Manual

MOTECK Actuator with **CANopen**

Revision	2025.9_V1.0

Technical changes may be made to improve the product without notice！

Content

1. Overview 3
2. Warning 3
3. Introduction to CANopen 3
4. Physical layer 4
   - 4.1 Bus connection 4
   - 4.2 Bit rate and bus length 4
   - 4.3 Wiring 5
   - 4.4 Wire definitions 5
   - 4.5 Off-bus operation 5
5. Application 6
   - 5.1 CAN-ID 6
   - 5.2 MOTECK supported communication objects 6
6. Network management 7
   - 6.1 Node-ID 7
   - 6.2 Hardware ID selection for MK35 and MK35L 7
   - 6.3 Actuator boot-up and NMT operational state 8
7. Command and status feedback 9
   - 7.1 RPDO 9
   - 7.2 TPDO 10
8. Object dictionary 11
   - 8.1 SDO 11
   - 8.2 Manufacturer objects 12

1. Overview

This manual contains information and instruction for readers to operate the MOTECK actuator with CANopen control option. These actuators with control board inside the housing are designated as N00 control option which are developed on the base of signal control option and can be physically connected to CAN bus than communicate through CANopen protocol.

As part of MOTECK intelligent actuator series, the CANopen actuators receive the commands to control the motion of actuator including the position, speed and the related setting relies on CiA 301 standard and transmit immediate position, current, speed ... and other status information to system. These actuators are focus on agriculture, construction, industrial automation and other application operated through CANopen network.

2. Warning

This manual assumes that the reader is familiar with the CiA 301 and ISO 11898 standards. Therefore, the terminology in these standards are used but not described in detail. The users who choose the N00 option must have already established or well prepared to establish the CANopen system to apply this actuator.

3. Introduction to CANopen

CANopen is one of the higher-layer CAN(Controller area network) protocols. The application is defined in CiA 301 standard which is released by CAN in AUTOMATION. With the following features:

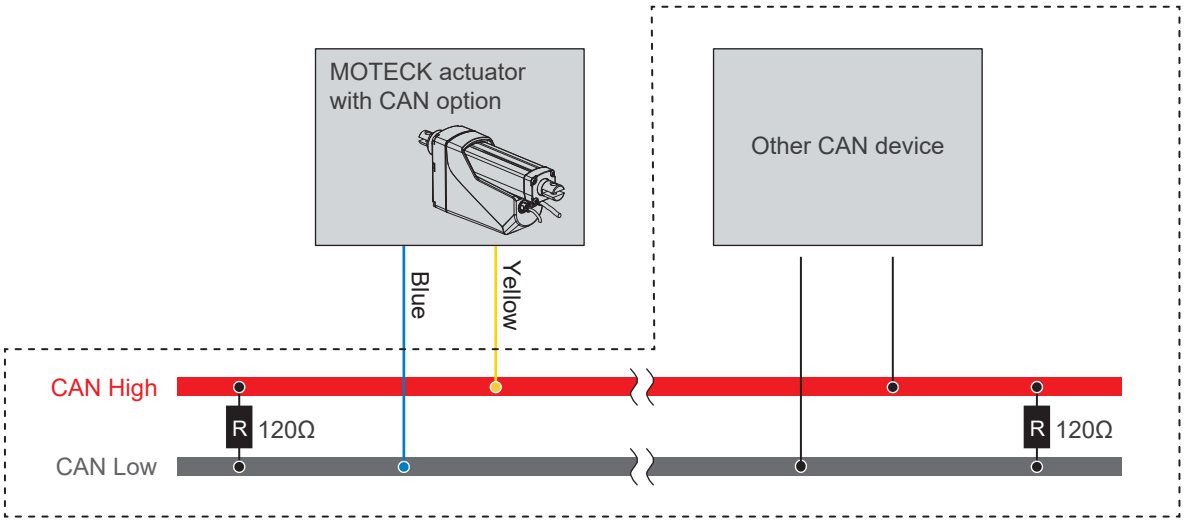
Base: Bosch CAN Specification 2.0A
Data Frame: Standard Frame
Identifier: 11 bits
Max. number of nodes in bus: One NMT master and up to 127 NMT slave nodes

MOTECK CANopen actuator is compliant with the Standard CiA 301 v4.20 which includes a command set for controlling the actuator and providing the feedback status.

4. Physical layer

4.1 Bus connection

Please follow the wiring guidelines of ISO-11898-1/11898-2 standard. The two ends of the CAN High / CAN Low harness should be connected with a 120Ω terminal resistor, as shown in the figure below. The wiring topology of a CAN network should be as close as possible to a single line structure in order to avoid cable-reflected waves.



4.2 Bit rate and bus length

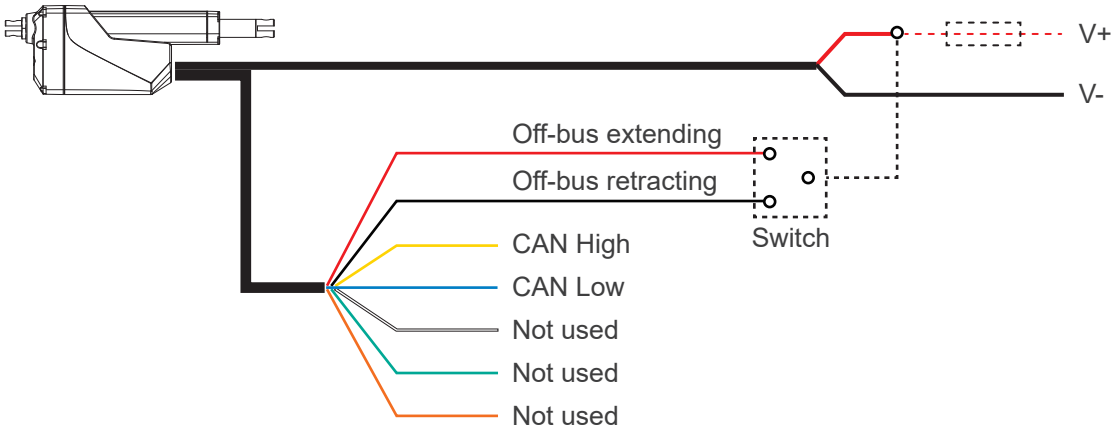
Three Bit rates can be set for MOTECK CANopen actuator: 250Kbps, 500Kbps, 1Mbps.
Default value: 500Kbps

(1) The Estimated bus lengths by CiA 301

Bit rate	Bus length
1Mbps	25 m
500Kbps	100 m
250Kbps	250 m

(2) ISO 11898-2 describes that the max. bus length 40M and max. stub length 0.3M at bit rate 1Mbps.
(3) CiA 303-1 states that bit rate 1Mbps with max. bus length 40M for CANopen networks with less than 64 nodes. The wire cross-section of bus cable would be 22AWG at least. For longer bus length, lower bit rate and larger wire cross-section are considered.
(4) MOTECK actuator tests are limited to consist of 3-meter bus cables and stub wires.

4.3 Wiring



Note: All dashed lines are connected by the customer.

4.4 Wire definitions

	Wire color	Definition	Description
Power cable	Red	V+	• Connect Red to positive • Connect Black to negative • Do not swap the polarity • Input voltage: According to actuator voltage specification ±10%
	Black	V-	
Signal cable	Red	Off-bus extending	Connect Red to positive (V+) to extend, input current <10mA.
	Black	Off-bus retracting	Connect Black to positive (V+) to retract, input current <10mA.
	Yellow	CAN High	
	Blue	CAN Low	
	White	Not used	
	Green	Not used	
	Orange	Not used	

Note: Please ensure that unused wires are well insulated to avoid damaging the internal circuit of the actuator.

4.5 Off-bus operation

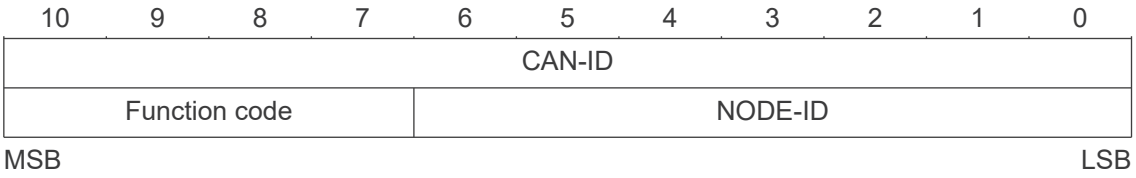
(1) When the actuator has the need to repair, maintain or test without the CAN commands. There are 2 separate wires to perform the off-bus operation which is the same as the S0L option for extending or retracting the actuator but without the EoS signal output.

(2) It is not necessary to disconnect the CAN High and CAN Low wires when the off-bus operation is required. The priority of CAN commands and off-bus control is that whose command comes first, it will be executed first, and the next command will be accepted after the execution is completed.

5. Application

5.1 CAN-ID

A message transferred by data frame through CANopen protocol is called an object or a communication object (COB). CAN-ID is a 11bit identifier which consist of 4 higher bits of Function code and 7 lower bits of Node-ID to identify an unique COB in a CAN network.



Note:
In CiA 301 v4.02, CAN-ID is represented as COB-ID; In CiA 301 v4.20, COB-ID specifically refers to the 32-bit parameter in the data object to cover the 29-bit extended identifier.

5.2 MOTECK supported communication objects

COB	Function code	Resulting CAN-IDs
NMT	0000b	0(0x000)
EMCY	0001b	129 (0x081) - 255 (0x0FF)
PDO1 (tx)	0011b	385 (0x181) - 511 (0x1FF)
PDO1 (rx)	0100b	513 (0x201) - 639 (0x27F)
SDO (tx)	1011b	1409 (0x581) - 1535 (0x5FF)
SDO (rx)	1100b	1537 (0x601) - 1663 (0x67F)
NMT error control	1110b	1793 (0x701) - 1919 (0x77F)

NMT: Network management object
EMCY: Emergency object
PDO: Process Data Objects
SDO: Service Data Objects

6. Network management

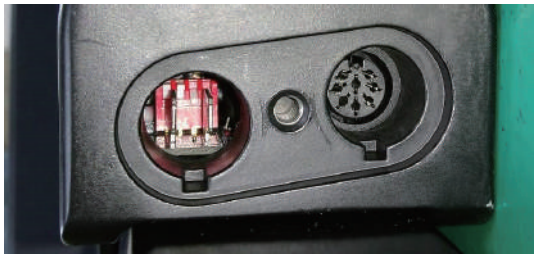
6.1 Node-ID

Each CANopen device shall have an unique Node-ID in the bus. The default Node-ID of MK35 and MK35L is 16 (0x10).

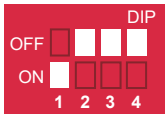
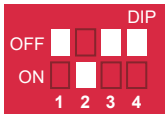
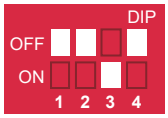
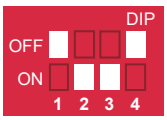
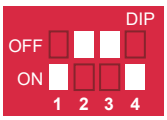
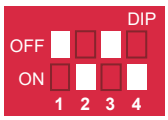
6.2 Hardware ID selection for MK35 and MK35L

If multiple MK35/MK35L actuators will be used in a CAN bus network, there is a DIP switch for ID selection. Up to 16 MK35/MK35L actuators can be connected and preset Node-ID.
Before connecting the actuators to the bus, please follow the steps below.

(1) Loosen the screw of the anti-pull lid, and then remove the lid.
(2) Pull out the power cable, you can see the red DIP switch as shown in the figure below.



(3) Use a non-conductive stick to slide the 4 DIP switches. The hardware ID settings are shown below, which represent values from 16 (Preset) to 31.

NODE-ID Value					
16 (Preset)	17	18	19	20	21
					
22	23	24	25	26	27
					
28	29	30	31		
					

(4) Plug in the power cable back.
(5) Put the anti-pull lid back correctly and tighten the screw to ensure that the cable plugs are secured and sealed.

6.3 Actuator boot-up and NMT operational state

After the CANopen actuator is connected to bus and powered up, It boots up and enters NMT Pre-operational state automatically. The actuator shall not transmit and receive PDOs until the host machine send the message to start the actuator entering NMT Operational state.

An example below shows the procedure of an actuator with NODE-ID 0x10

CAN-ID	Data	Function	Description
0x700+ Node-ID	0x00	Node-ID boot-up	Actuator goes online
0x700+ Node-ID	0x7F	Node-ID enters Pre-operational state	Actuator sends heartbeats per second until state transition
0x00	0x01 0x10	Set the Node-ID 0x10 start to enter Operational state	Data 0x01 0x00 will set all nodes on the bus enter Operational state
0x700+ Node-ID	0x05	Node-ID reports in Operational state	Actuator continues to sends heartbeats per second

7. Command and status feedback

7.1 RPDO

All Node in the same network can send actions or setting command to MOTECK actuator through a MOTECK-specific RPDO mapped as 0x200+NODE-ID with following 8-Byte data field.

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Action	**Position**		**Current limit**	**Speed**		**Soft start**	**Soft stop**

Data field definition:

Name	Function	Data bits	Details (values in decimal)
Action	Move	b0-b1	• 0 = Stop actuator moving • 1 = Extend the actuator • 2 = Retract the actuator • 3 = Move to designated position
	Clear error code	b2	• 0 = Do not clear error code • 1 = Clear the last error code
	Reserved	b3-b7	Not used. (suggest 0)
Position (1)	Go to position	Byte 1 & Byte 2	• 0~65535 • Resolution: 0.1mm/s per bit • Valid only when Move command = 3
Current limit (2)	Set current limit	Byte 3	• 0~254 • Resolution: 0.25A/ bit • 255 = Factory default
Speed (3)	Set run speed	Byte 4 & Byte 5	• 0~65534 • Resolution: 0.1mm/s per bit • 65535 = Factory default
Soft start (4)	Set the ramp time of soft start	Byte 6	• 0~254 • Resolution: 50ms/ bit • 255 = Factory default
Soft stop (4)	Set the ramp time of soft stop	Byte 7	• 0~254 • Resolution: 50ms/ bit • 255 = Factory default

Notes:

(1) The actual stop position is still limited by the mechanical stroke of the actuator. If the move command is not 3, the value for position is neglect.

(2) The factory default value of current limit depends on actuator model. If this parameter is set exceeding the factory default, the default value will be executed.

(3) The run speed is limited to the speed/load curve showed in the data sheet of actuator if the parameter is set exceeding the curve.

(4) Although 0 offset of the soft start/stop setting. There will be a little lag up to 100ms from command to the motor start due to firmware process.

7.2 TPDO

The feedback of MOTECK actuator will be provided every 100ms through a MOTECK-specific TPDO mapped as 0x180+NODE-ID with following 7-Byte data field.

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
Status flag	**Position**		**Current**	**Speed**		**Error flag**

Data field definition:

Name	Function description	Data bits	Details (values in decimal)
Status flag	Retracted EOS limit switch is triggered	b0	• 6 independent status bit indicators • 1 = Valid for the described situation • 0 = Otherwise
	Extended EOS limit switch is triggered	b1	
	Retracting	b2	
	Extending	b3	
	In soft starting	b4	
	In soft stopping	b5	
	Reserved	b6-b7	Always 0
Position [1]	The present stroke position	Byte 1 & Byte 2	• 0~65535 • Resolution: 0.1mm/s per bit
Current	The present electrical current	Byte 3	• 0~255 • Resolution: 0.25A/ bit
Speed	The present speed	Byte 4 & Byte 5	• 0~65535 • Resolution: 0.1mm/s per bit
Error flag [2]	Detected over currrent is lasted for 300ms	b0	• 6 independent error bit indicators. • 1 = Valid for the described situation and the actuator will stop immediately • 0 = Otherwise
	Position error (position unchanged while the actuator is moving status)	b1	
	Under voltage	b2	
	Over voltage	b3	
	Under temperature	b4	
	Over temperature	b5	
	Reserved	b6-b7	Always 0

Notes:

[1] The 0 position was preset by MOTECK when the retracted EOS limit switch is triggered.

[2] The allowable range of voltage and temperature depends on actuator model and listed in data sheet.

8. Object dictionary

8.1 SDO

Index	Sub-index	Name	Access	Data type	Default value
0x1000	0	Device type	ro	UNSIGNED32	0
0x1001	0	Error register	ro	UNSIGNED8	0
0x1014	0	COB-ID EMCY	ro	UNSIGNED32	0x80+ Node-ID
0x1015	0	Inhibit time EMCY	ro	UNSIGNED16	0
0x1016	Consumer heartbeat time				
	1~2	Consumer heartbeat time	rw	UNSIGNED32	0
0x1017	0	Producer heartbeat time	rw	UNSIGNED16	1000
0x1200	SDO server parameter				
	1	COB-ID client → server (rx)	ro	UNSIGNED32	0x600 +Node-ID
	2	COB-ID server → client (rx)	ro	UNSIGNED32	0x580 +Node-ID
0x1400	RPDO communication parameter				
	1	COB-ID used by → RPDO	ro	UNSIGNED32	0x200 +Node-ID
	2	Transmission type	rw	UNSIGNED8	254
	5	Event-timer	ro	UNSIGNED16	0
0x1600	RPDO mapping parameter				
	1	1st application object	ro	UNSIGNED32	0x20010008
	2	2nd application object	ro	UNSIGNED32	0x20020010
	3	3rd application object	ro	UNSIGNED32	0x20030008
	4	4th application object	ro	UNSIGNED32	0x20040010
	5	5th application object	ro	UNSIGNED32	0x20050008
	6	6th application object	ro	UNSIGNED32	0x20060008
0x1800	TPDO communication parameter				
	1	COB-ID used by → TPDO	ro	UNSIGNED32	0x180 +Node-ID
	2	Transmission type	rw	UNSIGNED8	254
	3	Inhibit time	ro	UNSIGNED16	500
	5	Event-timer	rw	UNSIGNED16	100
	6	SYNC start value	ro	UNSIGNED8	0
0x1A00	TPDO mapping parameter				
	1	1st application object	ro	UNSIGNED32	0x21010008
	2	2nd application object	ro	UNSIGNED32	0x21020010
	3	3rd application object	ro	UNSIGNED32	0x21030008
	4	4th application object	ro	UNSIGNED32	0x21040010
	5	5th application object	ro	UNSIGNED32	0x21050008

8.2 Manufacturer objects

Index	Sub-index	Name	Access	Data type	Default value
0x2001	0	Action	rw	UNSIGNED8	0
0x2002	0	Position	rw	UNSIGNED16	0
0x2003	0	Current limit	rw	UNSIGNED8	255
0x2004	0	Speed	rw	UNSIGNED16	65535
0x2005	0	Soft start	rw	UNSIGNED8	255
0x2006	0	Soft stop	rw	UNSIGNED8	255
0x20A0	0	Actual temperature	ro	INTEGER8	
0x20A1	0	Overload current	rw	UNSIGNED16	
0x20A2	0	Running time	ro	UNSIGNED32	
0x20A4	Error record				
	1	Overload error cnt	ro	UNSIGNED16	
	2	Voltage too low error cnt	ro	UNSIGNED16	
	3	Voltage too high error cnt	ro	UNSIGNED16	
	4	Position error cnt	ro	UNSIGNED16	
	5	Temperature too low error cnt	ro	UNSIGNED16	
	6	Temperature too high error cnt	ro	UNSIGNED16	
0x2101	0	Status flag	ro	UNSIGNED8	
0x2102	0	Position	ro	UNSIGNED16	
0x2103	0	Current	ro	UNSIGNED8	
0x2104	0	Speed	ro	UNSIGNED16	
0x2105	0	Error flag	ro	UNSIGNED8	

Notes:
ro: read only access
rw: read and write access

8.3 EDS file

Please consult MOTECK for the electronic data sheet(EDS) file that contains CANopen device informations of the actuator model.