

Kapitel 5

Test der Motorsteuerung

In diesem Kapitel werden nun die Testfälle bzw. Testsequenzen erläutert, die für den Test der Motorenkomponente des XCTL-Systems entwickelt worden sind.

Die primäre Sicht auf das Testobjekt Motorenkomponente ist eine *black box*-Sicht. D.h., es werden das von außen beobachtbare Verhalten und die nach außen angebotenen Dienste betrachtet. Solche nach außen zur Verfügung gestellten Dienste sind die in `m_layer.h` definierte Programmierschnittstelle und die Dialoge, die die Komponente implementiert. Das eine ist ein Dienst für die Benutzung durch einen Programmierer, das andere dient der Realisierung von Funktionen bzw. Anwendungsfällen für den Endbenutzer.

Die Motorenkomponente als Testobjekt ist sehr groß. Deshalb ist es sinnvoll und notwendig, einzelne Aspekte getrennt zu testen. Die Tests für die Komponente sind daher in mehrere Testpakete gegliedert. Die eben angesprochene Unterscheidung der Nutzer in Programmierer und Endbenutzer ist für diese Gliederung als Grundlage verwendet worden. D.h. es sind Testpakete zu entwerfen, die

- die einzelnen Dialoge und
- die Programmierschnittstelle testen.

Für den Test von Dialogen ist bisher *ein* Testpaket entworfen worden, das die Funktion „F2 Grundstellung Anfahren“ der Verhaltensspezifikation testet. Das Testpaket heißt *Referenzpunktlauf* (`m_rpl`), s. S. 165.

Die Programmierschnittstelle wird aktuell durch *zwei* Testpakete getestet. Zum einen durch das Paket *Initialisierung* (`m_init`), das die Initialisierung der Motorenkomponente aus den Konfigurationsdaten testet (s. S. 126). Hierbei sollen vor allem die Reaktionen der Komponente auf fehlende oder fehlerhafte Parameter überprüft werden. Mit der Annahme des erfolgreichen Tests durch dieses Paket, kann dann in den übrigen Paketen der Aspekt der *ungültigen Konfigurationsdaten* größtenteils ausgeblendet werden, was den dortigen Aufwand erheblich reduziert.

Zum anderen wird die Schnittstelle durch das Paket *C-Interface* (`m_layer`) getestet (s. S. 144). Hier werden die Ausgaben der einzelnen Funktionen der Schnittstelle in Abhängigkeit vom Zustand der Komponente getestet.

5.1 Testpaket *Initialisierung* (m_init)

Testobjekt

Initialisierung der Motorenkomponente aus den Parametern der Konfigurationsdateien; genauer der Initialisierungsvorgang, der durch die `m_layer.h`-Funktion `mInitializeMotorsDLL` ausgelöst wird, sowie der Zustand der Komponente nach der Initialisierung soweit er durch die `m_layer.h`-Schnittstelle sichtbar ist.

Es soll vor allem die Robustheit des Komponentenverhaltens im Bezug auf fehlende bzw. fehlerhafte Parameter sichergestellt werden.

Eingabedaten

Variable Eingaben sind die Parameter der Konfigurationsdateien der Sektionen `MotorX` (s. Verhaltensspezifikation in Anhang A) und `MOTORSIM` (s. Abschnitt 3.5).

Konstant ist dagegen im wesentlichen der Testablauf, d.h. die Reihenfolge und die Parameter der `m_layer.h`-Funktionen die verwendet werden.

Ausgabedaten

Ausgaben sind a) die Steuerungsanweisungen der Motorenkomponente für die Motorenhardware b) die Rückgabewerte und Variablenparameter der Funktionsaufrufe und c) die Konfigurationsparameter nach Beenden der Testanwendung.

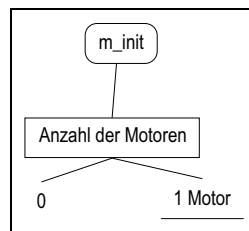


Abbildung 5.1: Klassifikationsbaum (m_init)

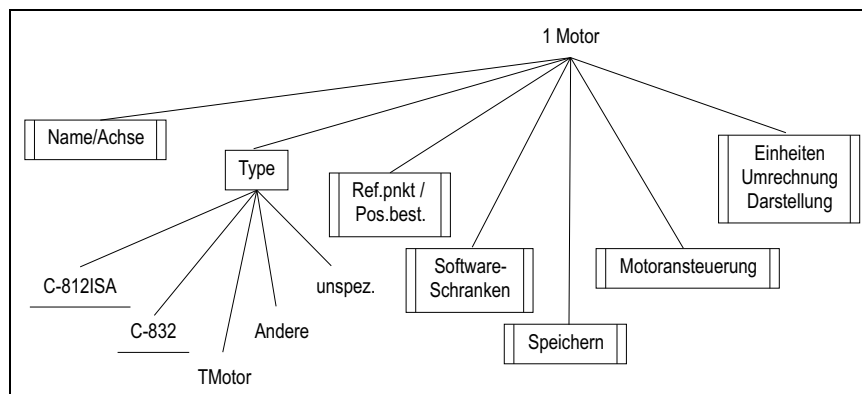


Abbildung 5.2: Klassifikationsbaum (m_init|1 Motor)

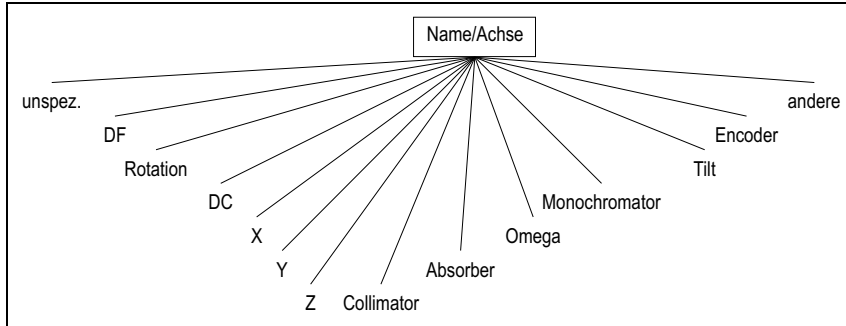


Abbildung 5.3: Klassifikationsbaum (m_init|1 Motor|Name/Achse)

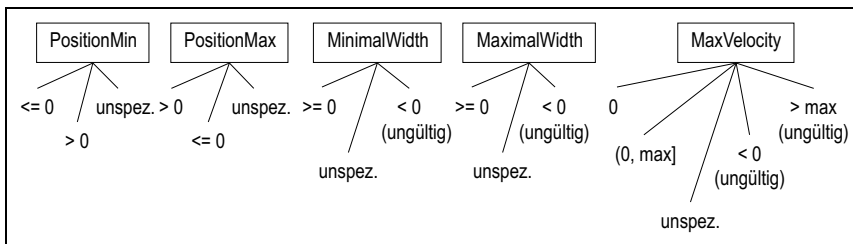


Abbildung 5.4: Klassifikationsbaum (m_init|1 Motor|Softwareschranken)

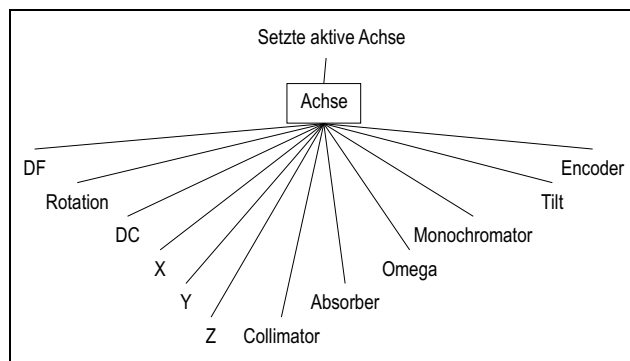


Abbildung 5.5: Klassifikationsbaum (m_init|1 Motor|Type|C-812ISA)

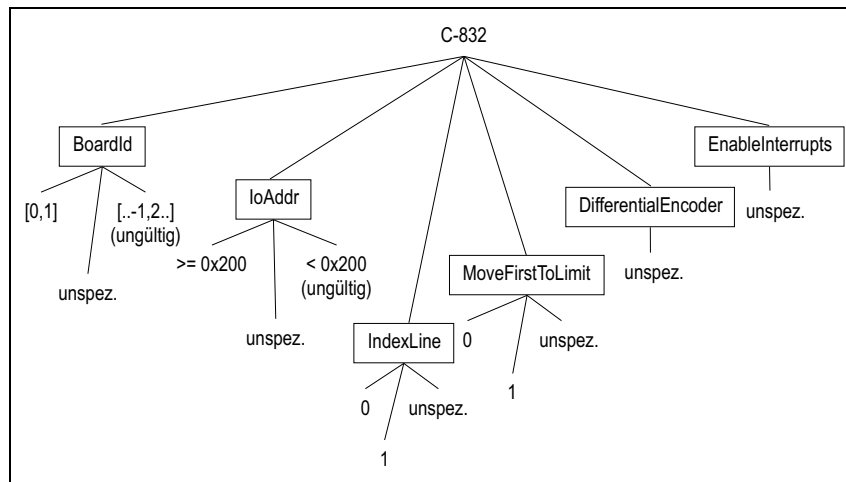


Abbildung 5.6: Klassifikationsbaum (m_init|1 Motor|Type|C-832)

Klassifikationsbaum

m_init

Attribute

%ini_file: hardware.ini

ScriptGenerator: m_init

ini| %ini_file| MOTORSIM|LogFile: %test_id.LOG

ini| %ini_file| MOTORSIM|LogLevel: 2

ini| %ini_file| MOTORSIM|StatusWindow: 1

ini| %ini_file| MOTORSIM|dll: msim.dll

Anzahl der Motoren

0

Beschreibung

Ini-Datei ohne Motor-Abschnitt.

1 Motor

Beschreibung

Ini-Datei mit genau einem Motor-Abschnitt.

Attribute

%ini_section: Motor0

Name/Achse

Beschreibung

Konfigurationsparameter: Name.

Attribute

ini| %ini_file| %ini_section|Name: %axis

DF

Attribute

%axis: DF

Rotation

Attribute

%axis: Rotation

DC

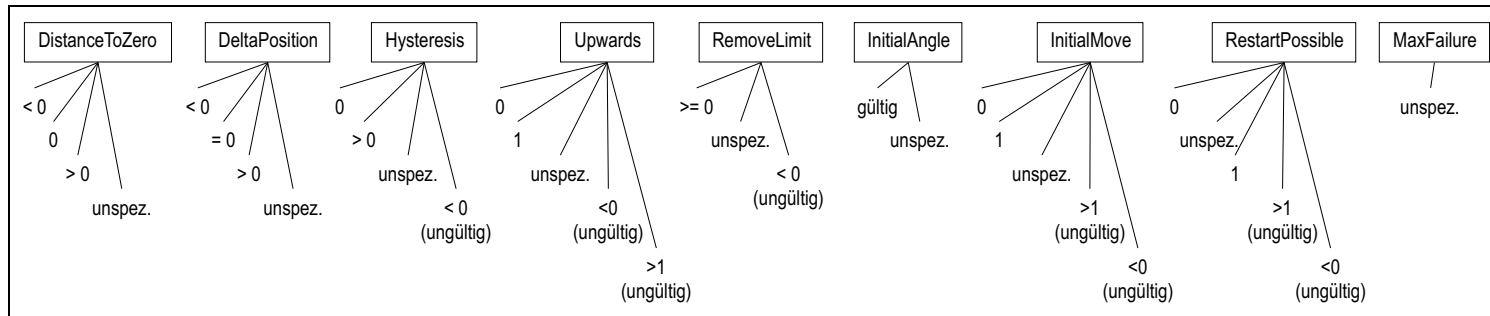


Abbildung 5.7: Klassifikationsbaum (m_init|1 Motor|Refpnt./Pos.best.)

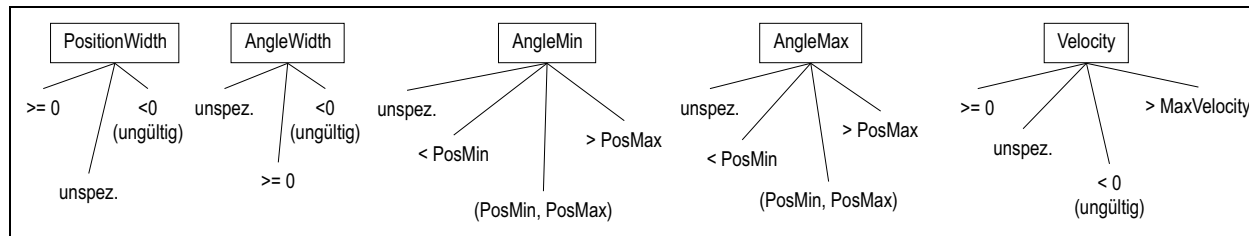


Abbildung 5.8: Klassifikationsbaum (m_init|1 Motor|Speichern)

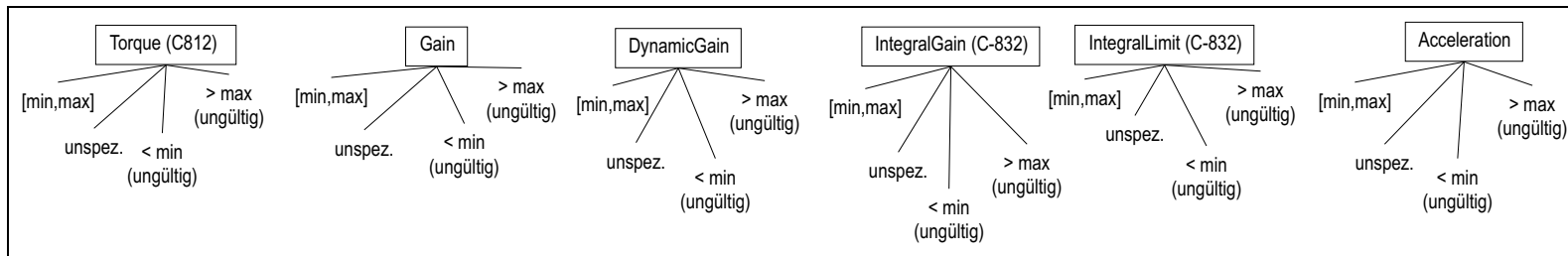


Abbildung 5.9: Klassifikationsbaum (m_init|1 Motor|Motoransteuerung)

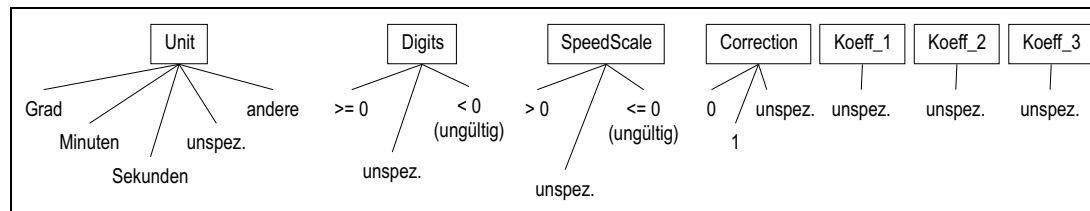


Abbildung 5.10: Klassifikationsbaum (m_init|1 Motor|Einheiten Umrechnung...)

Attribute
 %axis: DC
 X

Attribute
 %axis: X
 Y

Attribute
 %axis: Y
 Z

Attribute
 %axis: Z
 Collimator

Attribute
 %axis: Collimator
 Absorber

Attribute
 %axis: Absorber
 Omega

Attribute
 %axis: Omega
 Monochromator

Attribute
 %axis: Monochromator
 Tilt

Attribute
 %axis: Tilt
 Encoder

Attribute
 %axis: Encoder
 unspez.

Beschreibung
 kein Name im ini-file
Attribute
 %axis: NULL
 andere

Beschreibung
 ein nicht vordefinierter Name
Attribute
 %axis: andere

Type
Beschreibung
 Konfigurationsparameter: Type.
Attribute
 ini| %ini_file| %ini_section| Type: %MotorType
 C-812ISA

Attribute
 %MotorType: C-812ISA
BoardId
Attribute

ini|%ini_file|%ini_section|BoardID: %BoardID
[1..4]
Attribute
%BoardID: 2
[.0,5..] (ungültig)
Attribute
%BoardID: 5
unspez.
Attribute
%BoardID: NULL

RamAddr
Attribute
ini|%ini_file|%ini_section|RamAddr: %RamAddr
[0xC000..0xD000)
Attribute
%RamAddr: 0xC000
[0xD000..0xDE00]
Attribute
%RamAddr: 0xD000
unspez.
Attribute
%RamAddr: NULL
außerhalb [0xC000..0xDE00] (ungültig)
Attribute
%RamAddr: 0xB000

DeathBand
Attribute
ini|%ini_file|%ini_section|DeathBand: %DeathBand
< 0
Attribute
%DeathBand: -5
1
Attribute
%DeathBand: 1
unspez.
Attribute
%DeathBand: NULL

C-832
Attribute
%MotorType: C-832

BoardId
Attribute
ini|%ini_file|%ini_section|BoardId: %BoardId
[0,1]
Attribute
%BoardId: 1
unspez.
Attribute

%BoardId: NULL
 [-1,2..] (ungültig)

Attribute

%BoardId: -1

IoAddr

Attribute

ini|%ini_file|%ini_section|IoAddr: %IoAddr

>= 0x200

Attribute

%IoAddr: 0x200

unspez.

Attribute

%IoAddr: NULL

< 0x200 (ungültig)

Attribute

%IoAddr: 0x100

DifferentialEncoder

unspez.

EnableInterrupts

unspez.

IndexLine

Attribute

ini|%ini_file|%ini_section|IndexLine: %IndexLine

0

Attribute

%IndexLine: 0

1

Attribute

%IndexLine: 1

unspez.

Attribute

%IndexLine: NULL

MoveFirstToLimit

Attribute

ini|%ini_file|%ini_section|MoveFirstToLimit:

%MoveFirstToLimit

0

Attribute

%MoveFirstToLimit: 0

1

Attribute

%MoveFirstToLimit: 1

unspez.

Attribute

%MoveFirstToLimit: NULL

TMotor

Attribute

%MotorType: TMotor

Andere

Beschreibung

Nicht erlaubter Motortyp.

Attribute

%MotorType: Andere

unspez.

Beschreibung

Fehlende Angabe zum Motorentyp.

Attribute

%MotorType: NULL

Ref.punkt / Pos.best.

Beschreibung

Konfigurationsparameter zum Thema: Repezenzpunkt und Positionsbestimmung.

DistanceToZero

Beschreibung

?? auch negative Werte möglich, falls IndexLine=1

Attribute

ini|%ini_file|%ini_section|DistanceToZero: %DistanceToZero

> 0

Attribute

%DistanceToZero: 5000

unspez.

Attribute

%DistanceToZero: NULL

< 0

Attribute

%DistanceToZero: -1000

0

Attribute

%DistanceToZero: 0

InitialMove

Attribute

ini|%ini_file|%ini_section|InitialMove: %InitialMove

0

Attribute

%InitialMove: 0

1

Attribute

%InitialMove: 1

unspez.

Attribute

%InitialMove: NULL

<0 (ungültig)

Attribute

%InitialMove: -5
 >1 (ungültig)
Attribute
%InitialMove: 1000

InitialAngle

Attribute
ini|%ini_file|%ini_section|InitialAngle: %InitialAngle
 unspez.
Attribute
%InitialAngle: NULL
 gültig
Attribute
%InitialAngle: & %DeltaPositionEffective + 100

RemoveLimit

Attribute
ini|%ini_file|%ini_section|RemoveLimit: %RemoveLimit
 >= 0
Attribute
%RemoveLimit: 1000
 unspez.
Attribute
%RemoveLimit: NULL
 < 0 (ungültig)
Attribute
%RemoveLimit: -2000

Hysteresis

Attribute
ini|%ini_file|%ini_section|Hysteresis: %Hysteresis
 > 0
Attribute
%Hysteresis: 50
 unspez.
Attribute
%Hysteresis: NULL
 < 0 (ungültig)
Attribute
%Hysteresis: -100
 0
Attribute
%Hysteresis: 0

DeltaPosition

Attribute
%DeltaPositionEffective: ('%DeltaPosition' eq 'NULL') ? 0 :
%DeltaPosition
ini|%ini_file|%ini_section|DeltaPosition: %DeltaPosition
 < 0
Attribute
%DeltaPosition: -1000

unspec.
Attribute
%DeltaPosition: NULL
= 0
Attribute
%DeltaPosition: 0
> 0
Attribute
%DeltaPosition: 2000

Upwards

Attribute
ini|%ini_file|%ini_section|Upwards: %Upwards
0
Attribute
%Upwards: 0
1
Attribute
%Upwards: 1

unspec.
Attribute
%Upwards: NULL
<0 (ungültig)
Attribute
%Upwards: -5
>1 (ungültig)
Attribute
%Upwards: 10

RestartPossible

Attribute
ini|%ini_file|%ini_section|RestartPossible: %RestartPossible
0
Attribute
%RestartPossible: 0
1
Attribute
%RestartPossible: 1

unspec.
Attribute
%RestartPossible: NULL
<0 (ungültig)
Attribute
%RestartPossible: -1
>1 (ungültig)
Attribute
%RestartPossible: 100

MaxFailure

unspec.

Software- Schranken

Beschreibung

Konfigurationsparameter zu den Softwareschranken.

PositionMin

Attribute

*%PositionMinEffective: & ('%PositionMin' eq 'NULL') ? -100 :
%PositionMin*

ini|%ini_file|%ini_section|PositionMin: %PositionMin

<= 0

Attribute

%PositionMin: -6000

unspec.

Attribute

%PositionMin: NULL

> 0

Attribute

%PositionMin: 1000

PositionMax

Attribute

*%PositionMaxEffective: & ('%PositionMax' eq 'NULL') ? 100 :
%PositionMax*

ini|%ini_file|%ini_section|PositionMax: %PositionMax

<= 0

Attribute

%PositionMax: -500

unspec.

Attribute

%PositionMax: NULL

> 0

Attribute

%PositionMax: 10000

MinimalWidth

Attribute

*%MinimalWidthEffective: & ('%MinimalWidth' eq 'NULL') ?
10 : %MinimalWidth*

ini|%ini_file|%ini_section|MinimalWidth: %MinimalWidth

>= 0

Attribute

%MinimalWidth: 20

unspec.

Attribute

%MinimalWidth: NULL

< 0 (ungültig)

Attribute

%MinimalWidth: -200

MaximalWidth

Attribute

5. TEST DER MOTORSTEUERUNG

```
%MaximalWidthEffective: & ('%MaximalWidth' eq 'NULL') ?  
    10 : %MaximalWidth  
ini|%ini_file|%ini_section|MaximalWidth: %MaximalWidth  
>= 0  
    Attribute  
        %MaximalWidth: 100  
unspez.  
    Attribute  
        %MaximalWidth: NULL  
< 0 (ungültig)  
    Attribute  
        %MaximalWidth: -20
```

MaxVelocity

```
Attribute  
%MaxVelocityEffective: & ('%MaxVelocity' eq 'NULL') ? 8000  
    : %MaxVelocity  
ini|%ini_file|%ini_section|MaxVelocity: %MaxVelocity  
(0, max]  
    Attribute  
        %MaxVelocity: 8000  
unspez.  
    Attribute  
        %MaxVelocity: NULL  
< 0 (ungültig)  
    Attribute  
        %MaxVelocity: -100  
> max (ungültig)  
    Attribute  
        %MaxVelocity: 9000000  
0  
    Attribute  
        %MaxVelocity: 0
```

Speichern

PositionWidth

```
Attribute  
ini|%ini_file|%ini_section|PositionWidth: %PositionWidth  
>= 0  
    Attribute  
        %PositionWidth: 1000  
unspez.  
    Attribute  
        %PositionWidth: NULL  
<0 (ungültig)  
    Attribute  
        %PositionWidth: -2000
```

AngleMin

```
Attribute
```

```

    %AngleMinEffective: & ('%AngleMin' eq 'NULL') ? -1 :
        %AngleMin
    ini|%ini_file|%ini_section|AngleMin: %AngleMin
    un spez.
        Attribute
            %AngleMin: NULL
    (PosMin, PosMax)
        Attribute
            %AngleMin: & %PositionMinEffective + 100
    < PosMin
        Attribute
            %AngleMin: & %PositionMinEffective - 10000
    > PosMax
        Attribute
            %AngleMin: & %PositionMaxEffective + 10000

```

AngleMax

```

    Attribute
        %AngleMaxEffective: & ('%AngleMax' eq 'NULL') ? +1 :
            %AngleMax
    ini|%ini_file|%ini_section|AngleMax: %AngleMax
    (PosMin, PosMax)
        Attribute
            %AngleMax: & %PositionMaxEffective - 100
    un spez.
        Attribute
            %AngleMax: NULL
    < PosMin
        Attribute
            %AngleMax: & %PositionMinEffective - 10000
    > PosMax
        Attribute
            %AngleMax: & %PositionMaxEffective + 10000

```

AngleWidth

```

    Attribute
        %AngleWidthEffective: & ('%AngleWidth' eq 'NULL') ? 0.1 :
            %AngleWidth
    ini|%ini_file|%ini_section|AngleWidth: %AngleWidth
    >= 0
        Attribute
            %AngleWidth: 50
    un spez.
        Attribute
            %AngleWidth: NULL
    <0 (ungültig)
        Attribute
            %AngleWidth: -50

```

Velocity

```

    Attribute

```

5. TEST DER MOTORSTEUERUNG

```
ini| %ini_file| %ini_section| Velocity: %Velocity
>= 0
Attribute
%Velocity: 5000
unspez.
Attribute
%Velocity: NULL
< 0 (ungültig)
Attribute
%Velocity: -1000
> MaxVelocity
Attribute
%Velocity: & %MaxVelocity + 100
```

Motoransteuerung

Torque (C812)

```
Attribute
ini| %ini_file| %ini_section| Torque: %Torque
[min,max]
Attribute
%Torque: 50
unspez.
Attribute
%Torque: NULL
< min (ungültig)
Attribute
%Torque: 0
> max (ungültig)
Attribute
%Torque: 200
```

Gain

```
Attribute
ini| %ini_file| %ini_section| Gain: %Gain
[min,max]
Attribute
%Gain: 200
unspez.
Attribute
%Gain: NULL
< min (ungültig)
Attribute
%Gain: -10
> max (ungültig)
Attribute
%Gain: & ('%MotorType' eq 'C-812ISA') ? 300 :
(( '%MotorType' eq 'C-832') ? 33000 : 400000)
```

DynamicGain

```
Attribute
```



```

ini|%ini_file|%ini_section|DynamicGain: %DynamicGain
[min,max]
Attribute
%DynamicGain: 255
unspez.
Attribute
%DynamicGain: NULL
< min (ungültig)
Attribute
%DynamicGain: -100
> max (ungültig)
Attribute
%DynamicGain: & ('%MotorType' eq 'C-812ISA') ? 300 :
('%MotorType' eq 'C-832') ? 33000 : 40000

```

IntegralGain (C-832)

```

Attribute
ini|%ini_file|%ini_section|IntegralGain: %IntegralGain
[min,max]
Attribute
%IntegralGain: 1000
unspez.
Attribute
%IntegralGain: NULL
< min (ungültig)
Attribute
%IntegralGain: -100
> max (ungültig)
Attribute
%IntegralGain: 33000

```

IntegralLimit (C-832)

```

Attribute
ini|%ini_file|%ini_section|IntegralLimit: %IntegralLimit
[min,max]
Attribute
%IntegralLimit: 1000
unspez.
Attribute
%IntegralLimit: NULL
< min (ungültig)
Attribute
%IntegralLimit: -1
> max (ungültig)
Attribute
%IntegralLimit: 33000

```

Acceleration

```

Attribute
ini|%ini_file|%ini_section|Acceleration: %Acceleration
[min,max]

```

5. TEST DER MOTORSTEUERUNG

Attribute
%Acceleration: 10000
unspez.
Attribute
%Acceleration: NULL
< min (ungültig)
Attribute
%Acceleration: -1000
> max (ungültig)
Attribute
%Acceleration: & ('%MotorType' eq 'C-812ISA') ? 9000000
: (('%MotorType' eq 'C-832') ? 2000000000 : 232323)

Einheiten Umrechnung Darstellung

Unit

Attribute
ini|%ini_file|%ini_section|Unit: %Unit
Grad
Attribute
%Unit: Grad
Sekunden
Attribute
%Unit: Sekunden
Minuten
Attribute
%Unit: Minuten
unspez.
Attribute
%Unit: NULL
andere
Attribute
%Unit: Steps

Digits

Attribute
ini|%ini_file|%ini_section|Digits: %Digits
>= 0
Attribute
%Digits: 2
unspez.
Attribute
%Digits: NULL
< 0 (ungültig)
Attribute
%Digits: -5

SpeedScale

Attribute
ini|%ini_file|%ini_section|SpeedScale: %SpeedScale
> 0

Attribute
%SpeedScale: 1
unspez.

Attribute
%SpeedScale: NULL
<= 0 (ungültig)

Attribute
%SpeedScale: -1

Correction

Attribute
in| %ini_file| %ini_section| Correction: %Correction
0

Attribute
%Correction: 0
1

Attribute
%Correction: 0
unspez.

Attribute
%Correction: NULL

Koeff_1
unspez.

Koeff_2
unspez.

Koeff_3
unspez.

5.2 Testpaket *C-Interface* (`m_layer`)

Testobjekt

Testobjekt ist die Motorenkomponente aus der Sicht des in `m_layer.h` definierten Programmierinterfaces.

Aus dieser Sicht ist die Motorenkomponente ein großes Objekt und die Funktionen des C-Interfaces sind seine Methoden. Getestet werden Sequenzen von Methodenaufrufen zur Steuerung *eines* Motors.

Eingabedaten

Eingabe eines Methodenaufrufs sind die Methode selbst, deren Parameter sowie der Zustand des Objektes.

Der Ausgangszustand der Motorenkomponente wird durch die Parameter der Konfigurationsdateien gegeben. Alle weiteren Zustände sind jeweils das Ergebnis der vorhergehenden Methodenaufrufe. Zur Vereinfachung wird angenommen, dass die Komponente sich stets in einem gültigen Zustand befindet, insbesondere dass der durch die Konfigurationsparameter beschriebene Ausgangszustand korrekt ist.

Bei den Parametern der Methodenaufrufe sind dagegen auch ungültige Werte zu testen.

Außerdem werden nur Konfigurationen getestet, in denen das Verhältnis von Nutzereinheiten zu Motoreinheiten *eins zu eins* ist, d.h., dass die Parameter `Koeff_1`, `SpeedScale` und `Correction` konstant die Werte 1, 1 und 0 haben. Dies dient der Vereinfachung dieses ohnehin großen Testpakets, erfolgt aber auch, weil der Punkt "F1.4 Umrechnung von Einheiten" der Verhaltensspezifikation noch nicht ausreichend ausgeführt wurde. Diese Funktion sollte durch ein weiteres Testpaket getestet werden.

Ausgabedaten

Ausgaben sind wie im Paket `m_init` a) die Steuerungsanweisungen der Motorenkomponente für die Motorenhardware b) die Rückgabewerte und Variablenparameter der Funktionsaufrufe und c) die Konfigurationsparameter nach Beenden der Testanwendung.

Klassifikationsbaum

`m_layer`

Attribute

```
ScriptGenerator: m_layer
ini|hardware.ini|MOTORSIM|DLL: msim.dll
ini|hardware.ini|MOTORSIM|LogFile: %test_id.LOG
ini|hardware.ini|MOTORSIM|LogLevel: 2
ini|hardware.ini|MOTORSIM|StatusWindow: 1
```

Zustand

Beschreibung

Zustand des Motors, für den der Test durchgeführt wird.

Attribute

```
%MovableRange: 20000
%ini_file: hardware.ini
ini|%ini_file|%ini_section|AngleMax: %PositionMax
```

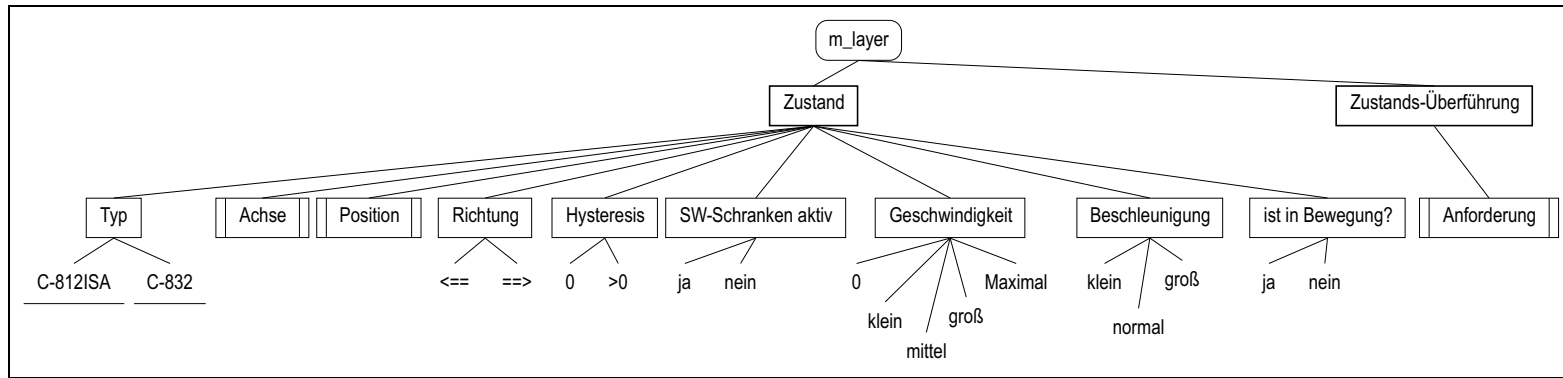


Abbildung 5.11: Klassifikationsbaum (m_layer)

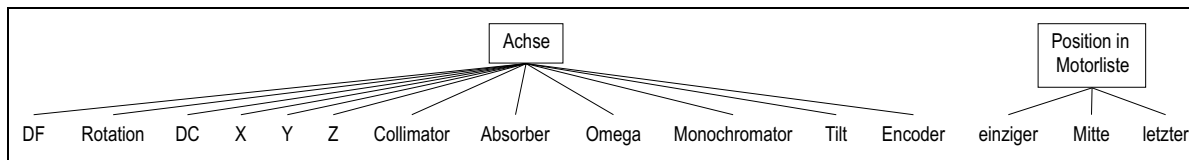


Abbildung 5.12: Klassifikationsbaum (m_layer|Achse)

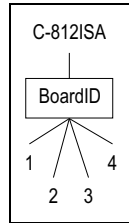


Abbildung 5.13: Klassifikationsbaum (m_layer|C-812ISA)

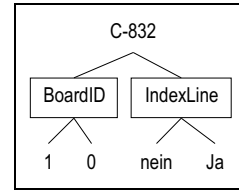


Abbildung 5.14: Klassifikationsbaum (m_layer|C-832)

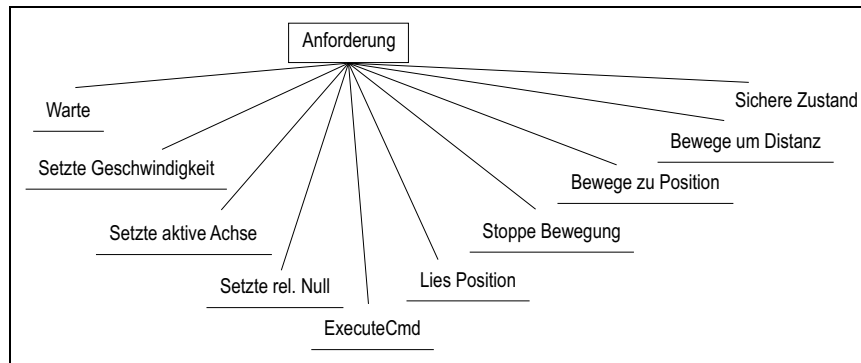


Abbildung 5.15: Klassifikationsbaum (m_layer|Anforderung)

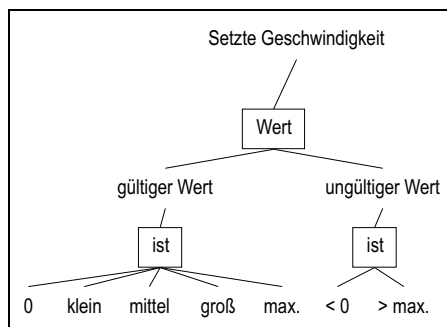


Abbildung 5.16: Klassifikationsbaum (m_layer|Anforderung|Setze Geschwindigkeit)

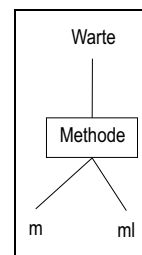


Abbildung 5.17: Klassifikationsbaum (m_layer|Anforderung|Warte)

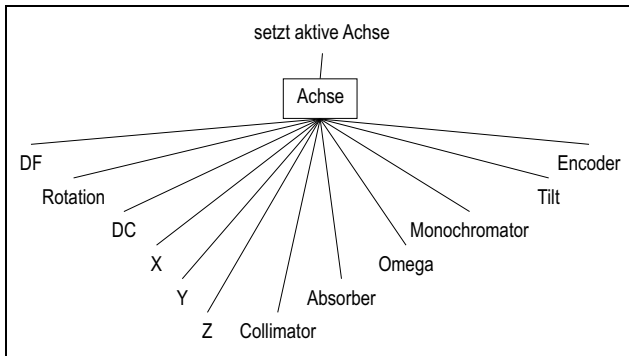


Abbildung 5.18: Klassifikationsbaum (m_layer|Anforderung|Setze aktive Achse)

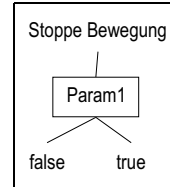


Abbildung 5.19: Klassifikationsbaum (m_layer|Anforderung|Stoppe Bewegung)

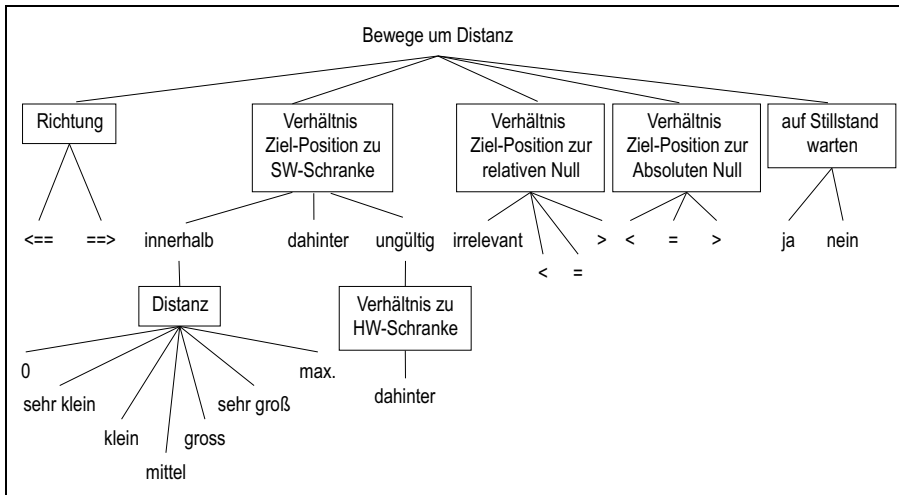


Abbildung 5.20: Klassifikationsbaum (m_layer|Anforderung|Bewege um Distanz)

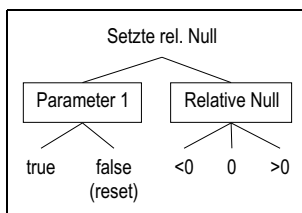


Abbildung 5.21: Klassifikationsbaum (m_layer|Anforderung|Setze Rel. Null)

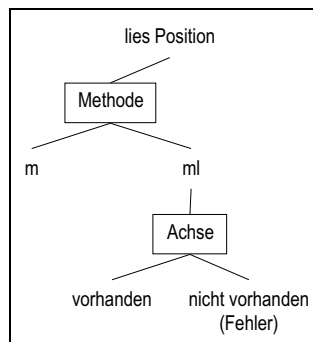


Abbildung 5.22: Klassifikationsbaum (m_layer|Anforderung|lies Position)

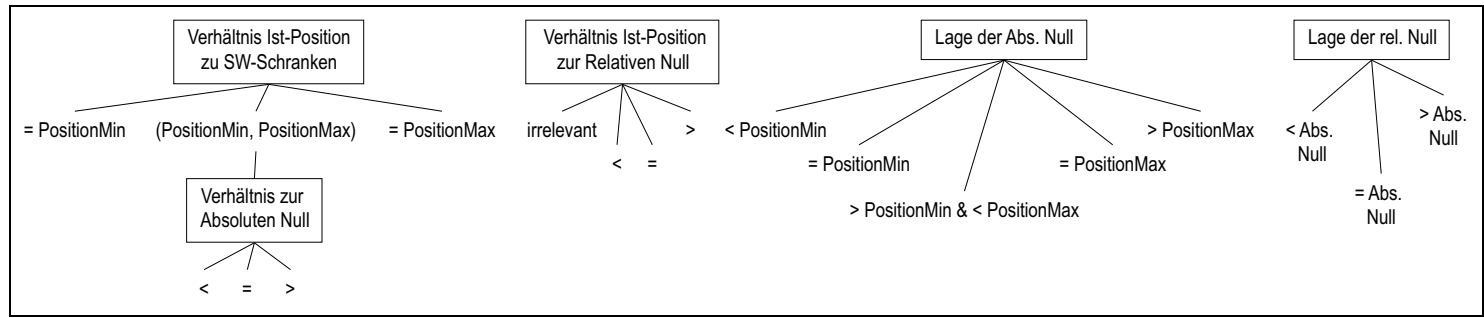


Abbildung 5.23: Klassifikationsbaum (m_layer|Position)

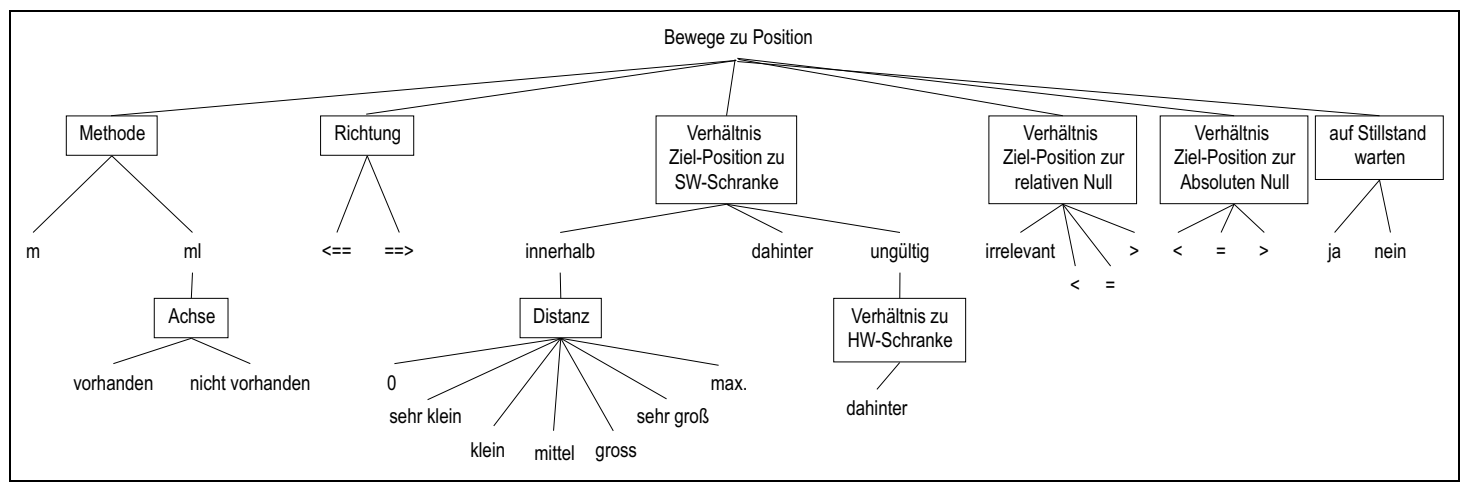


Abbildung 5.24: Klassifikationsbaum (m_layer|Anforderung|Bewegung zu Position)


```
ini|%ini_file|%ini_section|AngleMin: %PositionMin
ini|%ini_file|%ini_section|DeathBand: 1
ini|%ini_file|%ini_section|Koeff_1: 1
ini|%ini_file|%ini_section|RamAddr: 0xD000
ini|%ini_file|%ini_section|RemoveLimit: 4000
ini|%ini_file|%ini_section|Unit: Steps
```

ist in Bewegung?

Beschreibung

Zu testen via mIsMoveFinish oder mlIsMoveFinish.

ja

nein

Geschwindigkeit

Beschreibung

Zu setzten über mSetValue(Speed); zu lesen über mGetValue(Speed).

Attribute

%MaxVelocity: 20000

%SpeedScale: 1

```
ini|%ini_file|%ini_section|MaxVelocity: %MaxVelocity
```

```
ini|%ini_file|%ini_section|SpeedScale: %SpeedScale
```

```
ini|%ini_file|%ini_section|Velocity: %Velocity
```

klein

Attribute

%Velocity: 100

mittel

Attribute

%Velocity: 5000

groß

Attribute

%Velocity: 15000

0

Attribute

%Velocity: 0

Maximal

Attribute

%Velocity: *%MaxVelocity*

SW-Schranken aktiv

Beschreibung

Aktiv meint: die SW-Schranken liegen innerhalb der HW-Schranken (d.h. sind gültig).

Die Lage der HW-Schranken wird durch einen Parameter der Motorsimulation festgelegt.

ja

Attribute

```
ini|%ini_file|MOTORSIM|hswDistance: 4000
```

nein

Attribute

```
ini|%ini_file|MOTORSIM|hswDistance: -1000
```

Beschleunigung

Beschreibung

Kann nur im ini-File festgelegt werden.

Attribute

ini | *%ini_file* | *%ini_section* | *Acceleration: %Acceleration*

groß

Attribute

%Acceleration: 20000

normal

Attribute

%Acceleration: 8000

klein

Attribute

%Acceleration: 100

Position

Beschreibung

Ist-Position des Motors.

Verhältnis Ist-Position zu SW-Schranken

Attribute

assert | *Position1: (%PositionMin <= %DeltaPosition) &&*
(%DeltaPosition <= %PositionMax)

ini | *%ini_file* | *%ini_section* | *DeltaPosition: %DeltaPosition*
 = *PositionMin*

Beschreibung

Der Motor steht auf der linken SW-Schranke.

Attribute

%DeltaPosition: %PositionMin

(*PositionMin, PositionMax*)

Beschreibung

Der Motor steht zwischen linker und rechter SW-Schranke.

Verhältnis zur Absoluten Null

Beschreibung

Lage der Ist-Position im Koordinatensystem der Absoluten Positionen.

<

Beschreibung

Der Motor steht links der Abs. Null, d.h. absolute Positionsangaben sind negativ.

Attribute

%DeltaPosition: & (%test|step_nr == 1) ? -2000 :
(defined(%test|pred_step|newDeltaPosition) ?
%test|pred_step|newDeltaPosition :
%test|pred_step|DeltaPosition)

=

Attribute

%DeltaPosition: & (%test|step_nr == 1) ? 0 :
(defined(%test|pred_step|newDeltaPosition) ?
%test|pred_step|newDeltaPosition :

```

                                %test|pred_step|DeltaPosition)
>
    Beschreibung
        Der Motor steht rechts der Abs. Null, d.h. absolute Po-
        sitionsangaben sind positiv.
    Attribute
        %DeltaPosition: & (%test|step_nr == 1) ? 3000 :
            (defined(%test|pred_step|newDeltaPosition) ?
            %test|pred_step|newDeltaPosition :
            %test|pred_step|DeltaPosition)
= PositionMax
    Beschreibung
        Der Motor steht auf der rechten SW-Schranke.
    Attribute
        %DeltaPosition: %PositionMax

```

Lage der Abs. Null

```

    Beschreibung
        Lage der Abs. Null wird durch den ini-Parameter DistanceToZero
        gegeben, dem Abstand zwischen Referenzpunkt und Abs. Null.
        Kann von der Software nur durch ein Referenzpunktlauf verändert
        werden.
        In diesem Testpaket wird immer mit einem Referenzpunkt gear-
        beitet, der gleich der linken HW-Schranke ist.
    Attribute
        %PositionMax: & %PositionMin + %MovableRange
        %PositionMin: & -(%DistanceToZero)
        ini| %ini_file| %ini_section| DistanceToZero: %DistanceToZero
        ini| %ini_file| %ini_section| PositionMax: %PositionMax
        ini| %ini_file| %ini_section| PositionMin: %PositionMin
< PositionMin
    Beschreibung
        Abs. Null liegt links des verfahrbaren Bereiches.
    Attribute
        %DistanceToZero: -1000
= PositionMin
    Beschreibung
        Abs. Null ist gleich der linken SW-Schranke.
    Attribute
        %DistanceToZero: 0
> PositionMin & < PositionMax
    Beschreibung
        Abs. Null liegt innerhalb des verfahrbaren Bereiches.
    Attribute
        %DistanceToZero: 6000
= PositionMax
    Beschreibung
        Abs. Null ist gleich der rechten SW-Schranke.
    Attribute
        %DistanceToZero: %MovableRange
> PositionMax
    Beschreibung

```

Abs. Null liegt rechts des verfahrbaren Bereiches.

Attribute

`%DistanceToZero: & %MovableRange + 1000`

Lage der rel. Null

Beschreibung

Lage der Relativen Null im Verhältnis zur Abs. Null.

= Abs. Null

Beschreibung

d.h. nicht gesetzt; das ist stets nach der Initialisierung der Fall

Attribute

`%RelNullOffset: 0`

< Abs. Null

Attribute

`%RelNullOffset: %test|pred_step|newRelNullOffset`

> Abs. Null

Attribute

`%RelNullOffset: %test|pred_step|newRelNullOffset`

Verhältnis Ist-Position zur Relativen Null

Beschreibung

Nur relevant wenn die Rel. Null ungleich der Absoluten Null ist.

<

Beschreibung

Der Motor steht links der Rel. Null, d.h. relative Positionsangaben sind negativ.

Attribute

`assert|Position2: %DeltaPosition < %RelNullOffset`

=

Attribute

`assert|Position2: %DeltaPosition == %RelNullOffset`

>

Beschreibung

Der Motor steht rechts der Rel. Null, d.h. relative Positionsangaben sind positiv.

Attribute

`assert|Position2: %DeltaPosition > %RelNullOffset`

irrelevant

Attribute

`assert|Position2: %RelNullOffset == 0`

Hysteresis

Beschreibung

Das Motorspiel. Es ist nur im ini-File einstellbar.

Attribute

`ini|%ini_file|%ini_section|Hysteresis: %Hysteresis`

0

Attribute

`%Hysteresis: 0`

>0

Attribute

`%Hysteresis: 50`

Richtung*Beschreibung*

Richtung der letzten Bewegung.

*Attribute**ini*|%ini_file|%ini_section|Upwards: %Upwards

<==

Attribute

%Upwards: 0

==>

Attribute

%Upwards: 1

Achse**Achse***Beschreibung*

Die Achsenzuordnung erfolgt über den Namen des Motors.

*Attribute**ini*|%ini_file|%ini_section|Name: %axis

DF

Attribute

%axis: DF

Rotation

Attribute

%axis: Rotation

DC

Attribute

%axis: DC

X

Attribute

%axis: X

Y

Attribute

%axis: Y

Z

Attribute

%axis: Z

Collimator

Attribute

%axis: Collimator

Absorber

Attribute

%axis: Absorber

Monochromator

Attribute

%axis: Monochromator

Encoder

Attribute

%axis: Encoder

Omega

Attribute

%axis: Omega

Tilt

Attribute

%axis: Tilt

Position in Motorliste

Beschreibung

Die Position des Motors in der Motorenliste, wird durch den Namen des Abschnitts der ini-Datei bestimmt. Die Parameter des ersten Motors stehen im Abschnitt [Motor0].

einzigster

Beschreibung

Der Motor ist der einzige Motor in der Motorenliste.

Attribute

%ini_section: Motor0

letzter

Beschreibung

Der Motor ist der letzte Motor in einer Liste von wenigstens 2 Motoren.

Attribute

%ini_section: Motor1

ini| %ini_file| Motor0| Name: Dummy

ini| %ini_file| Motor0| Type: Any

Mitte

Beschreibung

Der Motor ist weder der erste noch der letzte Motor in einer Motorenliste mit wenigstens 3 Motoren.

Attribute

%ini_section: Motor1

ini| %ini_file| Motor0| Name: X

ini| %ini_file| Motor0| Type: TMotor

ini| %ini_file| Motor2| Name: Y

ini| %ini_file| Motor2| Type: TMotor

Typ

Beschreibung

Hardware-Typ des Motors.

C-812ISA

Attribute

ini| %ini_file| %ini_section| RamAddr: 0xD000

ini| %ini_file| %ini_section| Type: C-812ISA

BoardID

Attribute

ini| %ini_file| %ini_section| BoardID: %BoardID

1

Attribute

%BoardID: 1

2

Attribute

%BoardID: 2

3
Attribute
 %BoardID: 3

4
Attribute
 %BoardID: 4

C-832
Attribute
 in| %ini_file| %ini_section| Type: C-832

BoardID

Attribute
 in| %ini_file| %ini_section| BoardID: %BoardID

1
Attribute
 %BoardID: 1

0
Attribute
 %BoardID: 0

IndexLine

Beschreibung
 wohl nicht wirklich relevant, nur für Referenzpunktlauf
 nein

Beschreibung
 d.h. Referenzpunkt ist die Linke HW-Schranke; die Testfälle
 können bzgl. Position also wie beim C-812er konstruiert
 werden

Attribute
 in| %ini_file| %ini_section| IndexLine: 0

Ja
Attribute
 in| %ini_file| %ini_section| IndexLine: 1

Zustands-Überführung

Anforderung

Setzte Geschwindigkeit
Beschreibung
 testet mlSetValue(Speed)
Attribute
 step|1: TSTEP_SET_SPEED(%test|step_nr, %speed);

Wert

gültiger Wert

ist

groß
Attribute
 %speed: 15000

mittel
Attribute
 %speed: 5000

klein

Attribute
%speed: 100

0

Beschreibung
 Ist das ein gültiger Wert?

Attribute
%speed: 0

max.

Attribute
%speed: %MaxVelocity

ungültiger Wert

ist

> max.

Attribute
%speed: & %MaxVelocity + 1000

< 0

Attribute
%speed: -100

Setzte rel. Null

Beschreibung
 Testet Methode mSetRelativeZero.

Attribute
step|1: TSTEP_SET_RELZERO(%test|step_nr.1, %param1, %offset);
step|2: TSTEP_GET_RELOFFSET(%test|step_nr.2);

Parameter 1

Beschreibung
 Erster Parameter der Methode; gibt an ob die Relative Null gesetzt oder zurückgesetzt werden soll.

true

Attribute
%param1: 1

false (reset)

Beschreibung
 Zurücksetzten. Die Relative Null ist danach gleich der Absoluten Null.

Attribute
%param1: 0

Relative Null

Beschreibung
 Parameter, der die Verschiebung des Koordinatensystems der relativen Positionen zum aktuellen Koordinatensystem angibt.

Attribute
%newRelNullOffset: & %RelNullOffset + %offset

0

Attribute
%offset: 0

<0
Attribute
 %offset: -2000

>0
Attribute
 %offset: 3000

ExecuteCmd

Beschreibung
 TODO

Stoppe Bewegung

Beschreibung
 testet mStopDrive

Attribute
 step|1: TSTEP_STOP_MOTOR(%test|step_nr, %param);

Param1

false
Attribute
 %param: FALSE

true
Attribute
 %param: TRUE

Bewege zu Position

Beschreibung
 Testet mMoveToPosition oder mlMoveToPosition (s. Aspekt Methode) sowie mIsMoveFinish und mlIsMoveFinish (während Wartens auf Stillstand).

Attribute
 %newDeltaPosition: & (%newDeltaPosition1 > %PositionMax) ?
 %PositionMax : ((%newDeltaPosition1 < %PositionMin) ?
 %PositionMin : %newDeltaPosition1)
 %newDeltaPosition1: & %DeltaPosition + (%target_dir *
 %target_dist)
 %target_pos: & %DeltaPosition + (%target_dir * %target_dist) -
 %RelNullOffset

Verhältnis Ziel-Position zu SW-Schranke

Beschreibung
 meint SW-Schranke in Fahrtrichtung

innerhalb

Beschreibung
 Zielposition liegt vor der SW-Schranke in Fahrtrichtung.

Attribute
 assert|MoveToPos3: (%target_dir == 1) ? (%target_pos <=
 %PositionMax) : (%target_pos >= %PositionMin)

Distanz

Beschreibung
 Zurückzulegende Strecke.

0
Attribute
 %target_dist: 0

klein

Attribute
`%target_dist: 500`

mittel

Attribute
`%target_dist: 1000`

sehr klein

Attribute
`%target_dist: 100`

gross

Attribute
`%target_dist: 3000`

sehr groß

Attribute
`%target_dist: 5000`

max.

Beschreibung
 Zielposition liegt genau auf der SW-Schranke in Fahrtrichtung.

Attribute
`%target_dist: & (%target_dir > 0) ?
 abs(%PositionMax - %DeltaPosition) :
 abs(%PositionMin - %DeltaPosition)`

dahinter

Beschreibung
 Zielposition liegt hinter der SW-Schranke in Fahrtrichtung.

Attribute
`%target_dist: & (%target_dir > 0) ? abs(%PositionMax +
 1000 - %DeltaPosition) : abs(%PositionMin - 1000 -
 %DeltaPosition)`

ungültig

Beschreibung
 Ungültige Softwareschranken; d.h. die Softwareschranken liegen hinter den Hardwareschranken.

Verhältnis zu HW-Schranke

Beschreibung
 meint HW-Schranke in Fahrtrichtung

dahinter

Beschreibung
 Die SW-Schranke in Fahrtrichtung ist ungültig, die Zielposition liegt hinter der HW-Schranke.

Attribute
`%target_dist: 20000`

Verhältnis Ziel-Position zur relativen Null

Beschreibung
 Nur relevant wenn relative Null gesetzt.

<

Attribute
`assert|MoveToPos1: %RelNullOffset !=0 && %target_pos <
 0`

```

=
  Attribute
    assert|MoveToPos1: %RelNullOffset !=0 && %target_pos
      == 0
>
  Attribute
    assert|MoveToPos1: %RelNullOffset !=0 && %target_pos >
      0

```

irrelevant

Beschreibung
wenn relative Null nicht gesetzt

Attribute
assert|MoveToPos1: %RelNullOffset == 0

Verhältnis Ziel-Position zur Absoluten Null

```

<
  Attribute
    assert|MoveToPos2: %target_pos + %RelNullOffset < 0
=
  Attribute
    assert|MoveToPos2: %target_pos + %RelNullOffset == 0
>
  Attribute
    assert|MoveToPos2: %target_pos + %RelNullOffset > 0

```

Richtung

Beschreibung
Liegt Ziel-Position links oder rechts der Ist-Position?

```

<==
  Attribute
    %target_dir: -1
    assert|Richtung: %newDeltaPosition <= %DeltaPosition
==>
  Attribute
    %target_dir: 1
    assert|Richtung: %newDeltaPosition >= %DeltaPosition

```

Methode

```

m
  Beschreibung
    Test der Methode mMoveToPosition; es ist sicher zu stellen, dass die zu testenden Achse aktiviert wurde (SetAxis).
  Attribute
    %methode: 1
    %target_axis: %axis
    step|1: TSTEP_MOVE_TO_POSITION1(%test|step_nr.1,
      %target_pos);

```

ml

Beschreibung
Test der Methode mlMoveToPosition.

Attribute

```
%methode: 2
step|1: TSTEP_MOVE_TO_POSITION2(%test|step_nr.1,
    %target_axis, %target_pos);
```

Achse

vorhanden

Attribute

```
%target_axis: %axis
```

nicht vorhanden

Attribute

```
%target_axis: DF
```

```
assert|38: '%axis' ne '%target_axis'
```

auf Stillstand warten

Beschreibung

Soll nach dem auslösen der Bewegung auf deren Beendigung gewartet werden?

ja

Beschreibung

%timeout legt die max. Wartezeit in Millisekunden fest.

Attribute

```
%timeout: 10000
```

```
step|2: & (%methode == 1) ?
```

```
'TSTEP_WAIT_UNTIL_MOVE_FINISHED1(
    %test|step_nr.2, %timeout);' :
```

```
'TSTEP_WAIT_UNTIL_MOVE_FINISHED2(
    %test|step_nr.2, %target_axis, %timeout);'
```

nein

Bewege um Distanz

Beschreibung

testet mMoveByDistance sowie mIsMoveFinish (während Wartens auf Stillstand)

Attribute

```
%distance: & %target_dir * %target_dist
```

```
%newDeltaPosition: & (%newDeltaPosition1 > %PositionMax) ?
```

```
%PositionMax : ((%newDeltaPosition1 < %PositionMin) ?
```

```
%PositionMin : %newDeltaPosition1)
```

```
%newDeltaPosition1: & %DeltaPosition + (%target_dir *
    %target_dist)
```

```
%target_pos: & %DeltaPosition + (%target_dir * %target_dist) -
    %RelNullOffset
```

```
step|1: TSTEP_MOVE_BY_DISTANCE1(%test|step_nr.1,
    %distance);
```

Richtung

Beschreibung

Liegt Ziel-Position links oder rechts der Ist-Position?

<==

Attribute

```
%target_dir: -1
```

```
assert|Richtung: %newDeltaPosition <= %DeltaPosition
```

==>

*Attribute**%target_dir: 1**assert|Richtung: %newDeltaPosition >= %DeltaPosition***Verhältnis Ziel-Position zu SW-Schranke***Beschreibung*

meint SW-Schranke in Fahrtrichtung

innerhalb

Beschreibung

Zielposition liegt vor der SW-Schranke in Fahrtrichtung.

*Attribute**assert|MoveByDist3: (%target_dir == 1) ? (%target_pos
<= %PositionMax) : (%target_pos >= %PositionMin)***Distanz***Beschreibung*

Zurückzulegende Strecke.

0

*Attribute**%target_dist: 0*

sehr klein

*Attribute**%target_dist: 100*

klein

*Attribute**%target_dist: 500*

mittel

*Attribute**%target_dist: 1000*

gross

*Attribute**%target_dist: 3000*

sehr groß

*Attribute**%target_dist: 5000*

max.

Beschreibung

Zielposition liegt genau auf der SW-Schranke in Fahrtrichtung.

*Attribute**%target_dist: & (%target_dir > 0) ?
abs(%PositionMax - %DeltaPosition) :
abs(%PositionMin - %DeltaPosition)*

dahinter

Beschreibung

Zielposition liegt hinter der SW-Schranke in Fahrtrichtung.

*Attribute**%target_dist: & ((%target_dir > 0) ? abs(%PositionMax +
1000 - %DeltaPosition) : abs(%PositionMin - 1000 -
%DeltaPosition))*

ungültig

Beschreibung

Ungültige Softwareschranken; d.h. die Softwareschranken liegen hinter den Hardwareschranken.

Verhältnis zu HW-Schranke

Beschreibung

meint HW-Schranke in Fahrtrichtung

dahinter

Beschreibung

Die SW-Schranke in Fahrtrichtung ist ungültig, die Zielposition liegt hinter der HW-Schranke.

Attribute

%target_dist: 20000

Verhältnis Ziel-Position zur relativen Null

Beschreibung

nur relevant wenn relative Null gesetzt

irrelevant

Beschreibung

Nur relevant wenn relative Null gesetzt.

Attribute

assert|MoveByDist1: %RelNullOffset == 0

<

Attribute

assert|MoveByDist1: %RelNullOffset !=0 && %target_pos < 0

=

Attribute

assert|MoveByDist1: %RelNullOffset !=0 && %target_pos == 0

>

Attribute

assert|MoveByDist1: %RelNullOffset !=0 && %target_pos > 0

Verhältnis Ziel-Position zur Absoluten Null

<

Attribute

assert|MoveByDist2: %target_pos + %RelNullOffset < 0

=

Attribute

assert|MoveByDist2: %target_pos + %RelNullOffset == 0

>

Attribute

assert|MoveByDist2: %target_pos + %RelNullOffset > 0

auf Stillstand warten

Beschreibung

Soll nach dem auslösen der Bewegung auf deren Beendigung gewartet werden?

ja

Beschreibung

```

        %timeout legt die max. Wartezeit in Millisekunden fest.
        Attribute
        %timeout: 10000
        step|2:
            TSTEP_WAIT_UNTIL_MOVE_FINISHED1(%test|step_nr.2,
            %timeout);
    nein

```

Sichere Zustand

Beschreibung
testet mlSaveModuleSettings

Attribute
step|1: TSTEP_SAVE_MODULE_SETTINGS(%test|step_nr);

setzt aktive Achse

Beschreibung
Tested mlSetAxis.

Attribute
step|1: TSTEP_SET_AXIS(%test|step_nr, %activate_axis);

Achse

DF
Attribute
%activate_axis: DF

Rotation
Attribute
%activate_axis: Rotation

DC
Attribute
%activate_axis: DC

X
Attribute
%activate_axis: X

Y
Attribute
%activate_axis: Y

Z
Attribute
%activate_axis: Z

Collimator
Attribute
%activate_axis: Collimator

Absorber
Attribute
%activate_axis: Absorber

Omega
Attribute
%activate_axis: Omega

Monochromator
Attribute
%activate_axis: Monochromator

Tilt
Attribute
 %activate_axis: Tilt

Encoder
Attribute
 %activate_axis: Encoder

lies Position
Beschreibung
 Testet mlGetDistance und mGetDistance.
Attribute
 step|1: & (%methode == 1) ?
 'TSTEP_GET_POSITION1(%test|step_nr)' :
 'TSTEP_GET_POSITION2(%test|step_nr, %target_axis)'

Methode

m
Beschreibung
 Bestimmen der Position über mGetDistance.
Attribute
 %methode: 1
 %target_axis: %axis

ml
Beschreibung
 Bestimmen der Position über mlGetDistance.
Attribute
 %methode: 2

Achse

vorhanden
Attribute
 %target_axis: %axis

nicht vorhanden (Fehler)
Attribute
 %target_axis: DF
 assert|38: '%axis' ne '%target_axis'

Warte
Attribute
 %time: 1000

Methode

m
Attribute
 step|1: TSTEP_WAIT1(%test|step_nr, %time)

ml
Attribute
 step|1: TSTEP_WAIT2(%test|step_nr, %axis, %time)

5.3 Testpaket *Referenzpunktlauf (m_rpl)*

Testobjekt

Testobjekt ist der Dialog “Grundstellung anfahren”, der die Funktion F2 der Verhaltensspezifikation und deren Teilfunktionen realisiert und in der Motorenkomponente `motors.dll` implementiert ist.

Eingabedaten

Variable Eingaben sind zum einen eine Auswahl der Konfigurationsparameter, die im bezug auf die getestete Funktion relevant sind, sowie zum anderen die Nutzereingaben in den Steuerelementen des Dialoges.

Die relevanten Parameter, die im Test variiert werden, sind im `MotorX`-Abschnitt: `Type`, `Name`, `InitialMove`, `Velocity`, `MaxVelocity`, `Acceleration`, `DistanceToZero`, `DeltaPosition`, `IndexLine`, `MoveFirstToLimit`, `InitialAngle` und aus dem `MOTORSIM`-Abschnitt: `IndexPosition`.

Ausgabedaten

Ausgaben sind a) die Steuerungsanweisungen der Motorenkomponente für die Motorenhardware, b) die Konfigurationsdatei nach Beenden des Testes sowie c) einige Meldungen auf dem Dialog bzw. Message-Boxen die über Fortschritt und Status des Funktionsausführung informieren.

Klassifikationsbaum

Die zwei Hauptaspekte der Eingabedaten sind, wie gesagt, die Konfiguration der Motorenkomponente (gegeben in der Konfigurationsdatei) und die Nutzereingaben. Entsprechend ist der Klassifikationsbaum auf der ersten Ebene gegliedert. S. Abb. 5.26 und 5.27.

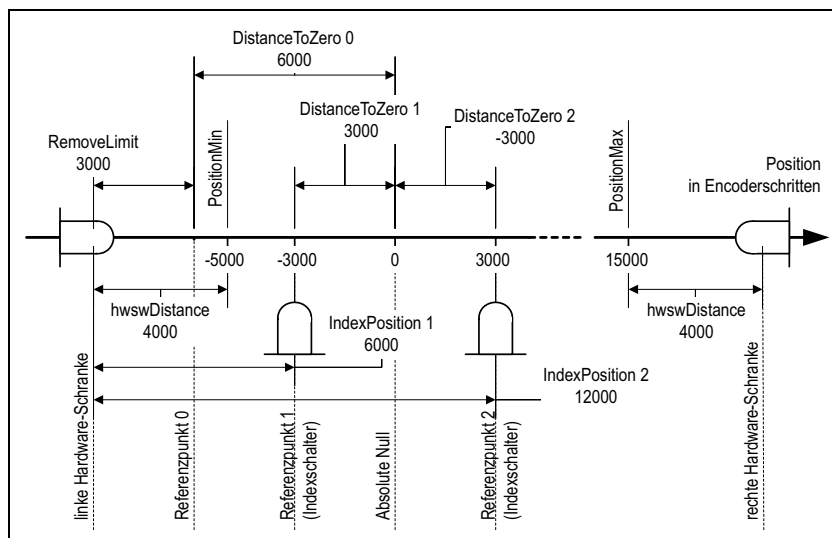


Abbildung 5.25: Angenommener Versuchsaufbau für die Testfälle

In der Konfiguration sind folgende Aspekte relevant: Hardware des Motors (`Type`) und die Achse, die der Motor im Versuchsaufbau einnimmt (`Name`, `Achse`); die Bewegungsparameter Geschwindigkeit und Beschleunigung; die

so genannte Grundstellung (**InitialAngle**); die Position des Motors in der Motorenliste; die Lage des Referenzpunktes (RP) und die Ausgangsposition des Motors. Zur Lage des RP werden im Test die folgenden drei verschiedenen Versuchsaufbauten angenommen (s. Abb. 5.25):

Aufbau 0 Versuchsaufbau ohne speziellen Indexschalter; RP ist die Position der linken Hardware-Schranke plus **RemoveLimit**; festgelegt durch **IndexLine=0** und **DistanceToZero0**; gilt für Motoren des Typs C-812 und C-832.

Aufbau 1 Versuchsaufbau mit speziellen Indexschalter der links der absoluten Null liegt; festgelegt durch **IndexLine=1**, **IndexPosition1** und **DistanceToZero1**; gilt nur für Motoren des Typs C-832.

Aufbau 2 Versuchsaufbau mit speziellen Indexschalter der rechts der absoluten Null liegt; festgelegt durch **IndexLine=1** und **IndexPosition2** und **DistanceToZero2**; gilt nur für Motoren des Typs C-832.

Referenzpunktlauf

Attribute

```
%ini_file: hardware.ini
ScriptGenerator: m_rpl
inj| %ini_file|MOTORSIM|LogFile: %test_id.LOG
inj| %ini_file|MOTORSIM|LogLevel: 2
inj| %ini_file|MOTORSIM|StatusWindow: 1
inj| %ini_file|MOTORSIM|dll: msim.dll
```

Aktion

Beschreibung

Aktion des Nutzers auf dem Dialog.

Click auf 'Keep Position'

Beschreibung

Der Nutzer klickt die Option 'Keep Position' an.

Attribute

```
step|1: TSTEP_CLICK_KEEP_POSITION(%test|step_nr);
```

Click auf 'All Drives'

Beschreibung

Der Nutzer klickt die Option 'Handle all Drives' an.

Attribute

```
step|1: TSTEP_CLICK_ALL_DRIVES(%test|step_nr);
```

Click auf 'Keep Cal. Data'

Beschreibung

Der Nutzer klickt die Option 'Keep Calibration Data' an.

Attribute

```
step|1: TSTEP_CLICK_KEEP_DATA(%test|step_nr);
```

Click auf 'Set abs. Zero'

Beschreibung

Der Nutzer klickt auf den Button 'Set Absolute Zero'.

Attribute

```
step|1: TSTEP_CLICK_SET_ABS_ZERO(%test|step_nr);
```

Click auf 'RPL'

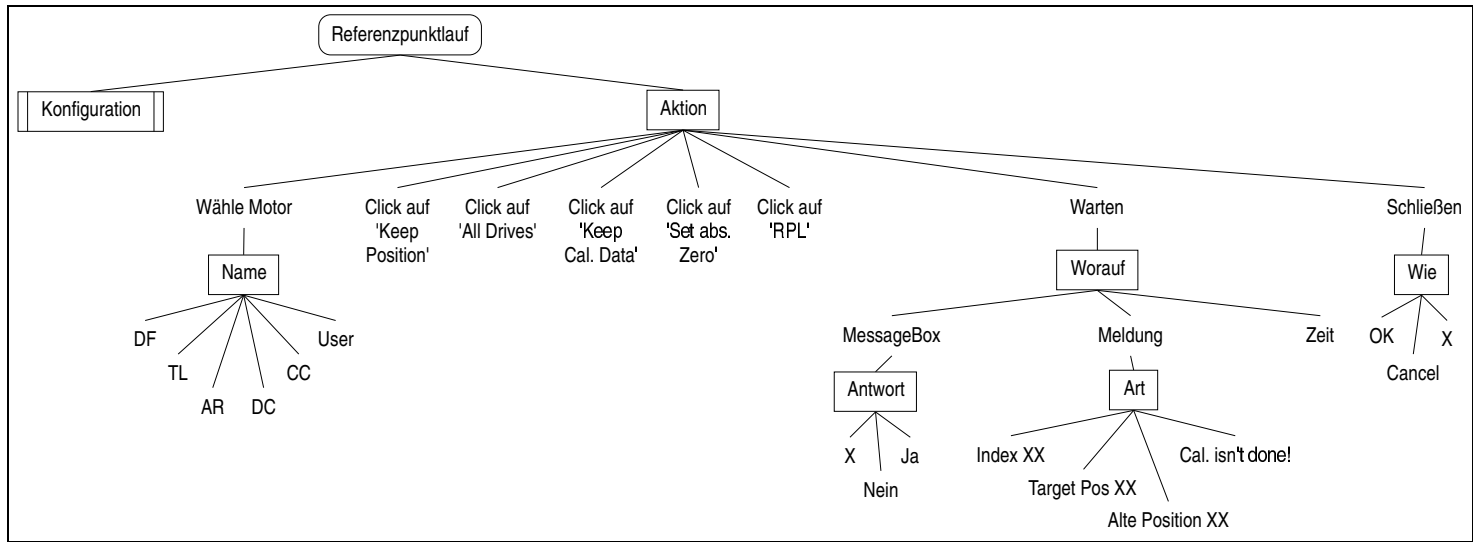


Abbildung 5.26: Klassifikationsbaum (m_rpl)

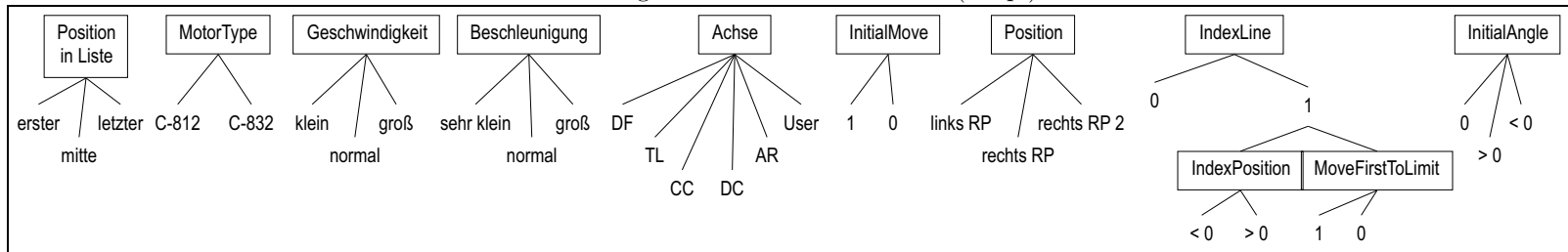


Abbildung 5.27: Klassifikationsbaum (m_rpl|Konfiguration)

Beschreibung

Der Nutzer klickt auf den Button 'Reference Run'.

Attribute

step|1: TSTEP_CLICK_RPL(%test|step_nr);

Warten

Worauf

Beschreibung

Die Zeit die max. gewartet werden soll ist als Variablen-Attribute %time in Sekunden (am entsprechenden Teststep) zu spezifizieren

Attribute

%time: 100

MessageBox

Beschreibung

Warten auf eine Messgebox mit dem Titel 'Message' oder 'Meldung' die das Ende des Referenzpunktllaufes signalisiert bzw. kund tut, dass dieser nicht möglich ist, bzw. fragt ob die Kalibrierung durchgeführt werden soll.

Antwort

X

Beschreibung

Der Nutzer schließt die MessageBox über X-Button in Titelleiste.

Attribute

step|1: TSTEP_WAIT_FOR_MSGBOX_CLOSE(
%test|step_nr, %time);

Ja

Beschreibung

Der Nutzer schließt die MessageBox über Ja-Button (IDYES).

Attribute

step|1: TSTEP_WAIT_FOR_MSGBOX_YES(
%test|step_nr, %time);

Nein

Beschreibung

Der Nutzer schließt die MessageBox über Ja-Button (IDNO).

Attribute

step|1: TSTEP_WAIT_FOR_MSGBOX_NO(
%test|step_nr, %time);

Meldung

Beschreibung

Warten auf eine Meldung in dem Static-Element des Dialoges das Informationen über den Fortschritt des RPL enthält. XX steht jeweils für den Namen der Achse.

Art

Index XX

Attribute

step|1: TSTEP_WAIT_FOR_MESSAGE(%test|step_nr
,Index %test|pred_step|selected_axis, %time);

Target Pos XX

Attribute
step|1: TSTEP_WAIT_FOR_MESSAGE(%test|step_nr
,Target Pos %test|pred_step|selected_axis, %time);

Alte Position XX

Attribute
step|1: TSTEP_WAIT_FOR_MESSAGE(%test|step_nr
,Alte Position %test|pred_step|selected_axis,
%time);

Cal. isn't done!

Attribute
step|1: TSTEP_WAIT_FOR_MESSAGE_1(%test|step_nr,
%time);

Zeit

Beschreibung
Warten auf ablauf einer Zeitspanne.

Attribute
step|1: TSTEP_WAIT(%test|step_nr, %time);

Schließen

Beschreibung
Der Nutzer schließt den Dialog.

Wie

OK

Beschreibung
Schließen des Dialoges über OK-Button.

Attribute
step|1: TSTEP_CLICK_OK(%test|step_nr);

Cancel

Beschreibung
Schließen des Dialoges über Candel-Button.

Attribute
step|1: TSTEP_CLICK_CANCEL(%test|step_nr);

X

Beschreibung
Schließen des Dialoges über X-Button in Titelleiste.

Attribute
step|1: TSTEP_CLICK_CLOSE(%test|step_nr);

Wähle Motor

Beschreibung
Der Nutzer wählen einen Motors in der Motorenliste aus.

Name

Beschreibung
Achsen der Topographie als Auswahl.

Attribute
%selected_axis: %select_motor
step|1: TSTEP_SELECT_MOTOR(%test|step_nr, %select_motor);

DF

Attribute
%select_motor: DF

TL

Attribute

%select_motor: TL

AR
Attribute
%select_motor: AR

DC
Attribute
%select_motor: DC

CC
Attribute
%select_motor: CC

User
Beschreibung
Ein Motor mit nicht vordefiniertem Namen, d.h. ohne Achsen Zuordnung.
Attribute
%select_motor: User

Konfiguration

Beschreibung

Konfiguration der Motorenkomponente im allgemeinen und des Motors, für den der RPL durchgeführt werden soll, im speziellen.

MotorType

Attribute

ini|%ini_file|%ini_section|Type: %MotorType

C-812

Attribute

%MotorType: C-812ISA

ini|%ini_file|%ini_section|BoardId: 2

C-832

Attribute

%MotorType: C-832

ini|%ini_file|%ini_section|BoardID: 1

Geschwindigkeit

Beschreibung

Für den RPL wird die maximale Geschwindigkeit verwendet, deshalb wird hier MaxVelocity variiert und Velocity auf einen kleineren Wert gesetzt um zu sehen wann welcher Wert verwendet wird.

Attribute

ini|%ini_file|%ini_section|MaxVelocity: %MaxVelocity

ini|%ini_file|%ini_section|Velocity: %Velocity

klein

Attribute

%MaxVelocity: 200

%Velocity: 100

normal

Attribute

%MaxVelocity: 8000

%Velocity: 7000

groß

Attribute
 %MaxVelocity: 15000
 %Velocity: 10000

Beschleunigung

Attribute
ini | %ini_file | %ini_section | Acceleration: %Acceleration
 sehr klein

Attribute
 %Acceleration: 10

normal

Attribute
 %Acceleration: 5000

groß

Attribute
 %Acceleration: 20000

IndexLine

Attribute
ini | %ini_file | %ini_section | IndexLine: %IndexLine
 0

Beschreibung
 Kein spezieller Schalter für den RP vorhanden; RP ist die linke
 HW-Schranke plus RemoveLimit.

Attribute
 %DistanceToZero: 6000
 %IndexLine: 0
 %IndexPosition: & -%DistanceToZero

1

Beschreibung
 Spezieller Schalter für den RP vorhanden.
 Nur sinnvoll wenn MotorType == C-832

Attribute
 %IndexLine: 1
 assert | IndexLine1: '%MotorType' eq 'C-832'

MoveFirstToLimit

1

Beschreibung
 Der Motor fährt zuerst zur linken Hardwareschranke und
 sucht von dort nach rechts fahrend den Index-Schalter.

Attribute
ini | %ini_file | %ini_section | MoveFirstToLimit: 1

0

Beschreibung
 Der Motor beginnt sofort nach links fahrend den Index-
 Schalter zu suchen. Fährt er auf die linke HW-Schranke
 ohne ihn zu finden sucht er ihn nach rechts fahrend.

Attribute
ini | %ini_file | %ini_section | MoveFirstToLimit: 0

IndexPosition

Beschreibung

Lage des Index-Schalters.

Attribute

ini|%ini_file|MOTORSIM|IndexPosition: %IndexPosition

< 0

Beschreibung

Index-Schalter liegt links der absoluten Null.

Attribute

%DistanceToZero: 3000

%IndexPosition: 6000

> 0

Beschreibung

Index-Schalter liegt rechts der absoluten Null.

Attribute

%DistanceToZero: -3000

%IndexPosition: 12000

Position in Liste

Beschreibung

Position des Motors der getestet werden soll in Liste von 5 Motoren. Die Position wird durch den Namen des ini-File-Abschnitts bestimmt (%ini_section). Motor0 ist der Name des Abschnitts des ersten Motors.

Die Attribute legen die Konfigurationsparameter der anderen Motoren fest (%ini_section_x). Die Namen der ini-File-Abschnitte werden durch die einzelnen Klassen festgelegt.

Der Motor im Abschnitt Motor0 ist der, der in der Motoren-Listbox standardmäßig ausgewählt ist (%selected_axis).

Attribute

ini|%ini_file|%ini_section_a|BoardID: 1

ini|%ini_file|%ini_section_a|DistanceToZero: 500

ini|%ini_file|%ini_section_a|InitialMove: 1

ini|%ini_file|%ini_section_a|Name: & ('%axis' eq 'DF') ? 'AR' : 'DF'

ini|%ini_file|%ini_section_a|PositionMax: 1000

ini|%ini_file|%ini_section_a|PositionMin: -1000

ini|%ini_file|%ini_section_a|RemoveLimit: 2000

ini|%ini_file|%ini_section_a|Type: C-812ISA

ini|%ini_file|%ini_section_b|BoardID: 3

ini|%ini_file|%ini_section_b|DistanceToZero: 1000

ini|%ini_file|%ini_section_b|InitialMove: 1

ini|%ini_file|%ini_section_b|Name: & ('%axis' eq 'TL') ? 'AR' : 'TL'

ini|%ini_file|%ini_section_b|PositionMax: 1000

ini|%ini_file|%ini_section_b|PositionMin: -1000

ini|%ini_file|%ini_section_b|RemoveLimit: 2000

ini|%ini_file|%ini_section_b|Type: C-812ISA

ini|%ini_file|%ini_section_c|BoardID: 4

ini|%ini_file|%ini_section_c|DistanceToZero: 1500

ini|%ini_file|%ini_section_c|InitialMove: 1

ini|%ini_file|%ini_section_c|Name: & ('%axis' eq 'CC') ? 'AR' : 'CC'

ini|%ini_file|%ini_section_c|PositionMax: 1000

ini|%ini_file|%ini_section_c|PositionMin: -1000

ini|%ini_file|%ini_section_c|RemoveLimit: 2000


```

ini| %ini_file| %ini_section_c| Type: C-812ISA
ini| %ini_file| %ini_section_d| BoardID: 0
ini| %ini_file| %ini_section_d| DistanceToZero: 2000
ini| %ini_file| %ini_section_d| InitialMove: 1
ini| %ini_file| %ini_section_d| Name: & ('%axis' eq 'DC') ? 'AR' : 'DC'
ini| %ini_file| %ini_section_d| PositionMax: 1000
ini| %ini_file| %ini_section_d| PositionMin: -1000
ini| %ini_file| %ini_section_d| RemoveLimit: 2000
ini| %ini_file| %ini_section_d| Type: C-832

```

erster

Beschreibung

Motor steht an erster Position in der Motorenliste.

Attribute

```

%ini_section: Motor0
%ini_section_a: Motor1
%ini_section_b: Motor2
%ini_section_c: Motor3
%ini_section_d: Motor4
%selected_axis: %axis

```

mitte

Beschreibung

Motor steht weder an erster noch an letzter Position in der Motorenliste.

Attribute

```

%ini_section: Motor1
%ini_section_a: Motor0
%ini_section_b: Motor2
%ini_section_c: Motor3
%ini_section_d: Motor4
%selected_axis: & ('%axis' eq 'DF') ? 'AR' : 'DF'

```

letzter

Beschreibung

Motor steht an letzter Position in der Motorenliste.

Attribute

```

%ini_section: Motor4
%ini_section_a: Motor0
%ini_section_b: Motor1
%ini_section_c: Motor2
%ini_section_d: Motor3
%selected_axis: & ('%axis' eq 'DF') ? 'AR' : 'DF'

```

Achse

Beschreibung

Name des Motors für den der RPL durchgeführt werden soll.

Attribute

```
ini| %ini_file| %ini_section| Name: %axis
```

DF

Attribute

```
%axis: DF
```

TL

Attribute

%axis: TL
AR
Attribute
%axis: AR
User
Beschreibung
Motor mit nutzerdefiniertem Namen, der keine Achsenzuordnung erzeugt.
Attribute
%axis: User
DC
Attribute
%axis: DC
CC
Attribute
%axis: CC

InitialMove

1
Attribute
ini|%ini_file|%ini_section|InitialMove: 1
0
Attribute
ini|%ini_file|%ini_section|InitialMove: 0

InitialAngle

Beschreibung
Absolute Position die nach dem RPL eingenommen werden soll.
Attribute
ini|%ini_file|%ini_section|InitialAngle: %InitialAngle
0
Attribute
%InitialAngle: 0
< 0
Attribute
%InitialAngle: -100
> 0
Attribute
%InitialAngle: 200

Position

Beschreibung
Position des Motors relativ zum RP.
Attribute
ini|%ini_file|%ini_section|DeltaPosition: %DeltaPosition
ini|%ini_file|%ini_section|DistanceToZero: %DistanceToZero
ini|%ini_file|%ini_section|PositionMax: 15000
ini|%ini_file|%ini_section|PositionMin: -5000
ini|%ini_file|%ini_section|RemoveLimit: 3000
ini|%ini_file|MOTORSIM|hswDistance: 4000
rechts RP 2

Beschreibung

Der Motor steht (weiter) rechts vom RP.

Attribute

`%DeltaPosition: & (-(%DistanceToZero) + 7000)`

links RP

Beschreibung

Der Motor steht links vom RP.

Attribute

`%DeltaPosition: & (-(%DistanceToZero) - 1000)`

`_assert|Position1: %IndexLine == 1`

rechts RP

Beschreibung

Der Motor steht rechts vom RP.

Attribute

`%DeltaPosition: & (-(%DistanceToZero) + 2000)`

5.4 Die Testergebnisse

Die Bewertung des Ist-Verhaltens der Motorenkomponente bei den entworfenen Testfällen hat eine Reihe von Fehlern bzw. Fehlergruppen sichtbar gemacht. Diese werden im Folgenden in der Reihenfolge ihrer Schwere aufgelistet und kurz erläutert. Die Fehlernummern beziehen sich auf die Fehlerdokumentation in den Entwicklerdokumenten.

“Stop smoothly“-Fehler (Nr. 6)

Die Implementation für den C-832er Motor verwendet an einigen Stellen zum stoppen des Motors den “Stop smoothly“-Befehl an statt des “Stop abruptly“-Befehls. D.h. es wird mit der programmierten an statt mit der maximalen Entschleunigung gebremst. Dies kann dazu führen – v.a. beim Referenzpunktlauf, wo *nach* dem Erreichen einer Hardware-Schranke der Motor zu stoppen ist – dass der Motor zu weit gegen die Endlagenschalter fährt.

fehlende Prüfung der Konfigurationsparameter (Nr. 11)

Die Parameter in der Konfigurationsdatei werden an fast keiner Stelle auf die Einhaltung von Wertebereichen bzw. auf Plausibilität überprüft. Das führt dazu, dass unsinnige Konfigurationen möglich sind und die Hardware möglicherweise mit unzulässigen Kommandos beschickt wird. Diese Fehler sind in fast allen Testfällen des Paketes `m_init` zu finden.

Es ist anzunehmen, dass in den Konfigurationsdaten an den realen Arbeitsplätzen Fehler enthalten sind, aber auch, dass die Entwickler mit falschen Konfigurationen arbeiten und testen¹.

Als Wenigstes müsste eine Überprüfung der Wertebereiche entsprechend den Angaben in Kapitel 3.2 der Verhaltensspezifikation vorgenommen werden. Grundlage für eine Plausibilitätskontrolle könnten Regeln in der Art des Abschnittes “Beziehungen/Bedingungen” im genannten Kapitel sein.

¹ Z.B. liegt im aktuellen CVS-Repository eine Konfigurationsdatei, in der es zwei Motor-Abschnitte für den gleichen C-832er Motor gibt.

fehlerhafte Hysteresisbehandlung (Nr. 10, 4)

Die Behandlung des Motorspiels (s. F1.3 in der Verhaltensspezifikation) ist fehlerhaft. Zur Erläuterung siehe in den *targetline*-Daten der Testfälle ML0002, ML0003, ML0006, ML0007, ML0008 und ML0013. Der Fehler führt zu Positionierungsfehlern in der Größe des Motorspiels.

Endlosschleife bei Suche nach fehlendem Index-Schalter (Nr. 12)

Wenn ein C-832er Motor zwar laut Konfigurationsdatei einen Index-Schalter hat dieser aber nicht anspricht oder nicht vorhanden ist, gerät der Referenzpunktlauf in eine Endlosschleife, die aber über den Cancel-Button zu unterbrechen ist. s. Testfall RPL022.