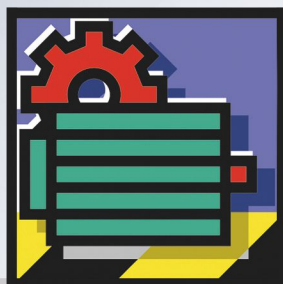


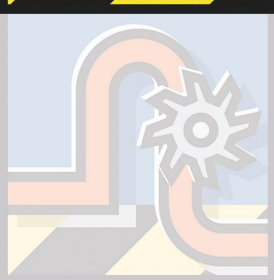
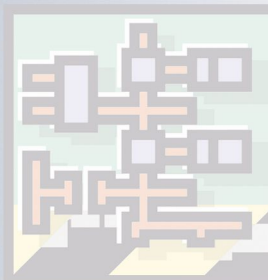
Handbuch | DE

# NC Flying Saw

TwinCAT 2



TwinCAT Supplement





# 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>Allgemein</b>	<b>8</b>
<b>3</b>	<b>Interfaces</b>	<b>10</b>
<b>4</b>	<b>Ablauf</b>	<b>11</b>
<b>5</b>	<b>Synchronbedingung</b>	<b>13</b>
<b>6</b>	<b>Stopp</b>	<b>15</b>
<b>7</b>	<b>Positionskompensation</b>	<b>17</b>
<b>8</b>	<b>Fliegende Diagonal-Säge</b>	<b>19</b>
<b>9</b>	<b>Fliegende Säge und fliegende Diagonal-Säge auf Geschwindigkeit</b>	<b>20</b>
<b>10</b>	<b>Fliegende Säge mit Online Umwandlung in Master PTP Achse</b>	<b>21</b>
<b>11</b>	<b>PLC-API</b>	<b>22</b>
11.1	Klassische Fliegende Säge	22
11.1.1	FlyingSawCoupleOnPositon	22
11.1.2	FlyingSawCoupleOnVelocity	23
11.1.3	FlyingSawStop	24
11.1.4	Get_TcNcFlyingSaw_Version	24
11.2	PLCopen Funktionsbausteine	25
11.2.1	MC_GearInPos	25
11.2.2	MC_GearInVelo	27
11.2.3	MC_ReadFlyingSawCharacteristics	28
11.2.4	Datenstruktur ST_SyncMode	29



# 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.



##### **Tipps oder Fingerzeige**

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 Allgemein

Die **Fliegende Säge (Parallel-Slave-Achse)** ist eine Slave-Achse, die auf eine fahrende Master-Achse an einer bestimmten Position – parallel zur Master-Achse fahrend - geschwindigkeitsmäßig auf die Mastergeschwindigkeit aufsynchronisiert wird. Unterstellt ist dabei, dass bei Kopplung der Slave-Achse an die Master-Achse die Master-Achse mit (konstanter) Sollgeschwindigkeit fährt und dass die Slave-Achse in ihrer Grundposition (Koppelposition) steht. Die Slave-Achse startet automatisch, wenn die Master-Achse die dynamische Koppelposition erreicht hat. Verwirklicht ist das Ganze teils als Streckensteuerung (Slave-Beschleunigungs-Phase (ruckbegrenzt) und Slave Parallel-Fahrt), sowie teils zeitgesteuert (Slave-Verzögerungs-Phase (ruckbegrenzt) und Slave Stillstand).

Die **Fliegende Diagonal-Säge(Diagonal-Slave-Achse)** ist eine Slave-Achse, die schräg (mit konstantem Winkel) zu einer Master-Achse auf die fahrende Master-Achse an einer gegebenen Position so geschwindigkeitsmäßig aufsynchronisiert, dass die Geschwindigkeitskomponente der Slave-Achse in Richtung der Mastergeschwindigkeit nach der Synchronisation gleich der Mastergeschwindigkeit ist.

### Interfaces

[Interfaces \[► 10\]](#)

Die Interfaces bestehen aus dem System Manager, SPS-Bausteinen und dem zyklischen NC-SPS-Achs-Interface.

### Ablauf

[Ablauf \[► 11\]](#)

Der Ablauf besteht aus den Phasen

1. Logisches Ankoppeln an die Master-Achse (die Slave-Achse bleibt stehen).
2. Dynamisches Ankoppeln an die Master-Achse: Beschleunigungsphase.
3. Parallelfahrt zur Master-Achse.
4. Stop: Dynamisches Abkoppeln von der Master-Achse: Bremsphase.
5. Stillstand.
6. Logisches Abkoppeln.

### Synchronbedingung

[Synchronbedingung \[► 13\]](#) Die Synchronbedingung des ruckbegrenzten/linearen Beschleunigungsalgorithmus impliziert eine **Zwangsbedingung zwischen den Koppelpositionen** von Master und Slave sowie den **Synchronpositionen** von Master und Slave.

### Start und Stop

[Start und Stop \[► 15\]](#)

Bei Achsstart wird nach Erreichen der dynamischen Koppelposition in jedem Aufruf der Satzausführungstask mittels der der Startparameter (Mastersynchronposition und Slavesynchronposition) und der generellen Dynamikparameter ein ruckbegrenzttes Beschleunigungsprofil mit der aktuellen Mastergeschwindigkeit als Sollgeschwindigkeit generiert, das streckengesteuert zur Sollwertgenerierung der Slave-Achse dient.

Bei Stopp wird mit den generellen Dynamikdaten ein zeitgesteuertes Bremsprofil berechnet, die Slave-Achse dynamisch vom Master abgekoppelt und Über das Bremsprofil abgebremst.

### Positions-Kompensation

[Positions-Kompensation \[► 17\]](#)

On-line Positionskorrektur der Fliegenden Säge, bei der auf einer vorgegebenen Strecke die Slave-Achse um eine bestimmte Strecke mehr oder weniger verfahren wird, als der Kopplung an die Master-Achse entspricht. Die Dynamik der Kompensation muss parametrisiert werden. Ist die gewünschte Kompensation mit



den gegebenen Kompensationsstartdaten nicht durchführbar, dann wird automatisch die maximal mögliche Kompensation durchgeführt. Die Kompensation kann über Stopp der Kompensation vorzeitig beendet werden. Die Kompensation sollte nur in der Synchronphase eingesetzt werden.

### ⚠ WARNUNG

Falls die Kompensationsdifferenz  $D$  negativ ist (Verzögern) sein, ist eine (doppelte) Bewegungsumkehr möglich.

## Fliegende Diagonal-Säge

[Fliegende Diagonal-Säge \[► 19\]](#)

Ist  $v$  die aktuelle Sollgeschwindigkeit der Parallelslave-Achse, dann ist die Geschwindigkeit der Diagonalslave-Achse  $v/\sin(\theta)$ , wobei  $\theta$ ,  $0.0^\circ < \theta \leq 90.0^\circ$ , den Komplementärwinkel zwischen Masterrichtung und Slaverichtung bezeichnet.

## Fliegende Säge und fliegende Diagonal-Säge auf Geschwindigkeit

[Fliegende Säge und fliegende Diagonal-Säge auf Geschwindigkeit \[► 20\]](#)

Die **Fliegende Säge auf Geschwindigkeit (Parallel-Slave-Achse)** ist eine Slave-Achse, die auf eine fahrende Master-Achse, parallel zur Master-Achse fahrend, maximal schnell geschwindigkeitsmäßig auf die Mastergeschwindigkeit aufsynchronisiert wird. Unterstellt ist dabei, dass bei Kopplung der Slave-Achse an die Master-Achse die Master-Achse mit (konstanter) Sollgeschwindigkeit fährt und dass die Slave-Achse in ihrer Grundposition (Koppelposition) steht. Verwirklicht ist das Ganze teils als Streckensteuerung (Slave-Beschleunigungs-Phase (ruckbegrenzt) und Slave Parallel-Fahrt), sowie teils zeitgesteuert (Slave-Verzögerungs-Phase (ruckbegrenzt) und Slave Stillstand).

Die **Fliegende Diagonal-Säge auf Geschwindigkeit (Diagonal-Slave-Achse)** ist eine Slave-Achse, die schräg (mit konstantem Winkel) zu einer Master-Achse auf die fahrende Master-Achse so schnell wie möglich so geschwindigkeitsmäßig aufsynchronisiert, dass die Geschwindigkeitskomponente der Slave-Achse in Richtung der Mastergeschwindigkeit nach der Synchronisation gleich der Mastergeschwindigkeit ist.

## Fliegende Säge mit Online Umwandlung in Master PTP Achse

[Fliegende Säge mit Online Umwandlung in Master PTP Achse \[► 21\]](#)

Dieser vielseitigste Slave-Typ wird als Fliegende Säge gestartet und koppelt am Synchronpunkt auf die Geschwindigkeit der Masterachse auf, kann dann in der Synchronphase per Positionskompensation erneut ortsmäßig synchronisiert werden und schließlich in der Synchronphase on-line vom Master abgekoppelt werden und selber in eine PTP Masterachse umgewandelt werden, die dann entweder kräftefrei gemacht wird, oder auf ein bei bei der Ankoppelinstruktion oder der Abkoppelinstruktion mitgegebenes Geschwindigkeitsniveau als Endlosachse weiterfährt und letztendlich per Instruktion 'Neue Endposition' auf ihre endgültige Zielposition fährt.

## 3 Interfaces

Der Benutzer kann die fliegende Säge über den System Manager oder SPS-Bausteine bedienen. Außerdem steht ihm (im System Manager und über die SPS) ein zyklisches NC-SPS Achs-Interface zur Verfügung, das das **Achs-Prozessabbild** beinhaltet und Zugriff auf diverse Variablen ermöglicht.

### **PLC-Document**

Es gibt eine besondere SPS-Dokumentation zur fliegenden Säge.

### **System Manager**

Der System Manager bietet den Zugriff auf alle Funktionalitäten der fliegenden Säge:

- Logisches Koppeln mit Angabe der Masterachse und Setzen der Synchronpositionen.
- Stopp.

### **SPS-Bausteine**

Die SPS Library bietet den Zugriff auf alle Slave-Achs Funktionalitäten :

- Logisches Koppeln mit Angabe der Masterachse und Setzen der Synchronpositionen.
- Stopp.

### **NC-SPS-Interface**

Das zyklische NC-SPS Achs-Interface beinhaltet das Achs-Prozessabbild und insbesondere Zugriff auf die Koppelinformation (Koppelflag).

## 4 Ablauf

Der Ablauf besteht aus den Phasen

1. Logisches Ankoppeln an die Master-Achse (die Slave-Achse bleibt stehen).
2. Dynamisches Ankoppeln an die Master-Achse: Beschleunigungsphase.
3. Parallelfahrt zur Master-Achse.
4. Stop: Dynamisches Abkoppeln von der Master-Achse: Bremsphase.
5. Stillstand
6. Logisches Abkoppeln.

### Logisches Ankoppeln

Durch das (logische) Ankoppeln werden der Slave-Achse fast alle Funktionalitäten gesperrt. Die Achse steht in Grundposition und wartet darauf, dass die Master-Achse die dynamische Koppelposition überfährt.

### Dynamisches Ankoppeln

Wenn die Master-Achse die dynamische Koppelposition überfährt startet die Slave-Achse **automatisch**. In jedem Aufruf der Satzausführungstask wird ein volles ruckbegrenzttes Beschleunigungsprofil von Geschwindigkeit  $0.0$  auf die **aktuelle Mastergeschwindigkeit** berechnet. Der jeweils aktuelle Wert wird über eine Streckensteuerung als Sollwert der Slave-Achse übergeben.

### Beschleunigungsphase

Die Beschleunigungsphase ist als **Streckensteuerung** ausgelegt. D.h. die Slave-Achse richtet sich sowohl geschwindigkeitsmäßig als auch positionsmäßig nach den aktuellen Daten der Master-Achse und folgt damit allen Geschwindigkeitsschwankungen und Bewegungsinversionen der Master-Achse. Wenn die Master-Achse ihre Synchronposition überfährt, hat auch die Slave-Achse ihre Synchronposition und damit die aktuelle Mastergeschwindigkeit erreicht und tritt in die Parallelphase ein.

### Parallelphase

In der Parallelphase ist die fliegende Säge wie im **Zustand einer linearen Slave-Achse** affin (mit Koppelfaktor  $1.0$ ) an den Master gekoppelt.

### Dynamisches Abkoppeln

Um die Parallelfahrt zu beenden wird der Slave vom Benutzer gestoppt. Dazu wird von dem aktuellen dynamischen Zustand (Beschleunigung, Geschwindigkeit) ein ruckbegrenzttes zeitgesteuertes Dynamikprofil auf Geschwindigkeit  $0.0$  erzeugt, der Slave dynamisch vom Master entkoppelt und auf dem berechneten Dynamikprofil abgebremst.

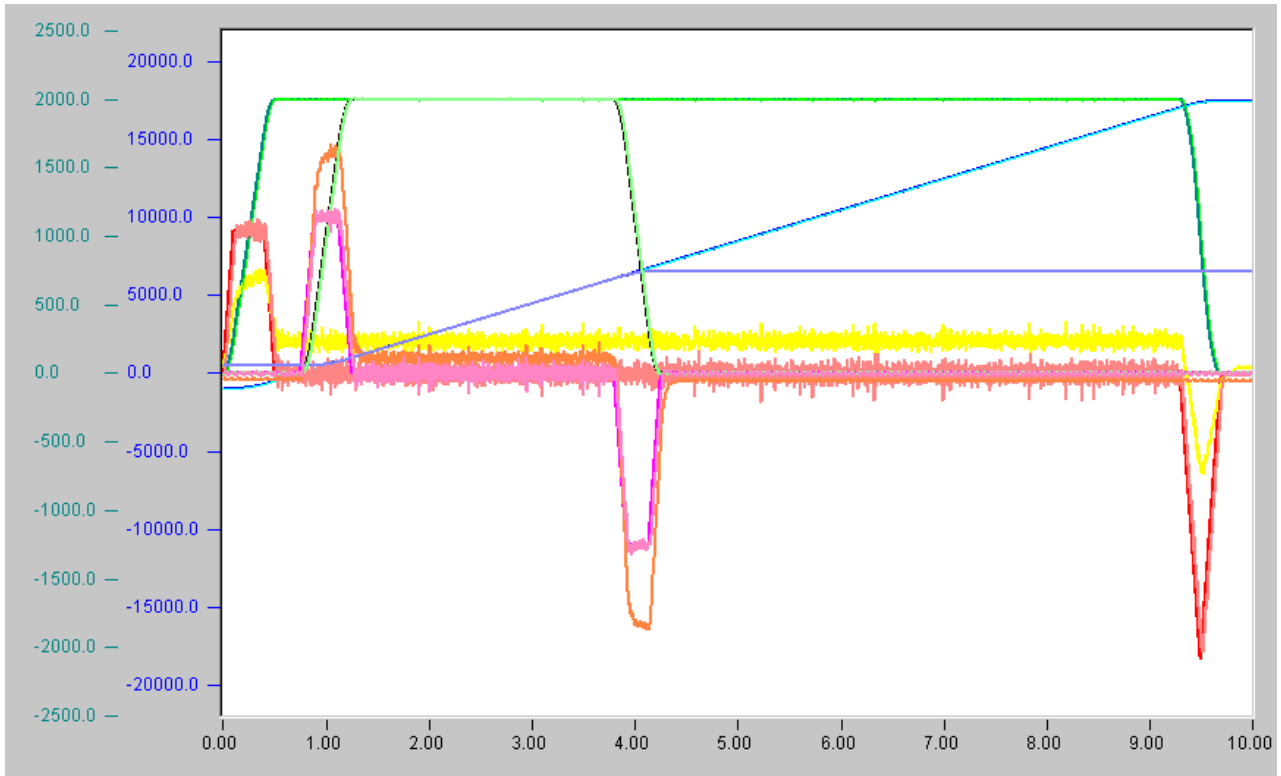
### Logisches Abkoppeln

Die Position an der der Slave zum Stillstand kommt, hängt neben der Geschwindigkeit wesentlich vom Beschleunigungszustand (im Augenblick des Aufrufs der Stop-Funktion) der Achse ab. Nach Stillstand kann die Slave-Achse von der Master-Achse abgekoppelt werden.

Das logische Abkoppeln ist jederzeit möglich, jedoch führt das Abkoppeln in Fahrt zu einem Schleppabstand.

### Rücklaufsperr

In den streckengesteuerten Phasen (Beschleunigung, Parallelfahrt) ist die Slave-Achse so mit der Master-Achse synchronisiert, dass sie jeden Bewegungsrücklauf mitmacht. Um zu verhindern, dass bei einem weiten Rücklauf die Slave-Achse über ihre Grundposition (Halterung) zurückfährt gibt es eine Rücklaufsperr, die den Rücklauf über die Grundposition hinaus verhindert. Diese Sperr wirkt instantan (Schleppabstand). Wenn der Master in der ursprünglichen Richtung wieder über seine alte Koppelposition fährt, läuft der Slave automatisch instantan wieder an.



Ablauf: Masterstart, Slavestart, Aufsynchronisieren, Synchronphase, Slavestopp, Masterstopp.

## 5 Synchronbedingung

Die der fliegenden Säge zugrundeliegende Achs-Synchronisierung hängt entscheidend vom Beschleunigungsalgorithmus der Slave-Achse ab. Erläutert wird hier die Achs-Synchronbedingung für den ruckbegrenzten 3-Phasen Algorithmus. Der Synchronbedingung liegen folgende Achsvariablen zugrunde (zu beachten ist dabei, dass Master und Slave bei gleichen Einheiten i.A. **verschiedene Koordinatensysteme** haben):

1. Slave-Synchronposition,
2. Slave-Grundposition = logische Slave-Koppelposition = dynamische Slave-Koppelposition,
3. Master-Synchronposition,
4. logische Master-Koppelposition (Position der Master-Achse bei Master-Slave-Kopplung),
5. dynamische Master-Koppelposition (Position der Master-Achse bei dynamischem Slave-Start).

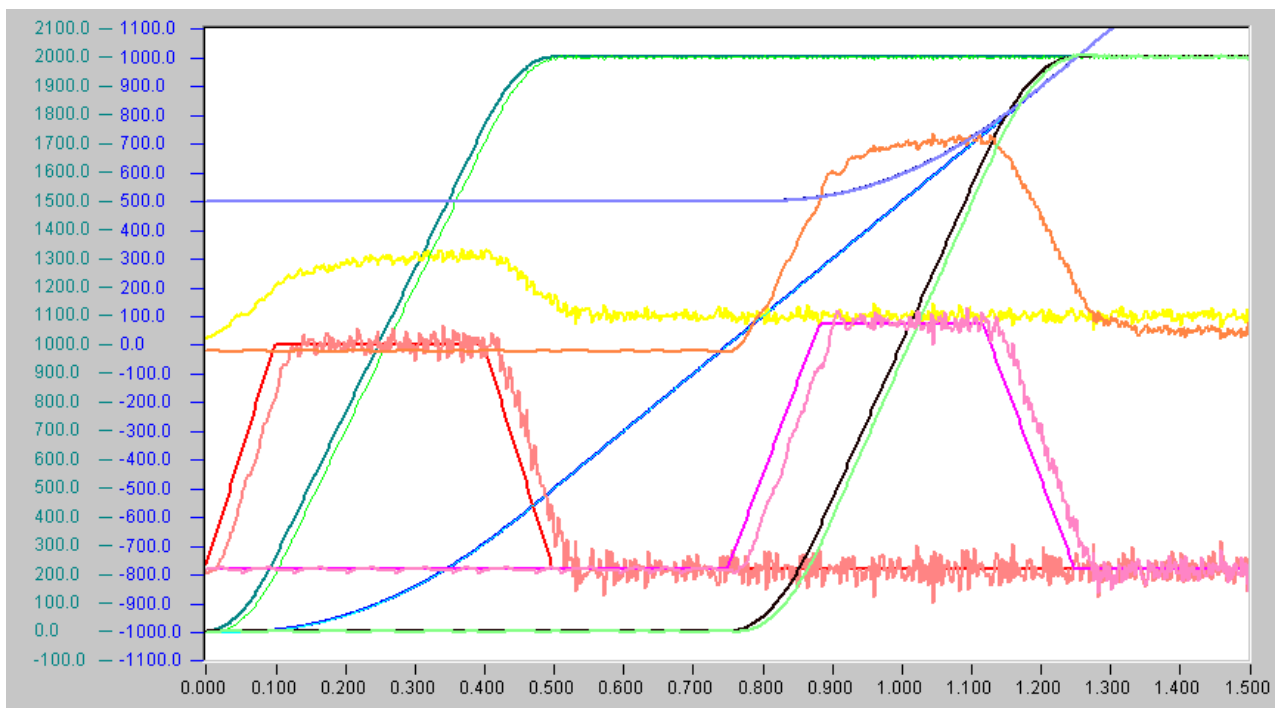
### Synchronbedingung des ruckbegrenzten Beschleunigungsalgorithmus

Die Slave-Achse braucht vom Anfangs-Zustand (Grundposition, Geschwindigkeit 0.0, Beschleunigung 0.0) zum End-Zustand (Synchronposition, Mastergeschwindigkeit, Masterbeschleunigung) eine bestimmte Zeit, in der die Master-Achse die Strecke von der dynamischen Master-Koppelposition bis zur Master-Synchronposition verfährt. D.h. von den vier Positionen (Slave-Synchronposition, Slave-Koppelposition, Master-Synchronposition, dynamische Master-Koppelposition) sind drei unabhängig und damit die vierte gegeben. Bei den zugrundeliegenden Algorithmen gilt für den Zusammenhang der Positionen:

**$|\text{dynamischen Master-Koppelposition} - \text{Master-Synchronposition}| = 2 \times |\text{Slave-Koppelposition} - \text{Slave-Synchronposition}|$ .**

Daraus folgt, dass der Abstand der dynamischen Master-Koppelposition bis zur Master-Synchronposition **größer** als zweimal dem Abstand Slave-Koppelposition - Slave-Synchronposition sein muss. Bei Start werden die beiden Synchronpositionen explizit mitgegeben, die Slave-Koppelposition ist intern bekannt, so dass daraus die dynamische Master-Koppelposition berechnet werden kann. Ist die Synchronisierung nicht durchführbar, weil der Master schon zu nah an der Koppelposition steht, wird das Koppeln mit Fehler abgebrochen.

Synchronbedingung der Fliegenden Diagonalsäge [\[► 19\]](#).



Synchronbedingung:  $|0.0 - 1000.0| = 2.0 \times |5000.0 - 1000.0|$

Masterstart bei -1000.0 mm, Masterkoppelposition-logisch -200.0 mm, Masterkoppelposition-dynamisch 0.0 mm, Mastersynchronposition 1000.0 mm.

Slavewarteposition = Slavekoppelposition-logisch = Slavekoppelposition-dynamisch 500.0 mm,  
Slavesynchronposition = 1000.0 mm.

Es reicht aus wenn die Masterkoppelposition-logisch echt kleiner als die Masterkoppelposition-dynamisch ist.

## 6 Stopp

Nach Erreichen der dynamischen Koppelposition wird in jedem Aufruf der Satzausführungstask mittels der der Startparameter (Mastersynchronposition und Slavesynchronposition) und der generellen Dynamikparameter ein ruckbegrenztes Beschleunigungsprofil mit der aktuellen Mastergeschwindigkeit als Sollgeschwindigkeit generiert das streckengesteuert zur Sollwertgenerierung der Slave-Achse dient.

Bei Stopp wird mit den generellen Dynamikdaten ein zeitgesteuertes Bremsprofil berechnet, die Slave-Achse dynamisch vom Master abgekoppelt und über das Bremsprofil abgebremst.

### Koppelparameter

$P_{ms}$  (Master-Synchronposition im Master-Koordinatensystem)>

$P_{ss}$  (Slave-Synchronposition im Slave-Koordinatensystem)

$\theta$  Diagonalwinkel

cf Koppelfaktor zur Mastergeschwindigkeit. Der Koppelfaktor darf die Richtung der Achse nicht umkehren.

interne Parameter: ( $P_{sc}$  Slave-Koppelposition,  $P_{mcd}$  dynamische Master-Koppelposition)

Es gibt **keinen expliziten Start**. Die Startdaten werden intern berechnet und der Start wird **automatisch** durchgeführt.

Tab. 1: Startparameter

Startparameter	Bedeutung und Randbedingungen
<b>Master-Synchronposition</b> $P_{ms}$	Position an der sich der Master (in seinem Koordinatensystem) befinden soll, wenn der Slave aufsynchronisiert sein soll, $ P_{ms} - P_{mcd}  = 2 \times  P_{ss} - P_{sc} $
<b>Slave-Synchronposition</b> $P_{ss}$	Position an der sich der Slave (in seinem Koordinatensystem) befinden soll, wenn er auf den Master aufsynchronisiert sein soll, $ P_{ms} - P_{mcd}  = 2 \times  P_{ss} - P_{sc} $
<b>Koppelfaktor</b> cf	Koppelfaktor $cf \neq 0.0$
<b>Diagonalwinkel</b> $\theta$	Komplementärwinkel zwischen Masterrichtung und Slaverichtung, $0.0^\circ < \theta \leq 90.0^\circ$ , Parallelslave: $\theta = 90.0^\circ$

### Interfaces

System Manager

SPS-Bausteine

### Stopp

Der Stopp hat **höchste Priorität** und muss in allen Dynamikphasen, bis auf die Stoppphase selber, funktionieren (nicht in der Vorphase und nicht in der Standphase). Der Stopp funktioniert auch während eines eventuellen Rücklaufs. Dem 3-Phasenalgorithmus der Stoppdynamik wird dann eine Zusatzphase vorgeschaltet, wenn die Slave-Achse nicht beschleunigungsfrei ist und damit keine Bewegungsinversion impliziert ist. In dieser Zusatzphase fährt die Slave-Achse von der aktuellen Beschleunigung auf Beschleunigung 0.0.



Der Stopp der fliegenden Säge darf zeitlich nicht länger andauern als die Positionierung der Masterachse ansonsten wird der Stopp abgebrochen.

Es gibt **keine expliziten Stoppparameter**. Diese (Verzögerung, Ruck) werden den globalen Dynamikdaten der Achse entnommen.

**Interfaces**

System Manager

SPS-Bausteine



## 7 Positionskompensation

On-line Positionskorrektur der Fliegenden Säge, bei der auf einer vorgegebenen Strecke die Slave-Achse um eine bestimmte Strecke mehr oder weniger verfahren wird, als der Kopplung an die Master-Achse entspricht. Die Dynamik der Kompensation muss parametrisiert werden. Ist die gewünschte Kompensation mit den gegebenen Kompensationsstartdaten nicht durchführbar, dann wird automatisch die maximal mögliche Kompensation durchgeführt. Die Kompensation kann über Stopp der Kompensation vorzeitig beendet werden. Die Kompensation sollte nur in der Synchronphase eingesetzt werden.

### ⚠️ WARNUNG

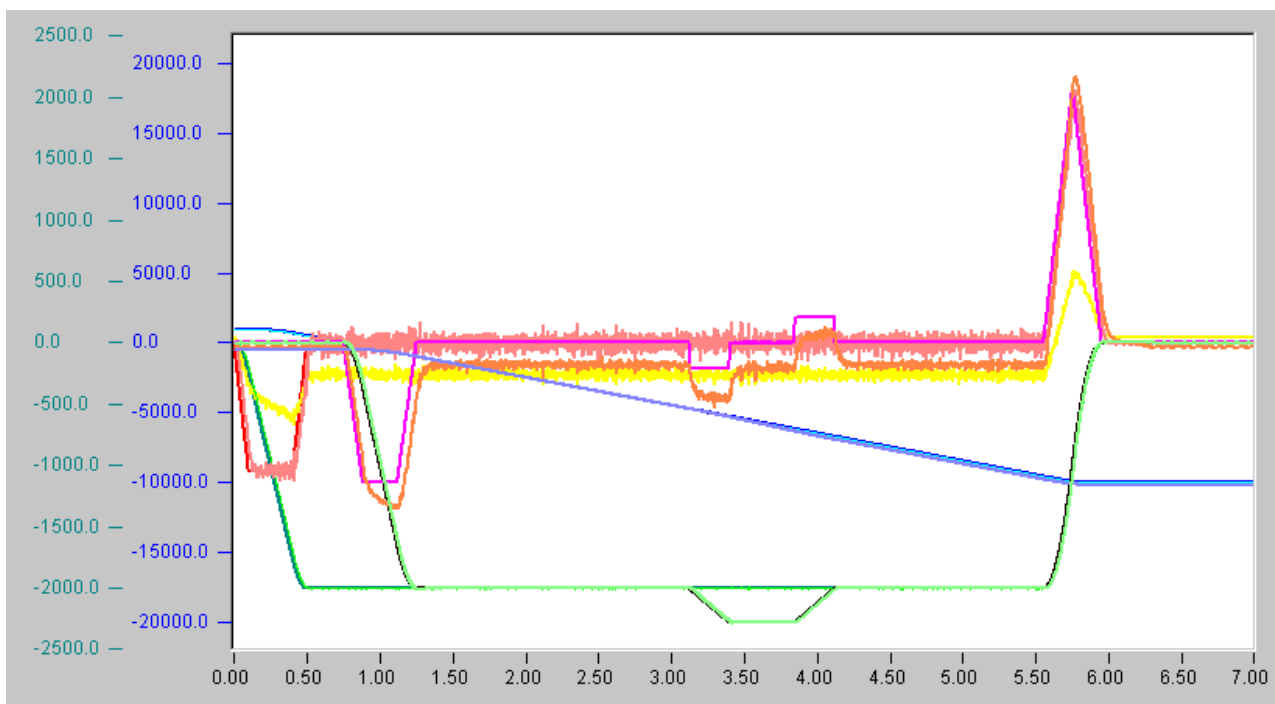
Falls die Kompensationsdifferenz D negativ ist (Verzögern) sein, ist eine (doppelte) Bewegungsumkehr möglich.



Die Positionskompensation von Slave-Achsen ist eine Variante der Positionskompensation von linearen Slave-Achsen.

### Besonderheiten

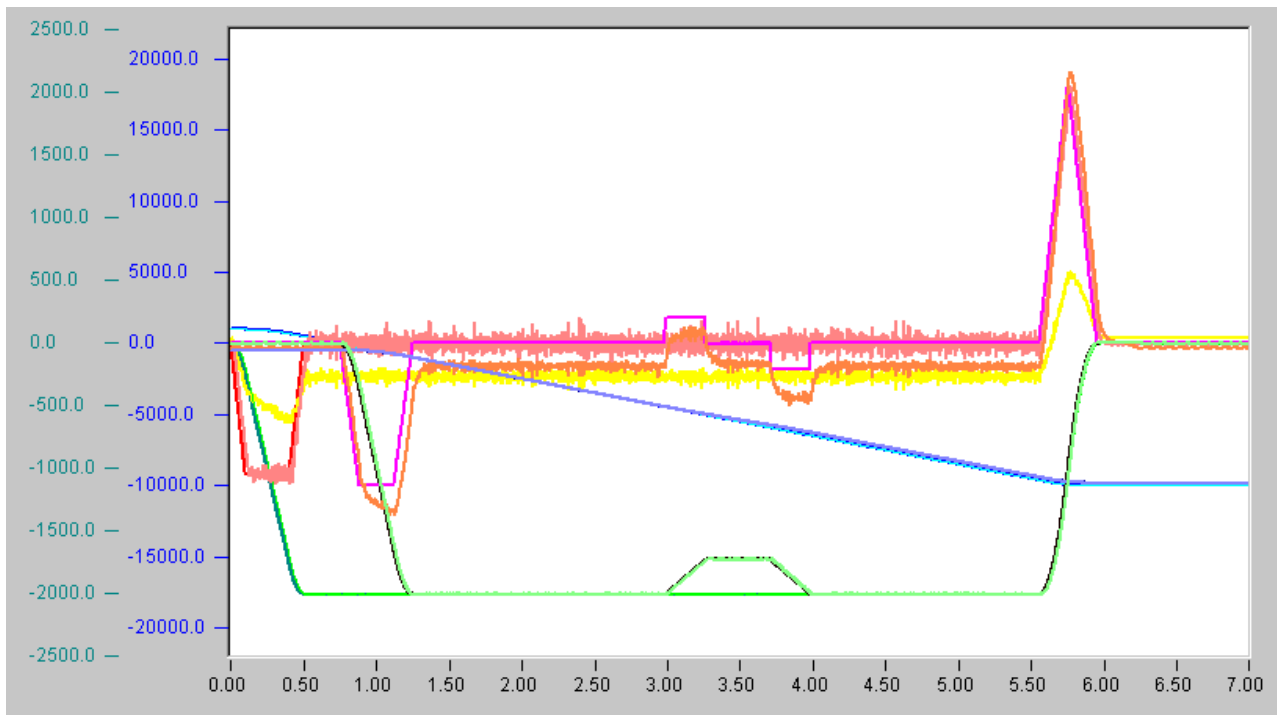
1. Die Kompensation sollte erst nach erfolgter Synchronisation (also nach Ende der Beschleunigungsphase) erfolgen.
2. Die Kompensation sollte vor Einleitung der Bremsphase beendet sein.
3. Die Kompensation berücksichtigt die Positionsdehnung bei der Diagonalsäge.



Masterdaten: Start von 1000.0 auf -10,000.0 mit Synchronposition -1000.0

Slavestartposition -500.0 mit Slavesynchronposition -1000.0

Kompensationsdaten:  $A_{\pm} = 1000.0$  mm/sec/sec,  $V = 500.0$  mm/sec,  $VP = 2000.0$  mm/sec,  
Kompensationsdifferenz  $\pm 200.0$  mm, Kompensationslänge 2000.0 mm.



## 8 Fliegende Diagonal-Säge

### Übersicht

Der **fliegende Diagonalsäge (Diagonal-Slave)** ist eine Slave-Achse, die schräg (mit konstantem Diagonalwinkel  $\theta$ ) zu einer Master-Achse auf die fahrende Master-Achse so geschwindigkeitsmäßig aufsynchronisiert, dass die Geschwindigkeitskomponente der Slave-Achse in Richtung der Mastergeschwindigkeit gleich der Mastergeschwindigkeit ist.

### Geometrie: Diagonalwinkel $\theta$

Sei  $v$  die aktuelle Sollgeschwindigkeit einer Parallelslave-Achse, dann ist die Geschwindigkeit der Diagonalslave-Achse  $v/\sin(\theta)$ , wobei  $\theta$ ,  $0.0^\circ < \theta \leq 90.0^\circ$ , den **Komplementärwinkel zwischen Masterrichtung und Slaverichtung** bezeichnet. In der Synchronphase ist daher die Slave-Achse mit dem Koppelfaktor  $1/\sin(\theta)$  an die Master-Achse gekoppelt.

Diagonalsäge: Geometrie

### Synchronbedingung des ruckbegrenzten/linearen diagonalen Beschleunigungsalgorithmus

Die Diagonal-Slave-Achse braucht vom Anfangs-Zustand (Grundposition, Geschwindigkeit  $0.0$ , Beschleunigung  $0.0$ ) zum End-Zustand (Synchronposition, Mastergeschwindigkeit, Masterbeschleunigung) eine bestimmte Zeit in der die Master-Achse die Strecke von der dynamischen Master-Koppelposition bis zur Master-Synchronposition verfährt. Bei den zugrundeliegenden Algorithmen gilt:

**$|\text{Slave-Koppelposition} - \text{Slave-Synchronposition}| = 0.5 \times |\text{dynamischen Master-Koppelposition} - \text{Master-Synchronposition}| / \sin(\theta)$ .**

Daraus folgt, dass der Abstand der dynamischen Master-Koppelposition bis zur Master-Synchronposition **größer** als  $2.0 \times \sin(\theta)$  mal dem Abstand Slave-Koppelposition - Slave-Synchronposition sein muss.

### Dynamik

Die aktuelle Geschwindigkeit der Slave-Achse nach Erreichen der Parallelphase ist gleich  $1.0/\sin(\theta)$  mal der aktuellen Geschwindigkeit der Masterachse. Diese Geschwindigkeit ist größer als die Geschwindigkeit der Master-Achse, da die Slave-Achse neben die Parallel-Komponente (Master-Geschwindigkeit) noch die Orthogonal-Komponente aufbringen muss.

# 9 Fliegende Säge und fliegende Diagonal-Säge auf Geschwindigkeit

## Übersicht

Die **Fliegende Säge auf Geschwindigkeit (Parallel-Slave-Achse)** ist eine Slave-Achse, die auf eine fahrende Master-Achse, parallel zur Master-Achse fahrend, maximal schnell geschwindigkeitsmäßig auf die Mastergeschwindigkeit aufsynchronisiert wird. Unterstellt ist dabei, dass bei Kopplung der Slave-Achse an die Master-Achse die Master-Achse mit (konstanter) Sollgeschwindigkeit fährt und dass die Slave-Achse in ihrer Grundposition (Koppelposition) steht. Verwirklicht ist das Ganze teils als Streckensteuerung (Slave-Beschleunigungs-Phase (ruckbegrenzt) und Slave Parallel-Fahrt), sowie teils zeitgesteuert (Slave-Verzögerungs-Phase (ruckbegrenzt) und Slave Stillstand).

Die **Fliegende Diagonal-Säge auf Geschwindigkeit (Diagonal-Slave-Achse)** ist eine Slave-Achse, die schräg (mit konstantem Winkel) zu einer Master-Achse auf die fahrende Master-Achse so schnell wie möglich so geschwindigkeitsmäßig aufsynchronisiert, dass die Geschwindigkeitskomponente der Slave-Achse in Richtung der Mastergeschwindigkeit nach der Synchronisation gleich der Mastergeschwindigkeit ist.

## Generelles

Bis auf die Art der Aufkopplung ist die Fliegende (Diagonal-) Säge auf Geschwindigkeit algorithmisch identisch mit der fliegenden (Diagonal-) Säge auf Position (und Geschwindigkeit). Die Richtung der Slaveachse ist dabei als Richtung der Masterachse festgelegt. Falls die Masterachse im Koppelzeitpunkt keine Richtung hat (z.B. eine EncoderAchse, die zufällig steht), wird das Koppeln abgebrochen.

## Koppelparameter

$\theta$  Diagonalwinkel

Die weiteren Startdaten werden intern berechnet und der Start wird sofort **automatisch** durchgeführt.

Tab. 2: Startparameter

Startparameter	Bedeutung und Randbedingungen
Diagonalwinkel $\theta$	Komplementärwinkel zwischen Masterrichtung und Slaverichtung, $0.0^\circ < \theta \leq 90.0^\circ$ , Parallelslave: $\theta = 90.0^\circ$

## Interfaces

System Manager

SPS-Bausteine

Die Slave-Achse koppelt fliegend so schnell wie möglich auf Geschwindigkeit auf.

# 10 Fliegende Säge mit Online Umwandlung in Master PTP Achse

Dieser vielseitigste Slave-Typ wird als Fliegende Säge gestartet und koppelt am Synchronpunkt auf die Geschwindigkeit der Masterachse auf, kann dann in der Synchronphase per Positionskompensation erneut ortsmäßig synchronisiert werden und schließlich in der Synchronphase on-line vom Master abgekoppelt werden und selber in eine PTP Masterachse umgewandelt werden, die dann kräftefrei gemacht wird und der dann alle Master-Funktionen zur Verfügung stehen..

## Voraussetzungen

Logische Voraussetzung der Umwandlung ist die Bedingung, dass (i) die fliegende Säge als Parallelslave gestartet worden ist und dass (ii) die aktuelle Richtung der fliegenden Säge mit der logischen Richtung bei Start übereinstimmt. Dynamisch ist es ist am günstigsten die fliegende Säge in der Synchronphase umzuwandeln.

Dynamische Voraussetzung einer Umwandlung ist, dass die neuen Dynamikparameter so beschaffen sind, dass die neue Masterachse jederzeit gestoppt werden kann, sodass i.A. die neuen Dynamikparameter (diejenigen der neuen Masterachse) nicht schwächer sein dürfen als die alten Dynamikparameter (in der Synchronphase: diejenigen der alten Masterachse der fliegenden Säge; ansonsten: diejenigen der alten Masterachse der fliegenden Säge) multipliziert mit dem Koppelfaktor.

Physikalische Voraussetzung einer dynamisch glatten Umwandlung ist, dass die Geschwindigkeit der fliegenden Säge zum Zeitpunkt der Umwandlung nicht allzu sehr fluktuiert.

## Zustand nach der Umwandlung

Nach der Umwandlung ist die fliegende Säge eine PTP-Masterachse mit voller Funktionalität. Die der Achse zum Zeitpunkt der Umwandlung mitgegebene Instruktion impliziert, dass nach der Umwandlung die Achse zuerst kräftefrei gemacht wird (also auf Beschleunigung 0 fährt), sodann die dabei erreichte Geschwindigkeit beibehält, per Definition Override 100% hat, und auf Zielposition  $\pm\infty$  fährt, wobei die Richtung im Zeitpunkt der Umwandlung beibehalten wird.

## Funktionalitäten nach der Umwandlung

Es stehen alle Masterfunktionen zur Verfügung: Override, Stop, Positionskompensation, 'Neue Endposition' und 'Neue Geschwindigkeit und neue Endposition'..

Tab. 3: Startparameter

Startparameter	Bedeutung und Randbedingungen
<b>Master-Synchronposition</b> $P_{ms}$	Position an der sich der Master (in seinem Koordinatensystem) befinden soll, wenn der Slave aufsynchronisiert sein soll, $ P_{ms} - P_{mcd}  = 2 \times  P_{ss} - P_{sc} $
<b>Slave-Synchronposition</b> $P_{ss}$	Position an der sich der Slave (in seinem Koordinatensystem) befinden soll, wenn er auf den Master aufsynchronisiert sein soll, $ P_{ms} - P_{mcd}  = 2 \times  P_{ss} - P_{sc} $
<b>Diagonalwinkel</b> $\theta$	Parallelslave: $\theta = 90.0^\circ$
<b>Koppelfaktor</b> $cf$	Koppelfaktor $cf \neq 0.0$

## Interfaces

System Manager

SPS-Bausteine

# 11 PLC-API

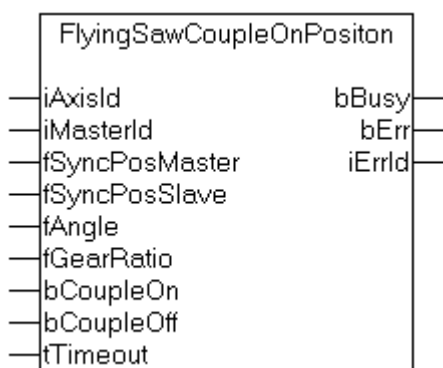
In vielen Anlagen werden Werkstücke während des Transportes bearbeitet. Dazu ist es notwendig, Werkzeuge und Werkstücke bezüglich Position und Geschwindigkeit zu synchronisieren, so dass das Werkzeug wie an einem stillstehenden Werkstück eingesetzt werden kann. Ein Beispiel für eine solche Anwendung ist eine Säge, die während der Fahrt einen Schnitt quer durch das transportierte Material führt (Fliegende Säge). Um diese Art der Anwendungen zu realisieren, stellt TwinCAT die Fliegende Säge zur Verfügung.

Zur einfachen Handhabung der Fliegenden Säge dient die TwinCAT PLC Library **TcNcFlyingSaw.lib**, die als *zusätzliches Produkt* erhältlich ist. Ein Beispielprogramm zum Thema Fliegende Säge verwendet diese Bibliothek.

Neben den klassischen Funktionsbausteinen (Präfix FlyingSaw...), die Beckhoff seit einigen Jahren anbietet, wurde die Bibliothek durch PLCopen konforme Funktionsbausteine (MC\_GearInPos, MC\_GearInVelo) ergänzt. In neuen Projekten sollten nach Möglichkeit diese Bausteine eingesetzt werden. Wegen der Kompatibilität zu bestehenden Projekten, sind die klassischen Bausteine unverändert in der Bibliothek enthalten.

## 11.1 Klassische Fliegende Säge

### 11.1.1 FlyingSawCoupleOnPositon



**FlyingSawCoupleOnPositon** koppelt eine Slave-Achse als Fliegende Säge an eine Master-Achse. Die Synchrongeschwindigkeit wird genau am Synchronpunkt erreicht.

#### Interface

```

VAR_INPUT
  iAxisId      : UINT; (* Flying saw axis ID *)
  iMasterId    : UINT;
  fSyncPosMaster : LREAL; (* absolute master synchronous position *)
  fSyncPosSlave : LREAL; (* absolute slave synchronous position *)
  fAngle       : LREAL; (* parallel = 90 degrees *)
  fGearRatio   : LREAL; (* 0.0 is invalid, then default is 1.0 *)
  bCoupleOn    : BOOL; (* couple *)
  bCoupleOff   : BOOL; (* decouple *)
  tTimeout     : TIME;
END_VAR
VAR_OUTPUT
  bBusy : BOOL;
  bErr  : BOOL;
  iErrId : UDINT;
END_VAR

```

**iAxisId** : Achs-ID der Slave-Achse, die als Fliegende Säge gekoppelt werden soll

**iMasterId** : Achs-ID der Master-Achse an die gekoppelt werden soll

**fSyncPosMaster** : Synchronposition der Master-Achse

**fSyncPosSlave** : Synchronposition der Slave-Achse

**fAngle** : Winkel der Fliegenden Säge (für eine Parallelachse 90°)

**fGearRatio** : Getriebefaktor. Ein Getriebefaktor 0 ist ungültig und wird als 1 interpretiert

**bCoupleOn** : Flankengetriggertes Signal zum Ausführen der Achskopplung

**bCoupleOff** : Flankengetriggertes Signal zum Entkoppeln der Achsen

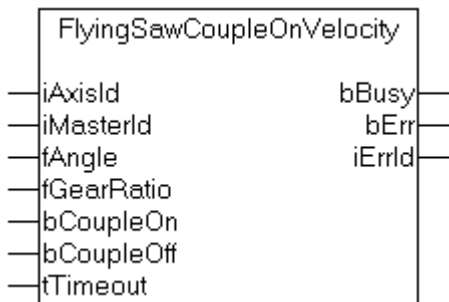
**tTimeout** : ADS-Timeout (ca. 1 Sekunde)

**bBusy** : wird mit steigender Flanke an bCoupleOn oder bCoupleOff TRUE und bleibt TRUE, solange der Baustein das Kommando ausführt

**bErr** : wird TRUE, wenn beim Ausführen des Kommandos ein Fehler aufgetreten ist.

**bErrId** : Fehlernummer (ADS- oder NC-Fehlernummer)

### 11.1.2 FlyingSawCoupleOnVelocity



**FlyingSawCoupleOnVelocity** koppelt eine Slave-Achse als Fliegende Säge an eine Master-Achse. Die Synchrongeschwindigkeit wird so schnell wie möglich erreicht.

#### Interface

```

VAR_INPUT
  iAxisId      : UINT; (* Flying saw axis ID *)
  iMasterId    : UINT;
  fAngle       : LREAL; (* parallel = 90 degrees *)
  fGearRatio   : LREAL; (* 0.0 is invalid, then default is 1.0 *)
  bCoupleOn    : BOOL; (* couple *)
  bCoupleOff   : BOOL; (* decouple *)
  tTimeout     : TIME;
END_VAR
VAR_OUTPUT
  bBusy        : BOOL;
  bErr         : BOOL;
  iErrId       : UDINT;
END_VAR
    
```

**iAxisId** : Achs-ID der Slave-Achse, die als Fliegende Säge gekoppelt werden soll

**iMasterId** : Achs-ID der Master-Achse an die gekoppelt werden soll

**fAngle** : Winkel der Fliegenden Säge (für eine Parallelachse 90°)

**fGearRatio** : Getriebefaktor. Ein Getriebefaktor 0 ist ungültig und wird als 1 interpretiert

**bCoupleOn** : Flankengetriggertes Signal zum Ausführen der Achskopplung

**bCoupleOff** : Flankengetriggertes Signal zum Entkoppeln der Achsen

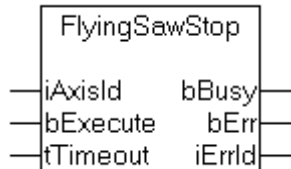
**tTimeout** : ADS-Timeout (ca. 1 Sekunde)

**bBusy** : wird mit steigender Flanke an bCoupleOn oder bCoupleOff TRUE und bleibt TRUE, solange der Baustein das Kommando ausführt

**bErr** : wird TRUE, wenn beim Ausführen des Kommandos ein Fehler aufgetreten ist.

**bErrId** : Fehlernummer (ADS- oder NC-Fehlernummer)

### 11.1.3 FlyingSawStop



**FlyingSawStop** stoppt einen als Fliegende Säge gekoppelten Slave. Der Stopp wird im gekoppelten Zustand ausgeführt und kann daher nicht mit dem Funktionsbaustein AXACT angestoßen werden.

#### Interface

```
VAR_INPUT
  iAxisId : UINT; (* Flying saw axis ID *)
  bExecute : BOOL; (* Stop Flying Saw *)
  tTimeout : TIME;
END_VAR
VAR_OUTPUT
  bBusy : BOOL;
  bErr : BOOL;
  iErrId : UDINT;
END_VAR
```

**iAxisId** : Achs-ID der Slave-Achse, die als Fliegende Säge gekoppelt ist

**iMasterId** : Achs-ID der Master-Achse

**bExecute** : Flankengetriggertes Signal zum Ausführen des Kommandos

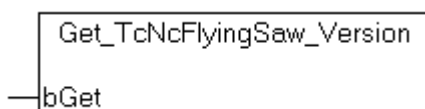
**tTimeout** : ADS-Timeout (ca. 1 Sekunde)

**bBusy** : wird mit steigender Flanke an bExecute TRUE und bleibt TRUE, solange der Baustein das Kommando ausführt

**bErr** : wird TRUE, wenn beim Ausführen des Kommandos ein Fehler aufgetreten ist.

**bErrId** : Fehlernummer (ADS- oder NC-Fehlernummer)

### 11.1.4 Get\_TcNcFlyingSaw\_Version



**Get\_TcNcFlyingSaw\_Version** ermittelt die Versionsnummer der SPS-Bibliothek TcNcFlyingSaw.lib. Der Funktionsaufruf liefert die Versionsnummer in einem String zurück.

#### Interface

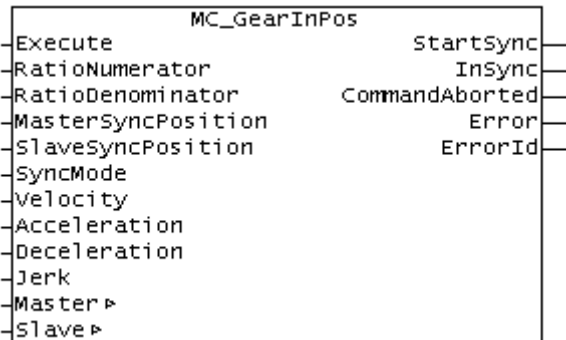
```
FUNCTION Get_TcNcFlyingSaw_Version: STRING(20)
VAR_INPUT
  bGet : BOOL;
END_VAR
```



bGet : Signal zum Ausführen des Kommandos

## 11.2 PLCopen Funktionsbausteine

### 11.2.1 MC\_GearInPos



**MC\_GearInPos** koppelt eine Slave-Achse als Fliegende Säge an eine Master-Achse. Die Synchrongeschwindigkeit wird genau am Synchronpunkt von Master und Slave erreicht.

Das Interface und die Funktion dieses Bausteins ist durch die PLCopen spezifiziert. Der Baustein ist damit als Ergänzung zur PLCopen Motion Control Library zu sehen.

#### VAR\_INPUT

```
VAR_INPUT
  Execute          : BOOL;
  RatioNumerator   : LREAL := 1;
  RatioDenominator : UINT  := 1;
  MasterSyncPosition : LREAL; (* absolute master synchronous position *)
  SlaveSyncPosition : LREAL; (* absolute slave synchronous position *)
  SyncMode         : ST_SyncMode;
  Velocity         : LREAL := 0;
  Acceleration     : LREAL; (* 0 : use default value *)
  Deceleration     : LREAL; (* 0 : use default value *)
  Jerk             : LREAL; (* 0 : use default value *)
END_VAR
```

- Execute :** Mit der steigenden Flanke wird die Synchronisation gestartet.
- RatioNumerator :** Zähler des Getriebefaktors (siehe [Hinweis \[▶ 25\]](#))
- RatioDenominator :** Nenner des Getriebefaktors (siehe [Hinweis \[▶ 25\]](#))
- MasterSyncPosition :** Synchronposition des Masters.
- SlaveSyncPosition :** Synchronposition des Slaves.
- SyncMode :** In der Variablen [SyncMode \[▶ 29\]](#) werden die Randbedingungen für die Synchronisierung festgelegt.
- Velocity :** Maximale Slave-Geschwindigkeit in der Synchronisierungsphase. Wenn keine Geschwindigkeit angegeben wird, wird die Maximalgeschwindigkeit der Achse aus den Systemmanager-Daten verwendet.  
**Die hier angegebene Geschwindigkeit wird nur überprüft, wenn diese Überprüfung mit der Variablen SyncMode aktiviert wird (GearInSync\_CheckMask\_MaxVelo=TRUE).**

<b>Acceleration :</b>	<p>Maximale Slave-Beschleunigung in der Synchronisierungsphase. Wenn keine Beschleunigung angegeben wird, wird die Maximalbeschleunigung der Achse aus den Systemmanager-Daten verwendet.</p> <p><b>Die hier angegebene Beschleunigung wird nur überprüft, wenn diese Überprüfung mit der Variablen SyncMode aktiviert wird (GearInSync_CheckMask_MaxAcc=TRUE).</b></p>
<b>Deceleration :</b>	<p>Maximale Slave-Verzögerung in der Synchronisierungsphase. Wenn keine Verzögerung angegeben wird, wird die maximale Verzögerung der Achse aus den Systemmanager-Daten verwendet.</p> <p><b>Die hier angegebene Verzögerung wird nur überprüft, wenn diese Überprüfung mit der Variablen SyncMode aktiviert wird (GearInSync_CheckMask_MaxDec=TRUE).</b></p>
<b>Jerk :</b>	<p>Maximale Slave-Ruck in der Synchronisierungsphase. Wenn kein Ruck angegeben wird, wird der maximale Ruck der Achse aus den Systemmanager-Daten verwendet.</p> <p><b>Der hier angegebene Ruck wird nur überprüft, wenn diese Überprüfung mit der Variablen SyncMode aktiviert wird (GearInSync_CheckMask_MaxJerk=TRUE).</b></p>

### ● Negative Werte

**i** Bei einem gewünschten Getriebefaktor von 1:3 wird der Zähler zu 1 und der Nenner zu 3 gewählt. Für den Zähler können auch negative Werte angegeben werden.

Die Slave-Achse kann mit dem Funktionsbaustein `MC_GearOut` aus der TcMC Motion Control Library abgekoppelt werden. Wird der Slave während der Fahrt abgekoppelt, so behält er seine Geschwindigkeit bei und kann mit `MC_Stop` angehalten werden.

### VAR\_OUTPUT

```
VAR_OUTPUT
  StartSync      : BOOL;
  InSync         : BOOL;
  CommandAborted : BOOL;
  Error          : BOOL;
  ErrorId        : UDINT;
END_VAR
```

<b>StartSync :</b>	Wird TRUE, sobald die Synchronisierung begonnen hat.
<b>InSync :</b>	Wird TRUE, sobald der Slave mit dem Master synchronisiert ist.
<b>CommandAborted :</b>	Wird TRUE, wenn die Kopplung nicht durchgeführt werden konnte.
<b>Error :</b>	Wird TRUE, sobald ein Fehler eintritt.
<b>ErrorID :</b>	Liefert bei einem gesetzten ERR-Ausgang die Fehlernummer

### VAR\_IN\_OUT

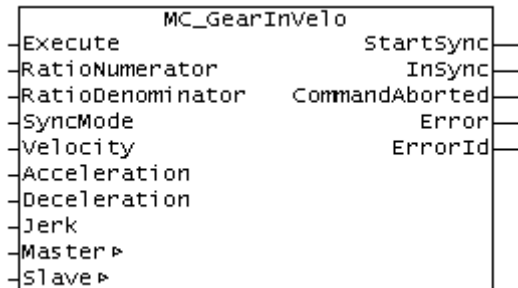
```
VAR_IN_OUT
  Master : NCTOPLC_AXLESTRUCT;
  Slave  : NCTOPLC_AXLESTRUCT;
END_VAR
```

<b>Master :</b>	Achsstruktur des Masters.
<b>Slave :</b>	Achsstruktur des Slaves.

Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken
TwinCAT v2.9	PC (i386)	TcNcFlyingSaw.Lib

11.2.2 MC\_GearInVelo



MC\_GearInVelo koppelt eine Slave-Achse als Fliegende Säge an eine Master-Achse.

Das Interface und die Funktion dieses Bausteins ist durch die PLCopen spezifiziert. Der Baustein ist damit als Ergänzung zur PLCopen Motion Control Library zu sehen.

VAR\_INPUT

```

VAR_INPUT
  Execute          : BOOL;
  RatioNumerator   : LREAL := 1;
  RatioDenominator : UINT  := 1;
  SyncMode         : ST_SyncMode;
  Velocity         : LREAL := 0;
  Acceleration     : LREAL; (* 0 : use default value *)
  Deceleration     : LREAL; (* 0 : use default value *)
  Jerk             : LREAL; (* 0 : use default value *)
END_VAR
    
```

- Execute:** Mit der steigenden Flanke wird die Master-Slave-Kopplung als Fliegende Säge gestartet.
- RatioNumerator:** Zähler des Getriebefaktors. (siehe [Hinweis \[▶ 27\]](#))
- RatioDenominator :** Nenner des Getriebefaktors. (siehe [Hinweis \[▶ 27\]](#))
- SyncMode:** In der Variablen [SyncMode \[▶ 29\]](#) werden die Randbedingungen für die Synchronisierung festgelegt.
- Velocity:** Maximale Slave-Geschwindigkeit in der Synchronisierungsphase. Wenn keine Geschwindigkeit angegeben wird, wird die Maximalgeschwindigkeit der Achse aus den Systemmanager-Daten verwendet  
**Die hier angegebene Geschwindigkeit wird nur überprüft, wenn dies Überprüfung mit der Variablen SyncMode aktiviert wird.**
- Acceleration:** Maximale Slave-Beschleunigung in der Synchronisierungsphase. Wenn keine Beschleunigung angegeben wird, wird die Maximalbeschleunigung der Achse aus den Systemmanager-Daten verwendet  
**Die hier angegebene Beschleunigung wird nur überprüft, wenn dies Überprüfung mit der Variablen SyncMode aktiviert wird.**
- Deceleration:** Maximale Slave-Verzögerung in der Synchronisierungsphase. Wenn keine Verzögerung angegeben wird, wird die maximale Verzögerung der Achse aus den Systemmanager-Daten verwendet.  
**Die hier angegebene Verzögerung wird nur überprüft, wenn dies Überprüfung mit der Variablen SyncMode aktiviert wird.**

**Jerk:** Maximale Slave-Ruck in der Synchronisierungsphase. Wenn kein Ruck angegeben wird, wird der maximale Ruck der Achse aus den Systemmanager-Daten verwendet.  
**Der hier angegebene Ruck wird nur überprüft, wenn dies Überprüfung mit der Variablen SyncMode aktiviert wird.**

**● Negative Werte**

**i** Bei einem gewünschten Getriebefaktor von 1:3 wird der Zähler zu 1 und der Nenner zu 3 gewählt. Für den Zähler können auch negative Werte angegeben werden.

Die Slave-Achse kann mit dem Funktionsbaustein MC\_GearOut aus der TcMC Motion Control Library abgekoppelt werden. Wird der Slave während der Fahrt abgekoppelt, so behält er seine Geschwindigkeit bei und kann mit MC\_Stop angehalten werden.

**VAR\_OUTPUT**

```
VAR_OUTPUT
  StartSync      : BOOL;
  InSync         : BOOL;
  CommandAborted : BOOL;
  Error          : BOOL;
  ErrorId        : UDINT;
END_VAR
```

**StartSync :** Wird TRUE, sobald die Synchronisierung begonnen hat.  
**InSync :** Wird TRUE, sobald der Slave mit dem Master synchronisiert ist.  
**CommandAborted :** Wird TRUE, wenn die Kopplung nicht durchgeführt werden konnte.  
**Error :** Wird TRUE, sobald ein Fehler eintritt.  
**ErrorID :** Liefert bei einem gesetzten ERR-Ausgang die Fehlernummer

**VAR\_IN\_OUT**

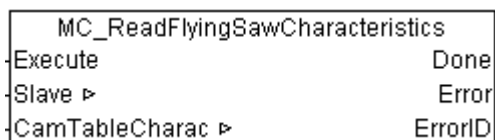
```
VAR_IN_OUT
  Master : NCTOPLC_AXLESTRUCT;
  Slave  : NCTOPLC_AXLESTRUCT;
END_VAR
```

**Master :** Achsstruktur des Masters.  
**Slave :** Achsstruktur des Slaves.

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken
TwinCAT v2.9	PC (i386)	TcNcFlyingSaw.Lib

**11.2.3 MC\_ReadFlyingSawCharacteristics**



Mit dem Funktionsbaustein **MC\_ReadFlyingSawCharacteristics** können die charakteristischen Kennwerte der Synchronisierungsphase der Universellen Fliegenden Säge ausgelesen werden.

**VAR\_INPUT**

```
VAR_INPUT
  Execute : BOOL;
END_VAR
```

**Execute:**

Mit einer steigenden Flanke wird das Auslesen der Kennwerte aus der TwinCAT-NC gestartet.

**● Verfügbarkeit der Informationen**

**i** Die berechneten Daten stehen erst nach dem Start der Universellen Fliegenden Säge zur Verfügung.

**VAR\_OUTPUT**

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

**Done:**

Wird TRUE, wenn der Datensatz erfolgreich gelesen worden ist.

**Error :**

Wird TRUE, sobald ein Fehler eintritt.

**ErrorID :**

Liefert bei einem gesetzten ERR-Ausgang die Fehlernummer

**VAR\_IN\_OUT**

```
VAR_IN_OUT
  Slave           : NCTOPLC_AXLESTRUCT;
  CamTableCharac : MC_FlyingSawCharacValues;
END_VAR
```

**Slave :**

Achsstruktur des Slaves.

**CamTableCharac: :**

Struktur mit den charakteristischen Kennwerten.

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken
TwinCAT v2.9	PC (i386)	TcNcFlyingSaw.Lib

**11.2.4 Datenstruktur ST\_SyncMode**

```
TYPE ST_SyncMode :
STRUCT
  (* mode *)
  GearInSyncMode           : ST_GearInSyncMode;

  (* 32 bit check mask ... *)
  GearInSync_CheckMask_MinPos       : BOOL;
  GearInSync_CheckMask_MaxPos       : BOOL;
  GearInSync_CheckMask_MaxVelo      : BOOL;
  GearInSync_CheckMask_MaxAcc       : BOOL;
  GearInSync_CheckMask_MaxDec       : BOOL;
  GearInSync_CheckMask_MaxJerk      : BOOL;
  GearInSync_CheckMask_OvershootPos  : BOOL;
  GearInSync_CheckMask_UndershootPos : BOOL;
  GearInSync_CheckMask_OvershootVelo : BOOL;
  GearInSync_CheckMask_UndershootVelo : BOOL;
  GearInSync_CheckMask_OvershootVeloZero : BOOL;
  GearInSync_CheckMask_UndershootVeloZero : BOOL;

  (* operation masks ... *)
  GearInSync_OpMask_RollbackLock     : BOOL;
  GearInSync_OpMask_InstantStopOnRollback : BOOL;
  GearInSync_OpMask_PreferConstVelo  : BOOL;
  GearInSync_OpMask_IgnoreMasterAcc  : BOOL;
  GearInSync_OpMask_IgnoreSlaveAcc   : BOOL;
  GearInSync_OpMask_DetailedErrorCodes : BOOL;
END_STRUCT
END_TYPE
```

```
TYPE ST_GearInSyncMode :  
(  
    GEARINSYNCMODE_POSITIONBASED, (* synchronization based on the master position, slave dynamics  
depend on master dynamics *)  
    GEARINSYNCMODE_TIMEBASED (* synchronization based on a standalone slave PTP profile, master  
independent slave dynamics *)  
);  
END_TYPE
```



Das zeitbasierte Bewegungsprofil (GEARINSYNCMODE\_TIMEBASED) ist zurzeit nur für den Funktionsbaustein MC\_GearInVelo implementiert.

Wirkungsweise der einzelnen Bits.



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

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)

