



## AKD Firmware Release Notes

These release notes for the AKD drive firmware (the software that executes inside the AKD drive) describe the changes between the previous firmware releases and this release.

Kollmorgen provides four different firmware files. Use the file that matches the model number of your drive.

Drive Type	Model Number	Firmware File
Base Analog	AKD-Bxxxxx-NAAN-0000	AKD-B-NAAN-V01-03-00-012.i00
Positioner Indexer Analog	AKD-Pxxxxx-NAAN-0000	AKD-P-NAAN-V01-03-00-012.i00
Positioner Indexer CANopen	AKD-Pxxxxx-NACN-0000	AKD-P-NACN-V01-03-00-012.i00
Positioner Indexer EtherCAT	AKD-Pxxxxx-NAEC-0000	AKD-P-NAEC-V01-03-00-012.i00

Please update to WorkBench version 1.3.0.14158 when using this firmware.

WorkBench is compatible with all previous AKD firmware versions. WorkBench forward compatibility, however, cannot be ensured. When a behavior in the firmware changes, an older WorkBench version may not function correctly with the new firmware. Kollmorgen recommends that you update WorkBench when you install new firmware. The table below describes compatibility between WorkBench versions and firmware versions. Any new firmware not listed here may have compatibility issues with WorkBench.

	M_01-01-00-xxx	M_01-02-00-xxx	M_01-03-00-xxx
WorkBench Version 1.1.x.x	Yes		
WorkBench Version 1.2.x.x	Yes	Yes	
WorkBench Version 1.3.x.x	Yes	Yes	Yes

### Release Summaries

Version: 01-03-00-012, January 07, 2011

#### + General Issues

Version: 01-03-00-011, November 15, 2010

#### + General Issues

Version: 01-03-00-008, September 03, 2010

#### + General Issues

Version: 01-03-00-006, August 26, 2010

#### + General Issues

Version: 01-03-00-004, August 02, 2010

#### + General Issues

Version: 01-03-00-012 Release Date: January 07, 2011

### General Issues

#### Fixed Bugs

- Fault reaction time is too slow (2767)**  
**Issue:**  
 Fault response time is not deterministic and can take over 50ms in a worst case scenario.  
**Solution:**  
 Response time is now deterministic and under 5ms
- Need to issue 2 drv.clrfaults comamnds in order to clear a safe torque off (STO) fault (1844)**  
**Issue:**  
 Need to issue 2 drv.clrfaults comamnds in order to clear a safe torque off (STO) fault  
**Solution:**  
 Made it possible to clear STO fault after one drv.clrfaults command

#### Known Issues

- BISS and Endat feedback errors must be cleared prior to enabling the drive. (2613)**  
**Issue:** (Identified in Version 01-03-00-004)  
 For BISS and Endat devices, uncontrolled motion may result from enabling the drive while feedback errors are present. Please be certain to clear feedback faults using DRV.CLRFAULTS or the Disable and Clear Faults button in Workbench, prior to enabling the drive.
- Home "found" bit incorrectly read as "ON" and remains on. (2439)**  
**Issue:** (Identified in Version 01-03-00-004)  
 When the drive powers up and it is connected to an SFD, resolver, EnDat, BiSS or Hiperface feedback, the drive incorrectly reports that the drive is homed (bit 1 of DRV.MOTIONSTAT is set). This setting means that the drive considers itself homed in whatever position it starts at, and motion tasks can be started even if the system is not truly homed. If homing is required before motion is started, you must verify that the homing move is made.

Version: 01-03-00-011 Release Date: November 15, 2010

### General Issues

#### Feature Changes

- Updated FPGA to 0100\_00\_03 to fix analog output offset issues. (2667)**  
**Issue:**  
 Analog output offset was larger than desired.  
**Solution:**  
 The new FPGA fixes the analog output offset.

Version: 01-03-00-008 Release Date: September 03, 2010

### General Issues

#### Fixed Bugs

- F703 occurs after disabling the drive for the first time. (2435)**  
**Issue:**  
 The drive issues an F703 fault (emergency timeout) when it receives a disable command for the first time after the boot process.

**Solution:**

A variable was not initialized and therefore the disable function did not work properly. This issue has been resolved and the fault does not occur.

- **The drive will intermittently not set up when using EnDat feedbacks. (2451)**

**Issue:**

When using EnDAT feedback and MOTOR.AUTOSET is 1, the drive powers on and intermittently does not get the motor identification data from the feedback device. The drive does not setup correctly under these conditions. Fault F534 occurs: "Failed to read motor parameters from feedback device". This fault can be cleared, and the drive will continue to operate correctly.

**Solution:**

The firmware has been revised so that data read from the EnDAT feedback is now consistently read correctly.

- **Drive performs erratically using EtherCAT. (2435)**

**Issue:**

When using EtherCAT communications the drive would perform erratically and the ASCII SDO would take a long time to respond to the master.

**Solution:**

Several performance improvements and stability changes were made to the firmware allowing the drive to operate reliably.

- **Drive failure occurs when running feedback auto-identify on devices that do not support auto-id. (2511)**

**Issue:**

The drive could intermittently fail during the auto-id process.

**Solution:**

The issue has been resolved.

- **Captured DCT values are incorrect if they are near zero. (2491)**

**Issue:**

The captured DCT value is incorrect if the value is near zero.

**Solution:**

The firmware has been changed to return the correct DCT values.

Version: 01-03-00-006 Release Date: August 26, 2010

## General Issues

### Fixed Bugs

---

- **Infrequent watchdog error upon receipt of drive discovery message from WorkBench. (2130)**

**Issue:**

The drive would occasionally experience watchdog errors upon receipt of a drive discovery message when WorkBench was opened on a connected computer.

**Solution:**

This issue has been corrected.

Version: 01-03-00-004 Release Date: August 02, 2010

## General Issues

### Feature Changes

---

- **Add drive support for the Modbus TCP protocol. (1669)**

**Issue:**

Modbus TCP is a common communication protocol used primarily for HMI interfaces and low-priority communication.

**Solution:**

The drive now fully supports access to drive parameters over Modbus TCP, in addition to the existing Telnet interface and WorkBench software. See the user guide and parameter reference for usage instructions. The implementation does not currently support block read/write. A future implementation will include block read/write access. The implementation also does not currently support partial reads of 64-bit parameters. A future implementation will include the ability to read 64-bit variables as 32-bit variables to help support more devices and programs.

- **Add support for single-ended pulse command inputs on X7 high speed opto inputs. (2135)**

**Issue:**

Support for single-ended pulse command inputs was needed for the X7 high speed opto inputs (pins 9 and 10).

**Solution:**

Support for single-ended pulse command inputs for the X7 high speed opto inputs has been added. This support was added to the firmware using the keywords FB2.XX. The new keywords are the source (FB2.SOURCE), the mode (FB2.MODE) and the resolution (FB2.ENCRESES). The emulated encoder input (X9 connector) modes can be defined using these keywords also, or with the previous keywords (DRV.EMUEMODE, DRV.EMUERES).

- **Add support for Endat 2.2 feedback. (1776)**

**Issue:**

Support for Endat 2.2 encoders is required. Support is required for rotary single turn and multi-turn encoders and linear encoders.

**Solution:**

Support for Endat 2.2 encoders has been added to the firmware. Rotary single turn and multi-turn encoders are currently supported. The drive does not support linear Endat 2.2 encoders yet.

- **Add Mode 22-command source/opmode change. (1603)**

**Issue:**

Applications require command source and opmode changes during operation via digital inputs.

**Solution:**

DINx.MODE 22 (command source/opmode change) was added.

- **Add support for Programmable Limit Switches for turning on outputs at programmed positions. (1673)**

**Issue:**

Additional support needed for Programmable Limit Switches.

**Solution:**

Added support for Programmable Limit Switches for turning on outputs at programmed positions.

- **Add DRV.HWENMODE. (1744)**

**Issue:**

A rising edge on the hardware enable input would always clear any faults on the drive. If you are using the drive with an EtherCAT or CANopen master, you may want to control the clearing of faults independently via the network.

**Solution:**

A new parameter DRV.HWENMODE has been added. This parameter controls whether or not the rising edge of the hardware enable clears any drive faults.

- 0 (default) = The rising edge of the hardware enable clears the drive faults.
- 1 = The rising edge of the hardware enable does not clear the drive faults.

The high/low state of the hardware enable is always used to control the active enable state of the drive.

- **Increase the maximum allowed position delta between two fieldbus position command values. (2136)**

**Issue:**

The motor moves in a wrong direction when the position delta between two fieldbus position command values is larger than 0.5 motor rev, which corresponds to  $2^{(FB1.PSCALE)} \text{Fieldbus-Counts}$ .

**Solution:**

Instead of only considering the position delta within one revolution (the lower FB1.PSCALE bits in the fieldbus position command value), the whole 32-bit position delta is now used for calculating a position delta within the drive. Thus the maximum position delta for the fieldbus command values is  $0.5 \cdot 2^{(32-FB1.PSCALE)} \text{motor rev}$ , which corresponds to  $2^{31} \text{Fieldbus-Counts}$ .

- **Added DRV.CMDDELAY to add delays in the command buffer for commands which take time to execute (such as DRV.EN). (2177)**

**Issue:**

Some command actions do not occur immediately (such as DRV.EN). When using the command buffer, this delay causes the en->move command sequence to fail.

**Solution:**

By adding DRV.CMDDELAY, you can insert a delay command into the command buffer. For DRV.EN, this delay allows the drive to enable before the motion command is given.

- **Update of the AKD XML File to version 1.09. (1335)**

**Issue:**

TwinCAT did not behave according to the ECAT conformance test tool during a Mailbox access (SDO exchange). The Mailbox counter was not incremented by TwinCAT.

**Solution:**

The AKD XML file has been updated in order to activate the missing feature in TwinCAT (incrementing the Mailbox counter for each Mailbox access).

- **Add DS402-mode 8 - cyclic synchronous position mode. (1432)**

**Issue:**

The EtherCAT-conformance tool demands one of the cyclic setpoint modes (position, velocity or torque). This mode works in user-scaled positions, whereas the interpolated position mode works in motor incremental units.

**Solution:**

This new mode is now implemented. Two new fixed PDO-mappings are created for this mode: In the receive-direction the mapping (0x1724) includes the object 0x607A sub 0 as target position. In the transmit-direction (0x1B24), the mapping includes the object 0x6064 sub 0 as position actual value. This mode is supported starting with version 1.08 of the AKD.xml-file.

- **STO fault behavior change. (1384)**

**Issue:**

The STO fault appears upon power up, which causes a safety chain deadlock.

**Solution:**

The STO fault will not occur unless software and hardware enable are on.

- **DS402-state machine should be influenced by the hardware-enable input. (1743)**

**Issue:**

No change of the state machine occurred if the hardware-enable signal was taken away and the drive was disabled.

**Solution:**

The default behavior is changed. If the hardware enable goes to 0, then the state of the state machine goes to "Switch On Disabled". This behavior can be switched off by setting bit 0 of FBUS.PARAM05 to 1.

- **Add DS402-support for parameters DOUTx.MODE, PL.ERRWTHRESH, IL.IFOLD, MOTOR.PHASE, REGEN.POWER, DRV.DISMODE, and MT.SET. (1796, 1913, 1919, 1931)**

**Issue:**

This feature addresses a customer request for SDO-access to parameters.

**Solution:**

New objects:

parameter	object (index/subindex)
DOUT1.MODE	0x35AF sub 0
DOUT2.MODE	0x35B2 sub 0
PL.ERRWTHRESH	0x3483 sub 0
IL.IFOLD	0x3425 sub 0
MOTOR.PHASE	0x359C sub 0
REGEN.POWER	0x3416 sub 0
DRV.DISMODE	0x35FF sub 0
MT.SET	0x3431 sub 0

- **Add DS402-support for parameters VL.KVFF, MOTOR.INERTIA, MOTOR.KT, VL.ERR, WS.MODE, WS.IMAX, WS.DISTMIN, WS.DISTMAX, WS.T, WS.TDELAY1, WS.TDELAY2, WS.TDELAY3, WS.VTHRESH, WS.STATE, and**

**IL.KACFF. (1719, 1990, 1991, 2033)****Issue:**

This feature addresses a customer request for SDO-access to parameters.

**Solution:**

New objects:

parameter	object (index/subindex)
VL.KVFF	0x354B sub 0
MOTOR.INERTIA	0x35AB sub 0
MOTOR.KT	0x3493 sub 0
VL.ERR	0x3407 sub 4
WS.MODE	0x3864 sub 0
WS.IMAX	0x3494 sub 1
WS.DISTMIN	0x36D1 sub 0
WS.DISTMAX	0x3494 sub 2
WS.T	0x36D0 sub 0
WS.TDELAY1	0x3863 sub 0
WS.TDELAY2	0x3865 sub 0
WS.TDELAY3	0x3494 sub 3
WS.VTHRESH	0x3494 sub 4
WS.STATE	0x3494 sub 5
WS.NUMLOOPS	0x36E2 sub 0
IL.KACFF	0x3426 sub 0

- **Add DS402-support for parameters AIN.PSCALE, AOUT.PSCALE, MOTOR.CTF0, MOTOR.LQLL, MOTOR.POLES, MOTOR.R and WS-commands. (2168)**

**Issue:**

This feature addresses a customer request for SDO-access to parameters.

**Solution:**

New objects:

parameter	object (index/subindex)
AIN.PSCALE	0x3472 sub 0
AOUT.PSCALE	0x3471 sub 5
MOTOR.CTF0	0x3454 sub 0
MOTOR.LQLL	0x3455 sub 0
MOTOR.POLES	0x359D sub 0
MOTOR.R	0x3456 sub 0
PLS.P1 .. PLS.P16	0x34A0 sub 1..8
PLS.WIDTH1 .. PLS.WIDTH8	0x34A1 sub 1..8
PLS.T1 .. PLS.T8	0x34A2 sub 1..8
PLS.EN	0x34a3 sub 1
PLS.RESET	0x34a3 sub 2
PLS.MODE	0x34a3 sub 3
PLS.STATE	0x34a3 sub 4
PLS.UNITS	0x34a4 sub 0

- **Allow EtherCAT firmware download with 512 byte and 1024 byte mailbox size. (2243)**

**Issue:**

The software download via KAS was not possible because the EtherCAT state transition from Init -> Bootstrap was denied by the drive. This state was denied because, originally, a reconfigured Mailbox size of 1024 bytes was required.

**Solution:**

The drive now allows a mailbox size of 512 bytes and 1024 bytes when switching from Init to Bootstrap state.

- **Correction of the electronic gearing behavior when hitting a hardware limit switch. (2362)**

**Issue:**

The electronic gearing class continued counting master pulses when hitting a hardware limit switch in a certain direction. This behavior led the following error grow. While this situation is pending, it would be problematic to release the hardware limit switch, since the position loop tries immediately to move the motor to PL.CMD. It appears as a run-away problem in the drive.

**Solution:**

The electronic gearing class no longer considers master signals, which try to move the motor in direction of an active hardware limit switch.

- **SDO access needed for WS.ARM. (2412)**

**Issue:**

SDO-Access needed for WS.ARM

**Solution:**

SDO 0x3494 sub 6 implemented for WS.ARM.

- **Added new homing mode. (1709)**

**Issue:**

This feature was added to address customer requests that a new homing mode (Homing Mode 11) be added to the already existing homing modes. This new mode searches for the index signal without the need to identify a hardware limit switch, a home switch, or a mechanical stop prior to the index-signal detection.

**Solution:**

The new homing mode searches for the index. The motor starts rotating in the homing direction as soon as the homing is triggered. If the limit switch is active before the index, the motor turns its direction of rotation and searches for the index. As soon it finds the index, home position found status should be set. The actual position and command position is set to the home position.

- **VBUS.OVWTHRESH was changed to R/W. (1251)**

**Issue:**

The over voltage warning was not operative.

**Solution:**

VBUS.OVWTHRESH changed to be R/W with default 0 (which inhibits the warning). The user can define this parameter to allow the warning to appear.

- **Increase the fault log size. (1594)**

**Issue:**

The fault log size is too small

**Solution:**  
Fault log size increased to 50 entries

- **VBUS.UVFTHRESH was changed to R/W. (1790)**

**Issue:**  
A customer requested an option to execute a controlled stop when main loss is discovered.

**Solution:**  
VBUS.UVFTHRESH was changed to be read/write type and can now be stored on the NV. VBUS.UVFTHRESH default value is the value set on the EE (which is also the parameter minimum value) and its maximum value is VBUS.OVFTHRESH.

- **Changed active disable behavior on INMODE 13 and on DRV.DIS. (1874)**

**Issue:**  
Confusion around the behavior of INMODE 13 and DRV.DIS resulted in a request to unite CS.TMAX and DRV.DISTO.

**Solution:**  
CS.TMAX was removed and DRV.DISTO is activated whenever active disable is requested.

## Fixed Bugs

---

- **Digital input mode 1 (fault reset) would clear the software enable. (1744)**

**Issue:**  
When you used only the digital inputs to control the drive, if you cleared a fault with a digital input configured as mode 1 the software enable would also be cleared and you could not enable the drive without sending a command via telnet.

**Solution:**  
DIN.MODE 1 (fault reset) was changed so that it does not clear the software enable.

- **DRV.HELP returned an inconsistent response. (1871)**

**Issue:** (Identified in Version 01-02-00-000)  
DRV.HELP for FB1.HALLSTATE, DRV.SETUPREQLIST and DRV.ONTIME returned a response that did not match the original command.

```
-->DRV.HELP FB1.HALLSTATE
Help for: PFB.HALLSTATE [Info] [Recordable]
-->
```

**Solution:**  
DRV.HELP for FB1.HALLSTATE, DRV.SETUPREQLIST and DRV.ONTIME now returns a consistent response.

- **Long strings would crash the drive. (1308)**

**Issue:** (Identified in Version 01-01-00-000)  
If you send a long string to the drive via telnet the drive will crash, either immediately or the next time you try and connect to the drive.

**Solution:**  
The firmware now detects long strings (greater than 256 characters) and will return "Error: Command was not found."

- **RxPDOs were not configured as event-triggered by default. (1881)**

**Issue:**  
All RxPDOs should be event-triggered by default.

**Solution:**  
Set the default configuration of the transmission type to event-triggered. Additionally, a bug in the event detection algorithm was fixed.

- **Electronic gearing was aborted when a HW or SW limit switch is triggered. (1974)**

**Issue:**  
The drive moves a motor in "Electronic Gearing" mode and triggers a limit switch while the motor is in motion. The electronic gearing was aborted and therefore it was not possible for the drive to move the motor out of the limit switch.

**Solution:**  
The electronic gearing function now remains active even if a limit switch has been triggered. This allows the drive to leave the limit switch in the opposite direction.

- **False brake fault conditions reported during brake enable/disable. (1857)**

**Issue:**  
When enabling or disabling a motor brake, a 306: Brake Short Circuit Fault condition occurred intermittently on a working brake.

**Solution:**  
The false brake faults no longer occur.

- **DRV.ENDEFAULT was not working. (1328)**

**Issue:**  
DRV.ENDEFAULT was ignored on CAN drives regardless of CMDSOURCE value.

**Solution:**  
Fix was made to inhibit CAN from disabling the drive upon power up. DRV.ENDEFAULT is now considered even if DRV.CMDSOURCE=1.

- **User could not enter max-value handling of acceleration/deceleration parameters. (2098)**

**Issue:**  
The user was not able to enter the max-value of acceleration parameters in UNIT.ACCROTARY=3.

**Solution:**  
The CUnitsConversion class has been corrected in order to allow setting the maximum allowed acceleration value, which is displayed in the DRV.HELP information.

- **Delay occurred when starting a Motion Task from the digital inputs. (2211)**

**Issue:**  
The user has a 2 to 12 ms delay in starting a motion task from the digital inputs. This delay occurs in digital inputs modes 2 and 4.

**Solution:**  
Digital input mode 2 and mode 4 were corrected to have a maximum delay of 1 ms.

- **Analog Position mode correction. (2224)**

**Issue:**

A problem existed with the interpolation mechanism of the analog position mode.

**Solution:**

A bug existed in the interpolation mechanism of the analog position mode. The flag for new position command was missing, causing the interpolation to repeat on the same command.

- **DRV.STOP behavior corrected for motion in opmode 2. (1907)**

**Issue:**

A problem occurred when a motion task was running with a very low PL.KP gain and a DRV.STOP command was triggered while the motion task was in the deceleration phase. The software scope exposed that the motion task was triggered again, although a stop-procedure had been triggered.

**Solution:**

A race-condition was the root-cause of this bug. The motion task class displayed that the stop-procedure is finished although the underneath located trajectory generator was still active. Therefore the motion task was triggered again when PL.CMD was set to PL.FB at the end of the stop procedure although the motion task was not completed by the trajectory generator. This condition has been fixed.

- **Corrected the "target position reached" bit in DRV.MOTIONSTAT. (2239)**

**Issue:**

The user adjusts a sequence of motion tasks that are consecutively called (one motion task calls the next one). The error was that the "target position reached" flag in DRV.MOTIONSTAT was high after finishing the first motion task, although further motion tasks follow. This behavior also adversely affected digital output mode 3, which considers this particular bit in DRV.MOTIONSTAT.

**Solution:**

The line in the motion task class source-code, which sets the target position reached flag too early, has been identified and has been removed.

- **Analog input low pass filter blocked input. (1730)**

**Issue:**

If the low pass filter is set to any value below 4, then the input signal is blocked and no response occurred to a changing input signal.

**Solution:**

The minimal value allowed to be written to low pass filter is now 1.

- **PDO scaling of DS402-object 0x606C sub 0 was wrong. (1681)**

**Issue:**

A still motor returned large values for the actual speed in PDO.

**Solution:**

The scaling was not correct for negative values. The value is now scaled in rpm/1000.

- **DRV.DISMODE=2 enables the drive automatically when HW-enable is set to 1. (2037)**

**Issue:**

The drive was creating motion when the hardware-enable was set to 0. After stopping motion and setting the HW-enable to 1 again, the axis was enabled automatically. This happened only with DRV.DISMODE=2.

**Solution:**

The definition of DRV.DISMODE=2 is "Use a controlled stop to ramp down and then disable the drive". This setting is now fixed.

- **When the TwinCAT-EtherCAT-master is re-started, the axis does not move (when set to operation enabled in profile velocity mode). (2170)**

**Issue:**

When the TwinCAT-EtherCAT-master is re-started, the axis does not move (when set to operation enabled in profile velocity mode). The drive gets a communication fault and cannot be re-started.

**Solution:**

The fault situation was not handled correctly. This problem is fixed and the axis moves.

- **Correction of the motion abort when moving out of the limit switches. (2246)**

**Issue:**

You create a motion in DRV.OPMODE 2 (service motion) into a hardware limit switch and afterwards the motion stops. When you try to move out of the hardware limit switch via a service motion in the opposite direction, the motion is immediately aborted after starting that motion due to the behavior of the hardware limit switch code.

**Solution:**

The hardware limit handler now aborts only motion in direction of the limit switch and does not abort the motion when the motor leaves the limit switch.

- **The DS402-state machine does not change its state when software enable is removed via Telnet. (2102)**

**Issue:**

The external software disable does not change the state of the DS402-state machine from operation enable.

**Solution:**

The state of the DS402-state machine is now changed by an external software disable. This behavior can be switched off by setting bit 3 of FBUS.PARAM05 to 1.

- **Some objects in the fixed EtherCAT-PDOs are not scaled. (1915)**

**Issue:**

The actual position given by object 6063 and object 60C1 sub 1 and the latched values 20A0 to 20A3 are not scaled by FB1.PSCALE. Also, the object 60F4 (following error) is not scaled correctly.

**Solution:**

The objects 6063, 60C1 and 20A0 to 20A3 are now scaled via object 608F sub 1, which is connected to the parameter FB1.PSCALE, and gives the increments per motor revolution (power of 2).

- **Bit 5 of the DS402-control word does not work correctly in profile position mode. (2223)**

**Issue:**

If trying to start more than one motion task, the first works correctly, but the next ones are not stored to the motion task 0.

**Solution:**

The bit 5 set to 0 works now correctly. When a first motion task is running and a second one is started with bit 5 = 0, the execution of it is delayed until the first reaches its target position.

- **Bit 13 of the DS402-status word was not supported. (1720)**

**Issue:**  
Bit 13 in the DS402-status word was not supported.

**Solution:**  
Bit 13 is set when the following fault (F439) occurs. This condition can happen in all positioning modes (profile position, homing, interpolated position mode, cyclic synchronous position mode)
- **Position mode capture returns raw data results instead of position values that are adjusted for scaling and homing. (2262)**

**Issue:**  
CAPx.PLFB returns raw data values that are not scaled or adjusted for homing as PL.FB is.

**Solution:**  
The value returned by CAPx.PLFB is now scaled and homed. It will match what the user expects from PL.FB.
- **Wrong Max-value check for the MT.TNUM parameter. (2347)**

**Issue:**  
A maximum value of 8 for the MT.TNUM parameter was allowed, but the AKD returns an error when sending the motion task with MT.TNUM=8 to the drive via the MT.SET command.

**Solution:**  
The maximum value check has been changed from 8 to 7 for the MT.TNUM parameter.
- **CANopen SDO objects for the velocity channel biquad filters were not responding. (1929)**

**Issue:**  
CANopen SDO objects for the velocity channel biquad filters were not responding to read or write requests.

**Solution:**  
The SDO objects have been fixed and now function properly.
- **HOME.SET prevents motion in Gearing mode (2269)**

**Issue:**  
HOME.SET should not prevent any subsequent motion if the drive is enabled.

**Solution:**  
Issuing HOME.SET no longer suppresses motion in electronic gearing mode.
- **DCT capture incomplete. (2133)**

**Issue:**  
Capture of the EtherCAT Distributed Clock (DCT) was not implemented.

**Solution:**  
Capture of the EtherCAT Distributed Clock is now supported.
- **SDO-access to object 60FD sub 0 does not work. (2327)**

**Issue:**  
The read access to 60FD sub 0 delivers 0 instead of the read value of the PDO.

**Solution:**  
The read access to 60FD sub 0 now delivers the correct value.
- **Disallow motion while brake is disengaging. (1241)**

**Issue:**  
Motion should not be allowed during the time period that the brake is disengaging.

**Solution:**  
The new drive parameter MOTOR.TBRAKERLS may be used to set a timeout period during which motion is disallowed, after the drive enables. This delay provides time for the motor brake to fully disengage.
- **Safe torque off warning is missing. (1384)**

**Issue:**  
Safe torque off does not issue a fault until the user tries to make the drive active. As soon as the user tries to make the drive active, the drive faults. The drive should issue a warning when the drive is inactive and the safe torque is off.

**Solution:**  
Added a warning to the state in which the drive is inactive and safe torque is off.

## Known Issues

---

- **Home "found" bit incorrectly read as "ON" and remains on. (2439)**

**Issue:**  
When the drive powers up and it is connected to an SFD, resolver, EnDat, BiSS or Hiperface feedback, the drive incorrectly reports that the drive is homed (bit 1 of DRV.MOTIONSTAT is set). This setting means that the drive considers itself homed in whatever position it starts at, and motion tasks can be started even if the system is not truly homed. If homing is required before motion is started, you must verify that the homing move is made.
- **Overhanging loads may fall briefly when drive is enabled. (1241)**

**Issue:**  
When enabling the drive for a brake motor connected to an overhanging vertical load, the load may drop between brake disengagement and full drive servoing due to delays in the drive enable process. The period when the load is not controlled will depend upon the mechanical time required for the brake to disengage and will generally be under 100 ms.