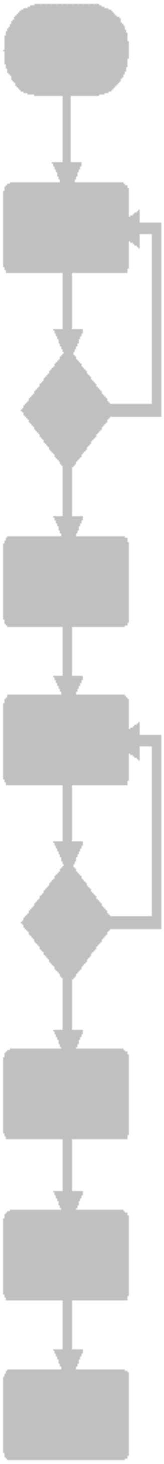

# JetMove 2xx
# Version Update
# from V2.10 to V2.11

Revision 1.00 28.04.2009/Printed in Germany

# Table of contents

# 1 Introduction

| Overview of Version Updates | | | |
|---|---|---|---|
| **Version** | **Function** | **Expanded** | **Corrected** |
| V2.07.0.0 | Operation of stepper motors | ✓ | |
| | "Safe Standstill" option | ✓ | |
| | Referencing on the fly | ✓ | |
| | Motor temperature monitoring | | ✓ |
| V2.09.0.0 | Incremental encoder simulation | ✓ | |
| | Incremental encoder evaluation | ✓ | |
| | Switching digital outputs to position X | ✓ | |
| | R196 *Linear/rotatory ratio* | | ✓ |
| | F02 + F28 after switching off the controller | | ✓ |
| | Sudden change after changing the encoder | | ✓ |
| V 2.10.0.0 | Triggering to float registers | ✓ | |
| | Dead time compensation for trigger input | ✓ | |
| | Trailing indicator for tracking error | ✓ | |
| | Command 14 | ✓ | |
| | R432 CamChange type | ✓ | |
| | Dig. outputs as a function of the actual | ✓ | |
| | JM_CNT module with incremental encoder | ✓ | |
| | JM_CNT module with EnDat 2.2 encoder | ✓ | |
| | JM_CNT module as second encoder system | ✓ | |
| | Torque deactivation | | ✓ |
| | Negative leading axis difference in the table | | ✓ |
| | Referencing with MC gantry axis | | ✓ |
| | Reference position with decimal places | | ✓ |
| | Hiperface reinitialization with MC | | ✓ |
| | Precision of speed scaling | | ✓ |
| | Machine referencing to the reference switch, | | ✓ |
| | Referencing towards zero pulse | | ✓ |
| | Current reduction | | ✓ |
| | Hold flag | | ✓ |
| V2.11.0.0 | Holding current for torque deactivation | ✓ | |
| | Limit switch monitoring | ✓ | |
| | Current pre-control | ✓ | |
| | Set and Clr registers for dig. outputs | ✓ | |
| | Set and Clr registers for capture command | ✓ | |
| | Sine modulation | ✓ | |
| | Current setpoint filter | ✓ | |

| Overview of Version Updates | | | |
|---|---|---|---|
| **Version** | **Function** | **Expanded** | **Corrected** |
| V2.11.0.0 | Osci function: Pre-trigger | ✓ | |
| | Support for new JetMove types | ✓ | |
| | Analog input deactivation | ✓ | |
| | Capture values for modulo axes | | ✓ |
| | Capture values influenced by dead time | | ✓ |
| | Overcurrent error | | ✓ |
| | "Ready for Operation" flag | | ✓ |
| | Offset correction for resolvers | | ✓ |
| | F05 for a deactivated motor cable test | | ✓ |
| | "Reference Set" flag | | ✓ |
| | Change of positioning speed | | ✓ |
| | Oscilloscope | | ✓ |
| | System bus initialization by JetControl 3xx | | ✓ |
| | Positioning | | ✓ |

# 2 Expansions

## 2.1 Holding current for torque deactivation

(# 680) As of version 2.10.0.01, the holding current for torque deactivation can be set separately.

| Register 607: Holding current | |
|---|---|
| **Function** | **Description** |
| Read/Write | Holding current after torque deactivation |
| Type/Unit | Float/[Aeff] |
| Value range | 0.0 to R502 (maximum value for output current from devices) |
| Value after reset | 0.0 |

After the axis has been stopped by torque deactivation, the holding current moves the axis against the obstacle until the user program switches off the axis, for example.

**Note:**

- The holding current can only be set as an amount of current.
- Motor constant $K_T$ [Nm/A] can be used to convert holding torque into torque **developed by the motor**.
- This approach is only recommended for torque deactivation in mode 2.

If the holding current has a value of 0.0, the current for the cut-off threshold (R137) is used as the holding current This ensures compatibility with older versions.

## 2.2 Limit switch monitoring

(# 708) As of version 2.10.0.03, only the limit switches required for the selected referencing process are monitored during referencing. This allows the limit switch inputs on the JetMove which are not required to be used for other purposes.

## 2.3 Current pre-control

(#329) As of version 2.10.0.06, there is a current pre-control facility to improve the dynamic performance and accuracy of the control.

The function is addressed by the following registers that have to be set according to the driveline characteristics:

- R616 motor torque constant $K_T$
- R628 Mass inertia of the driveline
- R629 Scaling factor for current pre-control

Registers R628 and R629 have been added and are described below:

| Register 628: Mass inertia of the driveline | |
|---|---|
| **Function** | **Description** |
| Read/Write | Mass inertia of the driveline at the motor shaft |
| Type/Unit | Float/[kgcm²] |
| Value range | 0.0 … |
| Value after reset | 0.0 |

| Register 629: Scaling for current pre-control | |
|---|---|
| **Function** | **Description** |
| Read/Write | Scaling factor for the current pre-control |
| Type/Unit | Float/[%] |
| Value range | 0.0 … 100.0 |
| Value after reset | 0.0 |

### 2.3.1 Setting the current pre-control

The current pre-control feature aims to improve the dynamic performance of the entire system for motion profiles with high acceleration values. This is achieved by relieving the speed controller's integral-action component of the responsibility for providing the current setpoint value needed for acceleration. This is because the integral-action component can only be changed via the setpoint-actual value difference at the controller input. The dynamic performance of this process is mainly determined by the integral-action time of the speed controller.

For correct setting of the current pre-control, the following steps are recommended:

- Record the behavior of register R125 *Current setpoint value* and R507 *Integral-action component* by oscilloscope
- Set R629 *Scaling the current pre-control* to 100.0%.
- Slowly increase the value of R628 *Inertia* from 0.0 kgcm² up to the known value. In this case, the integral-action component will influence acceleration less and less.
- Ideally, the integral-action component is now only responsible for the friction in the system. In other words, its behavior is roughly proportional to the speed. The target position is approached directly and in one go.
- Mass inertia is over-compensated when the axis starts exceeding and then tracking back to the target position. In this case, the oscilloscope shows how the integral-action component starts partially compensating the current pre-control, i.e. working against the acceleration current.

## 2.4 Set and Clr registers for dig. outputs

(# 766) As of version 2.10.0.10, the following registers have been made available for setting or resetting digital outputs directly. These registers also help to resolve the following problem:
When changing the digital outputs on the JetMove via register R515 *DigOut Status* using the controlling instructions BitSet( ) or BitClear( ), information can be lost if the trigger function is active at the same time, as the JetSym instructions BitSet( ) or BitClear( ) would be executed by the controller as follows:

- R515 *DigOut Status* is read by the JetMove
- Desired bit is set or cleared
- Contents of R515 *DigOut Status* written to the JetMove

During the period between controller reading and writing, the interrupt task in the JetMove can also change register R515 *DigOut Status*. When the controller gains write access this change is lost, because the controller writes back a former state with a changed bit.

Thanks to the Set and Clear registers, there is a smart solution to this problem. Register R515 *DigOut Status* can be changed as desired without its value being fed to the controller and back again.

- The bit assignment corresponds to R515
- =1: The set bits deactivate the relevant output (=0 V).
- = 0: The set bits have no effect.
- Values can be written to the register at any time with immediate effect.
- The initial write access switches the hardware driver to the active state.

| Register 596: DigOutStatus Set | |
|---|---|
| **Function** | **Description** |
| Read/Write | Register for setting digital outputs |
| Type/Unit | Integer/[-] |
| Value range | 0x0000 – 0x000F |
| Value after reset | 0x0000 |

- The bit assignment corresponds to R515.
- =1: The set bits activate the relevant output (=24 V).
- = 0: The set bits have no effect.

| Register 597: DigOutStatus Clear | |
|---|---|
| **Function** | **Description** |
| Read/Write | Register for resetting digital outputs |
| Type/Unit | Integer/[-] |
| Value range | 0x0000 – 0x000F |
| Value after reset | 0x0000 |

- The bit assignment corresponds to R515
- =1: The set bits deactivate the relevant output (=0 V).
- = 0: The set bits have no effect.

## 2.5 Set and Clr registers for capture command

(# 766) As of version 2.10.0.10, the following registers have been made available for setting or resetting register R519 *CaptureCommand* directly. These registers also help to resolve the following problem:
When using the BitSet( ) or BitClear( ) controlling instructions to change register R519 *CaptureCommand*, malfunctions can occur if the capture function is active at the same time, because the BitSet( ) or BitClear( ) JetSym instructions would be executed by the controller as follows:

- R519 *CaptureCommand* is read by the JetMove
- Desired bit is set or cleared
- Contents of R519 *CaptureCommand* written to the JetMove

During the period between controller reading and writing, the interrupt task in the JetMove can also change register R519 *CaptureCommand* at precisely the same time as a capture event occurs. When the controller gains write access this change is lost, because the controller writes back a former state with a changed bit.

Thanks to the Set and Clear registers, there is a smart solution to this problem. Register R519 *CaptureCommand* can be changed as desired without its value being fed to the controller and back again.
Settings of the registers mentioned below:

- The bit assignment corresponds to R519
- = 1: The set bits clear the corresponding bit in R519.
- = 0: The set bits have no effect.
- Values can be written to the register at any time with immediate effect.

| Register 631: CaptureCmd Set | |
|---|---|
| **Function** | **Description** |
| Read/Write | Register for setting R519 *CaptureCommand* |
| Type/Unit | Integer/[-] |
| Value range | 0x0000 – 0x010E |
| Value after reset | 0x0000 |

| Register 632: CaptureCmd Clear | |
|---|---|
| **Function** | **Description** |
| Read/Write | Register for resetting R519 *CaptureCommand* |
| Type/Unit | Integer/[-] |
| Value range | 0x0000 – 0x010E |
| Value after reset | 0x0000 |

## 2.6 Sine modulation

As of version 2.10.0.12, there is a new modulation method for generating voltage in the power section. It is particularly useful at lower speeds, as it makes axis behavior considerably smoother.
There are also, however, a number of disadvantages associated with sine modulation. It increases switching losses in the power section by 50%. In addition, the maximum output voltage for PWM is 15% less than with space vector modulation.

In order to take advantage of the benefits offered by sine modulation while still getting maximum power from the JetMove, an adaptation has been introduced which involves switching to sine modulation when the speed falls below a specifiable threshold. Space vector modulation remains active above this speed threshold.

| **Register 228: PWM adaptation type** | |
|---|---|
| **Function** | **Description** |
| Read/Write | Adaptation type for PWM modulation |
| Type/Unit | Integer/[-] |
| Value range | 0: Adaptation disabled<br>1: Adaptation between space vector and sine modulation |
| Value after reset | 0 |

| **Register 229: Switching threshold for adaptation** | |
|---|---|
| **Function** | **Description** |
| Read/Write | Threshold for switching between space vector and sine modulation The hysteresis amounts to 2% of the R118 *SpeedMax* |
| Type/Unit | Float/[rpm] |
| Value range | 0 – maximum speed (R118) |
| Value after reset | 300.0 |

The switching threshold for the adaptation can be modified at any time.

| **Register 227: Modulation type** | |
|---|---|
| **Function** | **Description** |
| Read/Write | Type of modulation currently effective for PWM |
| Type/Unit | Integer/[-] |
| Value range | 0: Space vector modulation<br>1: Sine modulation |
| Value after reset | 0 |

If the adaptation is disabled (R228 = 0), the modulation type can be changed online.

## 2.7 Current setpoint filter

| Register 497: Current setpoint filter | |
|---|---|
| **Function** | **Description** |
| Read/Write | Current time constant for the current setpoint filter |
| Type/Unit | Float/[ms] |
| Value range | 0.0 – 4.0 |
| Value after reset | 0.0 |

The current setpoint filter influences the setpoint value of the current control as a T1 constituent. As the setpoint value is within the speed control loop, the filter-time constant is directly integrated into the total of the minor time constants. This has to be considered when setting the parameters for the speed controller.

## 2.8 Oscilloscope function: Pre-trigger

(# 832) As of version 2.10.0.22, a pre-trigger is available in the osci function.

## 2.9 Support for new JetMove types

With version 2.11.0.00, the new JM 215B and JM-225 JetMove device types can now also be operated.

## 2.10 Analog input deactivation

(# 960) Analog input deactivation is applied if the analog input is to carry out an action within JetMove. When the first voltage threshold is exceeded, the motor is decelerated; when a subsequent voltage threshold is exceeded, the brake is locked. The reaction time is 2 ms max.

This function is available as of version 2.11.0.0.

# 3 Corrections

## 3.1 Capture values for modulo axes

(# 713) As of version 2.09.0.20, the capture values for modulo axes are no longer saved correctly. As a result of an error affecting modulo correction, the values for continuous operation exceed the definition range and continue to increase.

As of version 2.10.0.01, this problem has been resolved.

## 3.2 Capture values influenced by dead time

(# 743) Until now, capture values have been recorded internally with a dead time of 2 ms.

As of version 2.10.0.05, this problem has been resolved.

## 3.3 Overcurrent error

As of version 2.10.0.04, error F05 *Overcurrent* can occur sporadically even if current control has been set correctly.

As of version 2.10.0.21, this problem has been resolved.

## 3.4 "Ready for Operation" flag

(# 746) Until now, encoder reinitialization was started immediately after an F09 encoder error was acknowledged and the "Ready for Operation" flag R100.10 was set again.

With a resolver, reinitialization takes approximately 700 ms. The axis must not be activated during this time, as it may behave in an uncontrolled way. As a workaround, register R520.0 = 1 (encoder status.encoder init. = ok) has to be queried before issuing command 1.

As of version 2.10.0.10, the R100.10 "Ready for Operation" bit in the status register is linked with the R520.0 encoder status bit by an AND operation. This makes it possible to activate the axis immediately (after issuing command 8), if the R100.10 "Ready for Operation" bit is set.

## 3.5 Offset correction for resolvers

(# 290) As of version 2.07.0.08, there is only limited operational scope in terms of offset correction for resolver evaluation. This could mean that any offsets which might apply are not fully compensated.

As of version 2.10.0.12, this problem has been resolved.

## 3.6 F05 for a deactivated motor cable test

The following problem applies as of version 2.10.0.02:
After 24 V supply is switched on and with the motor cable test deactivated (R540.4 = 0), error F05 *Overcurrent* will occur as soon as the controller is enabled for the first time. This problem will not recur once the error has been acknowledged.

As of version 2.10.0.15, this problem has been resolved.

## 3.7 "Reference Set" flag

(# 871) Until now, the "Reference Set" bit in the R100 status register of the controller was always reset if registers R158 *Bus_fm_MasterPosMax* or R159 *Bus_fm_MasterPosMin* were written to in the relevant JetMove. As these two registers affect the leading axis configuration, the reference status of the axis must not change.

As of version 2.10.0.15, this situation has been given due consideration.

## 3.8 Change of positioning speed

(# 911) Since version 2.03, it has been possible to initiate positioning by changing the positioning speed (command 13).

This is no longer possible as of version 2.10.0.17.

## 3.9 Oscilloscope

(# 859) Since the first version, simultaneously writing and reading an oscilloscope record has proven capable of triggering a system reset.

As of version 2.10.0.17, this problem has been resolved.

## 3.10 System bus initialization by JetControl 3xx

(# 877) Until now and under certain circumstances, CAN bus initialization performed in conjunction with a JetControl 3xx would sometimes freeze.

As of version 2.10.0.18, this problem has been resolved.

## 3.11 Positioning

(# 944) Until now, very short travel distances and linear ramps could lead to sudden changes in position after positioning had been completed.

As of version 2.10.0.22, this problem has been resolved.