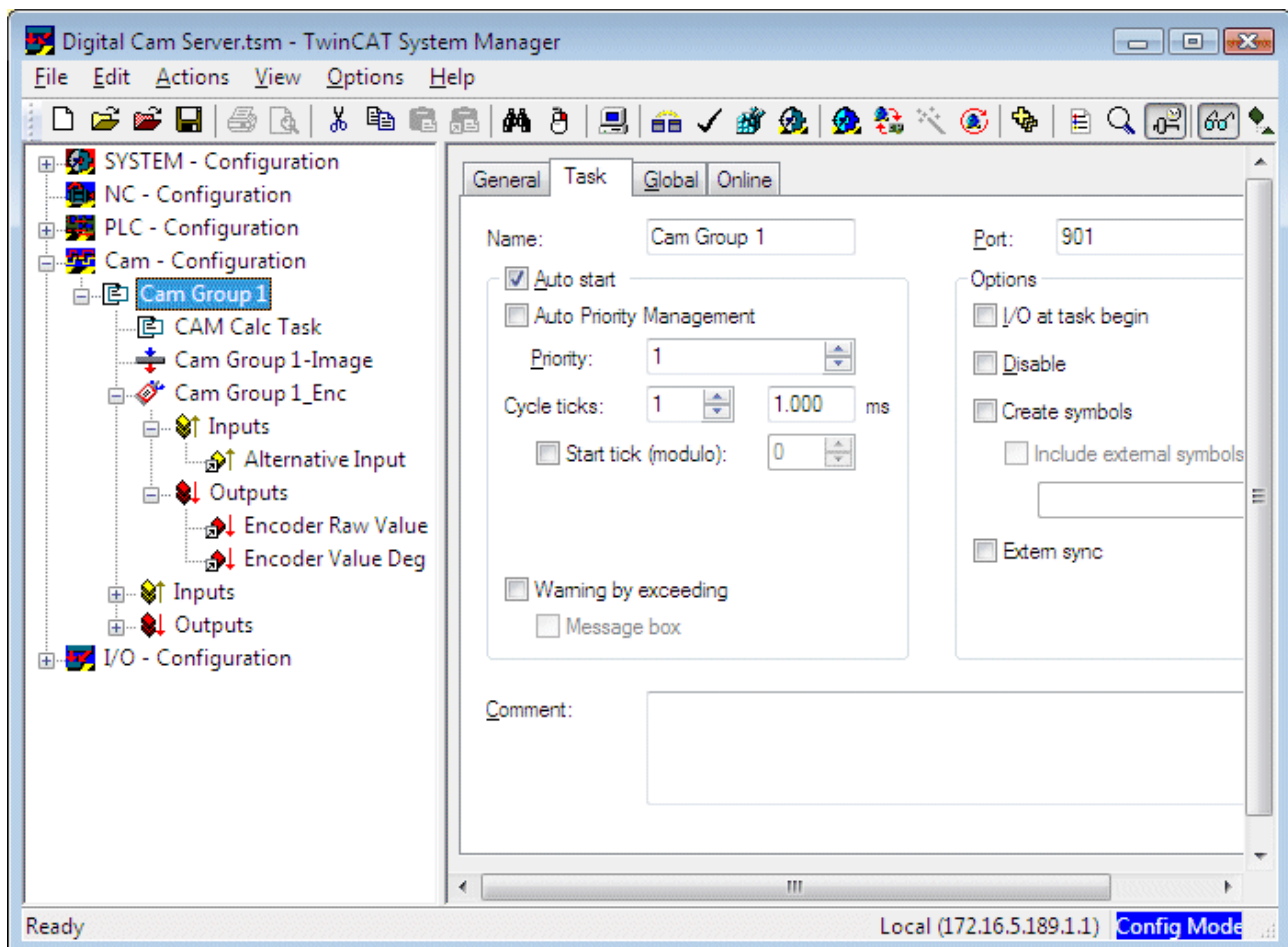


### 3 Konfiguration des Nockenschaltwerks

Der TwinCAT Digital Cam Server wird optional mit der Installation des TwinCAT-Systems installiert. Anschließend steht im TwinCAT System Manager eine Konfigurationsgruppe für das Nockenschaltwerk zur Verfügung. Durch einen rechten Mausklick auf diese Gruppe kann eine neues Nockenschaltwerk eingefügt werden.



Das neu eingefügte Nockenschaltwerk enthält zwei Tasks, eine schnelle Task zur Ausgabe der Nocken auf dem I/O-Bus und eine langsamere Task, die die Berechnung der Nockenschaltpunkte ausführt. Weiterhin ist ein logischer Encoder enthalten, der die Position des Nockenschaltwerks erfasst.



Die Zykluszeit wird auf dem Reiter *Task* eingestellt. Die Zykluszeit kann bis hinunter zur Basiszeit des TwinCAT Systems eingestellt werden (siehe *System-Konfiguration*).

Die Zykluszeit der schnellen Nocken-Ausgabe-Task (im Bild *Cam Group 1*) sollte so gewählt werden, dass bei maximaler Rotationsgeschwindigkeit ein maximal erlaubter Winkelfehler nicht überschritten wird.

Beispiel: Bei einer Hubzahl von 30 U/min (180 °/s) ist ein maximaler Winkelfehler der Nockenausgabe von 0.2 Grad zulässig

$$\text{CycleTime} = \text{PositionError} / \text{Velocity}$$

$$\text{Cycle Time} = 0.2^\circ / 180^\circ/\text{s} = 0.00111 \text{ s}$$

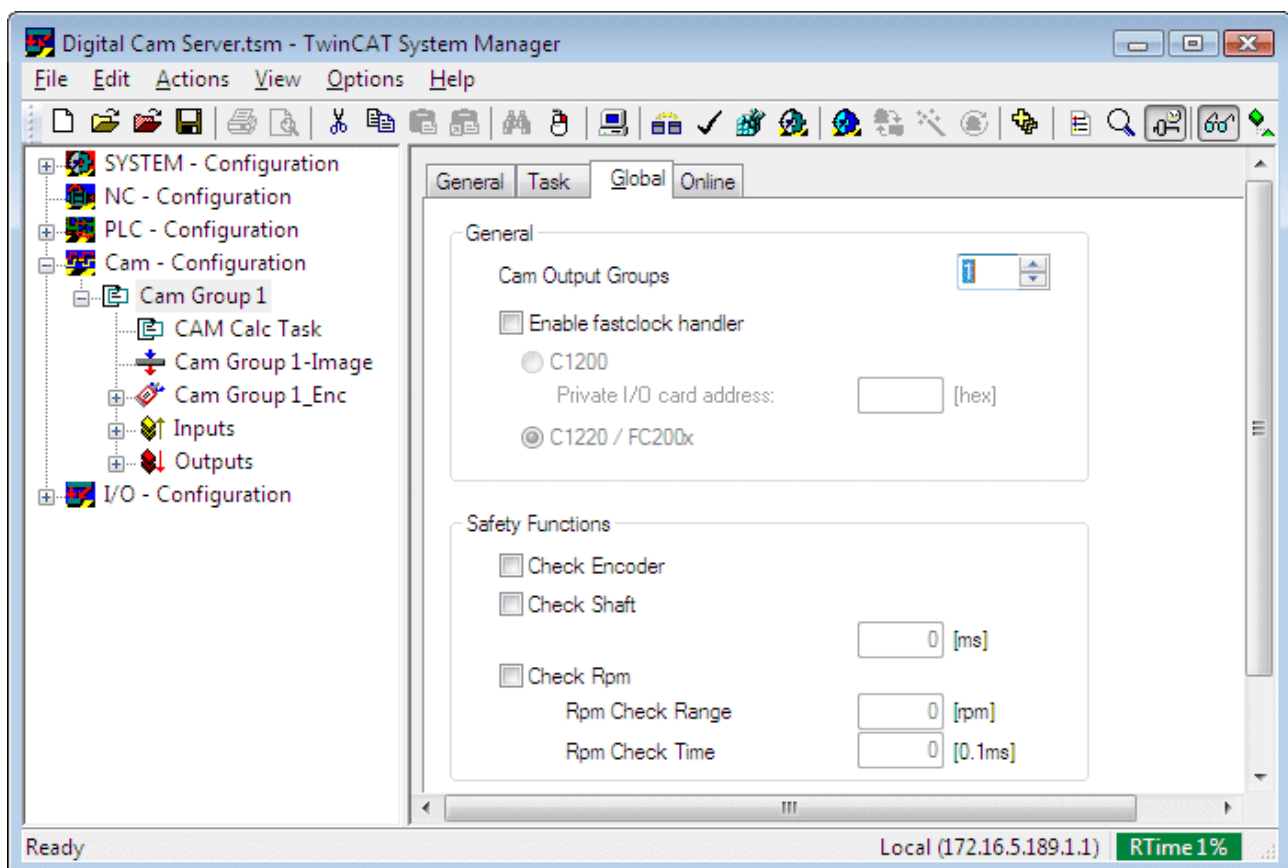
Ein Zykluszeit des Nockenschaltwerks von 1 ms ist in diesem Fall ausreichend.

Eine kürzere Zykluszeit kann die CPU der Steuerung nicht unerheblich belasten.

Die Zykluszeit der Berechnungstask (im Bild *CAM Calc Task*) wirkt sich nicht auf die Ausgabegenauigkeit aus. Da aufwändige Berechnungen in dieser Task bearbeitet werden, sollte die Zykluszeit deutlich länger sein (Default 10 ms).

Für Zykluszeiten unter einer Millisekunde sind zwei Konfigurationen möglich. Zum Einen kann man die Basiszeit des TwinCAT-Systems (siehe Echtzeitkonfiguration) unter eine Millisekunde reduzieren. Da die Nockenschaltwerks-Task bis hinunter zur Basiszeit eingestellt werden kann, kann dann auch diese Zykluszeit reduziert werden. Optional kann aber auch für die Nockenschaltwerks-Task eine separater Zeitgeber konfiguriert werden, um eine geringere CPU-Last des Gesamtsystems zu erreichen. Diese **Fast Clock** Option wird im nächsten Absatz beschrieben.

### Weitere Parameter des Nockenschaltwerks



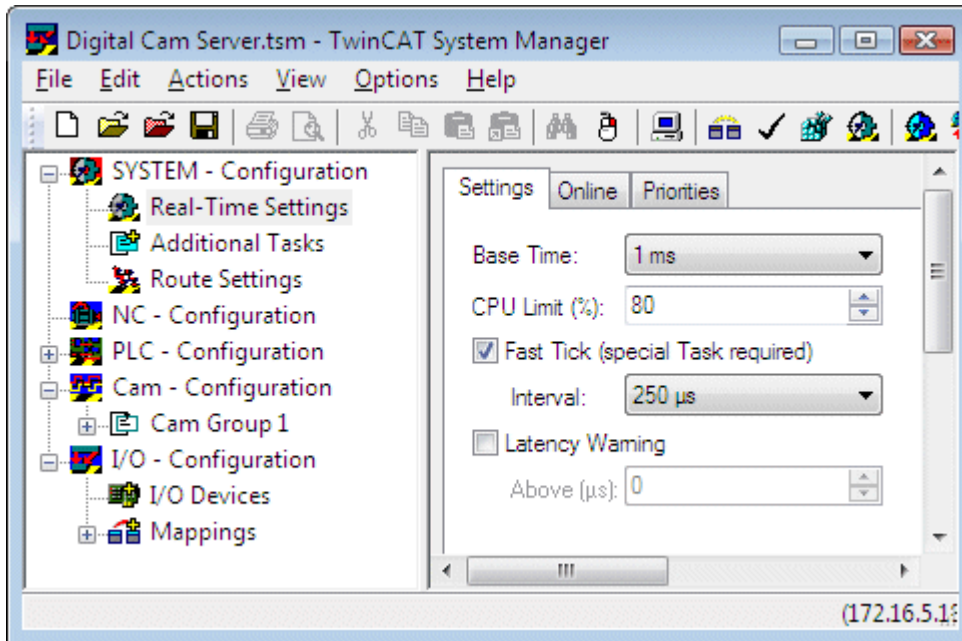
Auf dem Global-Reiter werden Nocken-unabhängige Betriebsparameter des Nockenschaltwerks festgelegt.

**Cam Output Groups:** Zunächst wird die benötigte Anzahl der Nockengruppen festgelegt. Eine Nockengruppe enthält 32 digitale Ausgänge. Das Nockenschaltwerk kann auf bis zu 10 Gruppen, also auf 320 digitale Ausgänge (Nockenspuren) erweitert werden.

### Fast Clock Handler

Das Nockenschaltwerk kann mit einer wesentlich schnelleren Zykluszeit betrieben werden, als andere Tasks im System (NC, SPS etc.). Unter Umständen liegt die geforderte Zykluszeit des Nockenschaltwerks sogar unterhalb der Basiszeit des Systems (typisch 1 ms). In diesem Fall kann es sinnvoll sein, das Nockenschaltwerk mit einem eigenen Taktgeber zu versehen.

Nachdem der *Fast Clock Handler* aktiviert wurde, läuft das Nockenschaltwerk außerhalb des TwinCAT-Task-Schedulers und ist somit von der Basiszeit des Systems unabhängig. Die Basiszeit des TwinCAT-Systems ist anschließend fest auf 1 ms eingestellt und die Zykluszeit des Nockenschaltwerkes kann jetzt in der Echtzeit-Konfiguration über das *Fast Tick* Intervall eingestellt werden.



Die Fast Clock Option ist nicht allein auf Lightbus-I/O-Karten (FC2000, C1220, C1200) beschränkt. Falls aber eine kaum mehr verwendete C1200 ISA Karte eingesetzt wird, so muss die entsprechende Option im Konfigurationsdialog angewählt werden, da diese Karte direkt durch das Nockenschaltwerk verwaltet wird. Diese Karte setzt auch die Fast Clock Option zwingend voraus.

Für eine C1220 Karte ist eine Firmware ab Version V4.18 notwendig.

## Sicherheitsfunktionen

Die Sicherheitsfunktionen dienen zur Überwachung der Maschinen- bzw. Encoder-Funktion. Die Überwachung wird aktiviert, sobald durch die SPS das *Coupled*-Signal im [Prozessabbild \[► 19\]](#) aktiviert wird. Ist das *Coupled*-Signal nicht aktiv, so geht die Steuerung davon aus, dass der Encoder mechanisch entkoppelt ist und schaltet damit auch die Überwachung aus.

- **Encoderüberwachung (Check Encoder):** Bei aktiver Überwachung wird die Änderung des Encoder-Zählwerts überprüft. Ändert sich der Wert nicht, so setzt das Nockenschaltwerk im [Prozessabbild \[► 19\]](#) die Variable *EncoderError*. Der Fehler kann von der SPS über die Variable *ResetEncoderError*, ebenfalls im Prozessabbild, zurückgesetzt werden.
- **Wellenbruchüberwachung (Check Shaft):** Bei aktiver Überwachung wird, ähnlich wie bei der Encoderüberwachung, die Änderung des Encoder-Zählwerts überprüft. Für diese Überwachung wird jedoch mit einer einstellbaren Filterzeit gefiltert. Ein Fehler wird im [Prozessabbild \[► 19\]](#) durch die Variable *ShaftError* signalisiert. Der Fehler kann von der SPS über die Variable *ResetShaftError*, ebenfalls im Prozessabbild, zurückgesetzt werden.
- **Hubzahlüberwachung (Check Rpm):** Die Hubzahlüberwachung überwacht eine Abweichung der Encoderdrehzahl von einem hier einstellbaren Wert. Die Überwachung kann mit einer einstellbaren Zeit gefiltert werden.
  - *Rpm Check Range*: maximale Drehzahlabweichung
  - *Rpm Check Time*: minimale Dauer der Drehzahlabweichung

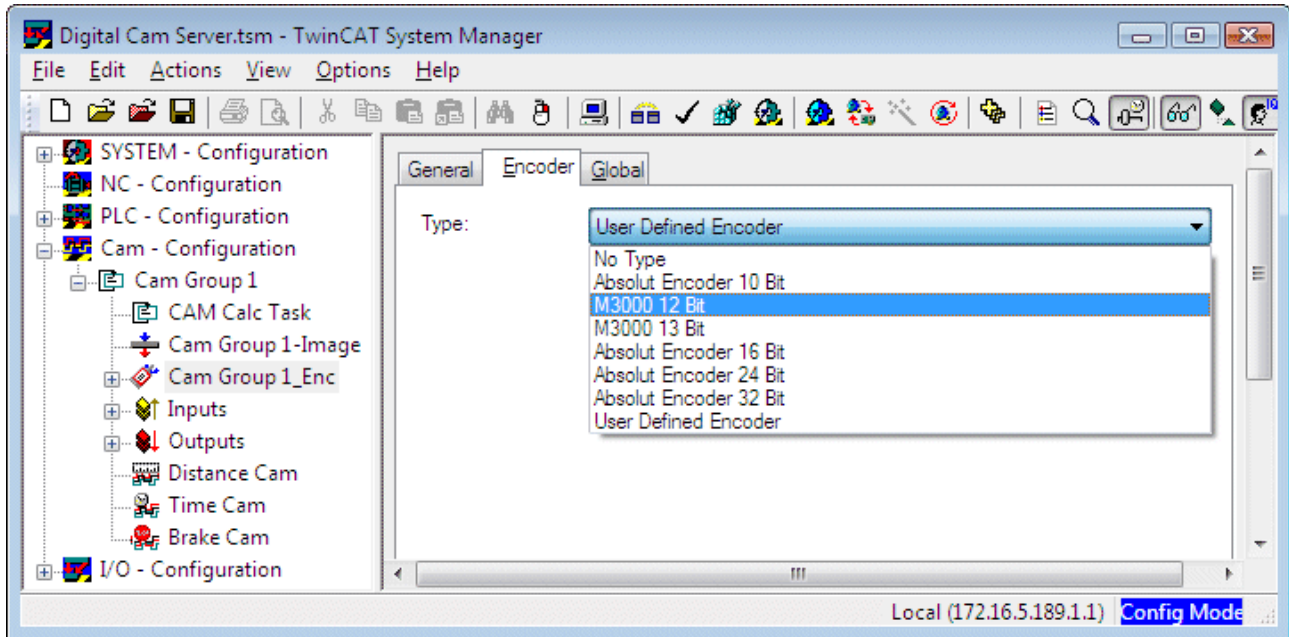
Im Fehlerfall setzt das Nockenschaltwerk die Variable *HPMError* im [Prozessabbild \[► 19\]](#). Der Fehler kann von der SPS über die Variable *ResetHPMError*, ebenfalls im Prozessabbild, zurückgesetzt werden.

## 4 Encoder-Konfiguration

Der Encoder für das Nockenschaltwerk kann über einen eigenen I/O-Bus eingelesen werden, der durch das Nockenschaltwerk selbst betrieben wird. Falls die Geschwindigkeit des Standard I/O-Busses ausreichend ist, können die Encoder-Daten auch von einem Encoder oder einem Antrieb an diesem Bus eingelesen werden.

### Encodertypen

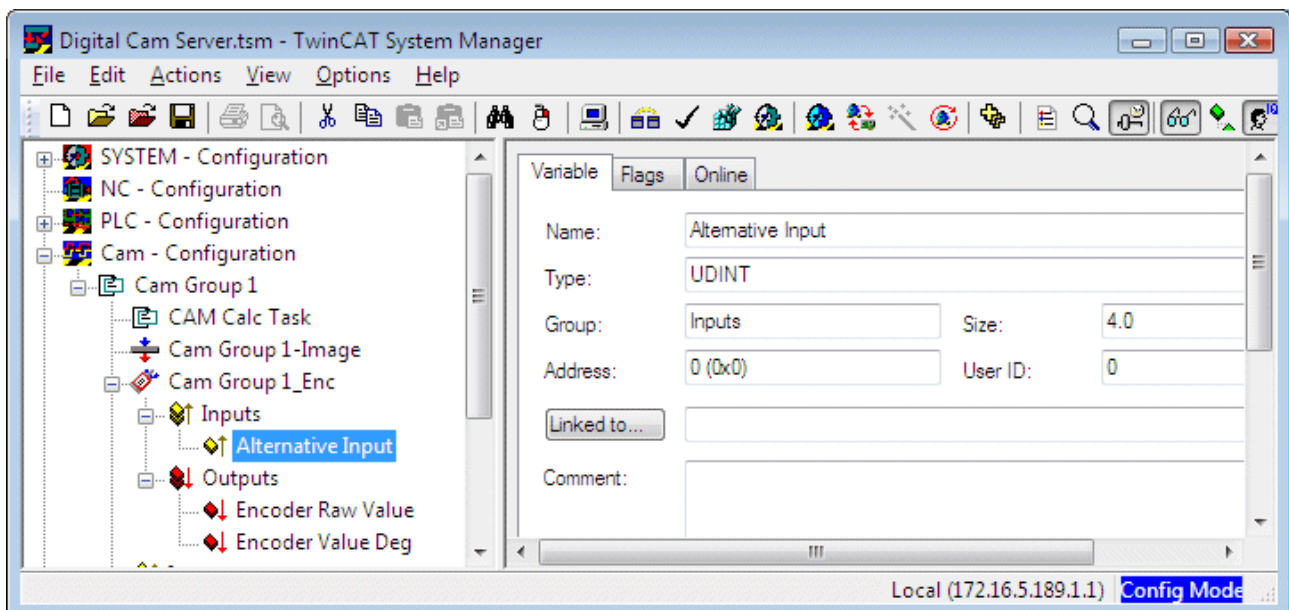
Das Nockenschaltwerk kann mit Encodern unterschiedlicher Auflösung arbeiten. Die im Auswahldialog angebotenen Encodertypen unterscheiden sich im wesentlichen durch die Bit-Breite, daher kann ein nicht gelisteter Encoder als *User Defined Encoder* eingefügt werden und es muss anschließend die Encoder-Maske passend zur tatsächlichen Bit-Breite konfiguriert werden (siehe *Global Dialog*).



### Encoder am Standard-I/O-Bus

Das Encoder-Datenwort wird mit dem alternativen Eingang des Nockenschaltwerks-Encoders verbunden.

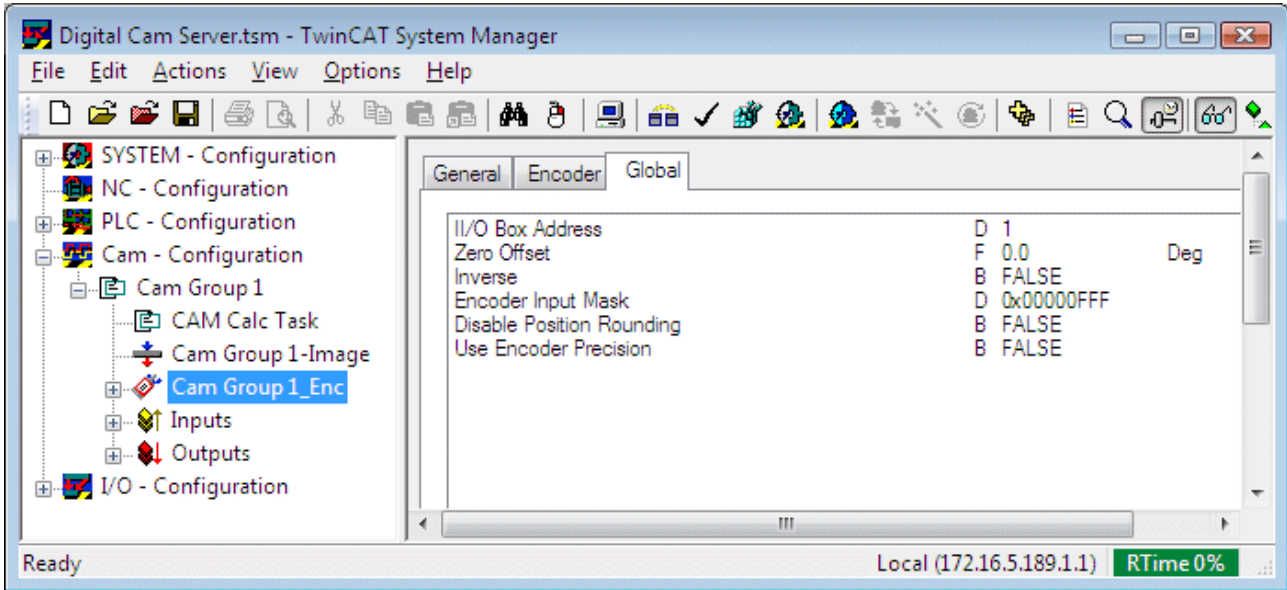
- **Alternative Input:** Eingang zur Verknüpfung des Encoder-Zählwertes von der Hardware mit dem Nockenschaltwerk



Die Ausgänge des Encoder-Prozessabbildes können optional mit der SPS verknüpft werden.

- **Encoder Raw Value:** aktueller Zählwert des Encoders
- **Encoder Value Deg:** aktuelle Encoderposition in Grad. Wenn der Parameter *Use Encoder Precision* FALSE ist, ist die Auflösung [1/10°], anderenfalls [1/1000 °]

Im Dialog *Global* wird die Parametrierung des Encoders festgelegt.



**I/O Box Address:** wird hier nicht benötigt.

Die *I/O Box Address* wird dann benötigt, wenn das Nockenschaltwerk über einen eigenen I/O-Bus (Option mit C1200, C1220 oder FC2000-Karte) verfügt. In diesem speziellen Fall wird der Bus durch das Nockenschaltwerk verwaltet und es muss hier die Adresse der Ausgangsbox angegeben werden.

**Zero Offset:** Offset zur Justierung der 0-Grad Position des Encoders

**Inverse:** Schaltet die Drehrichtung des Encoders um, falls die physikalische Zählrichtung nicht mit der gewünschten Drehrichtung übereinstimmt.

**Encoder Input Mask:** Über diese Maske wird der Wertebereich des Encoders bestimmt. Hier wird der größtmögliche Zählwert des Encoders als Hexadezimalwert angegeben. Für einen 12-Bit Encoder, der seinen Überlauf bei einem Zählwert von 4095 hat, wird beispielsweise 0x00000FFF eingetragen.

**Disable Position Rounding :** Durch die Positionsroundung wird die unterschiedliche Auflösung des Encoders (z. B. 4096 Inkremente) an die Auflösung des Nockenschaltwerks (z. B. 360 Grad in 1 Grad-Schritten) angepasst. Der tatsächliche Schaltpunkt einer Nocke liegt durch die Rundung in einem Fenster um die angegebene Schaltposition (z. B. 90 Grad +/- 0.5 Grad)

Die Rundung kann abgeschaltet werden, falls sie nicht gewünscht wird.

**Use Encoder Precision:** Das Nockenschaltwerk arbeitet standardmäßig mit einer Auflösung von 1 Grad, also mit 360 Schritten pro Umdrehung. Für den Fall, dass eine höhere Auflösung benötigt wird, kann das Nockenschaltwerk mit der Auflösung des Encoders betrieben werden. Bei einem 16-Bit Encoder stehen dann 65536 mögliche Schaltpunkte pro Umdrehung zur Verfügung. (Die Nocken-Schaltpositionen werden weiterhin in Grad angegeben.)

Durch die erhöhte Präzision ändern sich teilweise die Einheiten der Daten im Prozessabbild des Nockenschaltwerks [► 19]. Das nachträgliche Ändern dieser Einstellung kann daher das SPS-Programm beeinflussen.



Bei hochauflösenden Encodern (ab etwa 20 Bit) ist die Anzahl der möglichen Schaltpunkte extrem groß. In diesem Fall wird es notwendig, den Routerspeicher (siehe Echzeiteinstellungen) auf einen größeren Wert zu erweitern.

**Option: Encoder an dediziertem I/O-Bus**

Das Nockenschaltwerk kann mit einem eigens für das Nockenschaltwerk bestimmten I/O-Bus versehen werden. Diese spezielle Konfiguration arbeitet mit einem Beckhoff Lightbus und benötigt eine der I/O-Karten C1200, C1220 oder FC2000. Diese Option wurde für sehr kurze Zykluszeiten bis hinunter zu 80µs entwickelt und arbeitet mit einem Encoder und einer digitalen 32-Bit I/O-Box.

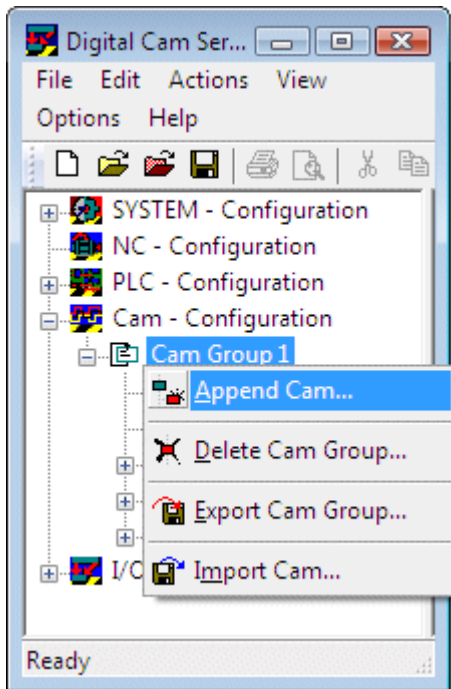
In vielen Fällen ist eine solche Konfiguration heute nicht mehr nötig und kann wie oben beschrieben, durch die Standardkonfiguration erreicht werden. In jedem Fall ist ein schneller I/O-Bus, wie z. B. EtherCAT notwendig.

## 5 Nocken-Konfiguration

Eine Nocke schaltet einen digitalen Ausgang innerhalb eines bestimmten Winkelbereiches während einer Umdrehung des Nocken-Encoders. Für einen Ausgang können mehrere Nocken definiert werden, so dass das Ausgangssignal während einer Umdrehung mehrfach geschaltet werden kann.

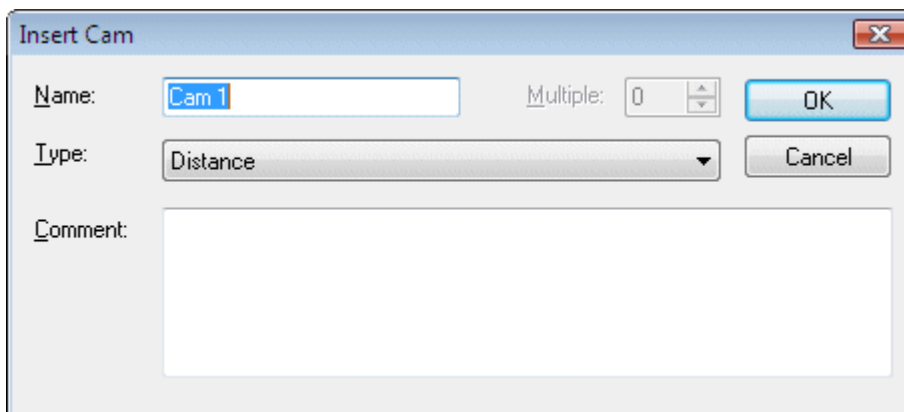
Es stehen drei verschiedene Nocken zur Verfügung. Eine Wegnocke (*auch Positionsnocke oder Distance Cam*) schaltet an einen Ausgang an einer bestimmten Position ein und an einer anderen Position aus. Eine Zeitnocke (*Time Cam*) schaltet einen Ausgang ebenfalls an einer bestimmten Position ein, hält ihn aber dann für eine bestimmte Zeit aktiv. Eine Bremsnocke (*Brake Cam*) hat eine Sonderfunktion während des Bremsvorgangs.

Mit einem rechten Mausklick auf die Nockengruppe *Cam Group 1* kann eine neue Nocke eingefügt werden.



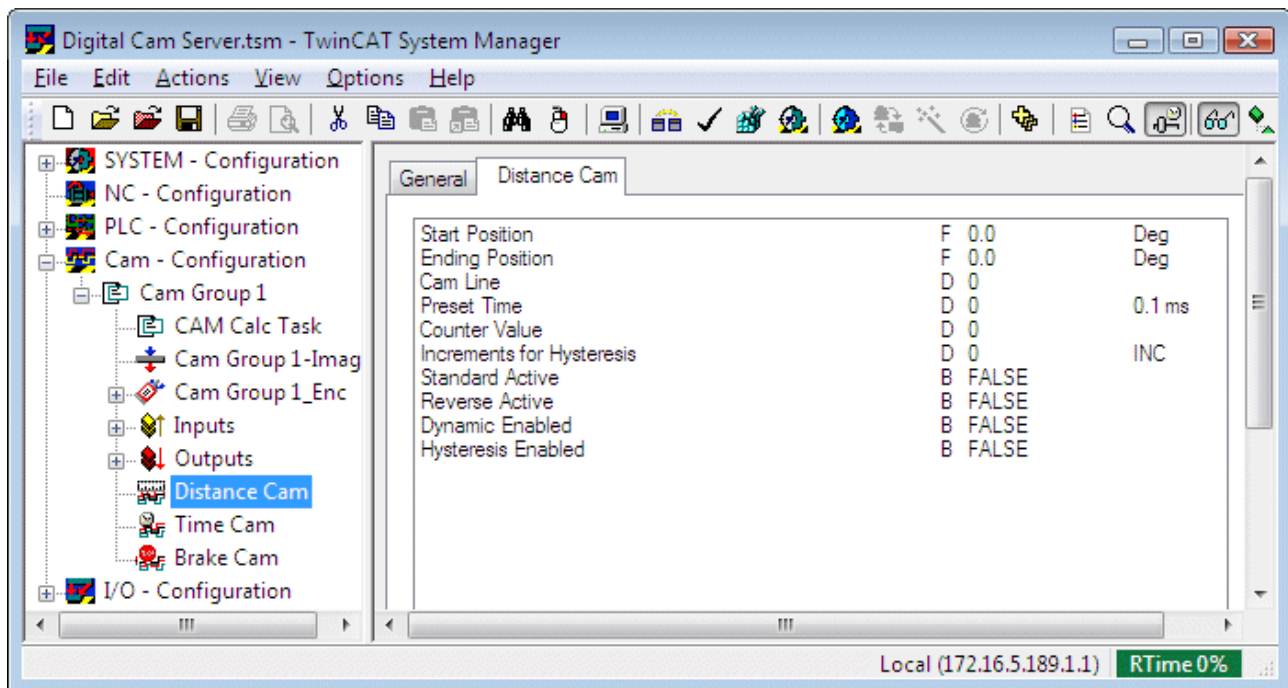
Es öffnet sich ein Dialogfenster, in dem der Name und der Typ der Nocke festgelegt wird.

- Wegnocke (Distance Cam)
- Zeitnocke (Time Cam)
- Bremsnocke (Brake Cam)



### Wegnocken

Nachdem eine Wegnocke eingefügt wurde, können die Parameter konfiguriert werden.

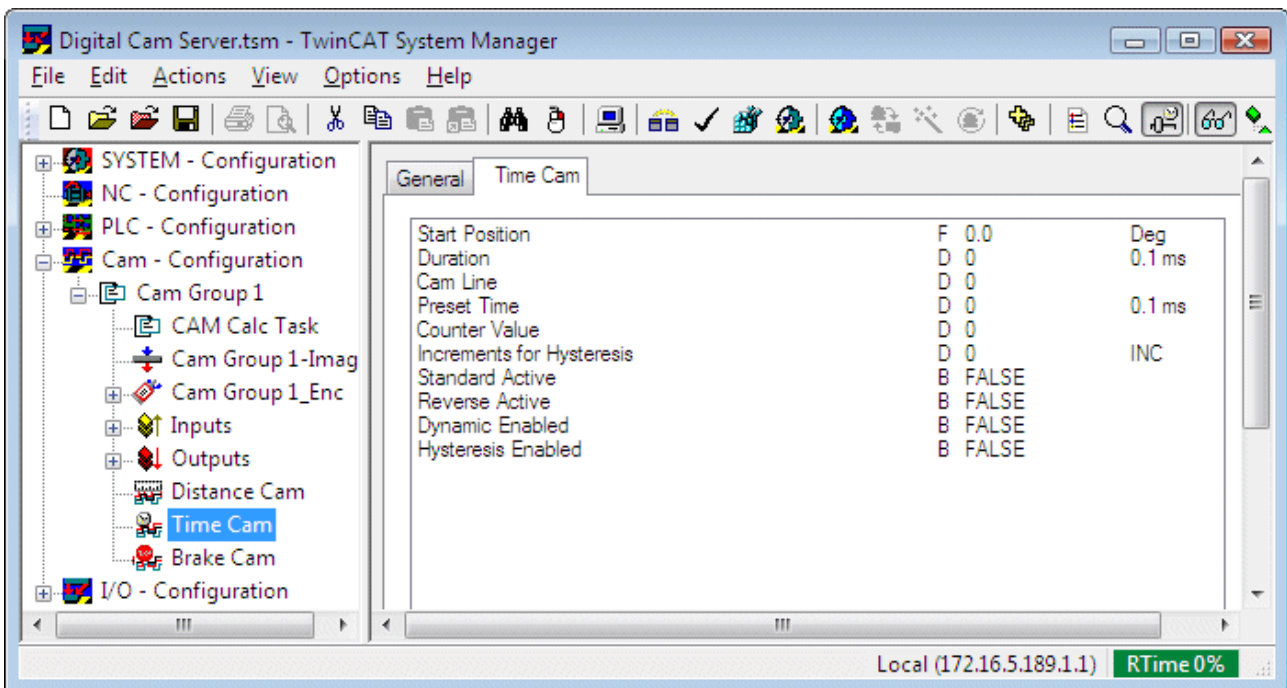


- **Start Position:** Winkelposition innerhalb einer Encoder-Umdrehung, an der ein digitaler Ausgang eingeschaltet wird.
- **Ending Position:** Winkelposition innerhalb einer Encoder-Umdrehung, an der der digitale Ausgang wieder ausgeschaltet wird
- **Cam Line:** Nummer des digitalen Ausgangs, der durch diese Nocke geschaltet wird. Die Ausgänge sind mit 0 beginnend durchnummeriert.
- **Counter Value:** Dieser Wert legt fest, ob die Nocke in jeder Umdrehung oder nur in jeder n-ten Umdrehung ausgegeben wird. Ein Wert von 0 schaltet die Nocke ganz aus. Es muss also für eine ordentliche Funktion mindestens der Wert 1 angegeben werden.
- **Standard Active:** Die Nocke wird in der Standard-Nockentabelle eingetragen wenn dieser Parameter TRUE ist.  
Das Nockenschaltwerk verwaltet zwei Nockentabellen. Die SPS legt über die Variable *ReverseMode* im Prozessabbild [► 19] fest, ob die Standard-Nockentabelle oder die Reverse-Nockentabelle aktiv ist (Standard wenn *ReverseMode*=0).
- **Reverse Active:** Die Nocke wird in der Reverse-Nockentabelle eingetragen, wenn dieser Parameter TRUE ist.  
Das Nockenschaltwerk verwaltet zwei Nockentabellen. Die SPS legt über die Variable *ReverseMode* im Prozessabbild [► 19] fest, ob die Standard-Nockentabelle oder die Reverse-Nockentabelle aktiv ist. (Reverse wenn *ReverseMode*=1).
- **Dynamic Enabled:** Das Nockenschaltwerk errechnet eine Geschwindigkeitsabhängige Positionskorrektur falls dieser Parameter TRUE ist. Bei höheren Geschwindigkeiten schaltet die Nocke also früher. Die Korrektur kann über *Preset Time* konfiguriert werden.
- **Preset Time:** Vorhaltezeit zum Ausgleich von Schaltzeiten eines an einem Ausgang angeschlossenen Aktors. (Ganzzahliger Wert in 100 µs Schritten). Die Vorhaltezeit wird erst durch Einschalten der Dynamik-Funktion (*Dynamic Enabled*) wirksam.
- **Hysteresis Enabled:** nicht implementiert
- **Increments for Hysteresis:** nicht implementiert

## Zeitnocken

Zeitnocken können nicht gemeinsam mit Weg- oder Bremsnocken auf demselben Ausgang konfiguriert werden. Abhängig von der Schaltzeit und Geschwindigkeit sollte auch nur eine Zeitnocke pro Ausgangsspur konfiguriert werden.



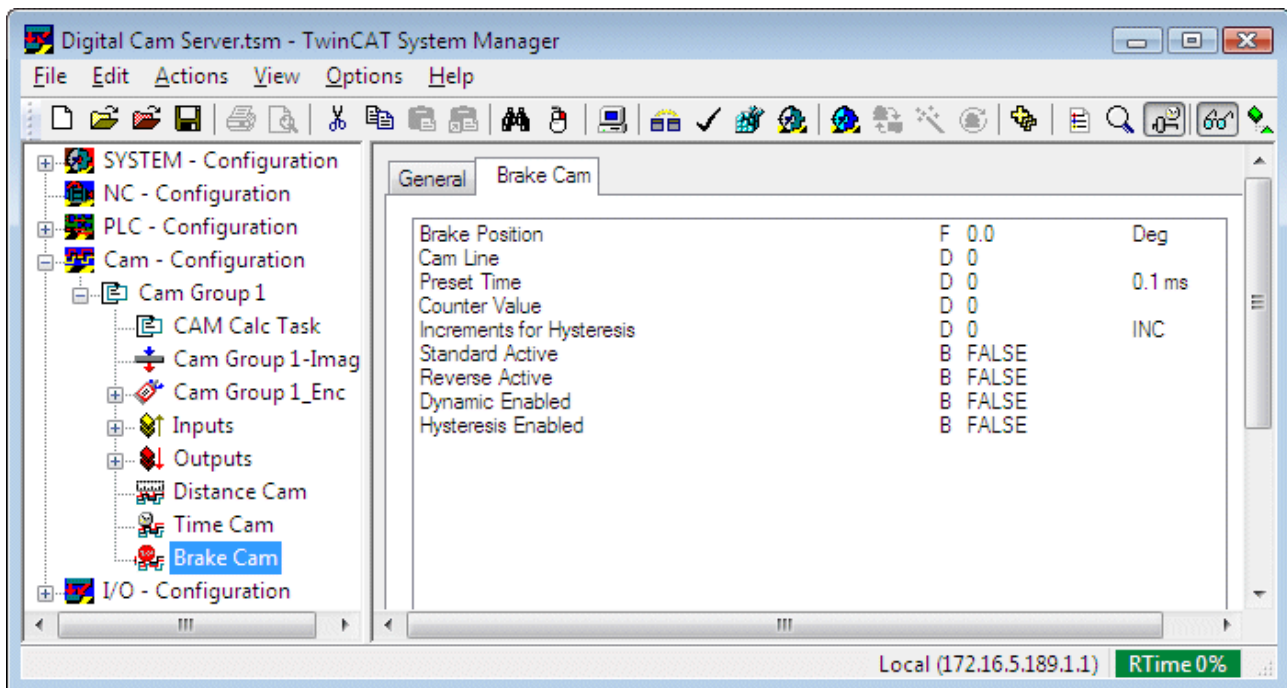


- **Start Position:** Winkelposition innerhalb einer Encoder-Umdrehung, an der ein digitaler Ausgang eingeschaltet wird.
- **Duration:** Einschaltdauer der Nocke als ganzzahliger Wert in 100 µs Schritten
- **Cam Line:** Nummer des digitalen Ausgangs, der durch diese Nocke geschaltet wird. Die Ausgänge sind mit 0 beginnend durchnummeriert.
- **Counter Value:** Dieser Wert legt fest, ob die Nocke in jeder Umdrehung oder nur in jeder n-ten Umdrehung ausgegeben wird. Ein Wert von 0 schaltet die Nocke ganz aus. Es muss also für eine ordentliche Funktion mindestens der Wert 1 angegeben werden.
- **Standard Active:** Die Nocke wird in der Standard-Nockentabelle eingetragen wenn dieser Parameter TRUE ist.  
Das Nockenschaltwerk verwaltet zwei Nockentabellen. Die SPS legt über die Variable *ReverseMode* im Prozessabbild [► 19] fest, ob die Standard-Nockentabelle oder die Reverse-Nockentabelle aktiv ist (Standard wenn *ReverseMode*=0).
- **Reverse Active:** Die Nocke wird in der Reverse-Nockentabelle eingetragen, wenn dieser Parameter TRUE ist.  
Das Nockenschaltwerk verwaltet zwei Nockentabellen. Die SPS legt über die Variable *ReverseMode* im Prozessabbild [► 19] fest, ob die Standard-Nockentabelle oder die Reverse-Nockentabelle aktiv ist. (Reverse wenn *ReverseMode*=1).
- **Dynamic Enabled:** Das Nockenschaltwerk errechnet eine Geschwindigkeitsabhängige Positionskorrektur falls dieser Parameter TRUE ist. Bei höheren Geschwindigkeiten schaltet die Nocke also früher. Die Korrektur kann über *Preset Time* konfiguriert werden.
- **Preset Time:** Vorhaltezeit zum Ausgleich von Schaltzeiten eines an einem Ausgang angeschlossenen Aktors. (Ganzzahliger Wert in 100 µs Schritten). Die Vorhaltezeit wird erst durch Einschalten der Dynamik-Funktion (*Dynamic Enabled*) wirksam.
- **Hysteresis Enabled:** nicht implementiert
- **Increments for Hysteresis:** nicht implementiert

**Bremsnocken**

Eine Bremsnocke hat eine Sonderfunktion. Der zugeordnete Ausgang (*Cam Line*) ist während des normalen Betriebes zunächst aktiv, solange die SPS die der Ausgangsspur zugeordnete *Break Enabled Variable* im Prozessabbild [► 19] gesetzt hat (Bremsfreigabe). Um einen Bremsvorgang einzuleiten, setzt die SPS *Break Enabled* auf 0. In Folge wird die Ausgangsspur an der parametrisierten Winkelposition abgeschaltet.

Bremsnocken können nicht gemeinsam mit Weg- oder Zeitnocken auf demselben Ausgang konfiguriert werden. Pro Ausgangsspur kann auch nur eine Bremsnocke konfiguriert werden.



- **Brake Position:** Winkelposition innerhalb einer Encoder-Umdrehung, an der ein digitaler Ausgang ausgeschaltet wird, sobald die Bremsfreigabe der SPS abgeschaltet wird (*Break Enabled* im Prozessabbild [► 19])
- **Cam Line:** Nummer des digitalen Ausgangs, der durch diese Nocke geschaltet wird. Die Ausgänge sind mit 0 beginnend durchnummeriert.
- **Counter Value:** Dieser Wert legt fest, ob die Nocke in jeder Umdrehung oder nur in jeder n-ten Umdrehung ausgegeben wird. Ein Wert von 0 schaltet die Nocke ganz aus. Es muss also für eine ordentliche Funktion mindestens der Wert 1 angegeben werden.
- **Standard Active:** Die Nocke wird in der Standard-Nockentabelle eingetragen wenn dieser Parameter TRUE ist.  
Das Nockenschaltwerk verwaltet zwei Nockentabellen. Die SPS legt über die Variable *ReverseMode* im Prozessabbild [► 19] fest, ob die Standard-Nockentabelle oder die Reverse-Nockentabelle aktiv ist (Standard wenn *ReverseMode*=0).
- **Reverse Active:** Die Nocke wird in der Reverse-Nockentabelle eingetragen, wenn dieser Parameter TRUE ist.  
Das Nockenschaltwerk verwaltet zwei Nockentabellen. Die SPS legt über die Variable *ReverseMode* im Prozessabbild [► 19] fest, ob die Standard-Nockentabelle oder die Reverse-Nockentabelle aktiv ist. (Reverse wenn *ReverseMode*=1).
- **Dynamic Enabled:** Das Nockenschaltwerk errechnet eine Geschwindigkeitsabhängige Positionskorrektur falls dieser Parameter TRUE ist. Bei höheren Geschwindigkeiten schaltet die Nocke also früher. Die Korrektur kann über *Preset Time* konfiguriert werden.
- **Preset Time:** Vorhaltezeit zum Ausgleich von Schaltzeiten eines an einem Ausgang angeschlossenen Aktors. (Ganzzahliger Wert in 100 µs Schritten). Die Vorhaltezeit wird erst durch Einschalten der Dynamik-Funktion (*Dynamic Enabled*) wirksam.
- **Hysteresis Enabled:** nicht implementiert
- **Increments for Hysteresis:** nicht implementiert

### Positionsauflösung

Anfangs- und Endpositionen der Nocken können mit einer Auflösung von 1/1000 Grad eingegeben werden. Es ist jedoch zu beachten, dass das Nockenschaltwerk intern mit zwei möglichen Auflösungen arbeitet und die definierten Position auf die interne Auflösung rundet. Je nach [Konfiguration des Encoders \[► 12\]](#) (siehe Parameter *Use Encoder Precision*) arbeitet das Nockenschaltwerk fest mit 360 Schritten oder aber mit der vollen Encoderauflösung (Encoderinkremente pro Umdrehung)

## 6 Prozessabbild des Nockenschaltwerks

Das Nockenschaltwerk stellt in seinem Prozessabbild Eingangs- und Ausgangsvariablen zur Verfügung, die mit der SPS verknüpft werden können. Über diese Schnittstelle kann die SPS die Funktion des Nockenschaltwerkes beeinflussen und den aktuellen Zustand überwachen.

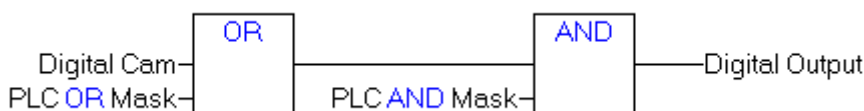
### Eingangs-Prozessabbild

Über das Eingangs-Prozessabbild kann die SPS zur Laufzeit die Funktion des Nockenschaltwerks beeinflussen. Auf Seite der SPS werden diese Variablen als Ausgänge deklariert und im System Manager mit dem Eingangsprozessabbild des Nockenschaltwerks verknüpft.

Name	Online		Type	Size	> Addr...	In/Out
Tool Safe	X	0x00000000 (0)	DWORD	4.0	4.0	Input
Reverse Mode	X	0x00000000 (0)	UDINT	4.0	8.0	Input
Coupled	X	0x00000000 (0)	UDINT	4.0	12.0	Input
HPM Preset	X	0x00000000 (0)	UDINT	4.0	16.0	Input
HPM Difference	X	0x00000000 (0)	UDINT	4.0	20.0	Input
Reset I/O Errors	X	0x00000000 (0)	UDINT	4.0	24.0	Input
Reset Encoder Error	X	0x00000000 (0)	UDINT	4.0	28.0	Input
Reset Shaft Error	X	0x00000000 (0)	UDINT	4.0	32.0	Input
Reset HPM Error	X	0x00000000 (0)	UDINT	4.0	36.0	Input
Enable Counters	X	0x00000000 (0)	UDINT	4.0	40.0	Input
Preset Counter 1	X	0x00000000 (0)	UDINT	4.0	44.0	Input
Preset Counter 2	X	0x00000000 (0)	UDINT	4.0	48.0	Input
Preset Counter 3	X	0x00000000 (0)	UDINT	4.0	52.0	Input
Preset Counter 4	X	0x00000000 (0)	UDINT	4.0	56.0	Input
Preset Counter 5	X	0x00000000 (0)	UDINT	4.0	60.0	Input
Max val counter 1	X	0x00000000 (0)	UDINT	4.0	64.0	Input
Max val counter 2	X	0x00000000 (0)	UDINT	4.0	68.0	Input
Max val counter 3	X	0x00000000 (0)	UDINT	4.0	72.0	Input
Max val counter 4	X	0x00000000 (0)	UDINT	4.0	76.0	Input
Max val counter 5	X	0x00000000 (0)	UDINT	4.0	80.0	Input
Brake Enabled	X	0x00000000 (0)	DWORD	4.0	84.0	Input
PLC AND Mask	X	0x00000000 (0)	DWORD	4.0	88.0	Input
PLC OR Mask	X	0x00000000 (0)	DWORD	4.0	92.0	Input

- **ToolSafe:** nicht implementiert
- **ReverseMode:** Das Nockenschaltwerk verwaltet zwei Nockentabellen (Standard und Reverse). Für jede Nocke kann parametrisiert werden in welcher Tabelle sie aktiv wird. Die SPS kann über die Variable *ReverseMode* festlegen welche der beiden Nockentabellen verwendet wird.
- **Coupled:** Das *Coupled* Signal muss von der SPS auf TRUE gesetzt werden, um die Überwachungsfunktionen des Nockenschaltwerk zu aktivieren. Es zeigt an, dass der Encoder mechanisch eingekoppelt ist und dass somit die Überwachungsfunktionen aktiv sind. Weiterhin wird die aktuelle Drehzahl *Actual HPM* erst dann berechnet und angezeigt, wenn *Coupled* TRUE ist. Anderenfalls wird *Actual HPM* mit dem Wert aus *HPM Preset* initialisiert.
- **HPM Preset:** Drehzahl-Sollwert bei konstanter Encoderdrehzahl nach dem Hochlauf. Dieser Wert wird insbesondere für die Berechnung der Zeitnocken benötigt und muss vor dem Start durch die SPS gesetzt werden.  
Wichtig: *HPM Preset* wird in [1/min] angegeben, wenn der *Encoder-Parameter* [▶ 12] *Use Encoder Precision* FALSE ist. Falls die erhöhte Encoder-Präzision aktiviert wurde, ist die Einheit [1/1000 min]

- **HPM Difference:** Maximale Drehzahlabweichung [1/min] vom Sollwert *HPM Preset*. Eine Abweichung führt zu einer Neuberechnung der Nockentabelle. Ist *HPM Difference* gleich 0, so wird keine automatische Neuberechnung ausgeführt.
- **Reset I/O Errors:** Über diese Variable kann die SPS einen vom Nockenschaltwerk angezeigten I/O-Fehler zurücksetzen. Dazu wird der Wert 1 und im nächsten Zyklus wieder 0 in die Variable geschrieben.
- **Reset Encoder Error:** Über diese Variable kann die SPS einen vom Nockenschaltwerk angezeigten Encoder-Fehler zurücksetzen. Dazu wird der Wert 1 und im nächsten Zyklus wieder 0 in die Variable geschrieben.
- **Reset Shaft Error:** Über diese Variable kann die SPS einen vom Nockenschaltwerk angezeigten Wellenbruch-Fehler (*Shaft Error*) zurücksetzen. Dazu wird der Wert 1 und im nächsten Zyklus wieder 0 in die Variable geschrieben.
- **Reset HPM Error:** Über diese Variable kann die SPS einen vom Nockenschaltwerk angezeigten Hubzahlüberwachungs-Fehler (HPM Error) zurücksetzen. Dazu wird der Wert 1 und im nächsten Zyklus wieder 0 in die Variable geschrieben.
- **Enable Counters:** Über die Bits 0..4 des Datenwortes wird einer der fünf Zähler aktiviert (siehe Ausgangs-Prozessabbild).
- **Preset Counter 1:** Anfangswert für Zähler 1.
- **Preset Counter 2:** Anfangswert für Zähler 2.
- **Preset Counter 3:** Anfangswert für Zähler 3.
- **Preset Counter 4:** Anfangswert für Zähler 4.
- **Preset Counter 5:** Anfangswert für Zähler 5.
- **Max val counter 1:** Maximalwert für Zähler 1. Nach diesem Wert läuft der Zähler auf 0 über.
- **Max val counter 2:** Maximalwert für Zähler 2. Nach diesem Wert läuft der Zähler auf 0 über.
- **Max val counter 3:** Maximalwert für Zähler 3. Nach diesem Wert läuft der Zähler auf 0 über.
- **Max val counter 4:** Maximalwert für Zähler 4. Nach diesem Wert läuft der Zähler auf 0 über.
- **Max val counter 5:** Maximalwert für Zähler 5. Nach diesem Wert läuft der Zähler auf 0 über.
- **Brake Enabled:** Bremsfreigabe. Das Datenwort ist in 32 Einzelvariablen für die ersten 32 Ausgangspuren aufgelöst. Die Variablen setzen die Bremsfreigabe für die Ausgangspuren 0 bis 31. Die Bremsfreigabe wird in Verbindung mit Bremsnocken verwendet. Für den Fall dass mehr als 32 Ausgänge konfiguriert wurden, kommen weitere Datenwörter (*Brake Enabled 1..9*) hinzu.
- **PLC AND Mask:** Über die UND-Maske kann die SPS einzelne Ausgangsspuren deaktivieren. Eine Spur kann nur dann aktiv werden, wenn das zugehörige Bit (0..31) den Wert 1 hat, anderenfalls ist der Ausgang dauerhaft inaktiv. Die Ausgangsspuren können durch ein SPS-Programm aktiviert werden, sobald der Nockenencoder seine konstante Drehzahl HPMpreset erreicht hat. Während der Hochlauf- oder Bremsphase sollte das SPS-Programm die Maske auf 0 setzen und damit die Ausgangsspuren deaktivieren. Die AND Maske hat Vorrang vor der OR Maske. Für den Fall dass mehr als 32 Ausgänge konfiguriert wurden, kommen weitere Datenwörter (*PLC AND Mask 1..9*) hinzu.
- **PLC OR Mask:** Über die ODER-Maske kann die SPS einzelne Ausgangsspuren explizit aktivieren. Eine Spur wird dauerhaft aktiv, wenn das zugehörige Bit (0..31) den Wert 1 hat. Die AND Maske hat Vorrang vor der OR Maske. Für den Fall dass mehr als 32 Ausgänge konfiguriert wurden, kommen weitere Datenwörter (*PLC OR Mask 1..9*) hinzu.



Ausgangs-Prozessabbild

Name	Online	Type	Size	>Addr...	In/Out
Actual HPM	X 0x00000000 (0)	UDINT	4.0	8.0	Output
Cams in use	X 0x00000000 (0)	UDINT	4.0	12.0	Output
I/O Errors	X 0x00000000 (0)	UDINT	4.0	16.0	Output
Encoder Error	X 0x00000000 (0)	UDINT	4.0	20.0	Output
Shaft Error	X 0x00000000 (0)	UDINT	4.0	24.0	Output
HPM Error	X 0x00000000 (0)	UDINT	4.0	28.0	Output
Timeout Rec. 1	X 0x00000000 (0)	UDINT	4.0	32.0	Output
Timeout Rec. 2	X 0x00000000 (0)	UDINT	4.0	36.0	Output
Counter 1	X 0x00000000 (0)	UDINT	4.0	40.0	Output
Counter 2	X 0x00000000 (0)	UDINT	4.0	44.0	Output
Counter 3	X 0x00000000 (0)	UDINT	4.0	48.0	Output
Counter 4	X 0x00000000 (0)	UDINT	4.0	52.0	Output
Counter 5	X 0x00000000 (0)	UDINT	4.0	56.0	Output
Counter ready	X 0x00000000 (0)	DWORD	4.0	60.0	Output
Cam Inputs	X 0x00000000 (0)	DWORD	4.0	64.0	Output
Cam Output	X 0x00000000 (0)	DWORD	4.0	68.0	Output

Ready Local (172.16.5.189.1.1) RTime 1%

- **Actual HPM:** Aktuelle Drehzahl des Nockenencoders. Die aktuelle Drehzahl wird erst dann berechnet und angezeigt, wenn *Coupled TRUE* ist. Anderenfalls wird *Actual HPM* mit dem Wert aus *HPM Preset* initialisiert.  
Wichtig: *Actual HPM* wird in [1/min] angegeben, wenn der *Encoder-Parameter*  $\triangleright$  *12|Use Encoder Precision FALSE* ist. Falls die erhöhte Encoder-Präzision aktiviert wurde, ist die Einheit [1/1000 min]
- **Cams in use:** nicht implementiert
- **I/O Errors:** Diese Variable wird gesetzt, wenn das Nockenschaltwerk einen I/O-Fehler feststellt.
- **Encoder Error:** Diese Variable wird gesetzt, wenn das Nockenschaltwerk einen Encoder-Zählfehler feststellt. (Der Encoderwert ändert sich nicht)
- **HPM Error:** Diese Variable wird gesetzt, wenn das Nockenschaltwerk eine abweichende Encoder-Drehzahl feststellt.
- **Timeout Rec. 1:** Nur für Sonderkonfiguration mit dedizierter Lightbus-I/O-Karte. In diesem Fall wird hier ein Diagnosewert der Karte signalisiert.
- **Timeout Rec. 2:** Nur für Sonderkonfiguration mit dedizierter Lightbus-I/O-Karte. In diesem Fall wird hier ein Diagnosewert der Karte signalisiert.
- **Counter 1:** Zählwert von Zähler 1. Der Wert wird mit einem von der SPS vorgegebenen Anfangswert initialisiert und mit jeder Encoderumdrehung inkrementiert solange das entsprechende Bit der Variablen *Enable Counters* im Eingangs-Prozessabbild 1 ist.
- **Counter 2:** Zählwert von Zähler 2.
- **Counter 3:** Zählwert von Zähler 3.
- **Counter 4:** Zählwert von Zähler 4.
- **Counter 5:** Zählwert von Zähler 5.
- **Counter ready:** Das Datenwort ist in einzelne Bits aufgeschlüsselt. Die Bits 0 bis 4 signalisieren jeweils, dass der Maximalwert für Zähler 1 bis 5 erreicht wurde.
- **Cam Inputs:** Nur für Sonderkonfiguration mit dedizierter Lightbus-I/O-Karte. In diesem Fall werden hier die digitalen Eingänge der angeschlossenen digitalen I/O-Box signalisiert.

- **Cam Output:** Aktueller Zustand der Ausgangsspuren. Das Datenwort ist in 32 Einzel-Bits aufgeschlüsselt, die mit den angeschlossenen digitalen Ausgängen verknüpft werden können. Falls mehr als 32 Ausgänge konfiguriert wurden, kommen weitere Datenwörter (*Cam Output 1..9*) hinzu.

## 7 Programmierung des Nockenschaltwerks durch die SPS

Um die Funktion des Nockenschaltwerks sicherzustellen, muss das SPS-Programm das Nockenschaltwerk und einzelne Ausgangsspuren freischalten. Weiterhin müssen Fehler überwacht und gegebenenfalls zurückgesetzt werden. Optional können auch Nockenparameter, wie Schaltpositionen, durch das SPS-Programm geändert werden. Zu diesem Zweck steht eine SPS-Bibliothek zur Verfügung.

<https://infosys.beckhoff.com/content/1031/tcdigitalcamserver/Resources/10985945739.zip>

### Definition von Variablen für das Prozessabbild

Um das [Prozessabbild \[► 19\]](#) des Nockenschaltwerks für das SPS-Programm verfügbar zu machen, werden im Programm Ein- und Ausgangs-Variablen gleichen Typs angelegt.

```
VAR_GLOBAL
  (*Digital Cam Server Input Process Image - PLC --> DigitalCamServer *)
  ReverseMode      AT %Q* : UDINT;
  Coupled          AT %Q* : UDINT;
  HPMpreset       AT %Q* : UDINT;
  HPMdifference    AT %Q* : UDINT;
  ResetIOErrors   AT %Q* : UDINT;
  ResetEncoderError AT %Q* : UDINT;
  ResetShaftError AT %Q* : UDINT;
  ResetHPMError   AT %Q* : UDINT;
  BrakeEnabled    AT %Q* : DWORD;
  PlcAndMask      AT %Q* : DWORD;
  PlcOrMask       AT %Q* : DWORD;

  (* Digital Cam Server Output Process Image - DigitalCamServer --> PLC *)
  ActualHPM       AT %I* : UDINT;
  IOerrors        AT %I* : UDINT;
  EncoderError    AT %I* : UDINT;
  ShaftError      AT %I* : UDINT;
  HPMError        AT %I* : UDINT;
  CamInputs       AT %I* : DWORD;
  CamOutput       AT %I* : DWORD;
END_VAR
```

Das hier beschriebene [Prozessabbild \[► 19\]](#) entspricht einer Konfiguration mit einer Ausgangsgruppe, also mit 32 digitalen Ausgängen. Falls mehr als eine Ausgangsgruppe (bis zu 10) angelegt werden, wird das Prozessabbild erweitert und es kommen einige Variablen hinzu. In diesem Fall sollten die Variablen *BrakeEnabled*, *PlcAndMask*, *PlcOrMask* und *CamOutput* als ARRAY angelegt werden.

Nachdem das SPS-Programm kompiliert wurde und in den TwinCAT System Manager importiert wurde, können diese Variablen mit den entsprechenden Variablen des Prozessabbildes des digitalen Nockenschaltwerks verknüpft werden. Das SPS-Programm hat dadurch Zugriff auf den Inhalt dieser Variablen.

### Programmierung des Nockenschaltwerks über das Prozessabbild

Um das Nockenschaltwerk zu aktivieren, sind einige Mindestschritte im SPS-Programm notwendig:

1. Die gewünschte Söldrehzahl des Nockenschaltwerkes wird in die Variable *HPMpresest* eingetragen (U/min). Diese Drehzahl dient als Grundlage für die Nockenberechnung und für Überwachungsfunktionen und sollte der tatsächlichen Drehzahl des Nockenencoders entsprechen.  
Wichtig: *HPM Preset* wird in [1/min] angegeben, wenn der [Encoder-Parameter \[► 12\]](#) *Use Encoder Precision* FALSE ist. Falls die erhöhte Encoder-Präzision aktiviert wurde, ist die Einheit [1/1000 min]
2. Die maximale Drehzahlabweichung wird in die Variable *HPMdifference* eingetragen (z. B. 1 U/min). Dieser Wert wird für Überwachungsfunktionen und zur [Neuberechnung \[► 29\]](#) der Nockenschaltpunkte benötigt.
3. Ausgangsspuren müssen über die Variable *PlcAndMask* aktiviert werden sobald der Nockenencoder seine konstante Drehzahl erreicht hat. Im einfachsten Fall wird die Variable *PlcAndMask* auf den Wert 16#FFFFFF gesetzt, um alle 32 Ausgangsspuren für das Nockenschaltwerk frei zu schalten. (Bei einer Konfiguration mit mehr als einer Ausgangsgruppe gibt es bis zu 10 *PlcAndMask*-Variablen (ARRAY) über die bis zu 320 Ausgänge aktiviert werden können.)

4. Falls Bremsnocken verwendet werden, werden die zugehörigen Ausgangsspuren durch Setzen der entsprechenden Bits der *BreakEnabled* Variablen aktiviert (ähnlich wie bei der *PlcAndMask* ).
5. Sobald der Encoder seine konstante Solldrehzahl erreicht hat, wird die Variable *Coupled* auf 1 gesetzt und somit die Überwachungsfunktionen des Nockenschaltwerks freigeschaltet.

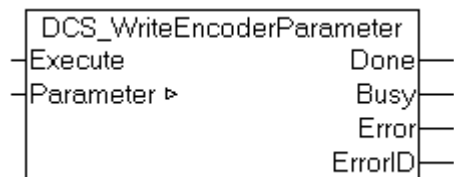
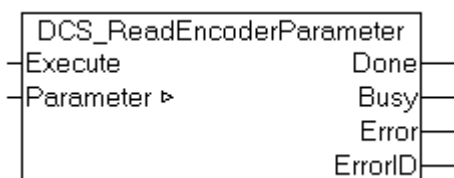
### SPS-Bibliothek TcDigitalCamServer.lib

Die SPS Bibliothek TcDigitalCamServer.lib enthält Funktionsbausteine um das Nockenschaltwerk durch die SPS zu konfigurieren.

<https://infosys.beckhoff.com/content/1031/tcdigitalcamserver/Resources/10985945739.zip>

### Lesen und Schreiben der Encoder-Konfiguration

Encoderparameter können mit jeweils einem Funktionsbaustein gelesen oder geschrieben werden. Um einzelne Parameter, wie z. B. den Encoder-Offset zu ändern, sollte der Encoderdatensatz zunächst gelesen werden und nach Ändern eines Wertes wieder zurückgeschrieben werden.



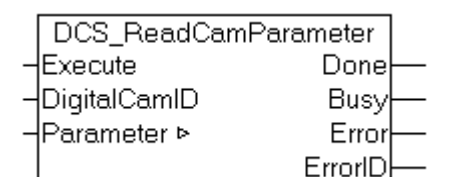
Die Parameter-Datenstruktur ist in der Bibliothek definiert.

```

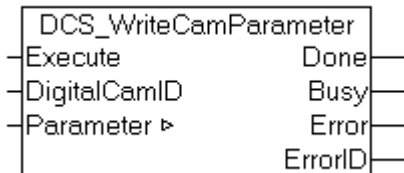
TYPE ST_DigitalCamEncoder :
STRUCT
  EncoderID      : UDINT; (* unique ID of the cam encoder *)
  EncoderType   : UDINT; (* type of the cam encoder *)
  EncoderName   : STRING(79);
  EncoderRevDir : UDINT; (* encoder uses inverted counting direction when TRUE *)
  EncoderBoxAddr : UDINT;
  EncoderZeroOffset : UDINT; (* zero offset of the encoder position in steps of [1/1000 degrees]
*)
  EncoderMask   : DWORD; (* encoder bit mask defines the maximum counter value passed to the
control system *)
END_STRUCT
END_TYPE
    
```

### Lesen und Schreiben von Nockenparametern

Der Parameter-Datensatz jeder Nocke kann gelesen und geschrieben werden. Auch hier sollte zum Ändern eines Parameters, wie z. B. einer Schaltposition, der gesamte Nockendatensatz gelesen und nach der Änderung wieder zurückgeschrieben werden.







Die Auflösung der Positionsdaten in der Parameter-Datenstruktur hängt von der parametrisierten Encoder-Auflösung ab. Falls der Parameter *Use Encoder Precision* in der Encoder- Konfiguration [► 12] gesetzt ist, werden die Positionsdaten in Einheiten zu [1/1000 Grad] angegeben; anderenfalls in Einheiten zu [1 Grad].

Die Parameter-Datenstruktur ist in der Bibliothek definiert./p>

```

TYPE ST_DigitalCamStruct :
STRUCT
  CamID          : UDINT; (* unique cam ID from 1 to 2048 *)
  CamType        : UDINT; (* type E_DigitalCamType: 1=distance cam, 2=time cam, 3=brake
cam *)
  CamLine        : UDINT; (* number of cam output 0 to 319 that is cam is active on *)
  PresetTime     : UDINT; (* preset time in steps of [1/10 ms] *)
  Counter        : UDINT; (* activation counter: 0=inactive, 1=every cycle, 2=every 2nd
cycle, etc. *)
  HysteresisInc  : UDINT; (* not supported *)
  StandardActive : UDINT; (* this cam is active when the standard cam table is activated
by the PLC *)
  ReverseActive  : UDINT; (* this cam is active when the reverse cam table is activated
by the PLC *)
  DynamicEnabled : UDINT; (* preset time is used when TRUE *)
  HysteresisEnabled : UDINT; (* not supported *)
  StartPosition : UDINT; (* start position of the cam in steps of [1/1000 or 1 degrees]
*)
  EndPositionOrDuration : UDINT; (* end position of distance cams in steps of [1/1000 or 1
degrees]
or duration of time cams in steps of [1/10 ms] *)
END_STRUCT
END_TYPE

```

Der Nockentyp *CamType* kann mit folgenden Werten belegt werden:

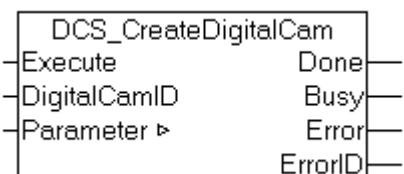
```

TYPE E_DigitalCamType :
(
  CAMTYPE_NONE, (* undefined cam type *)
  CAMTYPE_DISTANCE,
  CAMTYPE_TIME,
  CAMTYPE_BREAK
);
END_TYPE

```

**Nocken anlegen und löschen**

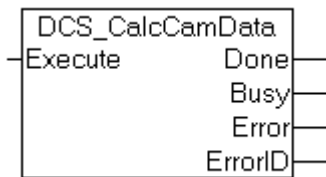
Üblicherweise werden Nocken im TwinCAT System Manager konfiguriert. Das SPS-Programm sollte dann nur noch die Parametrierung der Nocken ändern. Im Einzelfall kann es notwendig sein, neue Nocken durch das SPS-Programm anzulegen bzw. vollständig zu löschen. Die hier verwendete Parameter-Datenstruktur entspricht der beim Lesen und Schreiben der Nockendaten.





### Neuberechnen der Nockentabelle

Das digitale Nockenschaltwerk rechnet alle Nocken-Parameter in eine Nockentabelle zur schnellen Ausgabe um. Diese Nockentabelle wird für eine bestimmte vom Anwender vorgegebene Geschwindigkeit berechnet und nur bei Abweichungen von dieser Geschwindigkeit neu berechnet. Über den Funktionsbaustein DCS\_CalcCamData kann der Anwender eine Neuberechnung der Nockentabelle gezielt anstoßen und damit die Berechnung für ein bestimmtes Geschwindigkeitsniveau sicherstellen.

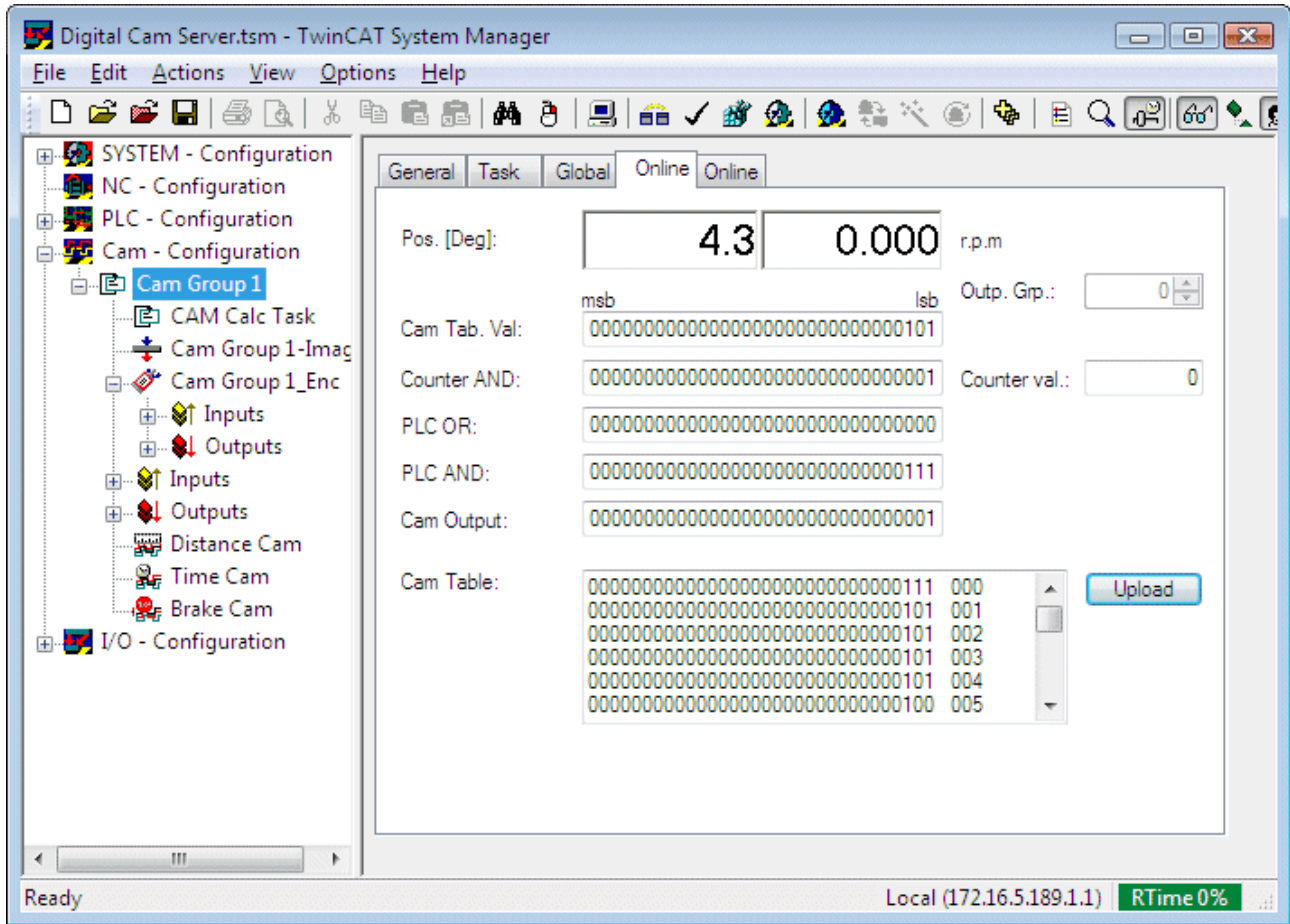


### Ausgangssignale der Funktionsbausteine

Alle beschriebenen Funktionsbausteine werden durch eine steigende Flanke am Eingang *Execute* getriggert. Anschließend wechselt der Ausgang *Busy* auf TRUE um anzuzeigen, dass die Funktion ausgeführt wird. Der Vorgang ist beendet, sobald *Busy* wieder FALSE ist. Gleichzeitig zeigt der Ausgang *Done* an, dass die Funktion erfolgreich abgeschlossen wurde. Anderenfalls wird der Ausgang *Error* TRUE um einen Fehler mit der Fehlernummer in *ErrorID* zu beschreiben. *Busy*, *Done* und *Error* sind exklusiv, es kann also nur eines der Signale gleichzeitig TRUE sein.

# 8 Diagnose des Nockenschaltwerks

Das Nockenschaltwerk berechnet aus allen konfigurierten Nocken eine Nockentabelle. Diese Nockentabelle kann zur Diagnose über den Online-Dialog des Nockenschaltwerks angezeigt werden.



### Diagnoseanzeigen im Online Dialog

Im oberen Teil des Dialog-Fensters wird die aktuelle Position des Nockenencoders in Grad und die aktuelle Encoderdrehzahl in [1/min] angezeigt. Darunter sind mehrere 32 Bit Binärwerte zu sehen, die folgendermaßen zu interpretieren sind. Bei nur einer Ausgangsgruppe mit 32 Spuren entspricht jedes Bit einem digitalen Ausgang. Das niederwertigste Bit ganz rechts ist dem Ausgang 0 zuzuordnen und das höherwertigste dem Ausgang 31. Sollte mehr als eine Ausgangsgruppe konfiguriert worden sein, so kann über das Dialogfeld *Output Group (Outp. Grp.)* die Gruppennummer eingestellt werden. Es kann also immer eine Gruppe zu 32 Ausgängen betrachtet werden. Die angezeigten Daten haben im Einzelnen folgende Bedeutung:

**Cam Tab. Val:** Die aktuelle Zeile aus der Nockentabelle, die dem angezeigten aktuellen Winkel entspricht.

**Counter AND:** Nocken können in jeder Umdrehung oder nur in jeder n-ten Umdrehung aktiv werden. Die Counter Maske zeigt an, dass ein Ausgang am angezeigten Winkel aktiv werden kann.

**PLC OR:** Aktuelle ODER Maske, die durch die SPS bestimmt wird (*PlcOrMask*).

**PLC AND:** Aktuelle UND Maske, die durch die SPS bestimmt wird (*PlcAndMask*).

**Cam Output:** Aktuell ausgegebener Datensatz für 32 Ausgangsspuren. Die Ausgabe entsteht durch Verknüpfen der Nockentabelle mit den beschriebenen Masken.

**Cam Table:** Gesamte Nockentabelle mit jeweils einer Zeile zu 32 Ausgängen für jede aufgelöste Winkelposition (zweite Spalte rechts neben dem 32-Bit Wert). Die Tabelle ist abhängig von der parametrisierten Auflösung 360 Einträge lang oder sie enthält die Anzahl Zeilen, die der Encoder-Auflösung entspricht (z. B. 4096). Die Nockentabelle enthält die aus allen Nocken berechneten Rohdaten für die Ausgabe der Nockenspuren. Berechnungsgrundlage ist die eingestellte Sollgeschwindigkeit *HPM Preset*.



Im Gegensatz zu den anderen hier beschriebenen Werten wird das Anzeigefeld Cam Table nicht selbständig aktualisiert, sondern nur durch manuelles Drücken des Upload Buttons. Die Nockentabelle kann sich nach Geschwindigkeitsänderungen ändern.

---

## 9 Arbeitsweise des Nockenschaltwerkes

Zum besseren Verständnis der Arbeitsweise des Nockenschaltwerkes, ist es wichtig zu wissen, wie die konfigurierten Nocken für eine möglichst schnelle Bearbeitung aufbereitet werden. Das Nockenschaltwerk rechnet die Nockeninformation aller Nocken in eine Nockentabelle um. Diese Nockentabelle enthält dann den Zustand aller Ausgänge an einem bestimmten Drehwinkel. Der Vorteil dieser Vorgehensweise liegt in einer sehr schnellen Nockenausgabe, da das Schalten der Nockenspuren von der aufwändigen Berechnung der Nocken inklusive Totzeitkompensation getrennt wird. Die Nockentabelle wird somit für eine vorher bestimmte Solldrehzahl korrekt berechnet. Bei wechselnden Geschwindigkeiten findet zwar eine Anpassung statt, die exakten Schaltpunkte können aber nur für konstante Drehzahl garantiert werden.

### Nockentabellen

Das Nockenschaltwerk legt intern eine Nockentabelle mit einer festen Zeilenzahl an. Die Zeilenzahl wird durch die konfigurierte Auflösung bestimmt. Somit besteht die Tabelle entweder aus 360 Zeilen oder die Anzahl entspricht der Anzahl Encoder-Inkremente (z. B. 4096). Im zweiten Fall berechnet sich der einer Zeile zugeordnete Winkel zu

$$\text{Winkel} = \text{Zeilennummer} / \text{Anzahl Encoder-Inkremente} * 360^\circ$$

z. B Winkel für Tabellenzeile 1000:  $1000 / 4096 * 360^\circ = 87.891^\circ$

In jeder Tabellenzeile wird der Zustand aller Nockenspuren (digitale Ausgänge) für den entsprechenden Encoder-Winkel abgelegt. Der Zustand ergibt sich aus allen konfigurierten Nocken, das heißt, aus Schaltpunkten, Schaltzeiten und Totzeiten.

Die Nockentabelle wird zunächst bei Systemstart einmal ausgeführt und wird später unter bestimmten Bedingungen, wie z. B. nach einer größeren Geschwindigkeitsänderung, neu berechnet. Eine Neuberechnung findet dabei immer bei 0 Grad statt, damit die Nockenausgabe innerhalb einer Umdrehung des Encoders konsistent ist.

Das Nockenschaltwerk verwaltet zwei umschaltbare Nockentabellen (*Variable Reverse Mode*). Jede Nocke kann für eine oder für beide Tabellen konfiguriert werden (Parameter *Standard Active* bzw. *Reverse Active*). Die Umschaltung zwischen beiden Tabellen wird durch die SPS veranlasst.

### Abbildung einer Wegnocke in der Nockentabelle

Eine Wegnocke ist zwischen einer Einschaltposition und einer Ausschaltposition definiert. In der Nockentabelle wird in allen Tabellenzeilen zwischen diesen beiden Positionen eine 1 eingetragen. Alle anderen Tabellenzeilen bleiben 0, wenn nicht weitere Nocken dieselbe Nockenspur ansprechen.

### Abbildung einer Zeitnocke in der Nockentabelle

Eine Zeitnocke ist ab einer Einschaltposition mit einer bestimmten Einschaltdauer definiert. In der Nockentabelle werden alle Tabellenzeilen ab der Einschaltposition bis zu einer um ein Grad größeren Position auf 1 gesetzt. Alle anderen Einträge sind 0. Obwohl die Zeitnocke wie eine Wegnocke mit einem 1 Grad breiten Schaltwinkel in die Nockentabelle eingetragen wird, ist die Ausschaltposition nicht von Bedeutung. Die Nockenausgabe wird tatsächlich zeitbasiert abgewickelt: Der Nockenausgang wird zunächst an der Einschaltposition eingeschaltet. Anschließend läuft im Nockenschaltwerk ein Timer ab, der den Ausgang nach Ablauf der definierten Schaltzeit ausschaltet.

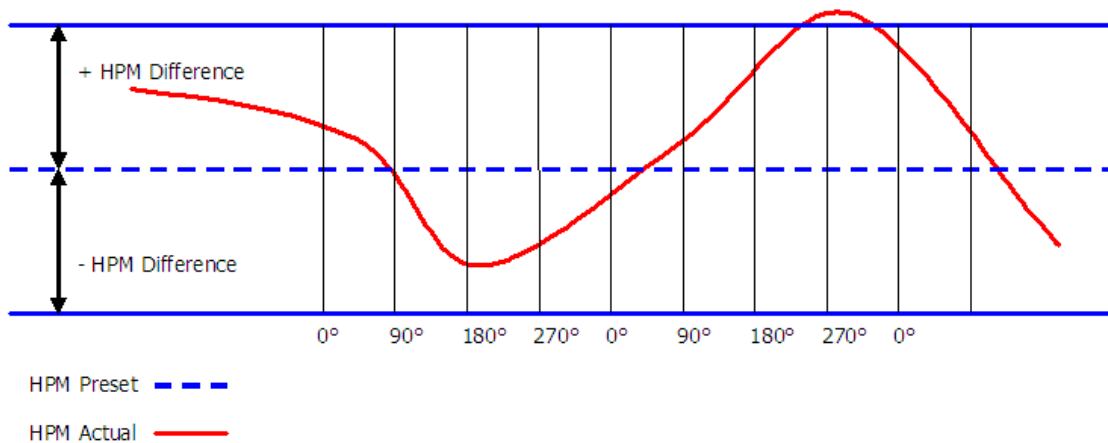
### Abbildung einer Bremsnocke in der Nockentabelle

Eine Bremsnocke ist mit einer bestimmten Bremsposition definiert. In der Nockentabelle werden alle Tabellenzeilen ab der Bremsposition bis zu einer um ein Grad größeren Position auf 0 gesetzt. Alle anderen Einträge sind 1. Obwohl die Bremsnocke wie eine Wegnocke mit einem 359 Grad breiten Schaltwinkel in die Nockentabelle eingetragen wird, ist die Einschaltposition nicht von Bedeutung. Die Nockenausgabe ist eine Sonderfunktion, die den Ausgang solange eingeschaltet lässt, wie die SPS die Bremsfreigabe auf 1 gesetzt hat. Sobald die Bremsfreigabe durch die SPS zurückgenommen wird, wird die Ausgangsspur an der definierten Bremsposition ausgeschaltet und bleibt ausgeschaltet.

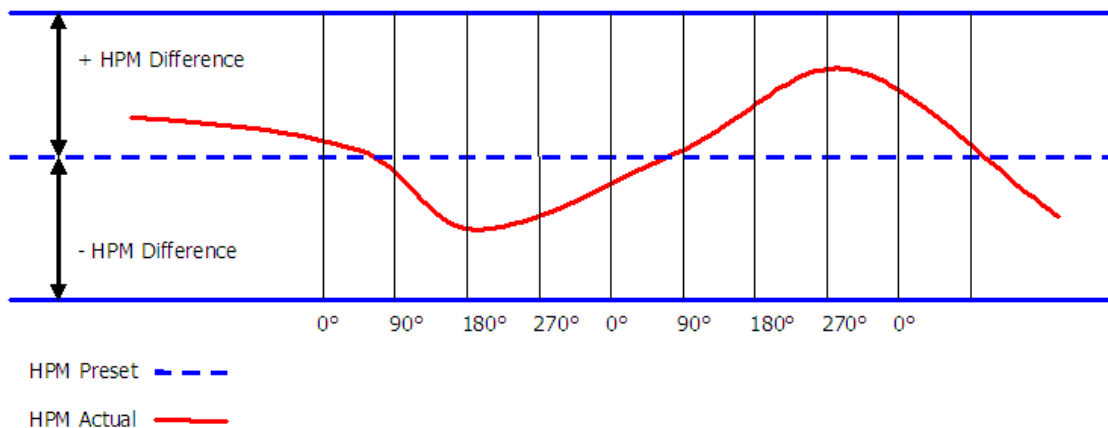
## Neuberechnung der Nockentabellen

Berechnungsgrundlage für die Nockentabelle ist die vom Anwender (SPS) angegebene Sollgeschwindigkeit des Nockenschaltwerkes (*HPM Preset* [1/min]). Solange das *Coupled* Flag von der SPS noch nicht gesetzt ist, wird die Nockentabelle auf Basis dieser Geschwindigkeit berechnet.

Der Anwender gibt ebenfalls eine Geschwindigkeitstoleranz an (*HPM Difference* [1/min]). Das Nockenschaltwerk prüft Abweichungen von der Sollgeschwindigkeit, sobald das *Coupled* Flag von der SPS gesetzt wurde. Wenn dann die aktuelle Drehzahl von der jeweils letzten Geschwindigkeit, mit der die Nockentabelle berechnet wurde, um den Differenz-Betrag abweicht, wird die Tabelle neu berechnet. Wird *HPM Difference* mit 0 angegeben, findet keine Neuberechnung statt.



Bei Geschwindigkeitsschwankungen innerhalb des Bandes  $\pm$  *HPM Difference* wird die Nockentabelle nicht sofort neu berechnet. *HPM Preset* sollte der mittleren zu erwartenden Drehzahl entsprechen und *HPM Difference* sollte so eingestellt werden, dass kleine zu erwartende Geschwindigkeitsschwankungen nicht zur Neuberechnung der Nockentabelle führen. Abweichungen innerhalb des Bandes  $\pm$  *HPM Difference* werden mit der Zeit aufintegriert, sodass auch bei einer kleinen mittleren Abweichung die Nockentabelle nach einiger Zeit neu berechnet und an die mittlere Geschwindigkeit angepasst wird.



Einzelne Nockenparameter, z. B. die Schaltpositionen, können durch das SPS-Programm geändert werden. Nach einer Änderung wird die Nockentabelle aktualisiert, ohne sie vollständig neu zu berechnen.

Die Neuberechnung der Nockentabelle kann optional vom SPS-Programm über einen Funktionsbaustein [► 23] angestoßen werden. So ist es z. B. sinnvoll, die Neuberechnung nach einer Geschwindigkeitsänderung anzustoßen, sobald das neue konstante Geschwindigkeitsniveau erreicht ist.

Um die Konsistenz der Nockenschaltpunkte zu gewährleisten, wird eine neu berechnete Nockentabelle jeweils erst beim nächsten Encoder-Nulldurchgang aktiv, nachdem die Berechnung abgeschlossen ist.



Mehr Informationen:  
**[www.beckhoff.de/ts5800](http://www.beckhoff.de/ts5800)**

Beckhoff Automation GmbH & Co. KG  
Hülshorstweg 20  
33415 Verl  
Deutschland  
Telefon: +49 5246 9630  
[info@beckhoff.de](mailto:info@beckhoff.de)  
[www.beckhoff.de](http://www.beckhoff.de)

