KannMOTION Manager 2

Manual

Document version 2.00
Application version 2.0.0.0

Overview

This is the user manual for KannMOTION Manager 2, an application used for controlling, configuring, programming, and performing other functions on KannMOTION devices. With this software, users can program personalized sequences to the device using either the intuitive Flow Chart Editor, where actions can be created by drag and drop, or by coding in ANSI-C.

⚠	Please note that certain features of our software require an internet connection to function properly. Be aware that firewall and other network settings may prohibit the necessary downloads and cause issues with the application. To ensure the smoothest experience possible, please verify that your internet connection and network settings are properly configured before using these features.

Versioning

Version	*Date*	*Name*	*Changes*
2.00	11.12.2024	TDU	First official version

INHALT

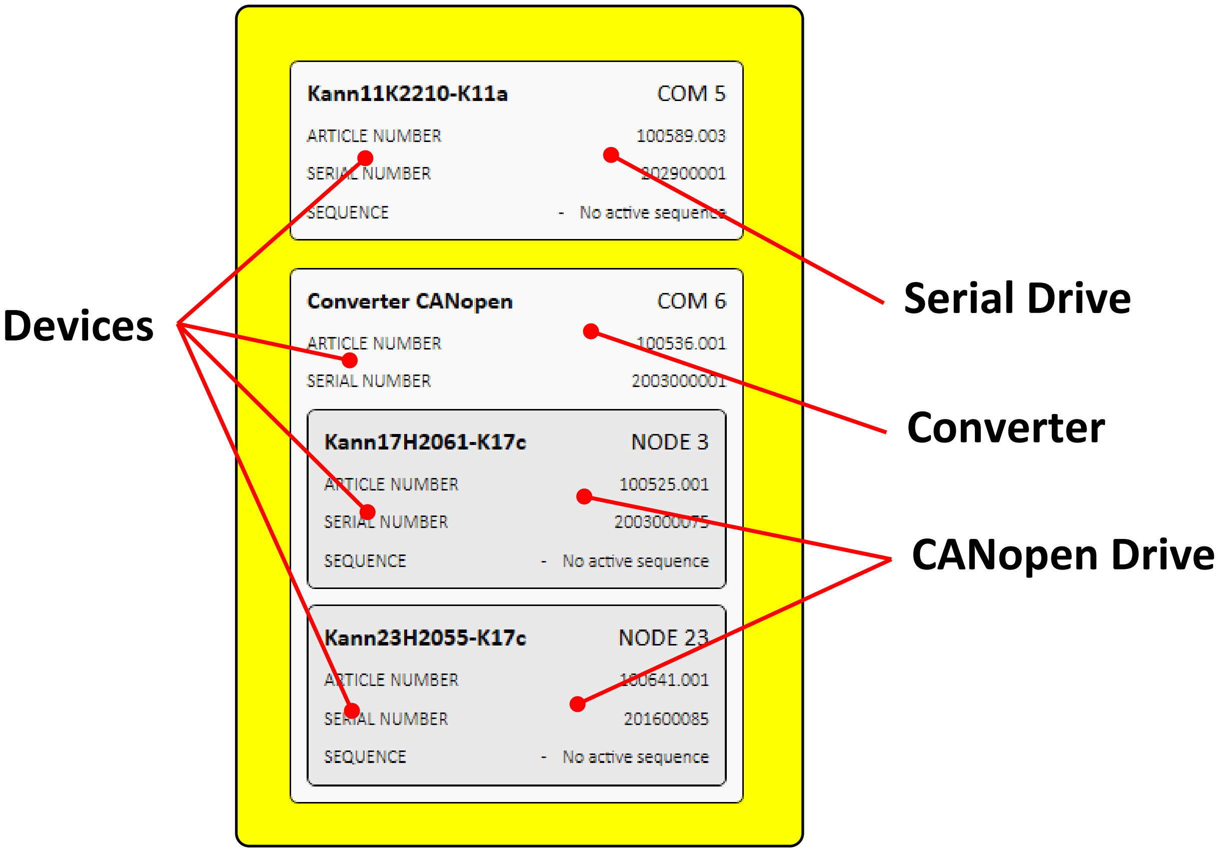
1 Connected devices and updates

1.1 General

To display a connected device for the first time, internet connection is needed.
Once a new device was displayed, internet connection is no longer necessary.

1.2 Overview devices

Within the KannMOTION Manager, there are three types of devices: serial drives, CANopen drives, and converters. It is possible for a converter to house multiple CANopen drives.



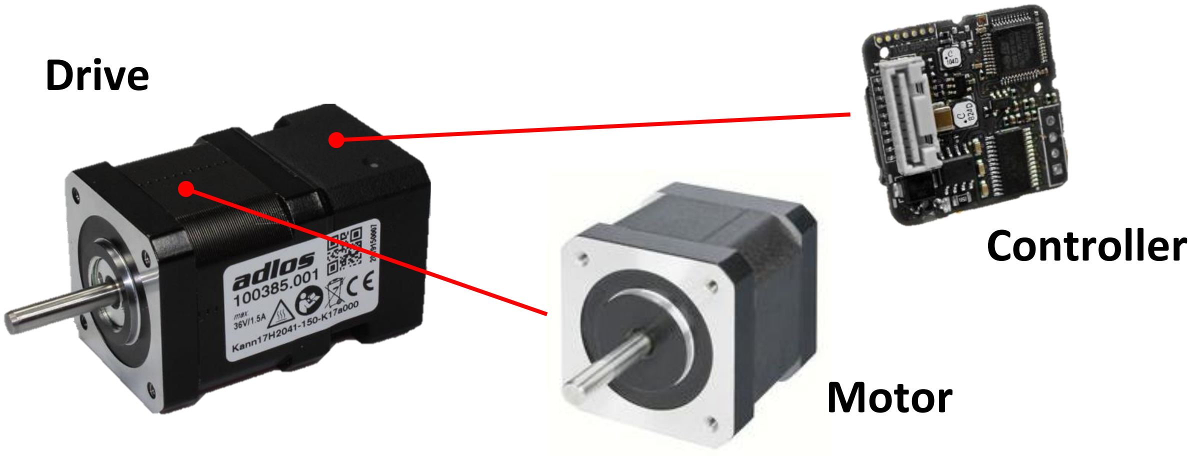
Kann11K2210-K11a COM 5
ARTICLE NUMBER 100589.003
SERIAL NUMBER 202900001
SEQUENCE - No active sequence

Converter CANopen COM 6
ARTICLE NUMBER 100536.001
SERIAL NUMBER 2003000001

Kann17H2061-K17c NODE 3
ARTICLE NUMBER 100525.001
SERIAL NUMBER 2003000075
SEQUENCE - No active sequence

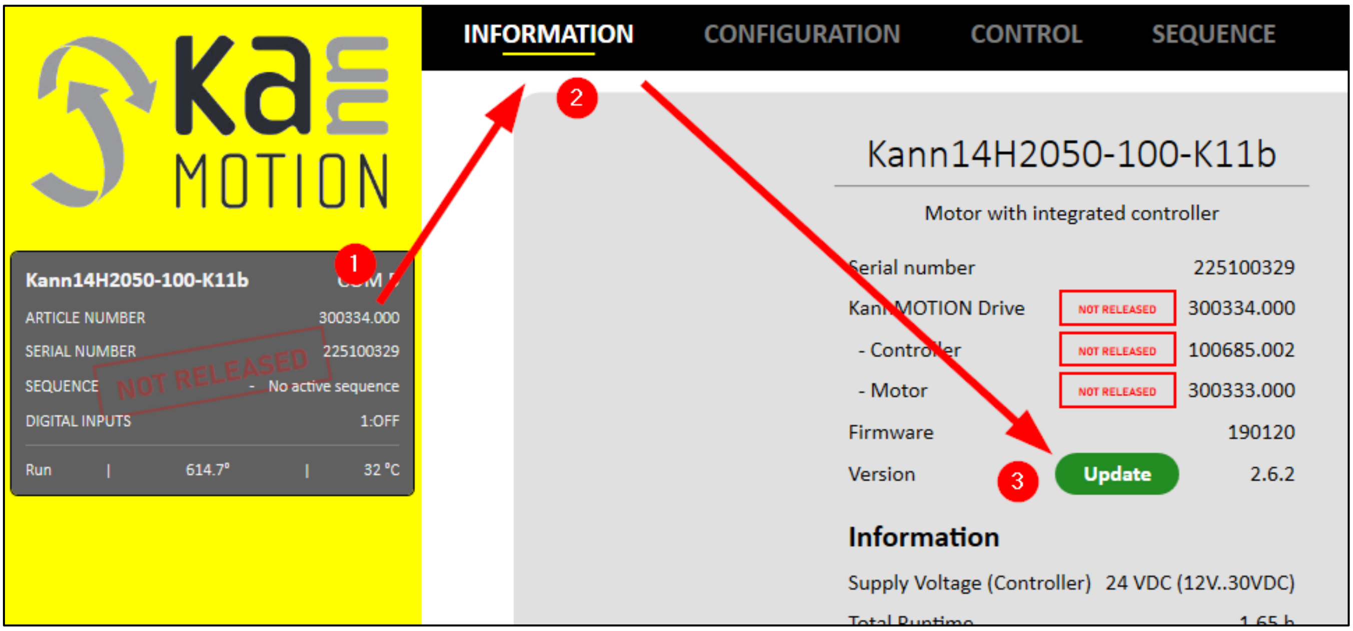
Kann23H2055-K17c NODE 23
ARTICLE NUMBER 100641.001
SERIAL NUMBER 201600085
SEQUENCE - No active sequence

A drive (motor with integrated controller) consists of a controller and a motor.



1.3 Update firmware

To update the firmware, select desired device, go to *INFORMATION* and click button *Update Firmware*.



Kann14H2050-100-K11b COM

ARTICLE NUMBER	300334.000
SERIAL NUMBER	225100329
SEQUENCE	- No active sequence
DIGITAL INPUTS	1:OFF

Run | 614.7° | 32 °C

Kann14H2050-100-K11b

Motor with integrated controller

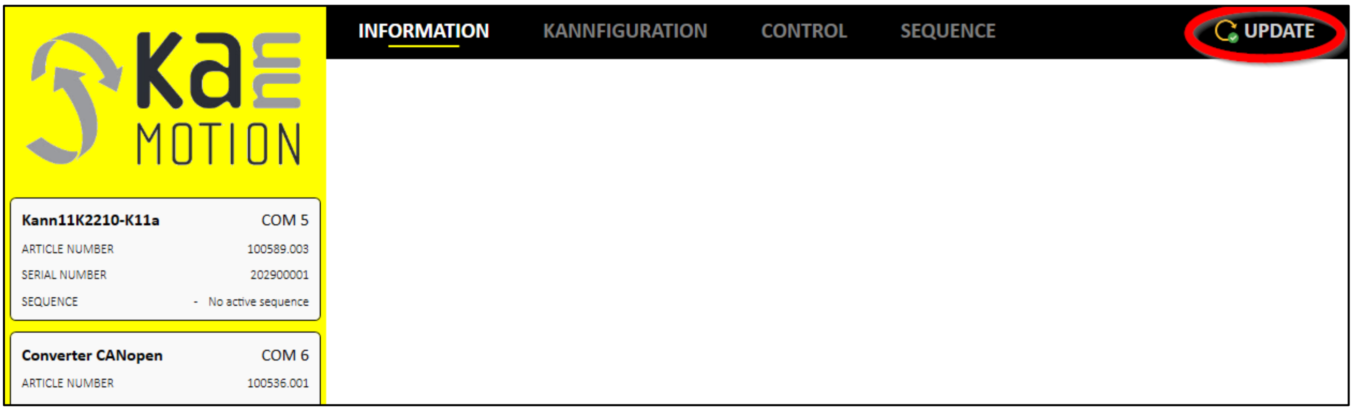
Serial number		225100329
KannMOTION Drive	NOT RELEASED	300334.000
- Controller	NOT RELEASED	100685.002
- Motor	NOT RELEASED	300333.000
Firmware		190120
Version	Update	2.6.2

Information

Supply Voltage (Controller)	24 VDC (12V..30VDC)
Total Runtime	1.65 h

1.4 Update application

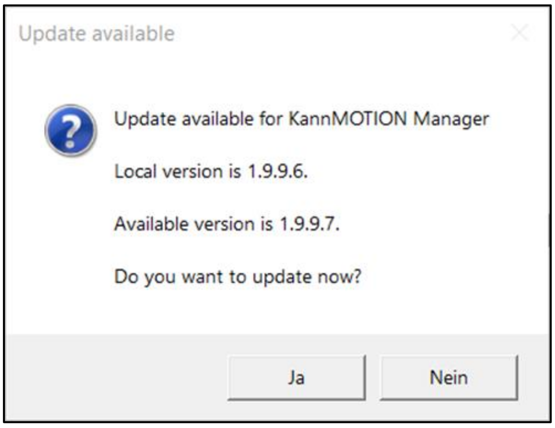
If the tab *UPDATE* (seen below within red circle) shows up, a new update for the application is available.



Kann11K2210-K11a COM 5
ARTICLE NUMBER 100589.003
SERIAL NUMBER 202900001
SEQUENCE - No active sequence

Converter CANopen COM 6
ARTICLE NUMBER 100536.001

With click on UPDATE, new available version is shown, and download/installation can be started.



Update available

Update available for KannMOTION Manager

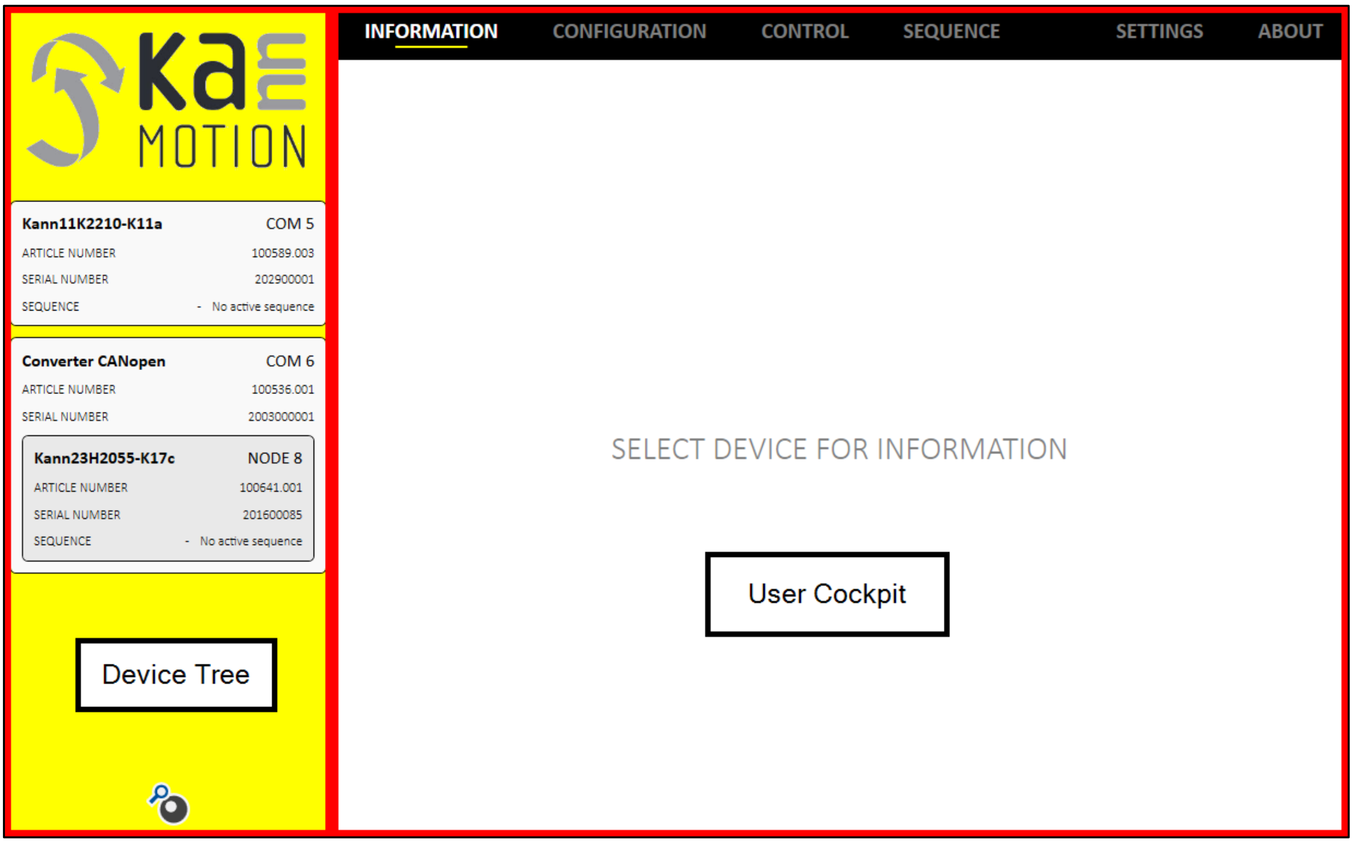
Local version is 1.9.9.6.

Available version is 1.9.9.7.

Do you want to update now?

2 User interface overview

The user interface consists of a device tree to display KannMOTION devices and the user cockpit to interact with the device.



Kann11K2210-K11a COM 5
ARTICLE NUMBER 100589.003
SERIAL NUMBER 202900001
SEQUENCE - No active sequence

Converter CANopen COM 6
ARTICLE NUMBER 100536.001
SERIAL NUMBER 2003000001

Kann23H2055-K17c NODE 8
ARTICLE NUMBER 100641.001
SERIAL NUMBER 201600085
SEQUENCE - No active sequence

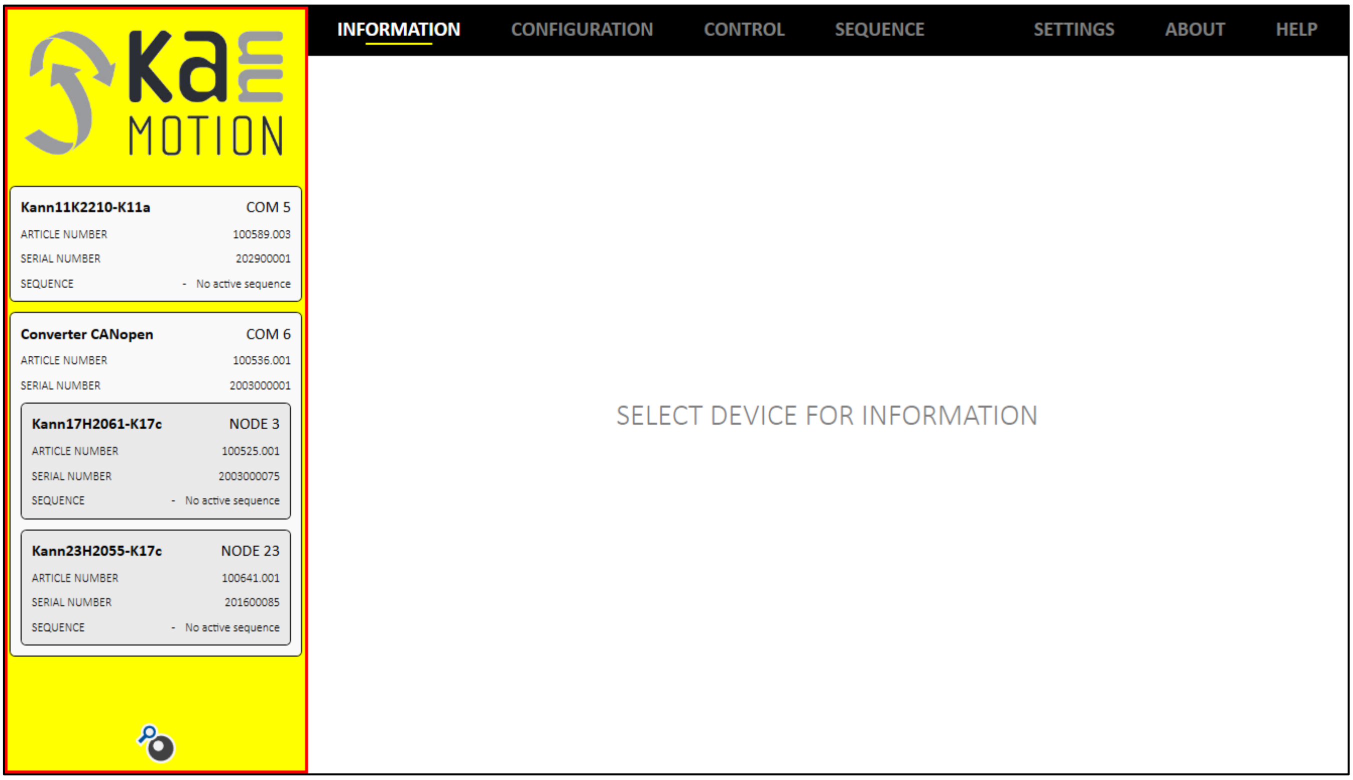
SELECT DEVICE FOR INFORMATION

3 Device tree

Attached devices can be found in the yellow column, the device tree.

Within the yellow column of the user interface there are differences in the presentation of two types of attached drives:

- Serial drives (RS232-TTL, RS232, RS485)
- CANopen drives
- CANopen converter



Kann11K2210-K11a COM 5
ARTICLE NUMBER 100589.003
SERIAL NUMBER 202900001
SEQUENCE - No active sequence

Converter CANopen COM 6
ARTICLE NUMBER 100536.001
SERIAL NUMBER 2003000001

Kann17H2061-K17c NODE 3
ARTICLE NUMBER 100525.001
SERIAL NUMBER 2003000075
SEQUENCE - No active sequence

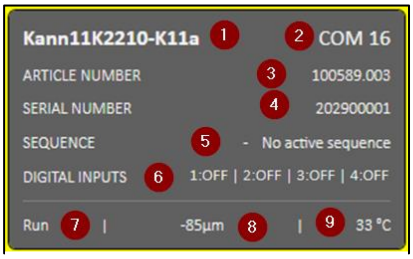
Kann23H2055-K17c NODE 23
ARTICLE NUMBER 100641.001
SERIAL NUMBER 201600085
SEQUENCE - No active sequence

SELECT DEVICE FOR INFORMATION

3.1 Selected device information

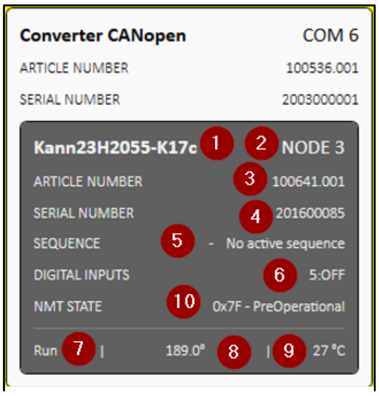
When left-clicking a device, it gets selected which is indicated by the darker gray color.

1. Product name
2. Identifier (COM-Port or node)
3. Article number
4. Serial number
5. Sequence version and description
6. Available digital inputs and states
7. Firmware state
8. Actual position
9. Temperature controller
10. NMT state (only CANopen devices)



Kann11K2210-K11a COM 16
ARTICLE NUMBER 100589.003
SERIAL NUMBER 202900001
SEQUENCE - No active sequence
DIGITAL INPUTS 1:OFF | 2:OFF | 3:OFF | 4:OFF
Run | -85µm | 33 °C

Serial device



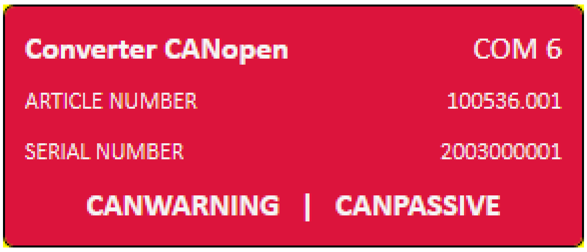
Converter CANopen COM 6
ARTICLE NUMBER 100536.001
SERIAL NUMBER 2003000001

Kann23H2055-K17c NODE 3
ARTICLE NUMBER 100641.001
SERIAL NUMBER 201600085
SEQUENCE - No active sequence
DIGITAL INPUTS 5:OFF
NMT STATE 0x7F - PreOperational
Run | 189.0° | 27 °C

CANopen device

3.2 Error displaying

If an error occurs with device, it gets colored red, and the error is displayed at the bottom.



Converter CANopen COM 6

ARTICLE NUMBER 100536.001

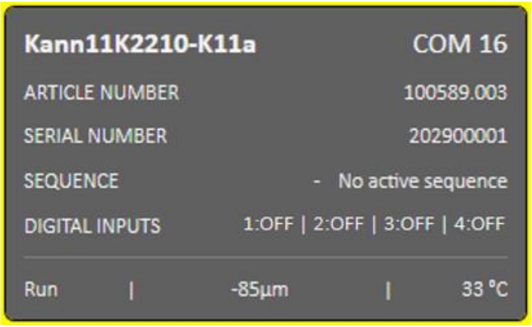
SERIAL NUMBER 2003000001

CANWARNING | CANPASSIVE

Most of the errors are fixed by the controller itself. Should an error stay, **disconnect and connect** the device again, same with converters.

3.3 Serial drives

Serial drives are connected over RS232 -TTL or RS485 -TTL converter and are displayed directly.



Kann11K2210-K11a COM 16

ARTICLE NUMBER 100589.003

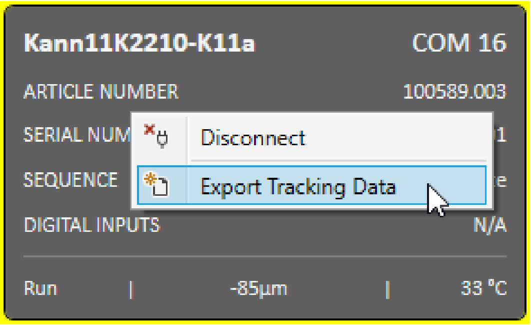
SERIAL NUMBER 202900001

SEQUENCE - No active sequence

DIGITAL INPUTS 1:OFF | 2:OFF | 3:OFF | 4:OFF

Run | -85µm | 33 °C

With righ-clicking on a selected serial drive, tracking data can be exported as a CSV file. Sometimes an advantage for solution finding.



Kann11K2210-K11a COM 16

ARTICLE NUMBER 100589.003

SERIAL NUM

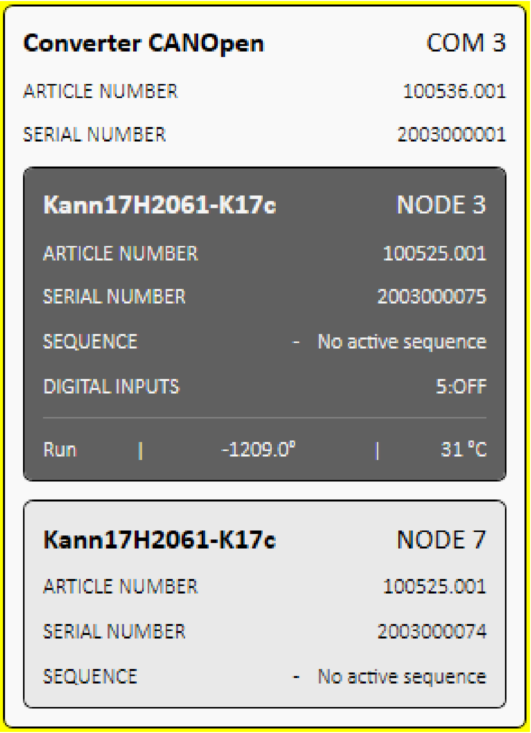
SEQUENCE

DIGITAL INPUTS N/A

Run | -85µm | 33 °C

3.4 Converter and CANopen drives

One or multiple CANopen drives can be displayed over a CANopen converter.



Converter CANOpen COM 3

ARTICLE NUMBER 100536.001

SERIAL NUMBER 2003000001

Kann17H2061-K17c NODE 3

ARTICLE NUMBER 100525.001

SERIAL NUMBER 2003000075

SEQUENCE - No active sequence

DIGITAL INPUTS 5:OFF

Run | -1209.0° | 31 °C

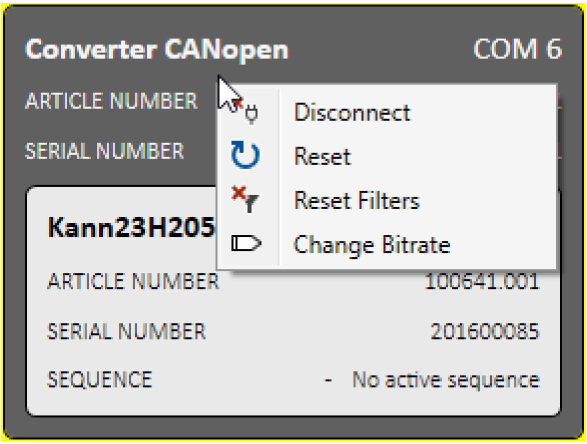
Kann17H2061-K17c NODE 7

ARTICLE NUMBER 100525.001

SERIAL NUMBER 2003000074

SEQUENCE - No active sequence

With right-clicking a selected converter, some commands are available.



Converter CANopen COM 6

ARTICLE NUMBER

SERIAL NUMBER

Kann23H205

ARTICLE NUMBER 100641.001

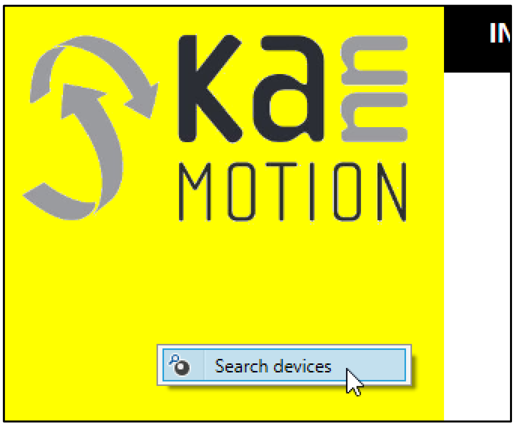
SERIAL NUMBER 201600085

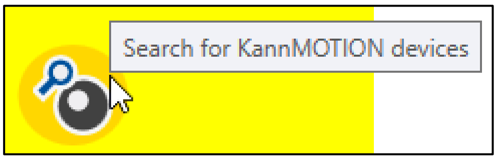
SEQUENCE - No active sequence

Disconnect	Disconnect device with all subdevices
Reset	Restarts the converter
Reset Filters	Sometimes needed after a firmware update, or if CANopen drives are not displayed
Change Bitrate	Change bitrate of converter, only possible if no drives connected

3.5 Search devices

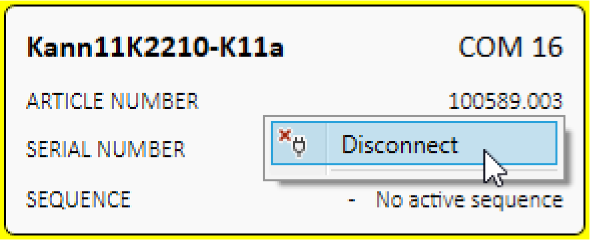
When starting the KannMOTION Manager, the KMM searches for devices automatically.
During runtime, right-click the yellow column or click the button at the bottom of the device tree to search for devices.





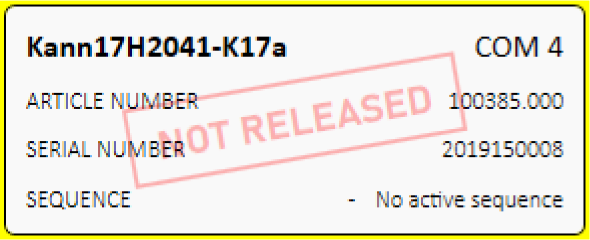
3.6 Disconnect devices

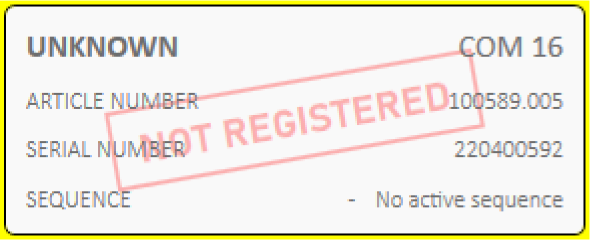
To disconnect a device, right-click devices and click on *Disconnect*.


Kann11K2210-K11a COM 16
ARTICLE NUMBER 100589.003
SERIAL NUMBER
SEQUENCE - No active sequence

3.7 Device status

If a device is not released or not registered, it is shown as following.


Kann17H2041-K17a COM 4
ARTICLE NUMBER 100385.000
SERIAL NUMBER 2019150008
SEQUENCE - No active sequence

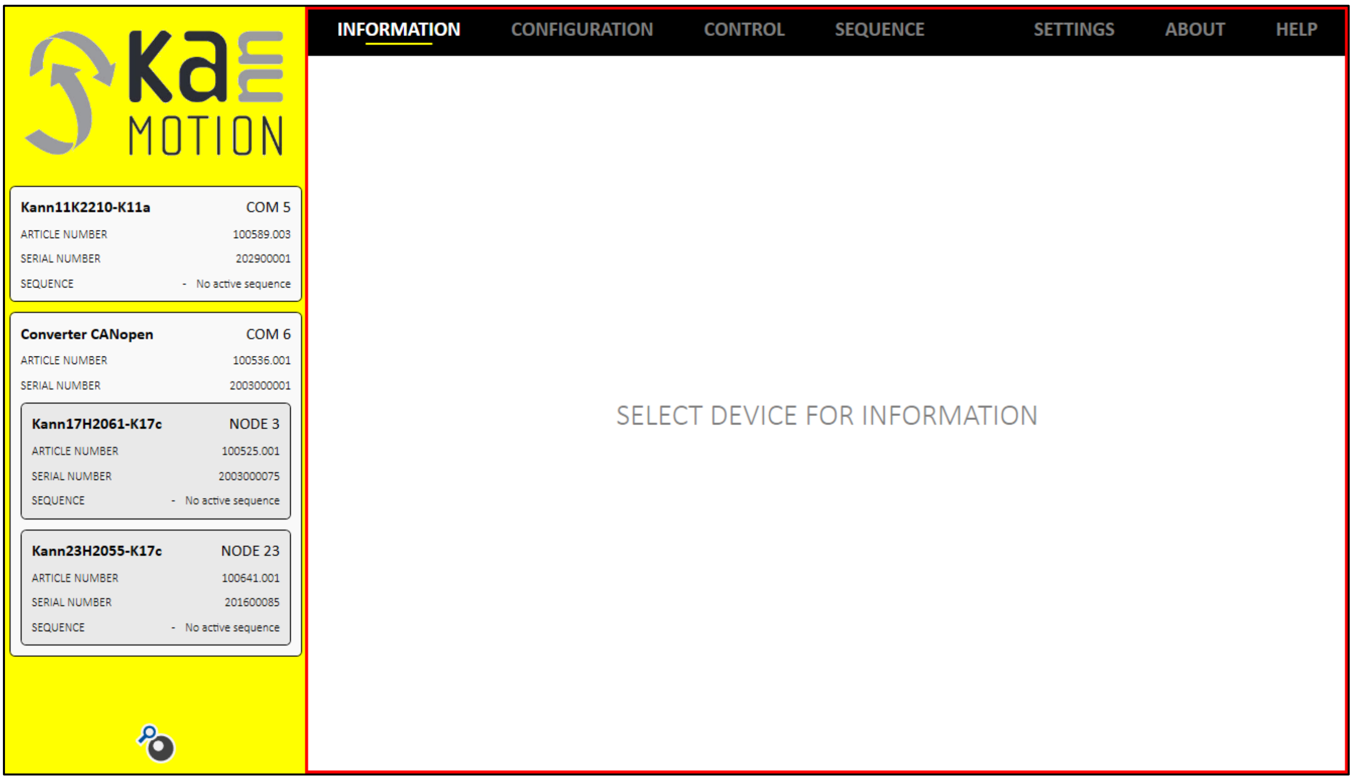

UNKNOWN COM 16
ARTICLE NUMBER 100589.005
SERIAL NUMBER 220400592
SEQUENCE - No active sequence

This means, either drive, controller and/or motor is not yet released by Adlos or it cannot be found in database.

⚠	**Not registered** devices cannot be interacted with. Contact Adlos for further support.

4 User cockpit

The user cockpit is for displaying information and controlling devices.



Kann11K2210-K11a COM 5
ARTICLE NUMBER 100589.003
SERIAL NUMBER 202900001
SEQUENCE - No active sequence

Converter CANopen COM 6
ARTICLE NUMBER 100536.001
SERIAL NUMBER 2003000001

Kann17H2061-K17c NODE 3
ARTICLE NUMBER 100525.001
SERIAL NUMBER 2003000075
SEQUENCE - No active sequence

Kann23H2055-K17c NODE 23
ARTICLE NUMBER 100641.001
SERIAL NUMBER 201600085
SEQUENCE - No active sequence

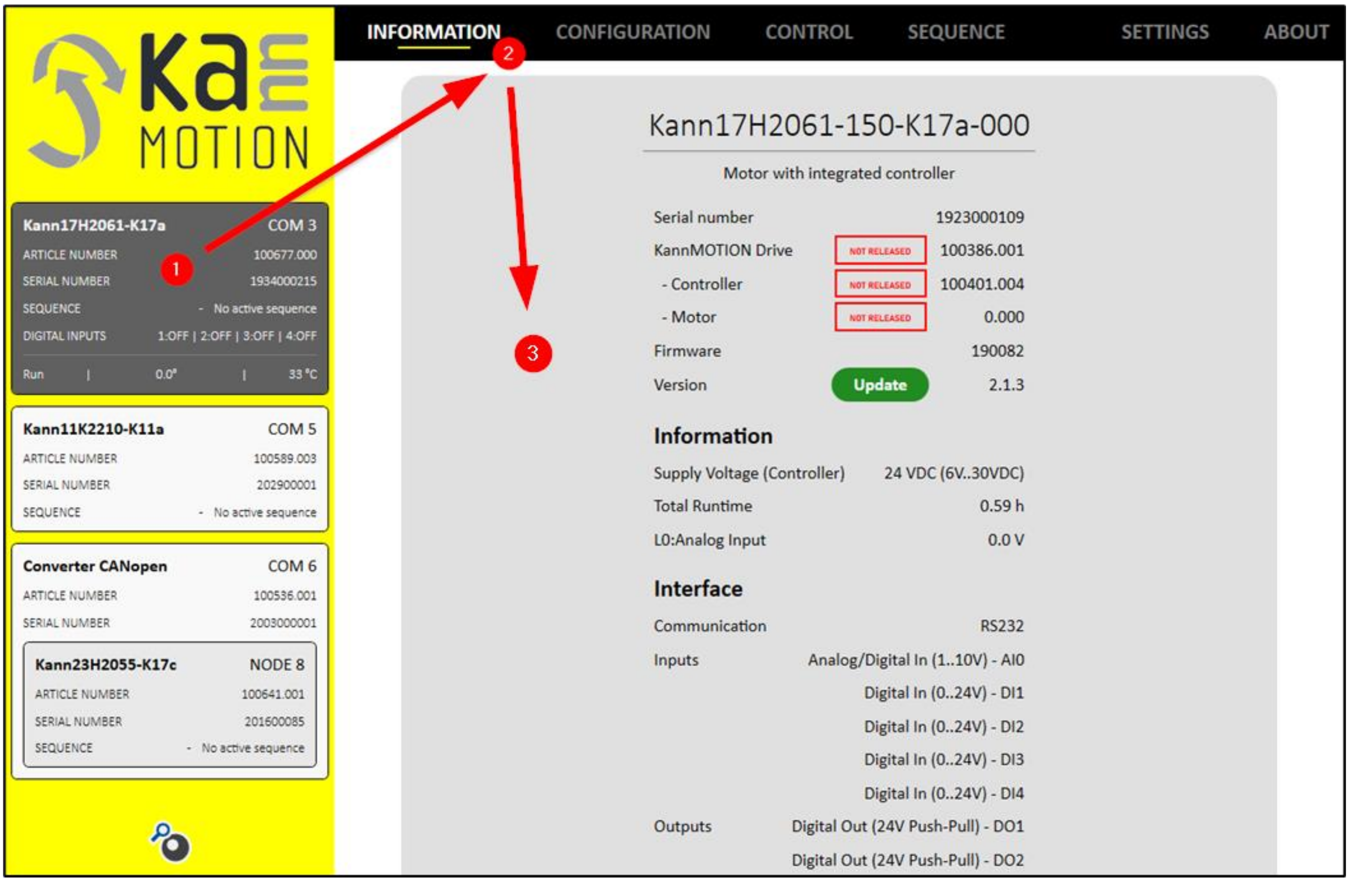
SELECT DEVICE FOR INFORMATION

The user cockpit consists of 6 parts:

Information	Contains further information about the selected drive
Configuration	Display and change configuration of selected drive
Control	Control the selected drive with commands
Sequence	Create and program sequences (PLC)
Settings	Application settings as: Application mode, Device tree, Control, Sequencer, Logging
About	Contains general app information, link to release notes, version numbers

4.1 Information

To get more detailed information about a drive or converter, select device in device tree an click the *Information* tab.



Kann17H2061-K17a COM 3
ARTICLE NUMBER 100677.000
SERIAL NUMBER 1934000215
SEQUENCE - No active sequence
DIGITAL INPUTS 1:OFF | 2:OFF | 3:OFF | 4:OFF
Run | 0.0° | 33 °C

Kann11K2210-K11a COM 5
ARTICLE NUMBER 100589.003
SERIAL NUMBER 202900001
SEQUENCE - No active sequence

Converter CANopen COM 6
ARTICLE NUMBER 100536.001
SERIAL NUMBER 2003000001

Kann23H2055-K17c NODE 8
ARTICLE NUMBER 100641.001
SERIAL NUMBER 201600085
SEQUENCE - No active sequence

Kann17H2061-150-K17a-000

Motor with integrated controller

Serial number 1923000109
KannMOTION Drive NOT RELEASED 100386.001
- Controller NOT RELEASED 100401.004
- Motor NOT RELEASED 0.000
Firmware 190082
Version Update 2.1.3

Information

Supply Voltage (Controller) 24 VDC (6V..30VDC)
Total Runtime 0.59 h
L0:Analog Input 0.0 V

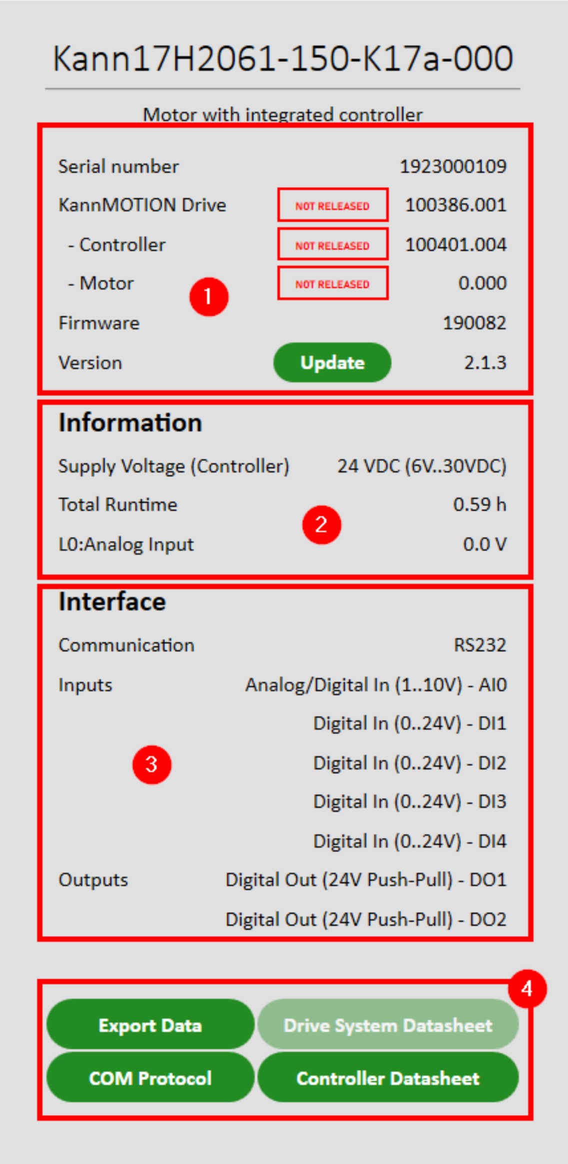
Interface

Communication RS232
Inputs Analog/Digital In (1..10V) - AI0
Digital In (0..24V) - DI1
Digital In (0..24V) - DI2
Digital In (0..24V) - DI3
Digital In (0..24V) - DI4
Outputs Digital Out (24V Push-Pull) - DO1
Digital Out (24V Push-Pull) - DO2

1	Select desired device
2	Click tab INFORMATION
3	See detailed information to device

4.1.1 Drive information

Information to the drive (serial or CANopen drive) are separated into 3 parts.



Kann17H2061-150-K17a-000

Motor with integrated controller

Serial number 1923000109

KannMOTION Drive NOT RELEASED 100386.001

- Controller NOT RELEASED 100401.004

- Motor NOT RELEASED 0.000

Firmware 190082

Version 2.1.3

Information

Supply Voltage (Controller) 24 VDC (6V..30VDC)

Total Runtime 0.59 h

L0:Analog Input 0.0 V

Interface

Communication RS232

Inputs Analog/Digital In (1..10V) - AI0
Digital In (0..24V) - DI1
Digital In (0..24V) - DI2
Digital In (0..24V) - DI3
Digital In (0..24V) - DI4

Outputs Digital Out (24V Push-Pull) - DO1
Digital Out (24V Push-Pull) - DO2

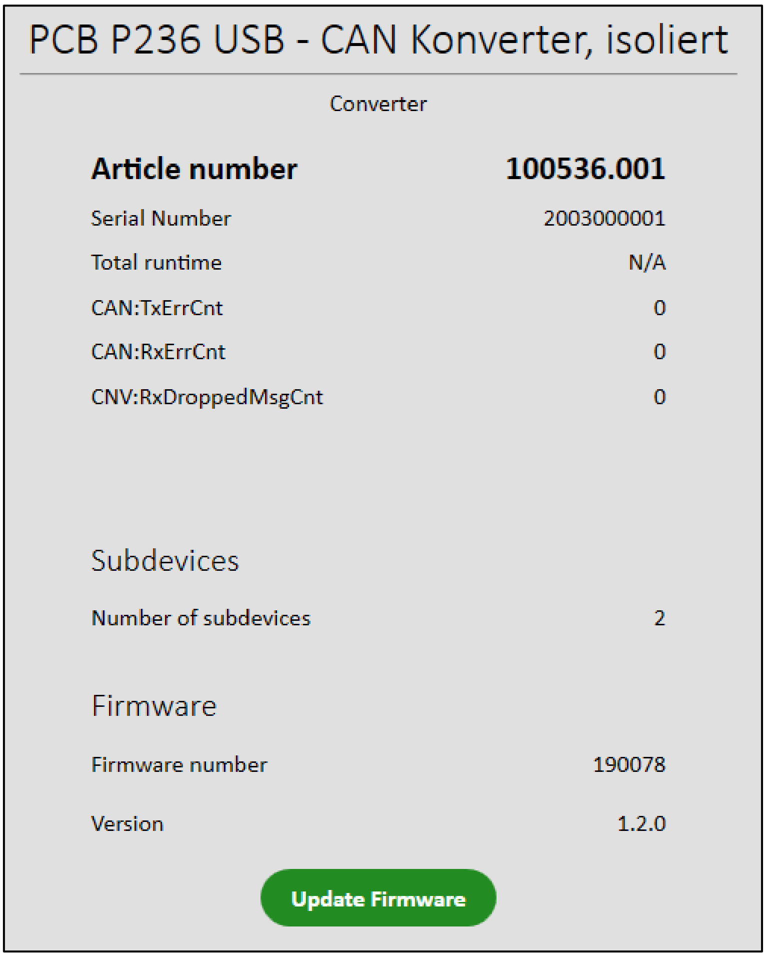
1	General information about the KannMOTION
2	Device information like supply voltage, runtime, etc.
3	Information about the interface
4	Extended information like manuals and datasheets

⚠	The information shown may vary across different devices.

4.1.2 Converter information

Information about the converter (CANopen converter).

The content includes general information, as well as error counters. Additionally, it shows the number of sub devices detected on the converter, and finally, the firmware details.



PCB P236 USB - CAN Konverter, isoliert

Converter

Article number	**100536.001**
Serial Number	2003000001
Total runtime	N/A
CAN:TxErrCnt	0
CAN:RxErrCnt	0
CNV:RxDroppedMsgCnt	0

Subdevices

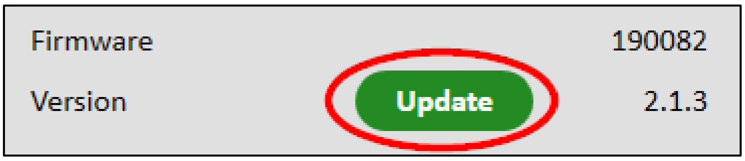
Number of subdevices	2

Firmware

Firmware number	190078
Version	1.2.0

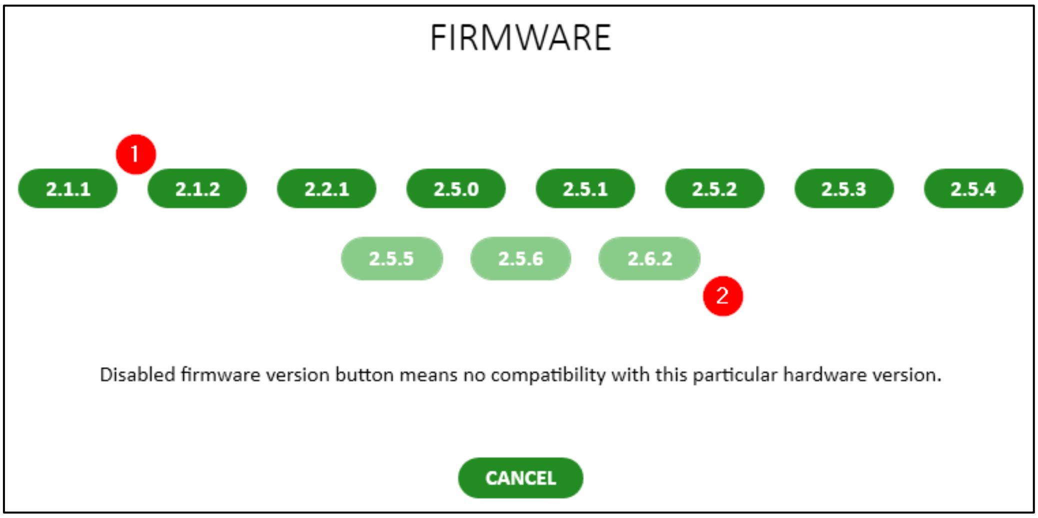
4.1.3 Firmware update

To update the firmware, click *Update Firmware* in the *Controller* section and choose the firmware version that should be programmed onto controller.



Firmware 190082

Version 2.1.3



FIRMWARE

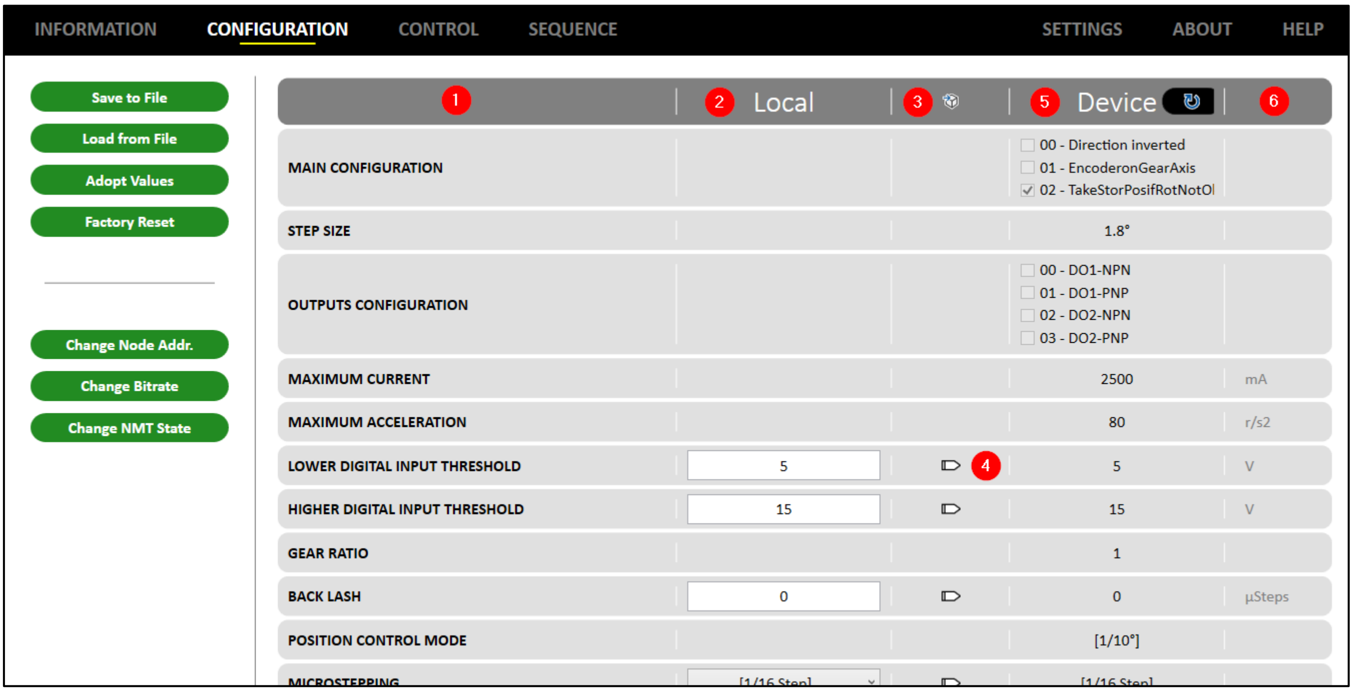
Disabled firmware version button means no compatibility with this particular hardware version.

1	Possible firmware versions for this controller
2	Disabled: Available firmware versions for controller, but not for this hardware version

⚠	It is recommended to work with the latest available firmware version.

4.2 Configuration

The *CONFIGURATION* is to configure your drive.



1	2 Local	3	5 Device	6
MAIN CONFIGURATION			□ 00 - Direction inverted □ 01 - EncoderonGearAxis ☑ 02 - TakeStorPosifRotNotOI	
STEP SIZE			1.8°	
OUTPUTS CONFIGURATION			□ 00 - DO1-NPN □ 01 - DO1-PNP □ 02 - DO2-NPN □ 03 - DO2-PNP	
MAXIMUM CURRENT			2500	mA
MAXIMUM ACCELERATION			80	r/s2
LOWER DIGITAL INPUT THRESHOLD	5	4	5	V
HIGHER DIGITAL INPUT THRESHOLD	15		15	V
GEAR RATIO			1	
BACK LASH	0		0	µSteps
POSITION CONTROL MODE			[1/10°]	
MICROSTEPPING	[1/16 Step]		[1/16 Step]	

Save to File	Save the configuration of the column «Local» to a CSV-File
Load from File	Import configuration from a CSV-File into the column «Local»
Adopt Values	Set configuration from the column «Device» to the column «Local»
Factory Reset	Factory reset of KannMOTION configuration.
Change Node Addr.	Change node address of selected device. *Only for CANopen devices!*
Change NMT State	Change the NMT state of the device. *Only for CANopen devices!*
Change Bitrate	Change the bitrate of the device. Changes bitrate on device and converter. Only possible if just one device at converter. *Only for CANopen devices!*

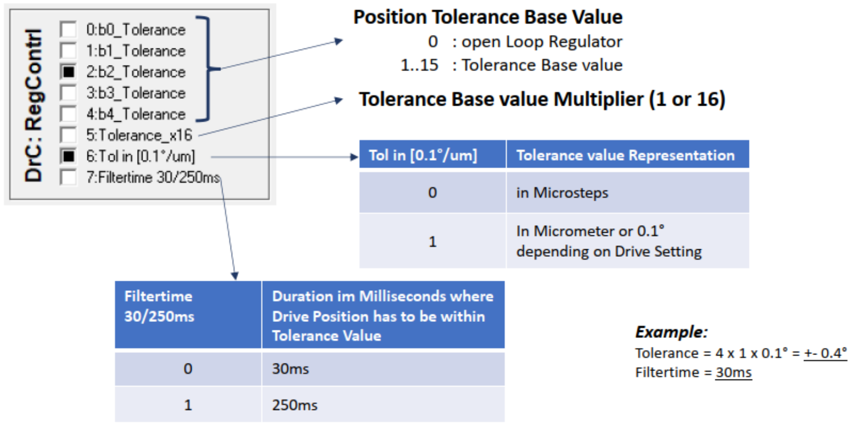
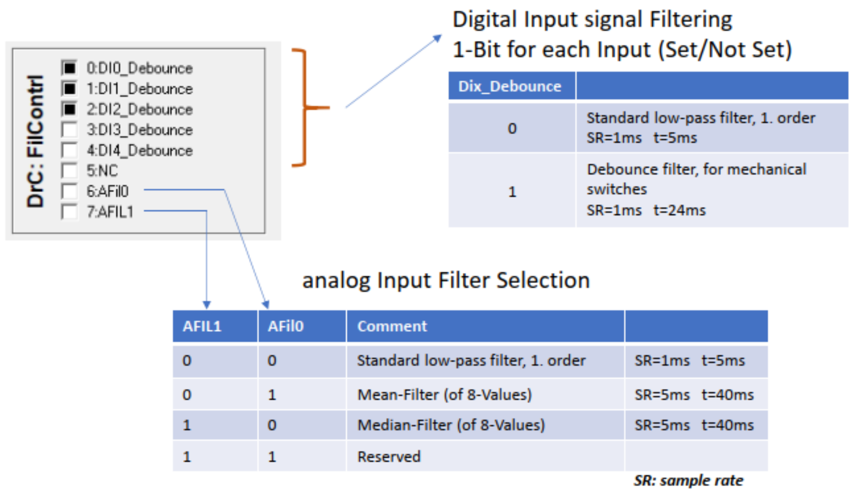
⚠	Factory Reset not available for all drives!

1	Description
2	Local data, to be written onto controller

3	Writing of **all changed configuration** from «Local»-column onto controller
4 ▷	Writing of **a single configuration** from «Local»-column controller
5	Configuration on controller ↻ → Read configuration from controller
6	Unit of configuration

For all KannMOTION drives, following configurations are available.

Configuration	***Access****	***Description***
Main configuration	RO	Main configuration bits
Step size	RO	Step size
Outputs configuration	RW	Configuration of digital outputs. If outputs are used, it is mandatory to configure at least one of them.
Maximum current	RO	
Maximum acceleration	RO	
Lower digital input threshold	RW	
Higher digital input threshold	RW	
Gear ratio	RO	
Back lash	RW	Back lash of gearbox
Position control mode	RO	Micrometer or 0.1°
Microstepping	RW	Number of micro steps per step
Minimum velocity	RW	
Maximum velocity	RW	
Holding torque	RW	Torque while motor is standing still, in % of maximum current
Acceleration torque	RW	Torque during acceleration phase, in % of maximum current
Run torque	RW	Torque while motor is running, in % of maximum current
Deceleration torque	RW	Torque during deceleration phase, in % of maximum current
Acceleration	RW	Acceleration, in % of maximum acceleration

Deceleration	RW	Deceleration, in % of maximum acceleration
Position regulator	RW	
Filter configuration	RW	

DrC: RegContrl

- ☐ 0:b0_Tolerance
- ☐ 1:b1_Tolerance
- ■ 2:b2_Tolerance
- ☐ 3:b3_Tolerance
- ☐ 4:b4_Tolerance
- ☐ 5:Tolerance_x16
- ■ 6:Tol in [0.1°/um]
- ☐ 7:Filtertime 30/250ms

Position Tolerance Base Value

0	: open Loop Regulator
1..15	: Tolerance Base value

Tolerance Base value Multiplier (1 or 16)

Tol in [0.1°/um]	Tolerance value Representation
0	in Microsteps
1	In Micrometer or 0.1° depending on Drive Setting

Filtertime 30/250ms	Duration im Milliseconds where Drive Position has to be within Tolerance Value
0	30ms
1	250ms

Example:
Tolerance = 4 x 1 x 0.1° = +- 0.4°
Filtertime = 30ms

DrC: FilContrl

- ■ 0:DI0_Debounce
- ■ 1:DI1_Debounce
- ■ 2:DI2_Debounce
- ☐ 3:DI3_Debounce
- ☐ 4:DI4_Debounce
- ☐ 5:NC
- ☐ 6:AFil0
- ☐ 7:AFIL1

Digital Input signal Filtering
1-Bit for each Input (Set/Not Set)

Dix_Debounce	
0	Standard low-pass filter, 1. order SR=1ms t=5ms
1	Debounce filter, for mechanical switches SR=1ms t=24ms

analog Input Filter Selection

AFIL1	AFil0	Comment	
0	0	Standard low-pass filter, 1. order	SR=1ms t=5ms
0	1	Mean-Filter (of 8-Values)	SR=5ms t=40ms
1	0	Median-Filter (of 8-Values)	SR=5ms t=40ms
1	1	Reserved	

SR: sample rate

** RO: read only / RW: read write*

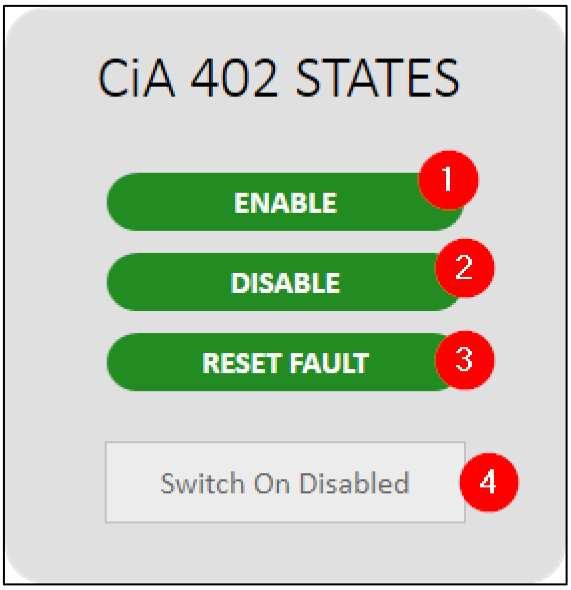
4.3 Control

Control the drive with different commands.

- CiA 402 States (only with CANopen Devices)
- Position control
- Velocity control
- Homing

4.3.1 CiA 402 States

Only available with CANopen devices with CiA 402 implemented (FW Version > 3.2.0).

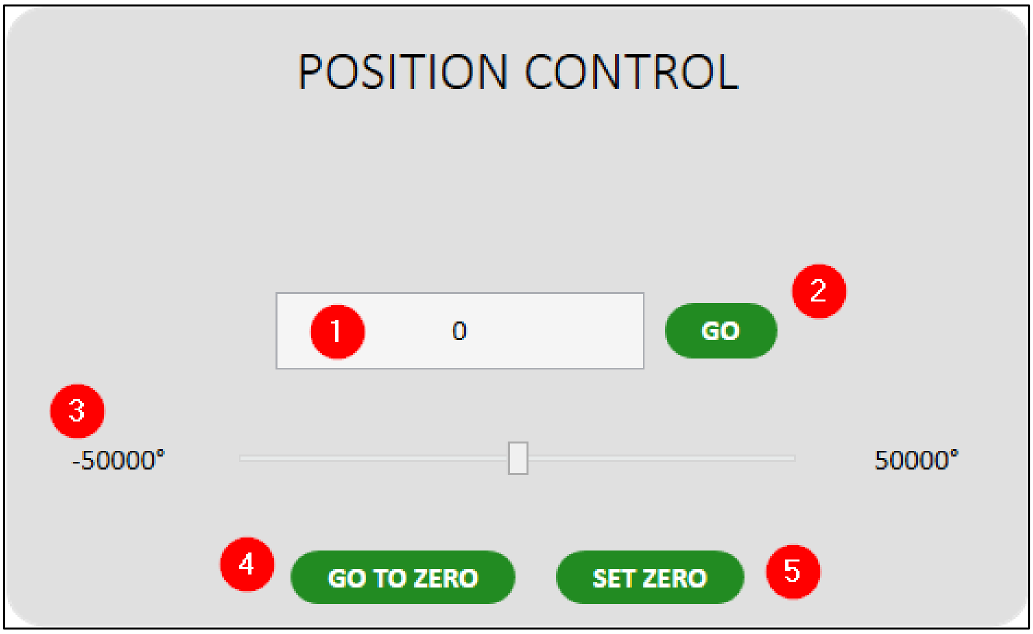


CiA 402 STATES

Switch On Disabled

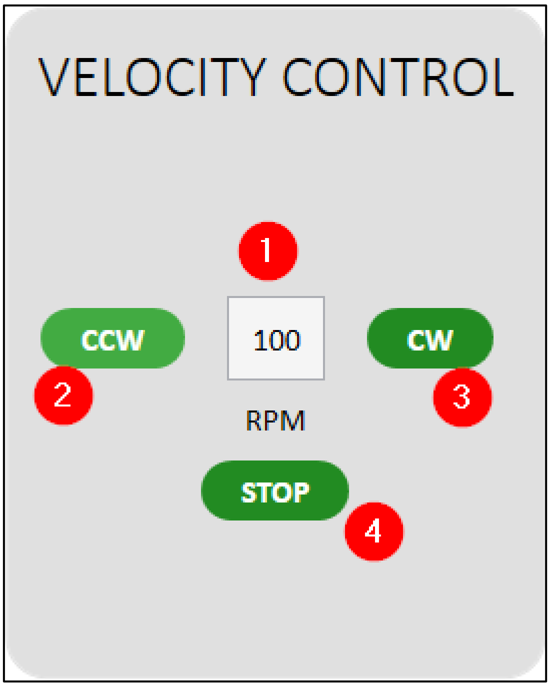
1	Switch from CiA 402 state "Switch ON Disabled" to "Operation Enable". Necessary to send commands.
2	Switch from CiA 402 state "Operation Enable" to "Switch ON Disabled"
3	Switch from CiA 402 state "Fault" to "Switch ON Disabled"
4	Current CiA 402 state displayed

4.3.2 Position control



1	Desired position in [µm] or [°]
2	Move to position defined in 1
3	Display of actual and target position with min/max (configurable in tab *SETTINGS*)
4	Move to position 0
5	Set current position to 0

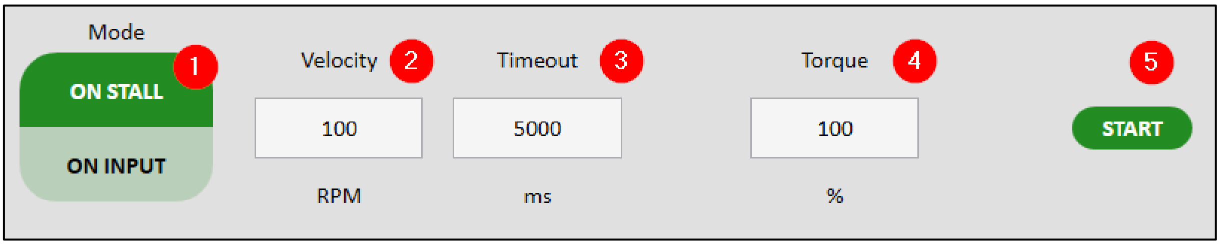
4.3.3 Velocity control



1	Desired speed in [rpm]
2	Move counter clockwise (CCW)
3	Move clockwise (CW)
4	Stop moving

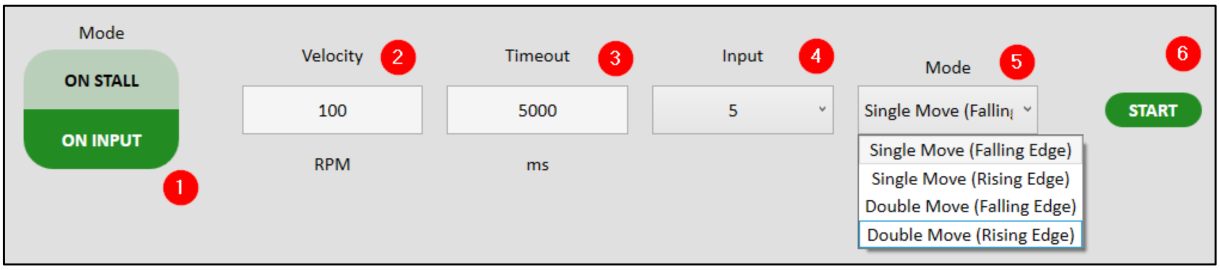
4.3.4 Homing

Homing on stall



1	Mode, here *on stall* active
2	Velocity to home with
3	Timeout, error if not stall within time
4	Torque in [%] of *torque run*
5	Start homing

Homing on Input

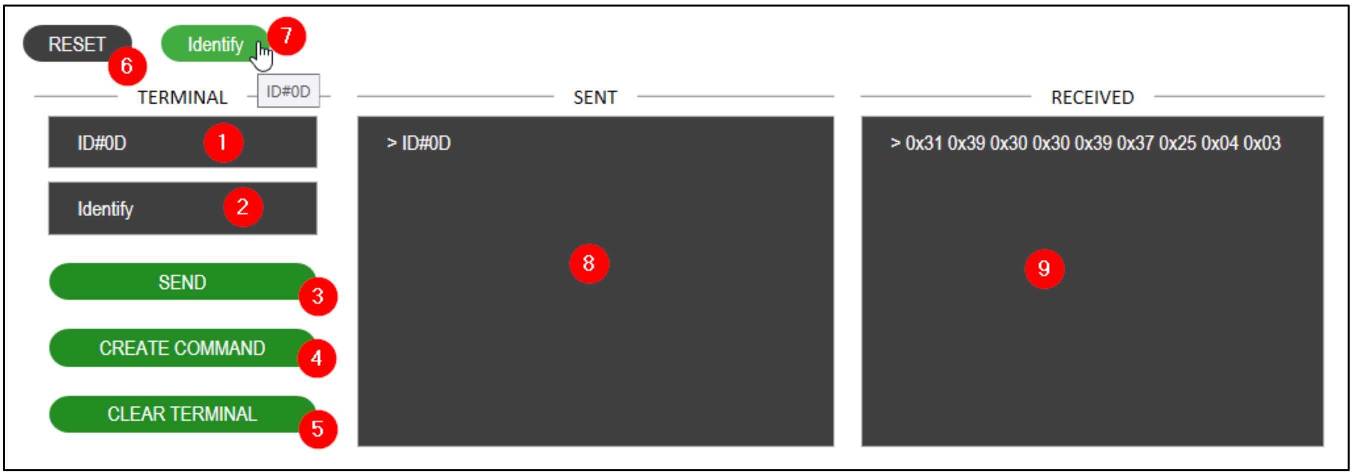

Mode
ON STALL
ON INPUT
1
Velocity
2
100
RPM
Timeout
3
5000
ms
Input
4
5
Mode
5
Single Move (Fallin
Single Move (Falling Edge)
Single Move (Rising Edge)
Double Move (Falling Edge)
Double Move (Rising Edge)
6
START

1	Mode, here *on input* active
2	Velocity to home with
3	Timeout, error if input not detected within time
4	Selected input to react to
5	Input detecting mode
6	Start homing

4.3.5 Command

4.3.5.1 Serial Terminal

The command part is only available, if *APPLICATION MODE* is set to *Expert* (change in tab *SETTINGS*).



ID#0D

Identify

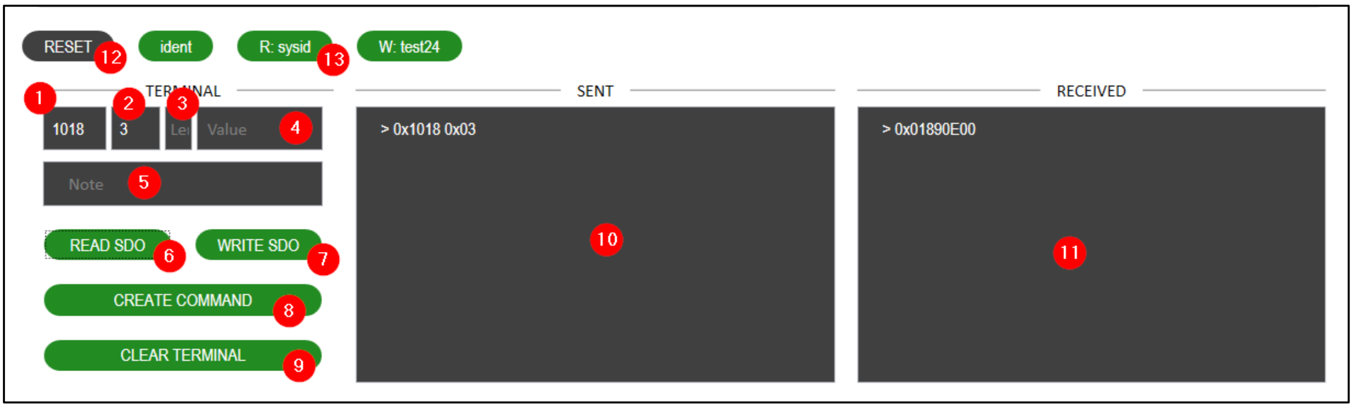
```
> ID#0D
```

```
> 0x31 0x39 0x30 0x30 0x39 0x37 0x25 0x04 0x03
```

1	Write command to be sent
2	Note to command (Content of button made with CREATE COMMAND)
3	Send command
4	Create command button with command in (1) + (2) as seen in (7)
5	Clear SENT and RECEIVED terminal
6	Predefined RESET command, click to send
7	Created ID#0D command as example, left click to send, right click to delete
8	SENT terminal
9	RECEIVED terminal

4.3.5.2 CANopen Terminal

The command part is only available, if *APPLICATION MODE* is set to *Expert* (change in tab *SETTINGS*).



```
> 0x1018 0x03
```

```
> 0x01890E00
```

1	SDO index in HEX w/o '0x'
2	SDO Subindex in HEX w/o '0x'
3	Length (number of bytes) in DEC (only needed if WRITE SDO)
4	Value in DEC (only needed if WRITE SDO)
5	Note to command (Content of button made with CREATE COMMAND)
6	Read SDO
7	Write SDO
8	Create command button with SDO Read/Write as seen in (13)
9	Clear SENT and RECEIVED terminal
10	SENT terminal
11	RECEIVED terminal
12	Predefined RESET command, click to send
13	Created SDO (R: Read, W: Write) as example, left click to send, right click to delete

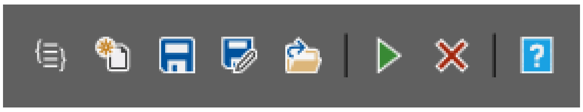
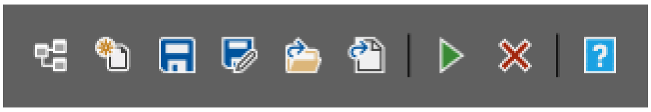
4.4 Sequence

Create sequences with the flow chart or c editor and program it onto your device.

Switch editors in *SETTINGS* → *Sequencer* or buttons in editor.

4.4.1 Menu bar

Create, load, program and more with the menu bar.

Flowchart Editor	C Editor
	

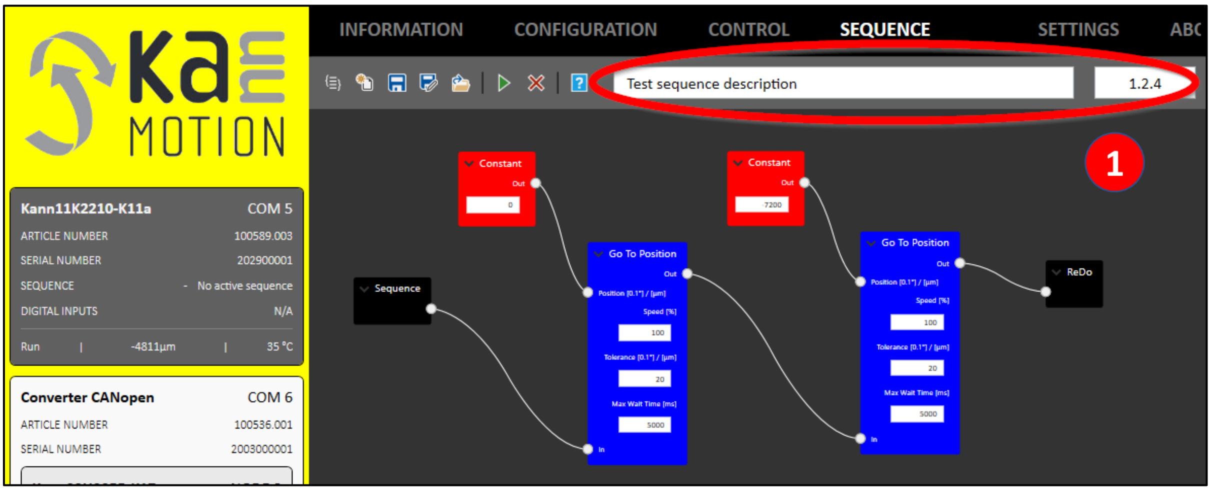
{≡}	Change to c-editor
	Change to flowchart-editor
	Create a new sequence
💾	Save sequence - Only enabled if it is an opened sequence
	Save sequence as…
📂	Load an existing sequence into the editor
	Load last compiled (programmed) sequence into editor. Only available in c-editor
▶	Program sequence onto drive Is this button disabled, the sequence is not programmable, maybe some block-connections (Flowchart Editor) are missing or no device selected.
✕	Delete sequence from drive
?	Open help

4.4.2 Versioning and description

⚠	It is recommended to add a version and a description to your sequence when programming, so you can determine later what is programmed on your drive.
⚠	Not all features are available on all devices. Check if the functions are possible for the existing device.

The sequence can be versioned and described. This will be programmed with the sequence and helps to identify an existing sequence on a drive.

In the Flow Chart Editor, the version and description can be defined in the menu bar (1).



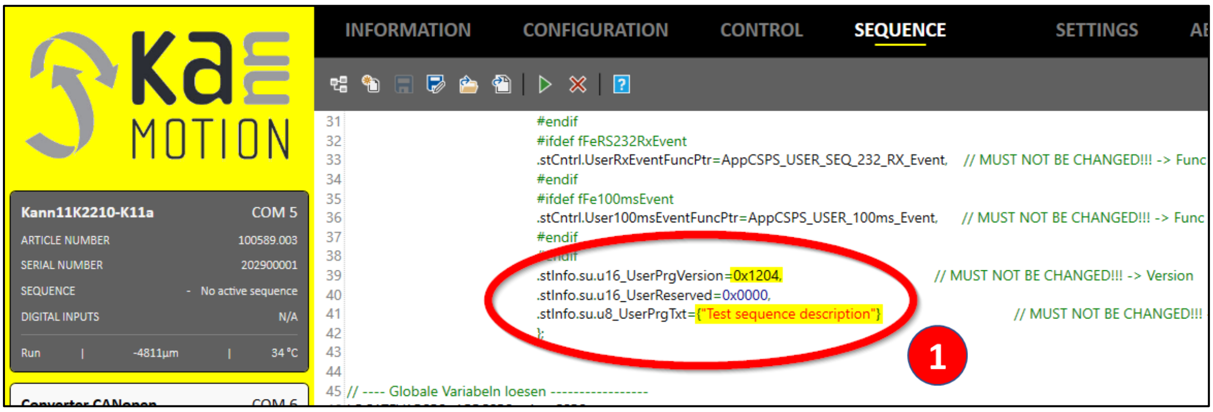
Test sequence description 1.2.4

Kann11K2210-K11a COM 5
ARTICLE NUMBER 100589.003
SERIAL NUMBER 202900001
SEQUENCE - No active sequence
DIGITAL INPUTS N/A
Run -4811µm 35 °C

Converter CANopen COM 6
ARTICLE NUMBER 100536.001
SERIAL NUMBER 2003000001

Constant: 0; Constant: 7200; Go To Position: 100, 20, 5000; Go To Position: 100, 20, 5000

In the C-Editor, the version and description can be defined as shown below.



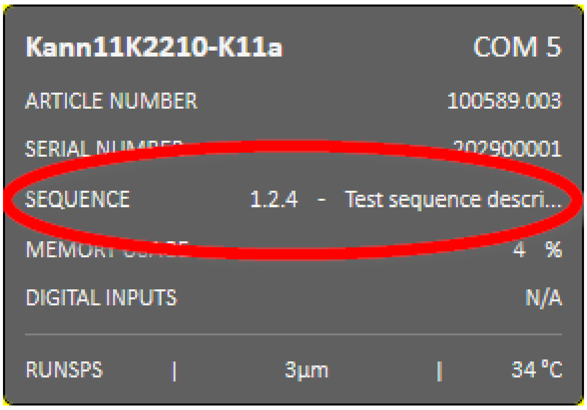
```
                #endif
                #ifdef fFeRS232RxEvent
                .stCntrl.UserRxEventFuncPtr=AppCSPS_USER_SEQ_232_RX_Event,   // MUST NOT BE CHANGED!!! -> Func
                #endif
                #ifdef fFe100msEvent
                .stCntrl.User100msEventFuncPtr=AppCSPS_USER_100ms_Event,     // MUST NOT BE CHANGED!!! -> Func
                #endif
                #endif
                .stInfo.su.u16_UserPrgVersion=0x1204,                        // MUST NOT BE CHANGED!!! -> Version
                .stInfo.su.u16_UserReserved=0x0000,
                .stInfo.su.u8_UserPrgTxt={"Test sequence description"}          // MUST NOT BE CHANGED!!!
                };


// ---- Globale Variabeln loesen -----------------
```

Kann11K2210-K11a COM 5
ARTICLE NUMBER 100589.003
SERIAL NUMBER 202900001
SEQUENCE - No active sequence
DIGITAL INPUTS N/A
Run -4811µm 34 °C

VERSION	Version of the sequence in HEX format. **Example**: Version 1.2.15 **1.2.15 → 0x120F**
DESCRIPTION	Description of the sequence. Maximum length is 32 characters.

Once programmed, the version and description will be shown in device tree.



Kann11K2210-K11a COM 5

ARTICLE NUMBER 100589.003

SERIAL NUMBER 202900001

SEQUENCE 1.2.4 - Test sequence descri...

MEMORY USAGE 4 %

DIGITAL INPUTS N/A

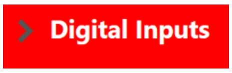
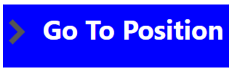
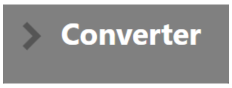
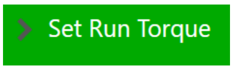
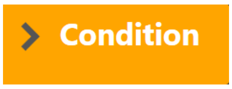
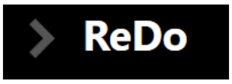
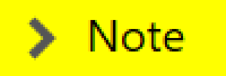
RUNSPS | 3µm | 34 °C

4.4.3 Flow Chart Editor

Choose the Flow Chart Editor to create sequences by drag and drop of command blocks.

4.4.3.1 Block types

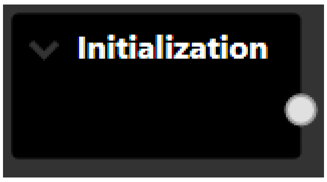
Each type of block is indicated by its color.

 Digital Inputs	Red – Input blocks
 Go To Position	Blue – Execution blocks
 Converter	Gray – Function blocks
 Set Run Torque	Green – Configuration blocks
 Condition	Orange – Condition blocks
 ReDo	Black – Start/Termination blocks
 Note	Yellow – Helping blocks

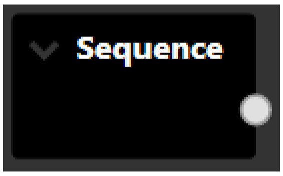
4.4.3.2 Blocks overview

Following, all available blocks are explained.

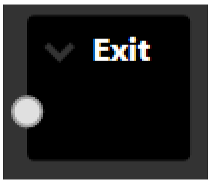
Initialization

Block	 Initialization
Description	Start of the initialization
Out	The first function of the initialization

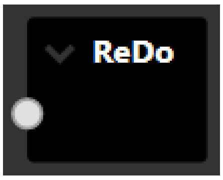
Sequence

Block	
Description	Start of the sequence
Out	The first statement of the sequence

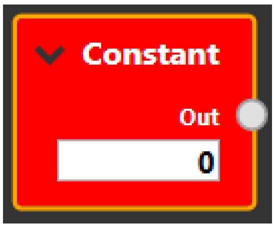
Exit

Block	
Description	Exit of the initialization part
In	The last statement of the initialization

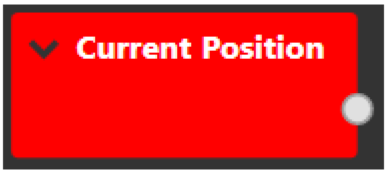
ReDo

Block	
Description	Starts the sequence loop from the beginning
In	The last statement of a sequence

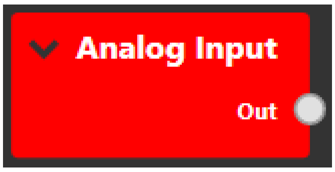
Constant

Block	
Description	A constant to go into another block Can be velocity, position, etc.
Out	Constant value
Input	Value in rpm / 0.1°/ µm / position

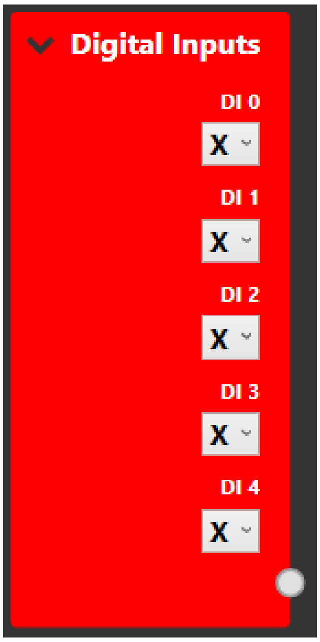
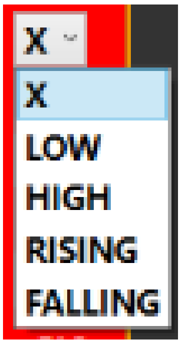
Current Position

Block	
Description	The current position to use as value for calculations or controlling tasks
Out	The position as value in the preset unit

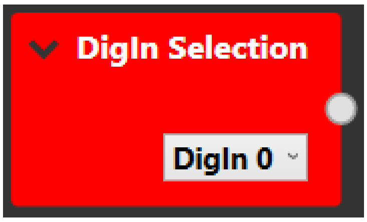
Analog Input

Block	
Description	The current value at the analog input
Out	Analog value in [mV] / [0.01mA]

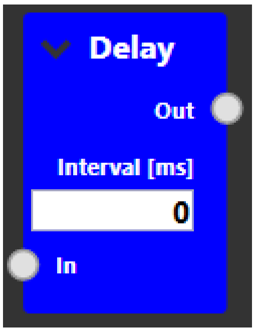
Digital Inputs

Block	 Digital Inputs DI 0 X DI 1 X DI 2 X DI 3 X DI 4 X
Description	Defines a state of the digital inputs which can be checked by a condition block
Out	State of the digital inputs
Input	X: Ignore LOW, HIGH, RISING, FALLING

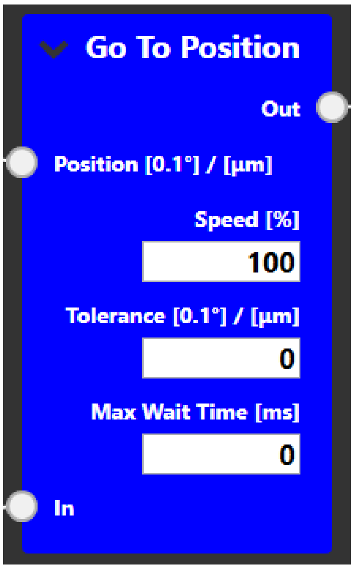
DigIn Selection

Block	 DigIn Selection DigIn 0
Description	Selects one of the digital inputs as sense channel
Out	Selected digital input
Input	Desired digital input

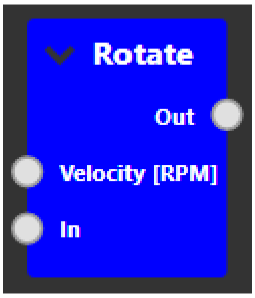
Delay

Block	 Delay Out Interval [ms] 0 In
Description	A delay of x milliseconds
In	Previous statement
Out	Following statement
Input	Delay interval in milliseconds

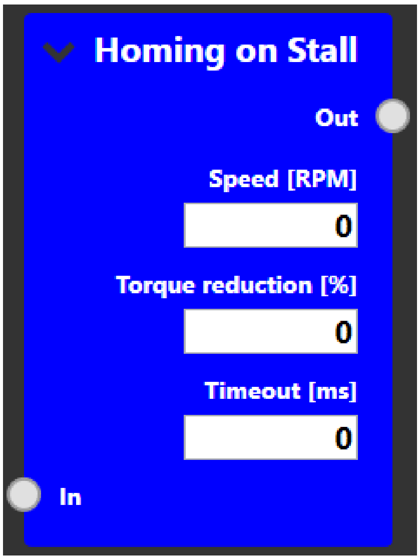
GoToPosition

Block	 Go To Position Out Position [0.1°] / [µm] Speed [%] 100 Tolerance [0.1°] / [µm] 0 Max Wait Time [ms] 0 In
Description	Go to position in 1/10 degree or micrometers, depending on wether the motor is rotative or linear Going to next statement if either position is reached inside given the tolerance or max wait time is up
In	Previous statement
Out	Following statement
Position	Position in 0.1° or µm (rotary or linear motor)
Input	1. Speed Speed in [%] of SpeedMax 2. Tolerance Position tolerance of a given position in which the drive is allowed to stop and go to the next statement 3. Max Wait Time Maximum time waited until going to next statement, if position is not already reached before

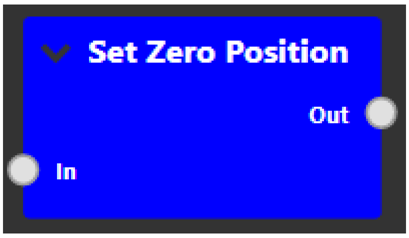
Rotate

Block	
Description	Rotate with the given velocity
In	Previous statement
Out	Following statement
Velocity	Velocity in rpm

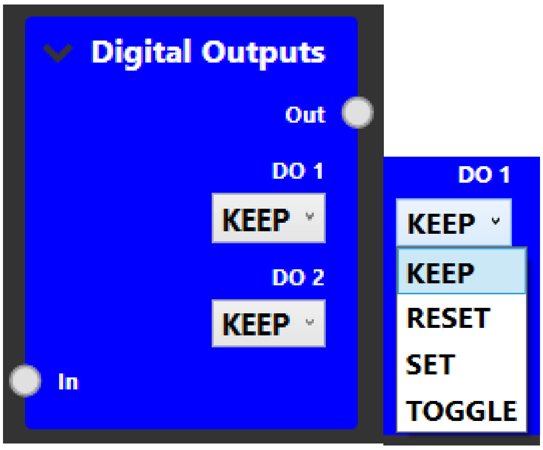
Homing on Stall

Block	 Homing on Stall Out Speed [RPM] 0 Torque reduction [%] 0 Timeout [ms] 0 In
Description	Homing on stall with set speed, torque reduction, and timeout
In	Previous statement
Out	Following statement
Input	- Speed in cycles per minute - Torque reduction in 0..100% - Timeout in milliseconds

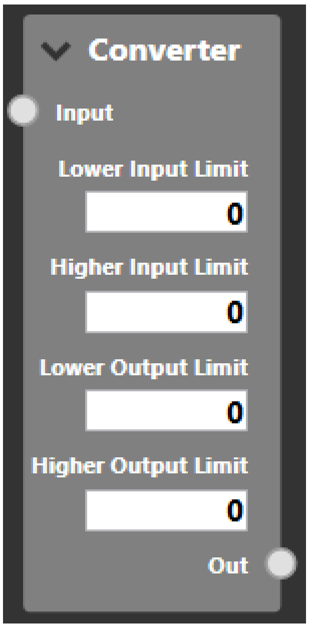
Set Zero Position

Block	
Description	Sets the actual position to 0 as long as the motor is on
In	Previous statement
Out	Following statement

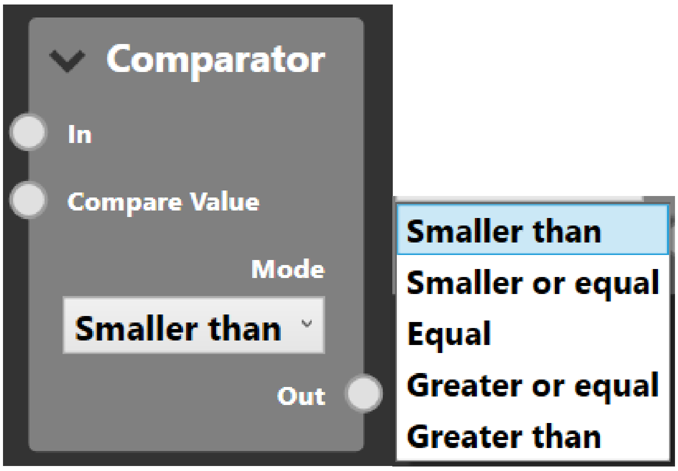
Digital Outputs

Block	 Digital Outputs Out DO 1 KEEP DO 2 KEEP In DO 1 KEEP KEEP RESET SET TOGGLE
Description	Sets the digital outputs on the selected value *KEEP:* keeps the value, nothing changes *RESET:* clears the output *SET:* sets the output *TOGGLE:* changes the value of the output Digital outputs only work if they are set in Motor Configuration Parameters (OUTPUTS CONFIGURATION)
In	Previous statement
Out	Following statement
Input	- State of digital output 1 - State of digital output 2

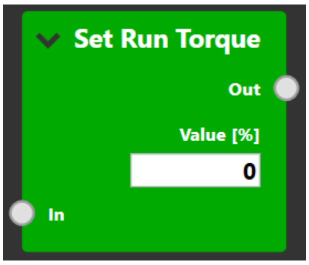
Converter

Block	 Converter Input Lower Input Limit 0 Higher Input Limit 0 Lower Output Limit 0 Higher Output Limit 0 Out
Description	Linear converting of a variable Calculation: $f(x) = m * x + q$ where $m = \frac{Limit_{HO} - Limit_{LO}}{Limit_{HI} - Limit_{LI}}$ $q = Limit_{LO} - m * Limit_{LI}$ with $Limit_{LI} = Lower\ Input\ Limit$ $Limit_{HI} = Higher\ Input\ Limit$ $Limit_{LO} = Lower\ Output\ Limit$ $Limit_{HO} = Higher\ Output\ Limit$
In	Value to convert
Out	Converted value
Input	- Lower Input Limit (min. volt. [mV] / curr. [mA]) - Higher Input Limit (max. volt. [mV] / curr. [mA]) - Lower Output Limit - Higher Output Limit

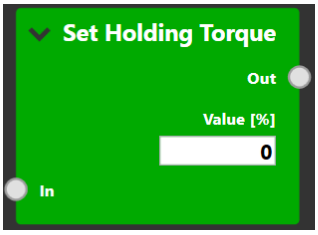
Comparator

Block	 Comparator In Compare Value Mode Smaller than Out Smaller than Smaller or equal Equal Greater or equal Greater than
Description	Comparator to compare an input with a value
In	Value to check (e.g. Analog In, Position)
Compare Value	Value to compare with (Constant)
Out	Boolean (true, false)
Input	Mode – the compare mode

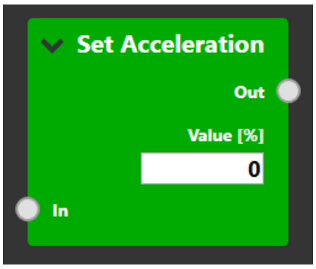
Set Run Torque

Block	
Description	Block for setting the run torque
In	Previous block
Out	Next block
Input	Value [%]: Run torque value (0..100%)

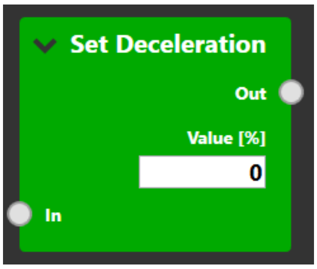
Set Holding Torque

Block	
Description	Block for setting the holding torque
In	Previous block
Out	Next block
Input	Value [%]: Holding torque value (0..100%)

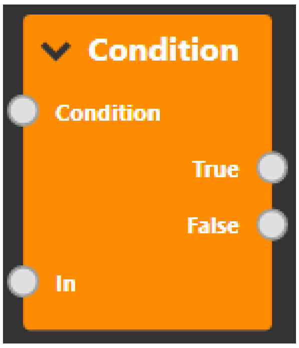
Set Acceleration

Block	
Description	Block for setting the acceleration
In	Previous block
Out	Next block
Input	Value [%]: Acceleration value (0..100%)

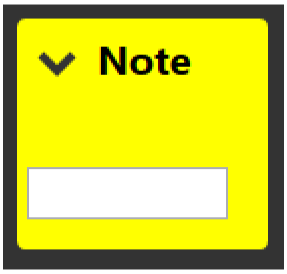
Set Deceleration

Block	
Description	Block for setting the deceleration
In	Previous block
Out	Next block
Input	Value [%]: Deceleration value (0..100%)

Condition

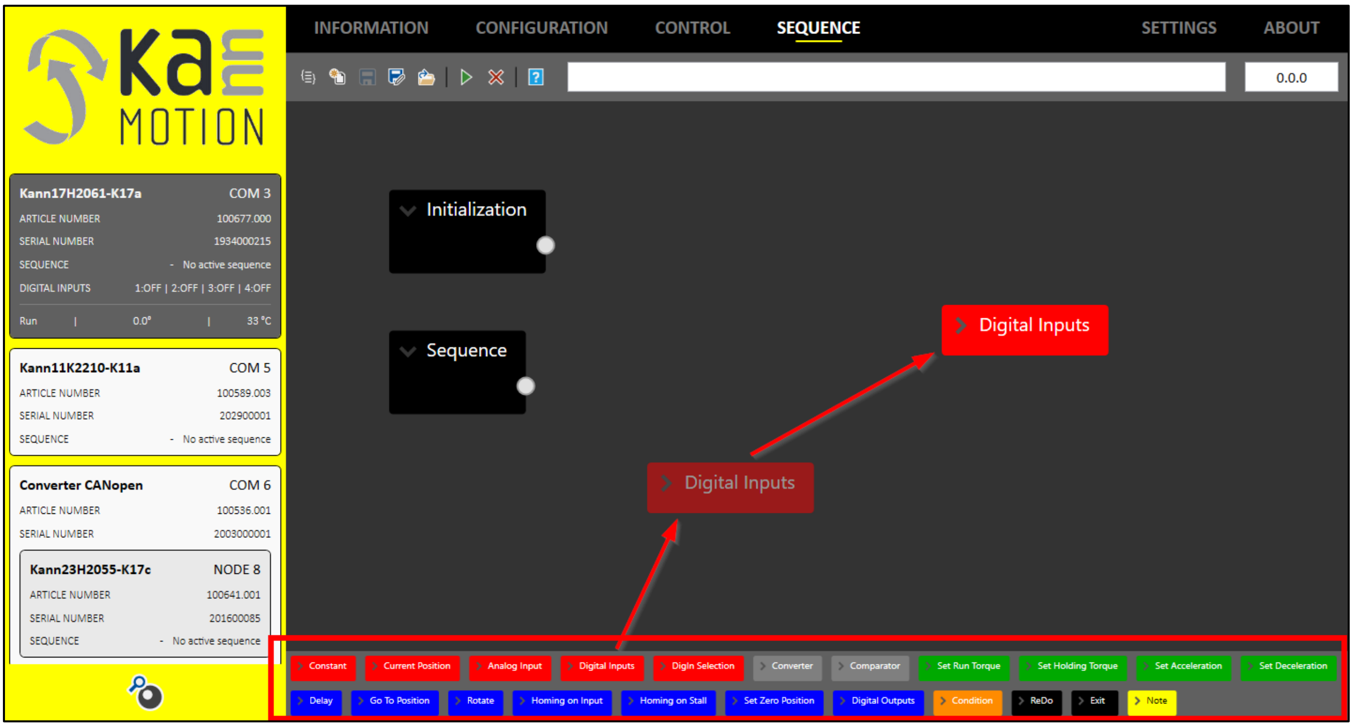
Block	
Description	A condition to continue different ways E.g. if Digital Input 3 is HIGH it's true, else false
In	Previous statement
True	Following statement if condition is true
False	Following statement if condition is false
Condition	A state to check (e.g,. the digital inputs)

Note

Block	
Description	Block for writing a note. Does not affect any of the sequence, only for explanations/information
In	–
Out	–

4.4.3.3 *Drag* & Drop

Drag a block from the bar at the bottom and drop it in the sequence area.



Kann17H2061-K17a COM 3
ARTICLE NUMBER 100677.000
SERIAL NUMBER 1934000215
SEQUENCE - No active sequence
DIGITAL INPUTS 1:OFF | 2:OFF | 3:OFF | 4:OFF
Run | 0.0° | 33 °C

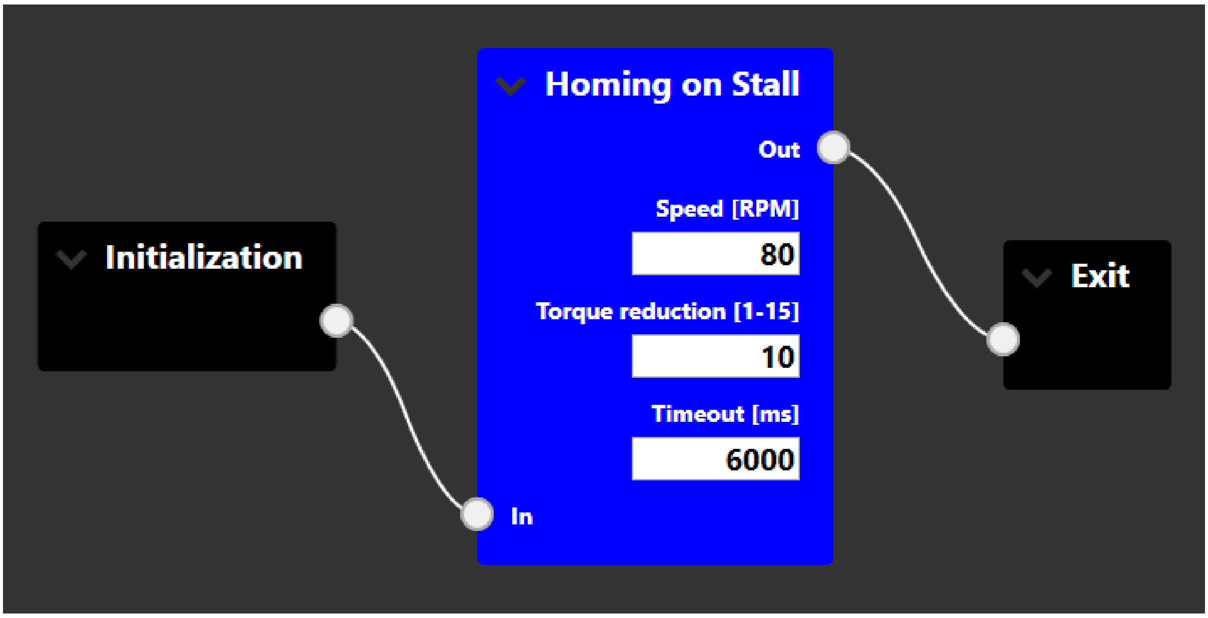
Kann11K2210-K11a COM 5
ARTICLE NUMBER 100589.003
SERIAL NUMBER 202900001
SEQUENCE - No active sequence

Converter CANopen COM 6
ARTICLE NUMBER 100536.001
SERIAL NUMBER 2003000001

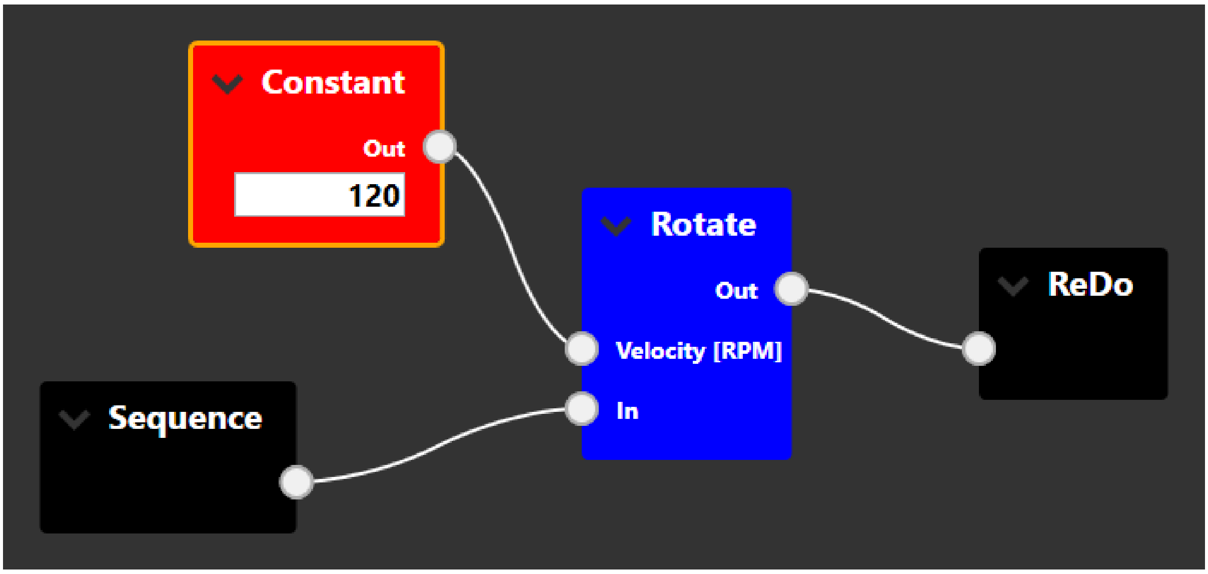
Kann23H2055-K17c NODE 8
ARTICLE NUMBER 100641.001
SERIAL NUMBER 201600085
SEQUENCE - No active sequence

4.4.3.4 Initialization and Sequence

An initialization has always to be terminated with an *Exit*-block.
This part of the sequence is done only once at the start of the sequence.

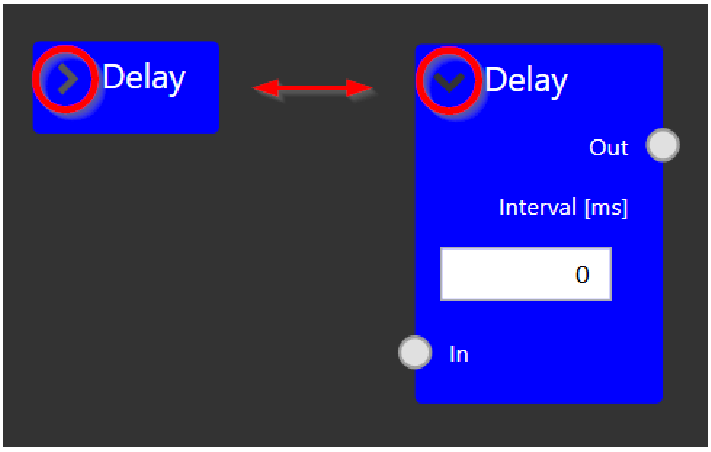


Every sequence must end with a *Redo*-block, which can be used multiple times if necessary.
This part will be looped on the drive every few milliseconds.

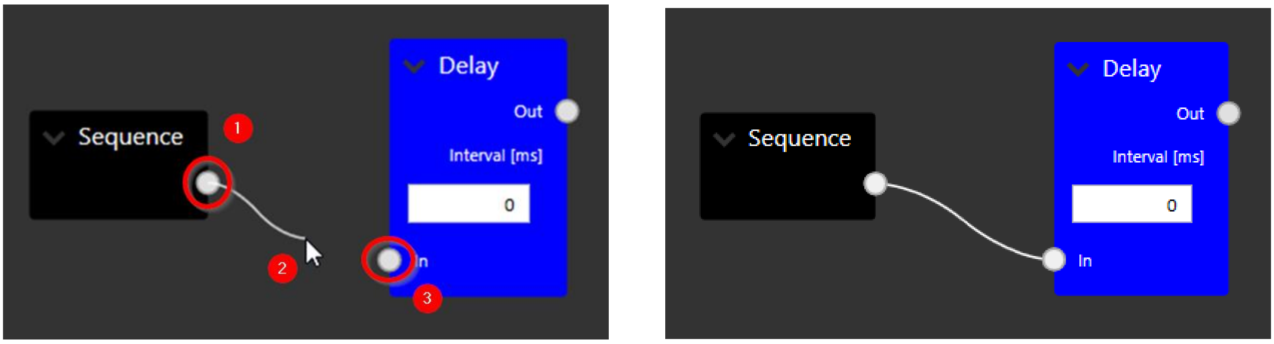


4.4.3.5 Connecting Blocks

Maximize/minimize a block with the arrow.



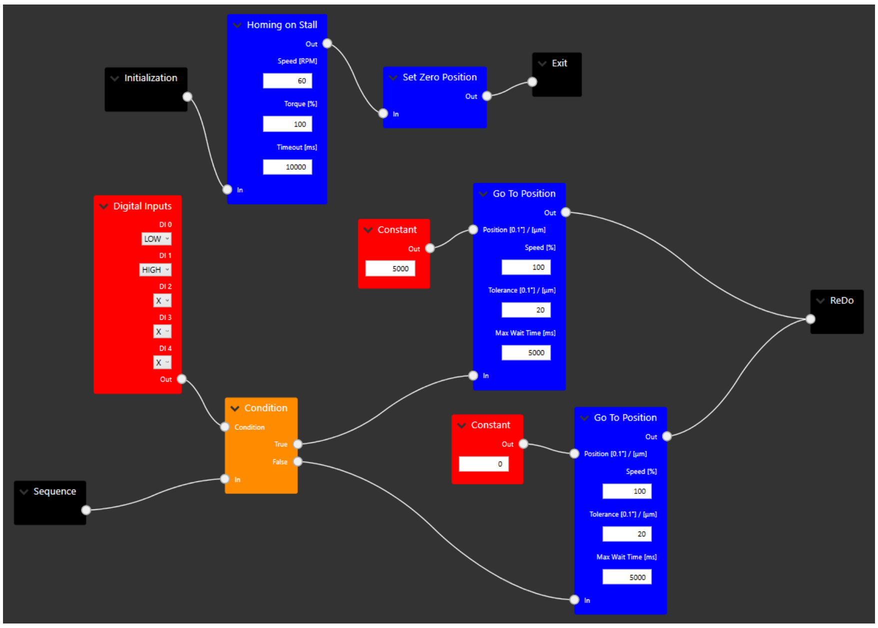
To connect two blocks, klick on circle of first block (1) and drag the line (2) to the circle of the next block (3).



To delete a connection, just click on the circle of the left block.

Following screenshot is an example of a simple sequence, which can be programmed to the drive.

- the upper sequence: starting by *Initialization* and ending by *Exit*
- the lower sequence: starting by *Sequence* and ending by *ReDo*



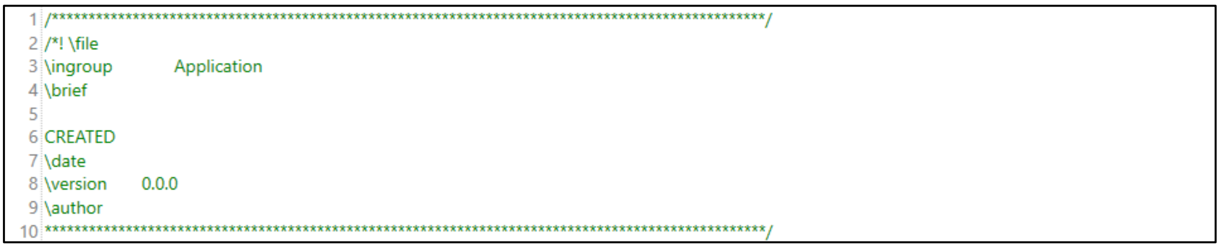
4.4.4 C Editor

Write the sequence in ANSI-C in predefined functions and create your applications like this.

See application notes 100631 and 100639 on https://www.kannmotion.com/downloads/ for more information and examples!

4.4.4.1 File description

To declare what's done with the sequence and add date, author and so on.

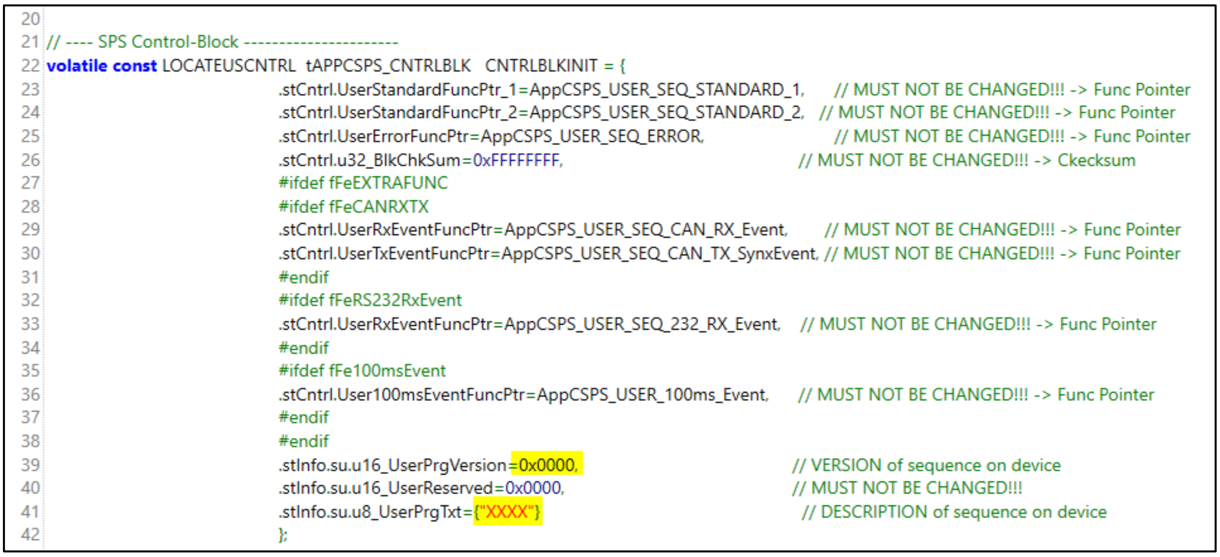


```
/*****************************************************************************************/
/*! \file
\ingroup         Application
\brief

CREATED
\date
\version      0.0.0
\author
*****************************************************************************************/
```

4.4.4.2 Control block

Control block with different ifdefs. Only sequence version and description should be changed by user.

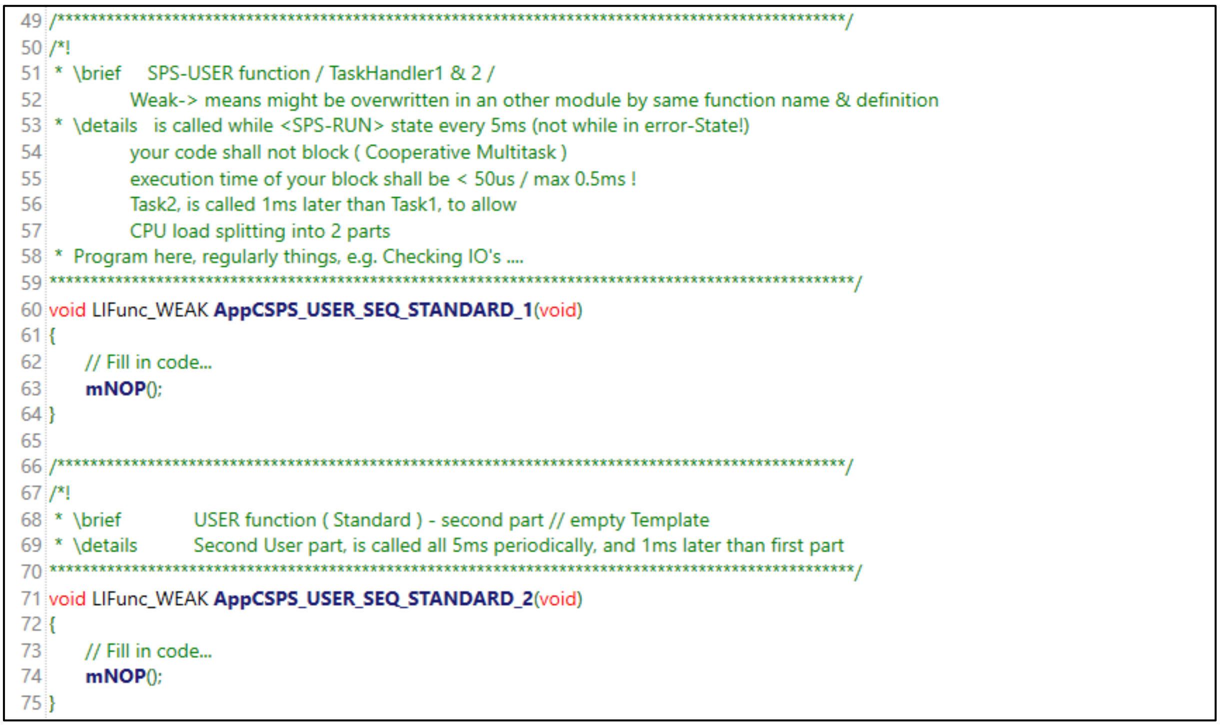


```

// ---- SPS Control-Block ----------------------
volatile const LOCATEUSCNTRL  tAPPCSPS_CNTRLBLK   CNTRLBLKINIT = {
                        .stCntrl.UserStandardFuncPtr_1=AppCSPS_USER_SEQ_STANDARD_1,     // MUST NOT BE CHANGED!!! -> Func Pointer
                        .stCntrl.UserStandardFuncPtr_2=AppCSPS_USER_SEQ_STANDARD_2,   // MUST NOT BE CHANGED!!! -> Func Pointer
                        .stCntrl.UserErrorFuncPtr=AppCSPS_USER_SEQ_ERROR,               // MUST NOT BE CHANGED!!! -> Func Pointer
                        .stCntrl.u32_BlkChkSum=0xFFFFFFFF,                           // MUST NOT BE CHANGED!!! -> Ckecksum
                        #ifdef fFeEXTRAFUNC
                        #ifdef fFeCANRXTX
                        .stCntrl.UserRxEventFuncPtr=AppCSPS_USER_SEQ_CAN_RX_Event,      // MUST NOT BE CHANGED!!! -> Func Pointer
                        .stCntrl.UserTxEventFuncPtr=AppCSPS_USER_SEQ_CAN_TX_SynxEvent, // MUST NOT BE CHANGED!!! -> Func Pointer
                        #endif
                        #ifdef fFeRS232RxEvent
                        .stCntrl.UserRxEventFuncPtr=AppCSPS_USER_SEQ_232_RX_Event,    // MUST NOT BE CHANGED!!! -> Func Pointer
                        #endif
                        #ifdef fFe100msEvent
                        .stCntrl.User100msEventFuncPtr=AppCSPS_USER_100ms_Event,     // MUST NOT BE CHANGED!!! -> Func Pointer
                        #endif
                        #endif
                        .stInfo.su.u16_UserPrgVersion=0x0000,                         // VERSION of sequence on device
                        .stInfo.su.u16_UserReserved=0x0000,                           // MUST NOT BE CHANGED!!!
                        .stInfo.su.u8_UserPrgTxt={"XXXX"}                              // DESCRIPTION of sequence on device
                        };
```

4.4.4.3 Main functions

Main functions for the sequence. These are called every 5ms, function 2 one millisecond later than function 1.
Implement your sequence here with a switch statement for example.

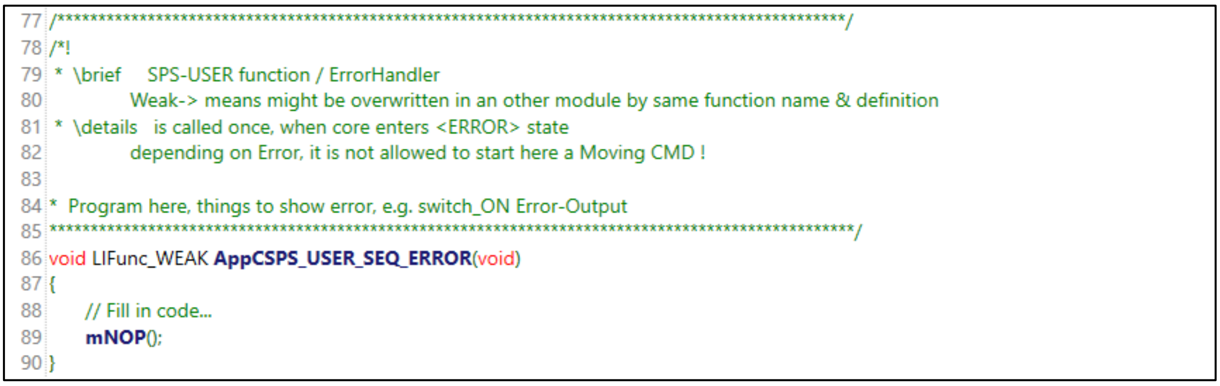


```
/******************************************************************************************/
/*!
 * \brief    SPS-USER function / TaskHandler1 & 2 /
             Weak-> means might be overwritten in an other module by same function name & definition
 * \details   is called while <SPS-RUN> state every 5ms (not while in error-State!)
             your code shall not block ( Cooperative Multitask )
             execution time of your block shall be < 50us / max 0.5ms !
             Task2, is called 1ms later than Task1, to allow
             CPU load splitting into 2 parts
 * Program here, regularly things, e.g. Checking IO's ....
*******************************************************************************************/
void LIFunc_WEAK AppCSPS_USER_SEQ_STANDARD_1(void)
{
    // Fill in code...
    mNOP();
}

/******************************************************************************************/
/*!
 * \brief          USER function ( Standard ) - second part // empty Template
 * \details        Second User part, is called all 5ms periodically, and 1ms later than first part
*******************************************************************************************/
void LIFunc_WEAK AppCSPS_USER_SEQ_STANDARD_2(void)
{
    // Fill in code...
    mNOP();
}
```

4.4.4.4 Error function

Is called exactly once when drive enters "ERROR" state. Can be used for error handling, maybe save some position, etc.

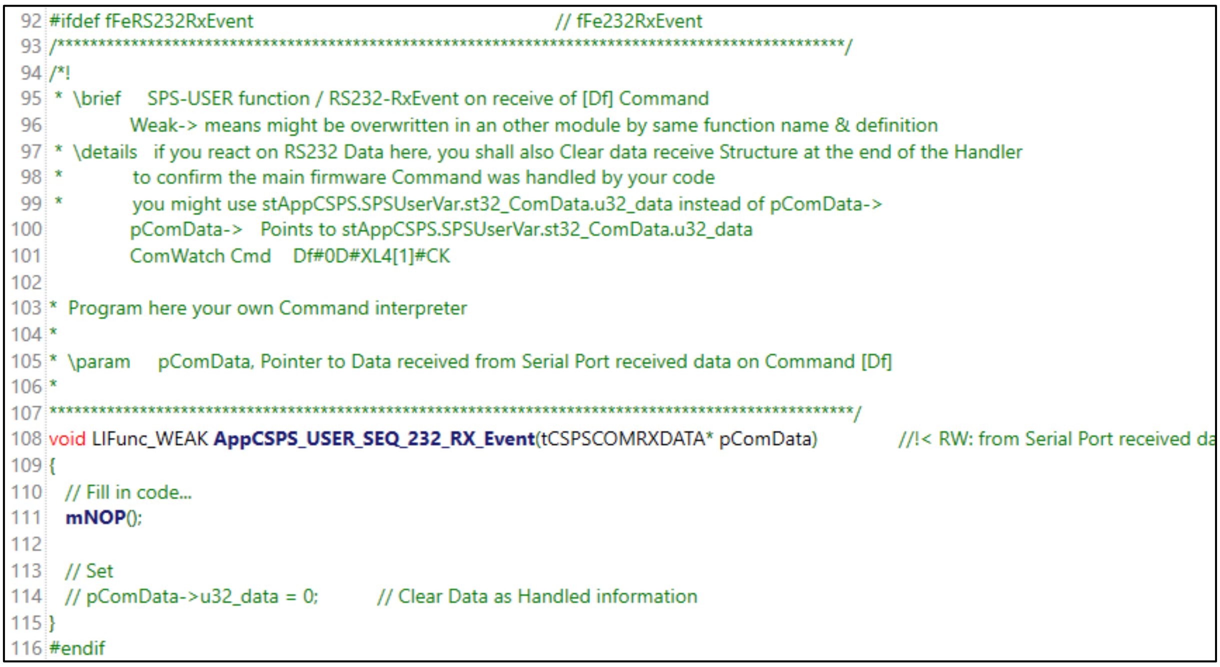


```
/******************************************************************************************/
/*!
 * \brief    SPS-USER function / ErrorHandler
             Weak-> means might be overwritten in an other module by same function name & definition
 * \details   is called once, when core enters <ERROR> state
             depending on Error, it is not allowed to start here a Moving CMD !

* Program here, things to show error, e.g. switch_ON Error-Output
*******************************************************************************************/
void LIFunc_WEAK AppCSPS_USER_SEQ_ERROR(void)
{
    // Fill in code...
    mNOP();
}
```

4.4.4.5 RS232 Rx Event function

RS232 Rx Event. Is called on reception of "Df" command. Available on most serial drives.
Can be used for example for own command interpreter.



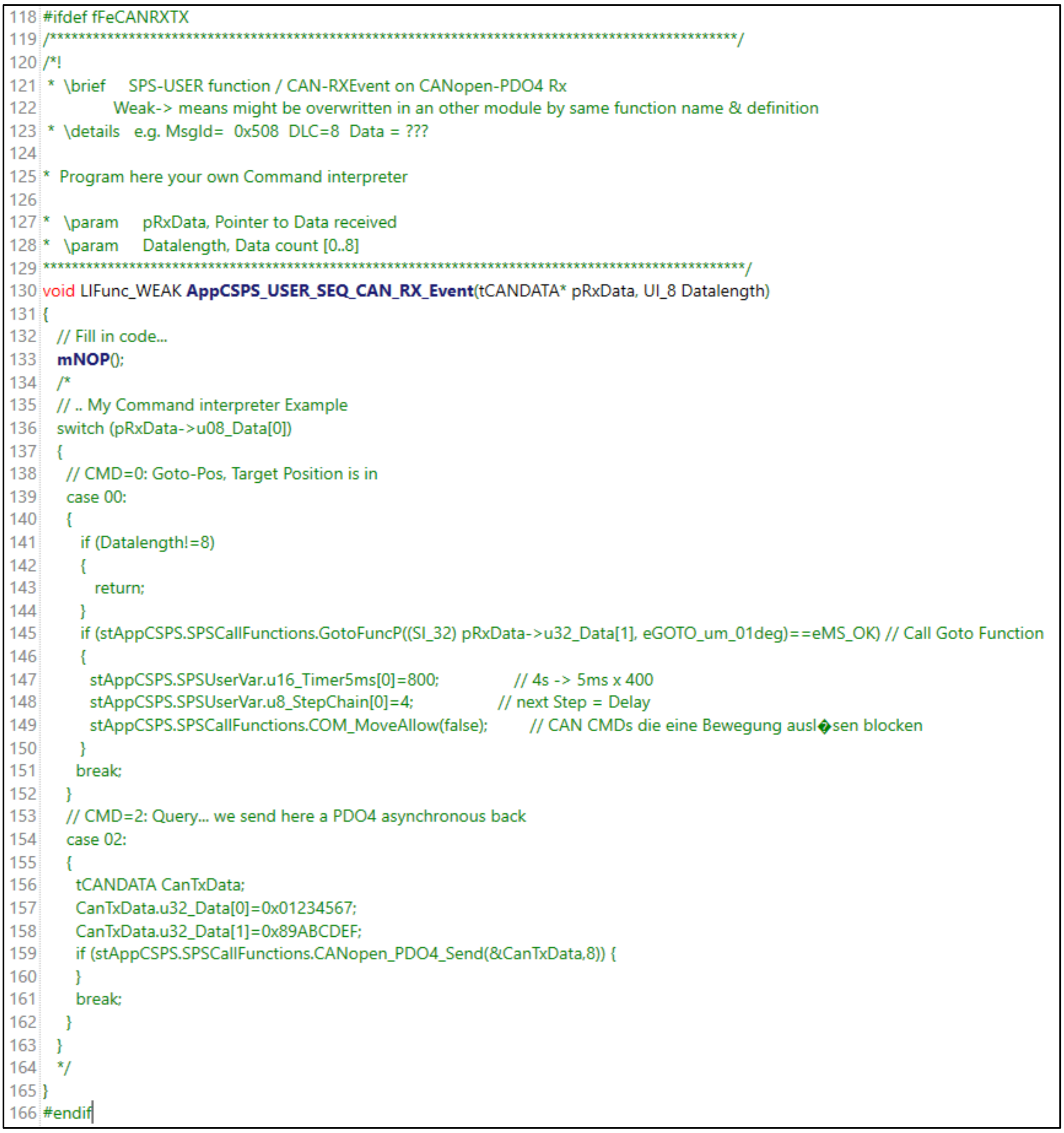
```
#ifdef fFeRS232RxEvent                                          // fFe232RxEvent
/******************************************************************************************/
/*!
 * \brief     SPS-USER function / RS232-RxEvent on receive of [Df] Command
             Weak-> means might be overwritten in an other module by same function name & definition
 * \details   if you react on RS232 Data here, you shall also Clear data receive Structure at the end of the Handler
 *           to confirm the main firmware Command was handled by your code
 *           you might use stAppCSPS.SPSUserVar.st32_ComData.u32_data instead of pComData->
             pComData->    Points to stAppCSPS.SPSUserVar.st32_ComData.u32_data
             ComWatch Cmd     Df#0D#XL4[1]#CK

*  Program here your own Command interpreter
*
*  \param     pComData, Pointer to Data received from Serial Port received data on Command [Df]
*
*******************************************************************************************/
void LIFunc_WEAK AppCSPS_USER_SEQ_232_RX_Event(tCSPSCOMRXDATA* pComData)          //!< RW: from Serial Port received da
{
  // Fill in code...
  mNOP();

  // Set
  // pComData->u32_data = 0;          // Clear Data as Handled information
}
#endif
```

4.4.4.6 CAN Rx Event function

The CAN Rx Event function. Is called on receive of CANopen-PDO4 Rx.
Can be used for example for own command interpreter.



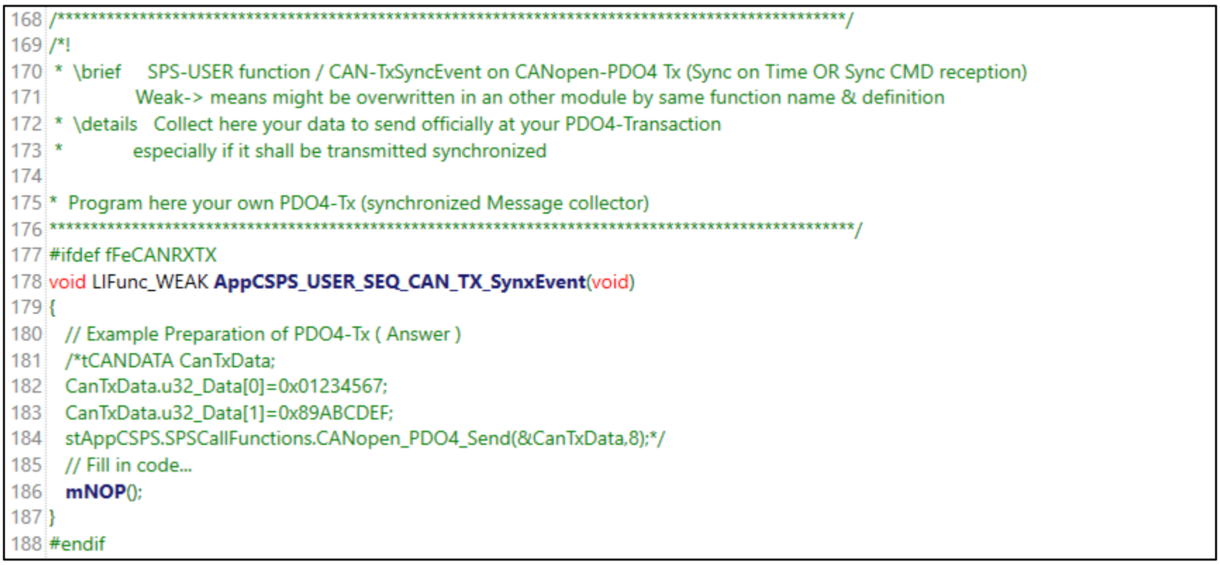
```
#ifdef fFeCANRXTX
/***************************************************************************************************/
/*!
 * \brief    SPS-USER function / CAN-RXEvent on CANopen-PDO4 Rx
             Weak-> means might be overwritten in an other module by same function name & definition
 * \details   e.g. MsgId=  0x508  DLC=8  Data = ???

*  Program here your own Command interpreter

*   \param     pRxData, Pointer to Data received
*   \param     Datalength, Data count [0..8]
****************************************************************************************************/
void LIFunc_WEAK AppCSPS_USER_SEQ_CAN_RX_Event(tCANDATA* pRxData, UI_8 Datalength)
{
  // Fill in code...
  mNOP();
  /*
  // .. My Command interpreter Example
  switch (pRxData->u08_Data[0])
  {
    // CMD=0: Goto-Pos, Target Position is in
    case 00:
    {
      if (Datalength!=8)
      {
        return;
      }
      if (stAppCSPS.SPSCallFunctions.GotoFuncP((SI_32) pRxData->u32_Data[1], eGOTO_um_01deg)==eMS_OK) // Call Goto Function
      {
        stAppCSPS.SPSUserVar.u16_Timer5ms[0]=800;             // 4s -> 5ms x 400
        stAppCSPS.SPSUserVar.u8_StepChain[0]=4;              // next Step = Delay
        stAppCSPS.SPSCallFunctions.COM_MoveAllow(false);        // CAN CMDs die eine Bewegung ausl�sen blocken
      }
      break;
    }
    // CMD=2: Query... we send here a PDO4 asynchronous back
    case 02:
    {
      tCANDATA CanTxData;
      CanTxData.u32_Data[0]=0x01234567;
      CanTxData.u32_Data[1]=0x89ABCDEF;
      if (stAppCSPS.SPSCallFunctions.CANopen_PDO4_Send(&CanTxData,8)) {
      }
      break;
    }
  }
  */
}
#endif
```

4.4.4.7 CAN Tx function

Is called when a PDO4-Tx Event occurs (Sync onTime OR SyncCMD).
Collect your data to send officially at your PDO4-Transaction, especially if it should be transmitted synchronized.

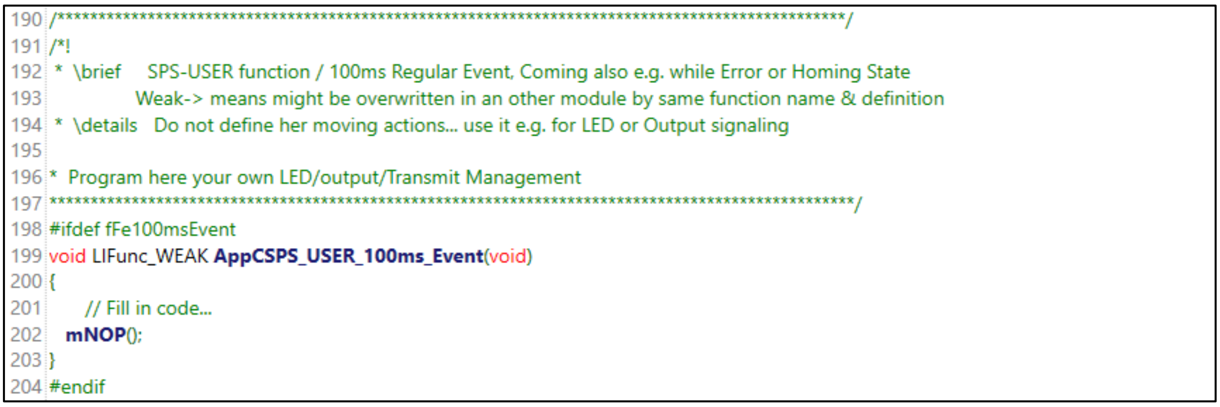


```
/**********************************************************************************************/
/*!
 * \brief    SPS-USER function / CAN-TxSyncEvent on CANopen-PDO4 Tx (Sync on Time OR Sync CMD reception)
             Weak-> means might be overwritten in an other module by same function name & definition
 * \details  Collect here your data to send officially at your PDO4-Transaction
 *           especially if it shall be transmitted synchronized

*  Program here your own PDO4-Tx (synchronized Message collector)
***********************************************************************************************/
#ifdef fFeCANRXTX
void LIFunc_WEAK AppCSPS_USER_SEQ_CAN_TX_SynxEvent(void)
{
  // Example Preparation of PDO4-Tx ( Answer )
  /*tCANDATA CanTxData;
  CanTxData.u32_Data[0]=0x01234567;
  CanTxData.u32_Data[1]=0x89ABCDEF;
  stAppCSPS.SPSCallFunctions.CANopen_PDO4_Send(&CanTxData,8);*/
  // Fill in code...
  mNOP();
}
#endif
```

4.4.4.8 100ms Event function

Called every 100ms, even while in error or homing state.



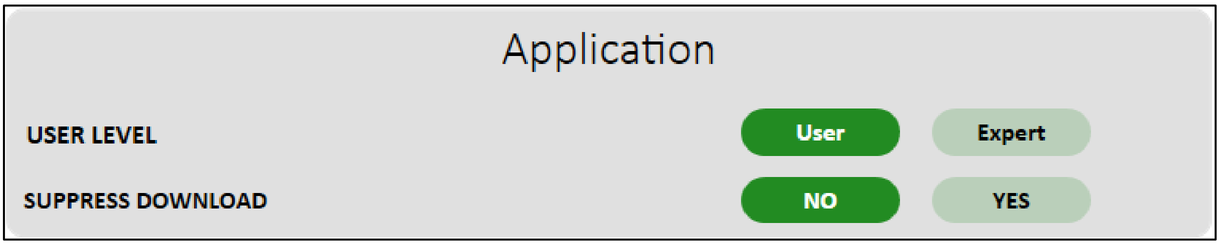
```
/**********************************************************************************************/
/*!
 * \brief    SPS-USER function / 100ms Regular Event, Coming also e.g. while Error or Homing State
             Weak-> means might be overwritten in an other module by same function name & definition
 * \details  Do not define her moving actions... use it e.g. for LED or Output signaling

*  Program here your own LED/output/Transmit Management
***********************************************************************************************/
#ifdef fFe100msEvent
void LIFunc_WEAK AppCSPS_USER_100ms_Event(void)
{
    // Fill in code...
  mNOP();
}
#endif
```

4.5 Settings

All the application settings that can be changed and adapted.

4.5.1 Application



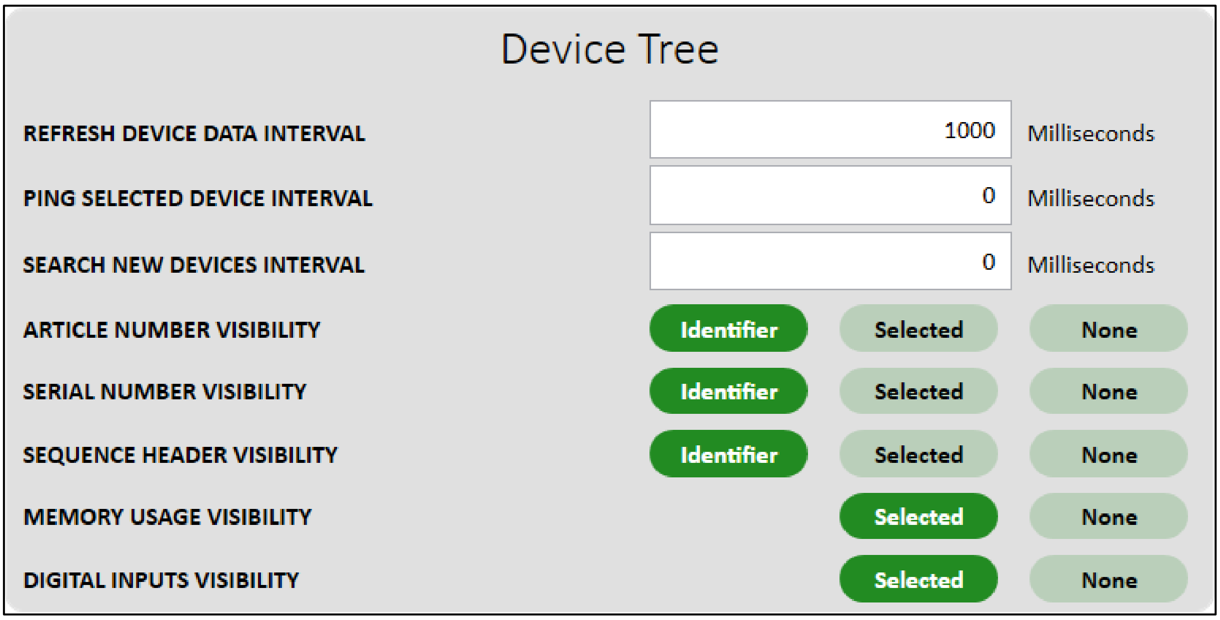
Application

USER LEVEL [✓ User] Expert

SUPPRESS DOWNLOAD [✓ NO] YES

User Level	If in "Expert" Mode, more information in INFORMATION and written commands can be executed from *CONTROL*
Suppress download	Suppress refreshing of drive data when starting app. Local data will be used if available.

4.5.2 Device Tree



Device Tree

REFRESH DEVICE DATA INTERVAL 1000 Milliseconds

PING SELECTED DEVICE INTERVAL 0 Milliseconds

SEARCH NEW DEVICES INTERVAL 0 Milliseconds

ARTICLE NUMBER VISIBILITY [✓ Identifier] Selected None

SERIAL NUMBER VISIBILITY [✓ Identifier] Selected None

SEQUENCE HEADER VISIBILITY [✓ Identifier] Selected None

MEMORY USAGE VISIBILITY [✓ Selected] None

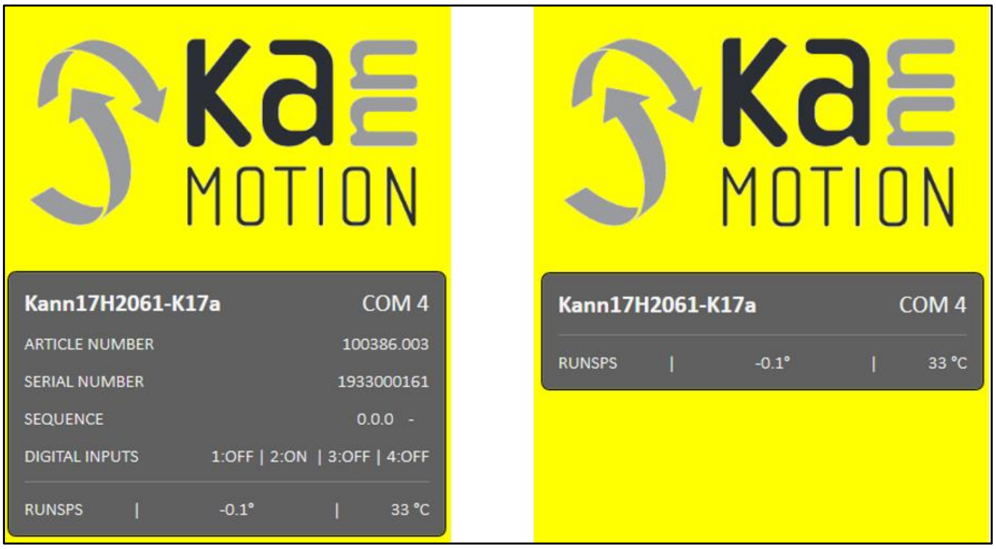
DIGITAL INPUTS VISIBILITY [✓ Selected] None

Refresh device data interval	Interval for updating device information (e.g. status, position, temperature, DINs, Ain, …)
Ping selected device interval	Interval for checking if device is still connected (0 means deactivated, min is 5s)
Search new devices interval	Interval for searching for new devices (0 means deactivated, min is 5s)
Article number visibility	Defines if/when article number is shown

Serial number visibility	Defines if/when serial number is shown
Sequence Header visibility	Defines if/when sequence version and description is shown
Memory usage visibility	Defines if/when memory usage of sequence is shown
Digital inputs visibility	Defines if/when status of digital inputs are shown

Identifier	Will always be displayed
Selected	Will only be displayed if device is selected
None	Will not be displayed

For example, on the left is *Identifier*/*Selected*, on the right is *None*.



Kann17H2061-K17a COM 4

ARTICLE NUMBER 100386.003

SERIAL NUMBER 1933000161

SEQUENCE 0.0.0 -

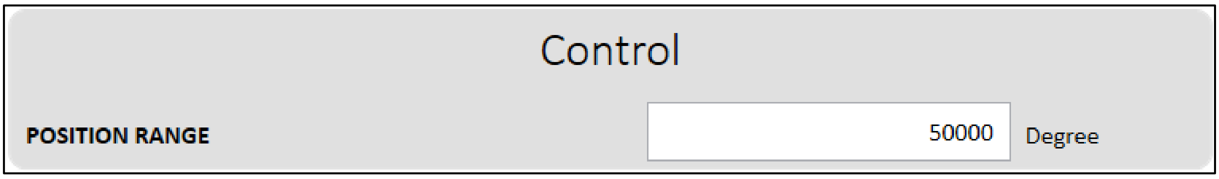
DIGITAL INPUTS 1:OFF | 2:ON | 3:OFF | 4:OFF

RUNSPS | -0.1° | 33 °C

Kann17H2061-K17a COM 4

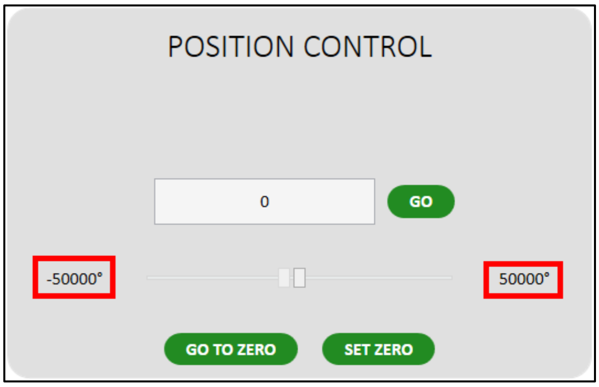
RUNSPS | -0.1° | 33 °C

4.5.3 Control

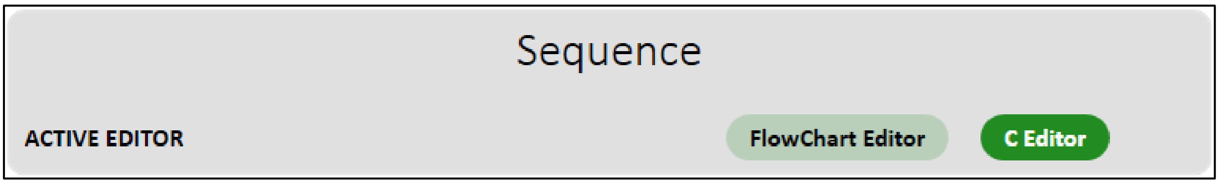


Control

POSITION RANGE 50000 Degree

Position range	Displayed position range for position control.  POSITION CONTROL 0 GO -50000° 50000° GO TO ZERO SET ZERO

4.5.4 Sequence

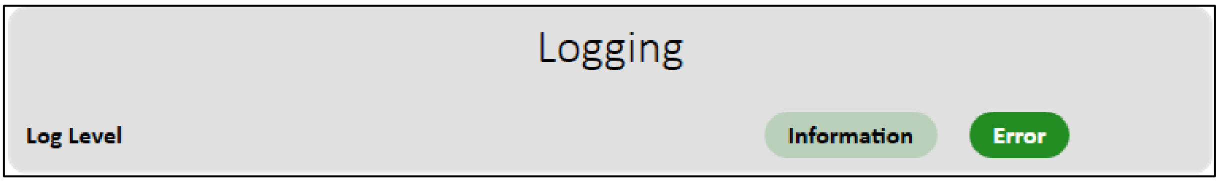


Sequence

ACTIVE EDITOR FlowChart Editor C Editor

Active editor	Active editor in *SEQUENCE*

4.5.5 Logging



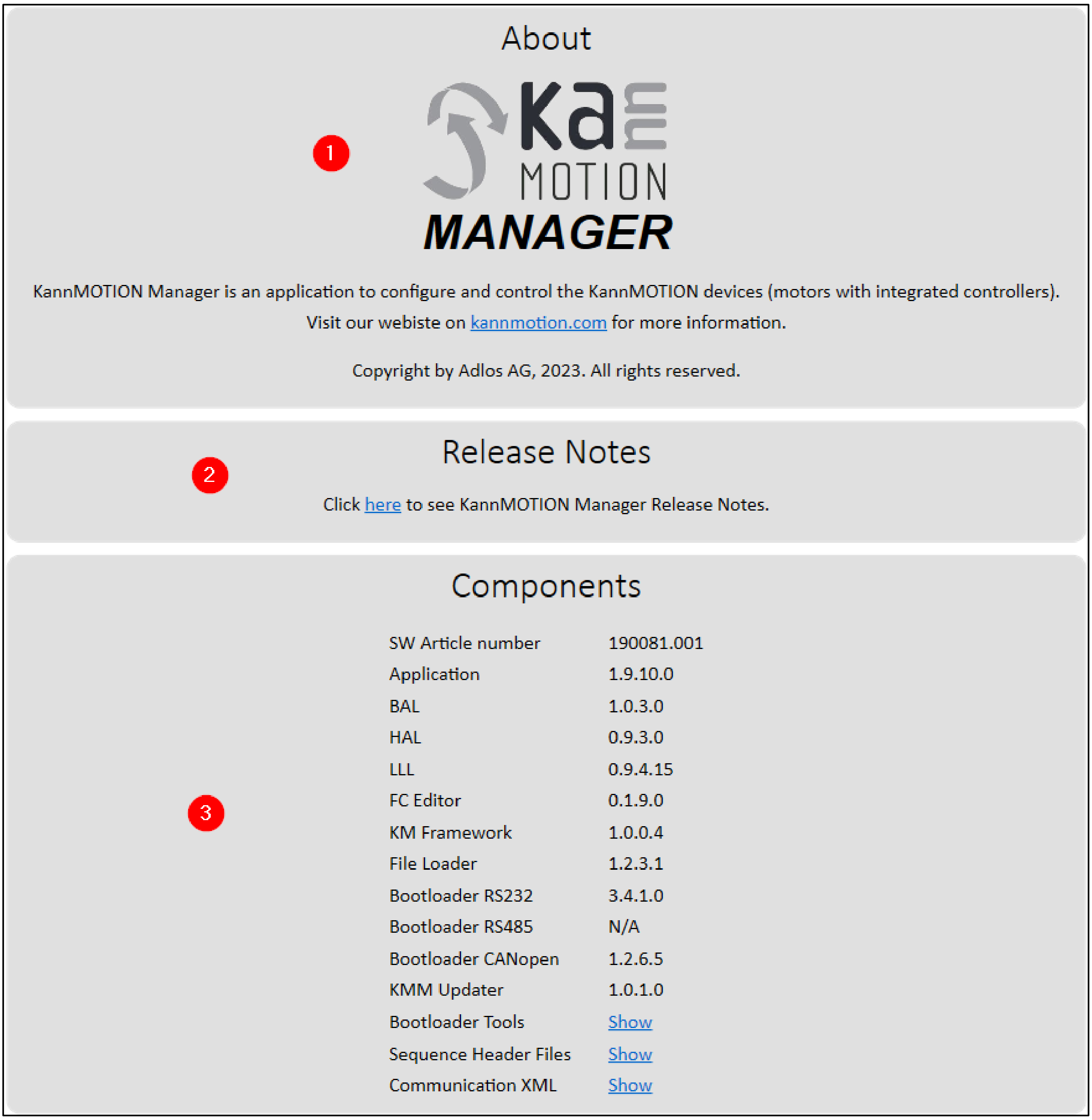
Logging

Log Level Information Error

Log level	Log level for application log, to be found in: *C:\Users\<user>\AppData\Local\Adlos\KannMOTION Manager\Logfiles*

4.6 About

Here, all the information about the application can be found.



About

KannMOTION Manager is an application to configure and control the KannMOTION devices (motors with integrated controllers). Visit our webiste on kannmotion.com for more information.

Copyright by Adlos AG, 2023. All rights reserved.

Release Notes

Click here to see KannMOTION Manager Release Notes.

Components

SW Article number	190081.001
Application	1.9.10.0
BAL	1.0.3.0
HAL	0.9.3.0
LLL	0.9.4.15
FC Editor	0.1.9.0
KM Framework	1.0.0.4
File Loader	1.2.3.1
Bootloader RS232	3.4.1.0
Bootloader RS485	N/A
Bootloader CANopen	1.2.6.5
KMM Updater	1.0.1.0
Bootloader Tools	Show
Sequence Header Files	Show
Communication XML	Show

1	General information about the application
2	Link to release notes
3	Version numbers of libraries and further needed applications

4.7 Help

Clicking on HELP opens the CHM Help File, which provides a comprehensive explanation of all the features available in the KannMOTION Manager.

Proper use

Do not connect or disconnect motor during operation!

Motor cable and motor inductivity might lead to voltage spikes when the motor is disconnected / connected while energized. These voltage spikes might exceed voltage limits of the driver MOSFETs and might permanently damage them. Therefore, always disconnect power supply before connecting / disconnecting the motor

Keep the power supply voltage below the upper limit!

Otherwise, the driver electronics will seriously be damaged! Especially, when the selected operating voltage is near the upper limit a regulated power supply is highly recommended.

Check your mechanical system, is it able to drive the motor, avoid motor being used as generator

Every motor could be operated as a voltage generator, so take care about generated voltage, this might damage your electronics by overvoltage. Add some voltage limiter units to keep supply voltage in range.

Back-EMF

When a motor rotates in the reverse direction, stops or slows down abruptly, a current flow back to the motor's power supply due to the effect of back-EMF. If the current sink capability of the power supply is small, the device's motor power supply and output pins might be exposed to conditions beyond absolute maximum ratings. To avoid this problem, take the effect of back-EMF into consideration in system design

Contact information

Adlos AG
Föhrenweg 14
FL-9496 Balzers

Thomas Vogt
thomas.vogt@adlos.com
Tel: +423 263 63 63

Countries: CH, A, LI, SK, IT

www.adlos.com

KOCO MOTION GmbH
Niedereschacher Straße 54
D-78083 Dauchingen

Olaf Kämmerling
o.kaemmerling@kocomotion.de
Tel: +49 7720/995858-0

Countries: DE, BE, NL, LU

www.kocomotion.de