

JetMove 200

JetSym-Funktionen

Benutzer-Information



Dieses Dokument besitzt nur in Verbindung mit dem zugehörigen Dokument zu den Sicherheitshinweisen volle Gültigkeit.

Die Firma JETTER AG behält sich das Recht vor, Änderungen an ihren Produkten vorzunehmen, die der technischen Weiterentwicklung dienen. Diese Änderungen werden nicht notwendigerweise in jedem Einzelfall dokumentiert.

Dieses Handbuch und die darin enthaltenen Informationen wurden mit der gebotenen Sorgfalt zusammengestellt. Die Firma JETTER AG übernimmt jedoch keine Gewähr für Druckfehler oder andere daraus entstehende Schäden.

Die in diesem Buch genannten Marken und Produktnamen sind Warenzeichen oder eingetragene Warenzeichen der jeweiligen Titelhälter.

Inhaltsverzeichnis

1	Einleitung	4
1.1	Produktbeschreibung	4
1.2	Systemvoraussetzungen	4
2	Software und Programmierung	5
2.1	Einbindung der Funktionen in ein Projekt	5
2.2	Funktionen	6
2.2.1	JM_MotionLoadParameter	6
2.2.2	JM_MotionPower	6
2.2.3	JM_MotionConfigHoming	7
2.2.4	JM_MotionHome	7
2.2.5	JM_MotionMovePtp	8
2.2.6	JM_MotionMoveVelocity	9
2.2.7	JM_MotionStop	9
2.2.8	JM_MotionClearError	10
2.2.9	JM_MotionReadStatus	10
2.2.10	JM_MotionReadErrorType	11
2.3	Beispiel	15

1 Einleitung

**Diese Benutzer-Information beschreibt die Funktionalität des Produktes
JetMove 200 mit der Betriebssystem-Version V2.02.**

Dieses Dokument beschreibt die Verwendung der bereit gestellten Funktionen für die Programmiersprache JetSym der Jetter AG. Diese Funktionen sollten benutzt werden, wenn **nicht** die neue Programmiersprache JetSym-ST verwendet wird.

Bei der Programmiersprache JetSym-ST werden die Anwendungen mit Motion-Befehlen programmiert, bei Programmiersprache JetSym können die hier beschriebenen JetSym-Funktionen benutzt werden.

Wenn die Funktionen unverändert benutzt werden, so können **keine** Gleitkommazahlen benutzt werden, da die Parameter von JetSym-Funktionen nur Ganzzahlwerte sein dürfen.

Ergänzend zu diesem Dokument sind die Dokumentationen zu den JetMoves und die Beschreibung der Online-Hilfe gültig.

1.1 Produktbeschreibung

Mit Hilfe der JetSym-Funktionen kann ein Anwenderprogramm erstellt werden, das die speicherprogrammierbar Steuerung im Betrieb abarbeitet. Die JetMove sind Leistungsendstufen für Servomotoren oder ähnlicher Motoren.

1.2 Systemvoraussetzungen

Folgende Software-Versionen werden benötigt.

Software-Versionen	
Steuerung / Software	ab SW-Version
JetControl 24x	V3.11
JetControl 647	V3.00
JetMove 2xx	V2.02
JetSym	V2.2

2 Software und Programmierung

2.1 Einbindung der Funktionen in ein Projekt

Mit einem #INCLUDE-Befehl kann die Programmdatei und die Symboldefinitionen mit den JetSym-Funktionen in ein Projekt eingebunden werden:

z.B.:

Programmdatei:

```
#include "Lib\JM_AX_Functions.spf"
```

Symboldatei:

```
#include "Lib\JM_Lib.ssf"
```

Zur Definition der verwendeten Achse ist eine weitere Symboldefinition nötig:

z.B.:

Definition:

```
axTest 12000
```

Am Beispiel des Registers REG **1x000** wird demonstriert, nach welchem Schema die Registerdefinition erfolgt.

- Die Registernummer ist eine fünfstellige Nummer.
- Die erste Ziffer ist 1.
- Die zweite Ziffer x ergibt sich aus der Modulnummer, an dem sich das Modul JetMove befindet.
- Die Ziffern drei, vier und fünf sind 0.



Hinweis!

Zur Ermittlung der Modulnummer werden nur die intelligenten Module gezählt, nicht jedoch nichtintelligenten Module, vgl. nachfolgende Tabelle.

Nummern-
belegung
der
Module

JC-24x Grundgerät	JetMove 200	JX2-ID8 Eingangs- modul	JetMove 200
Modul 1	Modul 2	Modul 3	Modul 4
Eingänge 101 .. 108	Registernummer 12000	Eingänge 201 ..208	Registernummer 13000

Parameter nicht ändern:

Wenn ein Parameter einer Funktion nicht verändert werden soll, so kann das Define „x“ an dessen Stelle benutzt werden.

Beispiel:

```
JM_MotionPower (axTest, JM_FUNC_POWER_ENABLE, x)
```

2.2 Funktionen

Folgende Funktionen stehen zur Verfügung:

```
JM_MotionLoadParameter
JM_MotionPower
JM_MotionConfigHoming
JM_MotionHome
JM_MotionMovePtp
JM_MotionMoveVelocity
JM_MotionStop
JM_MotionClearError
JM_MotionReadStatus
JM_MotionReadErrorType
JM_MotionWriteParameter
JM_MotionWriteParameterBit
JM_MotionReadParameter
```

2.2.1 JM_MotionLoadParameter

```
DEF_FUNCTION (JM_MotionLoadParameter, jmloadpara)
Par:  AxisOffset
```

Diese Funktion lädt ein Parameter-File, das mit der Motion-Setup erstellt und mit FTP auf die JetControl übertragen wurde, in die entsprechende JetMove-Endstufe.

Beispiel:

```
JM_MotionLoadParameter axTest
```

2.2.2 JM_MotionPower

```
DEF_FUNCTION (JM_MotionPower, jmpow)
Par:  AxisOffset
      Function      JM_FUNC_POWER_ENABLE
                        JM_FUNC_POWER_DISABLE
      Mode          x
                        JM_MODE_POSITION_CONTROLLED
                        JM_MODE_SPEED_CONTROLLED
                        JM_MODE_CURRENT_CONTROLLED
```

Mit dieser Funktion kann die JetMove-Endstufe ein- oder ausgeschaltet werden. Beim Einschalten der JetMove kann der Modus (Position-, Geschwindigkeit- oder Stromvorgabe) eingestellt werden. Wenn ein Fehler ansteht, so muß der zuerst mit der Funktion JM_ClearError gelöscht werden.

Beispiel :

```
JM_MotionPower (axTest, JM_FUNC_POWER_ENABLE,
               JM_MODE_CURRENT_CONTROLLED)
```

2.2.3 JM_MotionConfigHoming

DEF_FUNCTION (JM_MotionConfigHoming, jmconfhom)

Par:	AxisOffset	
	Direction	x JM_REF_DIRECTION_POS JM_REF_DIRECTION_NEG
	RefCamType	x JM_REF_CAM_LIMIT_CAM: nur Referenzschalter JM_REF_CAM_LIMIT: nur Endschalter JM_REF_CAM_CAM: End- und Referenzschalter
	CamSearchSpeed	Geschwindigkeit in [mm/s] oder [°/s]
	CamAcceleration	Beschleunigung in [mm/s ²] oder [°/s ²]
	CamSearchDistMax	Maximaler Suchweg in [mm] oder [°]
	K0Type	x JM_REF_K0_TYPE_WITH_K0 JM_REF_K0_TYPE_WITHOUT_K0
	K0SearchSpeed	Geschwindigkeit in [mm/s] oder [°/s]
	K0SearchDistMax	Maximaler Suchweg in [mm] oder [°]

Mit dieser Funktion wird die Referenzfahrt konfiguriert. Die Referenzfahrt ist in zwei Abschnitte aufgeteilt: Nockensuche und K0-Suche. Es wird darauf geachtet, daß der Nocken immer von der selben Seite angefahren wird, bevor auf K0-Suche umgeschaltet wird.

Beispiel:

M_MotionConfigHoming (axTest, JM_REF_DIRECTION_POS, JM_REF_CAM_LIMIT, 100, 2000, 1000, JM_REF_K0_TYPE_WITH_K0, 20, 360)

2.2.4 JM_MotionHome

DEF_FUNCTION (JM_MotionHome, jmhome)

Par:	AxisOffset	
	RefType	x JM_REF_TYPE_SET_REF JM_REF_TYPE_SEARCH_REF
	HomeMovement	Referenzbewegung in [mm] oder [°]
	HomePosition	Referenzposition in [mm] oder [°]

Mit dieser Funktion wird das Referenzieren gestartet:

Das Referenz-Setzen benutzt nur den Parameter „HomePosition“;

das Referenz-Suchen benutzt auch die Parameter der Funktion JM_MotionConfigHoming.

Beispiel:

JM_MotionHome (axTest, JM_REF_TYPE_SET_REF, x, 0)

2.2.5 JM_MotionMovePtp

DEF_FUNCTION (JM_MotionMovePtp, jmmoveptp)

Par:	AxisOffset	x
	Function	JM_FUNC_MOVE_ABSOLUTE_ALL JM_FUNC_MOVE_ABSOLUTE_TARGET (wird verwendet, wenn nur die Zielposition geändert werden soll) JM_FUNC_MOVE_RELATIVE_ALL JM_FUNC_MOVE_RELATIVE_TARGET (wird verwendet, wenn nur die Zielposition geändert werden soll) JM_FUNC_MOVE_SPEED (wird verwendet, wenn nur die Geschwindigkeit geändert werden soll) JM_FUNC_MOVE_ACCELERATION (wird verwendet, wenn nur die Beschleunigung geändert werden soll) JM_FUNC_MOVE_DECELERATION (wird verwendet, wenn nur die Verzögerung geändert werden soll)
	TargetPosition	Zielposition in [mm] oder [°]
	Speed	Sollgeschwindigkeit in [mm/s] oder [°/s]
	Acceleration	Beschleunigung in [mm/s ²] oder [°/s ²]
	Deceleration	Verzögerung in [mm/s ²] oder [°/s ²]
	DestinationWindow	Zielfenster in [mm] oder [°]
	RampType	x JM_RAMP_TYPE_LINEAR JM_RAMP_TYPE_SIN2
	MovementType	x JM_MOVEMENT_TYPE_ABSOLUTE JM_MOVEMENT_TYPE_MODPOS JM_MOVEMENT_TYPE_MODNEG JM_MOVEMENT_TYPE_MODALAUTO (wird nur bei Modulo-Achsen verwendet)
	BaseType	x JM_BASE_TYPE_LAST_TARGET JM_BASE_TYPE_ACT_POS (wird nur bei Relativpositionierung verwendet. Hier wird angegeben, ob sich die Position auf die aktuelle Istposition oder auf die letzte Sollposition beziehen soll)

Mit Hilfe dieses Befehls kann eine Absolut- oder Relativpositionierung gestartet werden. Der Befehl wird aber auch benutzt, wenn z.B. nur die Geschwindigkeit geändert werden soll.

Beispiel:

JM_MotionMovePtp (axTest, JM_FUNC_MOVE_ABSOLUTE_ALL, 1000, 100, x, 10000, 1, JM_RAMP_TYPE_SIN2, x, x)

2.2.6 JM_MotionMoveVelocity

DEF_FUNCTION (JM_MotionMoveVelocity, jmmovevelo)

Par:	AxisOffset	
	Function	x JM_FUNC_MOVE_VELOCITY JM_FUNC_MOVE_SPEED (wird verwendet, wenn nur die Geschwindigkeit geändert werden soll) JM_FUNC_MOVE_ACCELERATION (wird verwendet, wenn nur die Beschleunigung geändert werden soll)
	SpeedDirection	x JM_SPEED_DIRECTION_POS JM_SPEED_DIRECTION_NEG
	Speed	Sollgeschwindigkeit in [mm/s] oder [°/s]
	Acceleration	Beschleunigung in [mm/s ²] oder [°/s ²]
	RampType	x JM_RAMP_TYPE_LINEAR JM_RAMP_TYPE_SIN2

Mit dieser Funktion wird eine Endlospositionierung gestartet.

Beispiel:

```
JM_MotionMoveVelocity (axTest, JM_FUNC_MOVE_VELOCITY,
                      JM_SPEED_DIRECTION_POS, 200, x, x)
```

2.2.7 JM_MotionStop

DEF_FUNCTION (JM_MotionStop, jmstop)

Par:	AxisOffset	
	StopType	x JM_STOP_TYPE_MAX_RAMP JM_STOP_TYPE_USER_RAMP
	Deceleration	Verzögerung in [mm/s ²] oder [°/s ²] (wird nur benötigt, wenn der Stopp-Typ JM_STOP_TYPE_USER_RAMP ist)

Mit dieser Funktion kann eine aktive Bewegung angehalten werden. Mit Hilfe der Parameter kann noch unterschieden werden, ob mit maximaler Rampe oder mit einer Bediennrampe angehalten werden soll.

Beispiel:

```
JM_MotionStop (axTest, JM_STOP_TYPE_USER_RAMP, 2000)
```

2.2.8 JM_MotionClearError

DEF_FUNCTION (JM_MotionClearError, jmclerr)

Par: AxisOffset

Diese Funktion muß benutzt werden, um einen anstehenden Fehler zu löschen.

Beispiel:

JM_MotionClearError axTest

2.2.9 JM_MotionReadStatus

DEF_FUNCTION (JM_MotionReadStatus, jmreadstatus)

Par: AxisOffset

Status	JM_STATUS_REF_SET
	JM_STATUS_STOPPED
	JM_STATUS_DEST_WINDOW
	JM_STATUS_LIMIT_HARD_NEG
	JM_STATUS_LIMIT_HARD_POS
	JM_STATUS_REF_CAM
	JM_STATUS_LIMIT_SOFT_NEG
	JM_STATUS_LIMIT_SOFT_POS
	JM_STATUS_POWER_ENABLED
	JM_STATUS_BUSY
	JM_STATUS_AT_MAX_SPEED
	JM_STATUS_ACCELERATION
	JM_STATUS_DECELERATION
	JM_STATUS_MESSAGE
	JM_STATUS_ERROR
	JM_STATUS_WARNING
	JM_STATUS_PULSE_ENABLE

Mit dieser Funktion kann ein Statusbit abgefragt werden.

Beispiel:

WHEN

JM_MotionReadStatus (axTest, JM_STATUS_DEST_WINDOW)

THEN

2.2.10 JM_MotionReadErrorType

DEF_FUNCTION (JM_MotionReadErrorType, jmreaderrortype)

```

Par:   AxisOffset
      Status          JM_STATUS_ERROR
                        JM_STATUS_WARNING
      Error_Warning  JM_WARNING_BALLAST
                        JM_WARNING_DEVICE_TEMP
                        JM_WARNING_MOTOR_TEMP
                        JM_WARNING_OVERLOAD_PFC
                        JM_WARNING_INPUT_OVERCURRENT
                        JM_WARNING_BOARD_TEMP
                        JM_ERROR_HARDWARE
                        JM_ERROR_INTERNAL_SUPPLY
                        JM_ERROR_PHASE_FAILED
                        JM_ERROR_MOTOR_OR_CABLE
                        JM_ERROR_DC_LINK_OVERVOLTAGE
                        JM_ERROR_CURRENT_OVERLOAD
                        JM_ERROR_INTERNAL_BALLAST
                        JM_ERROR_DEVICE_TEMP
                        JM_ERROR_MOTOR_TEMP
                        JM_ERROR_ENCODER
                        JM_ERROR_OVERSPEED
                        JM_ERROR_CURRENT_OVERRANGE
                        JM_ERROR_EARCH_FAULT
                        JM_ERROR_AVR_EEPROM
                        JM_ERROR_AVR_TIMEOUT
                        JM_ERROR_PULSE_ENABLE
                        JM_ERROR_INPUT_OVERCURRENT
                        JM_ERROR_LIMIT_HARD
                        JM_ERROR_LIMIT_SOFT
                        JM_ERROR_TIMEOUT_EXT_REACTION
                        JM_ERROR_DC_LINK_MIN
                        JM_ERROR_DC_LINK_MAX
                        JM_ERROR_DRIVE_BLOCKED
                        JM_ERROR_TRACKING
                        JM_ERROR_SUPPLY_24V
                        JM_ERROR_SUPPLY_15V
                        JM_ERROR_SUPPLY_5V
                        JM_ERROR_SUPPLY_AVR
                        JM_ERROR_MOTION_K0_DIST
                        JM_ERROR_MOTION_REF_CAM_DIST
                        JM_ERROR_MOTION_LIMIT_POS
                        JM_ERROR_MOTION_LIMIT_NEG
  
```

Mit dieser Funktion kann eine spezielle Warn- oder Fehlermeldung ausgelesen werden.

Beispiel:

```

IF
    JM_MotionReadErrorType (axTest, JM_STATUS_ERROR,
                            JM_ERROR_ENCODER)
THEN
  
```

2.2.11 JM_MotionWriteParameter

DEF_FUNCTION (JM_MotionWriteParameter, jmwritepara)

Par: AxisOffset
Parameter siehe Parameterliste
Value neuer Wert für den Parameter

2.2.12 JM_MotionWriteParameterBit

DEF_FUNCTION (JM_MotionWriteParameterBit, jmwriteparabit)

Par: AxisOffset
Parameter siehe Parameterliste
Bit Parameterbit, siehe Parameterliste
Value 0, 1

2.2.13 JM_MotionReadParameter

DEF_FUNCTION (JM_MotionReadParameter, jmreadpara)

Par: AxisOffset
Parameter siehe Parameterliste

2.3 Programmierung

Die JetSym-Funktionen, die eine Aktion starten, benutzen intern Kommandos. Bis diese Kommandos intern verarbeitet werden, können bis zu 2 Millisekunden vergehen. Während dieser Zeit ist das Busy-Bit im Status gesetzt. Um weitere Aktionen auszuführen oder davon betroffene Parameter zu benutzen, muß also auf das Rücksetzen des Busy-Bits gewartet werden.

Bei folgenden Befehlen ist eine Abfrage des Busy-Bits notwendig:

JM_MotionHome
 JM_MotionMovePtp
 JM_MotionMoveVelocity
 JM_MotionStop
 JM_MotionClearError

2.4 Parameterliste

Für die Benutzung von Parametern wurden in Anlehnung an den Motion-Setup folgende Symbole angelegt:

JM_REG_AXIS_TYPE	mit folgenden Möglichkeiten: JM_AXIS_TYPE_LINEAR JM_AXIS_TYPE_ROTATORY
JM_REG_MODULO_AXIS	mit folgenden Möglichkeiten: JM_MODULO_AXIS_NOT_MODULO JM_MODULO_AXIS_MODULO
JM_REG_GEAR_RATIO_MOTOR	
JM_REG_GEAR_RATIO_MECHANIC	
JM_REG_LINEAR_ROTATION_RATIO	
JM_REG_MAXIMUM_POSITION_NEGATIVE	
JM_REG_MAXIMUM_POSITION_POSITIVE	
JM_REG_MAXIMUM_SPEED	
JM_REG_MAXIMUM_ACCELERATION	
JM_REG_MAXIMUM_JERK	
JM_REG_DIGITAL_INPUT_POLARITY	folgende Bits sind definiert: JM_BIT_POLARITY_EPOS JM_BIT_POLARITY_ENEG JM_BIT_POLARITY_REF_CAM
JM_REG_DEVICE_RATED_VOLTAGE	
JM_REG_DEVICE_RATED_CURRENT	
JM_REG_MAINS_VOLTAGE	
JM_REG_DEVICE_INPUT_CURRENT	
JM_REG_ACTUAL_DC_LINK	
JM_REG_BALLAST_LOAD	
JM_REG_DEVICE_TEMPERATURE	
JM_REG_BOARD_TEMPERATURE	
JM_REG_SOFTWARE_VERSION	
JM_REG_DELAY_AFTER_LOCKING_BREAK	
JM_REG_DELAY_AFTER_RELEASE_BREAK	
JM_REG_MOTOR_VOLTAGE_CONSTANT	
JM_REG_COMMUTATION_OFFSET	
JM_REG_POLE_PAIR_NUMBER	
JM_REG_MOTOR_SHAFT_POSITION	
JM_REG_MOTOR_TEMPERATURE	

JM_REG_DRIVE_MODE1 folgende Bits sind definiert:
JM_BIT_DRIVE_MODE1_AUTO_BREAK
JM_BIT_DRIVE_MODE1_AUTO_FAN
JM_BIT_DRIVE_MODE1_PHASE_MONITORING
JM_BIT_DRIVE_MODE1_TEST_MOTOR_CABLE
JM_BIT_DRIVE_MODE1_INVERT_DIRECTION
JM_BIT_DRIVE_MODE1_ENABLE_LIMIT_SWITCH_SOFT
JM_BIT_DRIVE_MODE1_ENABLE_LIMIT_SWITCH_HARD

JM_REG_DC_LINK_MAX_TRIP
JM_REG_DC_LINK_MIN_TRIP
JM_REG_TRACKING_ERROR_LIMIT
JM_REG_TRACKING_WINDOW_TIME
JM_REG_ACTUAL_TRACKING_ERROR
JM_REG_SOFTWARE_LIMIT_NEGATIVE
JM_REG_SOFTWARE_LIMIT_POSITIVE
JM_REG_BLOCKING_TRIPPING_TIME
JM_REG_EMERGENCY_STOP_RAMP_TIME

JM_REG_POSITION_CONTROLLER_KP
JM_REG_POSITION_SET_POINT
JM_REG_POSITION_CONTROLLER_FEED_FORWARD
JM_REG_MAXIMUM_MOTOR_SPEED
JM_REG_SPEED_CONTROLLER_PRESET_VALUE
JM_REG_SPEED_CONTROLLER_SET_POINT
JM_REG_SPEED_CONTROLLER_KP
JM_REG_SPEED_CONTROLLER_TN
JM_REG_SMOOTHING_FEEDBACK_TF
JM_REG_ACTUAL_MOTOR_SPEED
JM_REG_MAXIMUM_OUTPUT_CURRENT
JM_REG_CURRENT_LIMITATION
JM_REG_CURRENT_CONTROLLER_SET_POINT
JM_REG_CURRENT_CONTROLLER_KP
JM_REG_CURRENT_CONTROLLER_TN
JM_REG_ACTUAL_CURRENT

JM_REG_TARGET_POSITION
JM_REG_SPEED
JM_REG_ACCELERATION
JM_REG_DECELERATION
JM_REG_DESTINATION_WINDOW
JM_REG_ACTUAL_POSITION
JM_REG_ACTUAL_MECHANICAL_SPEED

Bei diesen Parametern handelt es sich je nach Verwendung um:
Zahlzahl-Parameter
Gleitkomma-Parameter
Binäre Parameter.

Die Beschreibung zu den Parameter befindet sich in der Online-Hilfe des JetSym unter „JetSym ST Projekte / JetSym Motion“.

2.5 Beispiel

Programmfile:

```
#include "Lib\JM_AX_Functions.spf"
```

```
TASK 0
```

```
LABEL 0
```

```
-FLAG 2201 WHEN FLAG 2201 THEN
```

```
    JM_MotionPower (axTest, JM_FUNC_POWER_DISABLE, x)
```

```
    JM_MotionLoadParameter axTest
```

```
    JM_MotionWriteParameterBit (axTest, JM_REG_DRIVE_MODE1,
                                JM_BIT_DRIVE_MODE1_INVERT_DIRECTION, 1)
```

```
    JM_MotionWriteParameter (axTest, JM_REG_GEAR_RATIO_MOTOR, 3)
```

```
    JM_MotionWriteParameter (axTest, JM_REG_GEAR_RATIO_MECHANIC, 1)
```

```
    REG 100 = JM_MotionReadParameter (axTest, JM_REG_ACTUAL_POSITION)
```

```
    JM_MotionClearError axTest
```

```
    JM_MotionPower (axTest, JM_FUNC_POWER_ENABLE,
                    JM_MODE_POSITION_CONTROLLED)
```

```
    JM_MotionConfigHoming (axTest, JM_REF_DIRECTION_POS,
                            JM_REF_CAM_LIMIT, 100, 2000, 1000,
                            JM_REF_K0_TYPE_WITH_K0, 20, 360)
```

```
    JM_MotionHome (axTest, JM_REF_TYPE_SET_REF, x, 0)
```

```
    WHEN
```

```
        JM_MotionReadStatus (axTest, JM_STATUS_REF_SET)
```

```
    THEN
```

```
        DELAY 10
```

```
        JM_MotionMovePtp (axTest, JM_FUNC_MOVE_ABSOLUTE_ALL, 1000,
                           100, x, 10000, 1, JM_RAMP_TYPE_SIN2, x, x)
```

```
    WHEN
```

```
        JM_MotionReadStatus (axTest, JM_STATUS_DEST_WINDOW)
```

```
    THEN
```

```
    IF
```

```
        JM_MotionReadStatus (axTest, JM_STATUS_ERROR)
```

```
    THEN
```

```
        REG 100 = JM_MotionReadErrorType (axTest, JM_STATUS_ERROR,
                                           JM_ERROR_ENCODER)
```

```
    THEN
```

```
    GOTO 0
```

Deklarationsfile:

```
#include "Lib\JM_Lib.ssf"
```

```
axTest 12000
```