



angewandte Microelectronic

Haller + Erne GmbH

MSTKN-HM-UM

Bedienungsanleitung MSTKN Stecknussköcher

HEI-24-0314

Revision R21

Revisionen

Revision	History	Author	Date
R20	Übernommen aus alter Doku R19, ergänzt um EtherCAT	he	2024-03-14

Über dieses Dokument

Dieses Dokument enthält Informationen zu dem MSTKN-Stecknussköchersystem in Form einer Bedienungsanleitung.

Inhalt

1	Einleitung	5
1.1	Systemübersicht	5
1.2	Bestellnummernschlüssel	6
1.2.1	Bauform-Matrix	6
1.3	Technische Daten	7
1.4	Schnittstellen	7
1.4.1	DB25M 24V I/O	7
1.4.2	M12 Profibus	8
1.4.3	M12 Ethernet	8
1.4.4	M12 Stromversorgung	8
1.4.5	M12 Interne Busverbindung	8
1.4.6	Konfigurationsschnittstelle	9
2	Funktionsbeschreibung	10
2.1	Allgemeines	10
2.1.1	EA24	10
2.1.2	Profibus	10
2.1.3	Ethernet	10
2.2	Kalibrierung	11
2.2.1	Einzelne Stecknussköcher	11
2.2.2	Mehrere Stecknussköcher (Erweiterungsbus)	12
2.2.3	Unbenutzte Steckplätze	13
2.3	Betriebsarten	13
2.4	Aufbau der E/A-Signale EA24	14
2.4.1	Übersicht	14
2.4.2	Geräteoptionen	14
2.4.3	Betriebsarten	16
2.5	Aufbau der E/A-Signale Profibus/ProfiNet	20
2.5.1	Intelligenter Modus	20
2.5.2	Benutzermodus (user mode)	21
2.5.3	Beispielkonfiguration Siemens SPS	22
2.6	Aufbau der E/A-Signale Ethernet/WLan	24
2.6.1	Modbus-Kommunikation	24
2.6.2	Intelligenter Modus	26
2.6.3	Benutzermodus	27
2.6.4	Python Beispielcode	28
2.7	Aufbau der E/A-Signale EtherCAT	29
2.7.1	Übersicht	29
2.7.2	Eingangsdaten (MSTKN → SPS)	29
2.7.3	Ausgangsdaten (MSTKN → SPS)	30
2.7.4	Inbetriebnahme mit CtrlX	30
3	Anzeigen	38
3.1	Sensor-Anzeigemuster	38
3.2	Hauptmodul-Anzeigen	39
3.2.1	Ok/NOK-LEDs	39
4	Diagnose und Konfiguration	40
4.1	Übersicht	40
4.2	Software-Installation	40
4.3	Werkzeugerkennung parametrieren	40
4.3.1	Verbindung herstellen	41
4.3.2	Übersicht Werkzeugparameter	42
4.3.3	Parametrierung Werkzeugerkennung	43

4.3.4	Prüfen der Werkzeugerkennung	44
4.4	Gruppierung parametrieren	45
4.4.1	Verbindung mit Stecknussköcher herstellen	45
4.4.2	Gruppierung parametrieren	46
4.4.3	Gruppenparametrierung testen	47
4.5	Firmwareupdate durchführen	47
4.5.1	Verbindung mit dem Stecknussköchersystem herstellen	47
4.5.2	Stecknussköcherart wählen	48
4.5.3	Firmwarepaket auswählen und Update durchführen	49
5	Mechanische Daten	51
5.1	Aufnahmeblöcke und Stopfen	51
5.1.1	Maße für Stopfen S14 und S25	51
5.1.2	Maße für T18 Aufnahmeblock	51
5.1.3	Maße für Stopfen Qxx (70mm)	52
5.2	Gehäuseabmessungen	52
5.3	Nummerierung der Steckplätze	54
6	Konformitätserklärung	56

1 Einleitung

1.1 Systemübersicht

Das MSTKN-Stecknussköchersystem bietet eine flexible Lösung für die Überwachung von Werkzeugentnahmen und für die Werkerführung bei Operationen, die ein manuelles Wechseln von Werkzeugen erfordern. Das MSTKN-Stecknussköchersystem bietet insbesondere:



Verschiedene Gehäusevarianten für die Nutzung von Werkzeugen mit Durchmesser im Bereich von 5...65mm. Varianten mit variablen (Prismenschieber) oder festen Werkzeugaufnahmen sind verfügbar.

- Verschiedene Feld-Schnittstellen: 24V-E/A und Feldbussysteme (Profibus, ProfiNet, EtherCAT, Ethernet/IP, IndraLogic-CAA/Modbus-UDP/TCP)
- Flexible Erweiterbarkeit auf bis zu 60 Werkzeuge über internes Bussystem

Alle Geräte des MSTKN-Systems bestehen grundsätzlich aus einer Kombination von zwei verschiedenen Grundelementen, die in verschiedenen Varianten miteinander kombiniert und in unterschiedliche Gehäuse integriert werden. Die Grundelemente des Systems sind:

- **Gatewayelement:** Ein Gatewayelement stellt die Verbindung zwischen einer Feld-Schnittstelle (bzw. Feldbus-Schnittstelle) und dem internen Bussystem her. Gatewayelemente sind für 24V-E/A und für verschiedene Feldbus-Systeme (Profibus-DP, ProfiNet, Ethernet/IP, IndraLogic-CAA, ...) verfügbar und stellen die Verbindung zwischen Stecknussköchersystem und der übergeordneten Prozesssteuerung (bzw. dem Schraubsystem) her.
- **Sensorelement:** Ein Sensorelement enthält Sensoren, welche die Anwesenheit von Werkzeugen in den zugehörigen Werkzeugablagen detektieren und über das interne Bussystem melden. Sensorelemente sind in verschiedenen Bauformen (Anzahl der Sensoren, große/kleine Sensoren für verschiedene Werkzeuggrößen) und Technologien (für die Anwesenheits- bzw. Belegter Kennung oder auch Werkzeugidentifikation) verfügbar.

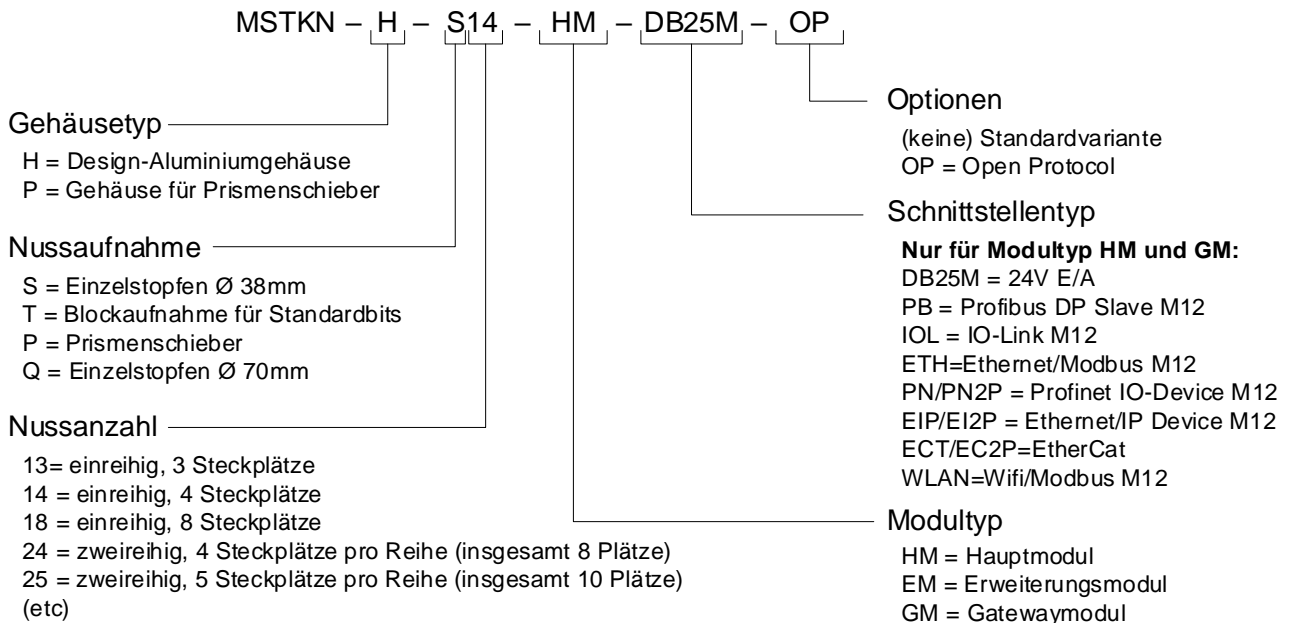
Die Geräte des MSTKN-Systems werden daraus wie folgt zusammengesetzt:

- **Hauptmodul (HM):** Ein Hauptmodul kombiniert ein Gatewayelement mit einem oder mehreren Sensorelementen und ist in eines der Gehäuse des MSTKN-Systems eingebaut. Ein Hauptmodul ist ein eigenständiger und voll funktionsfähiger Stecknussköcher, der z. B. direkt an eine SPS oder ein Schraubsystem angeschlossen werden kann. Über das interne Bussystem kann ein Hauptmodul um weitere Erweiterungsmodule erweitert werden, um die Gesamtanzahl überwachter Werkzeuge zu erhöhen.
- **Erweiterungsmodul (EM):** Ein Erweiterungsmodul besteht aus einem oder mehreren Sensorelementen, eingebaut in eines der Gehäuse des MSTKN-Systems. Ein Erweiterungsmodul muss immer an das interne Bussystem des MSTKN-Systems angeschlossen werden, entweder an ein Gatewaymodul oder ein Hauptmodul.
- **Gatewaymodul (GM):** Ein Gatewaymodul besteht aus einem Gatewayelement, eingebaut in ein Gehäuse für die Schaltschrankmontage (Tragschienengehäuse). Das Gatewaymodul ermöglicht die Anbindung von einem oder mehreren Erweiterungsmodulen an eine 24V-E/A-Schnittstelle oder an ein Feldbussystem.

Alle Geräte des MSTKN-Systems können miteinander kombiniert werden, da alle über ein gemeinsames internes Bussystem miteinander kommunizieren. In einem Stecknussköchersystem muss dabei immer ein Gatewayelement und mindestens ein Sensorelement enthalten sein, unabhängig davon, welche Bauformen, Gehäusetyper oder Sensorvarianten eingesetzt werden.

1.2 Bestellnummernschlüssel

Die Bestellnummern der Stecknussköcher der MSTKN-Familie sind wie folgt aufgeschlüsselt:



Für die hier beschriebenen Stecknussköcher sind Haupt- und Erweiterungsmodule typischerweise in vier verschiedenen Gehäusevarianten verfügbar. Für alle Hauptmodule ist neben der Standardvariante die Option „OP“ für die direkte Ansteuerung der Stecknussköcher über die im Schraubsystem 350 enthaltene IM24-Schnittstelle für die integrierte „Open Protocol“-Implementierung und eine „XML“-Variante mit veränderten Signalen und anderer Farbcodierung der LEDs, welche für internationale Audi-Produktionsstandards angepasst ist, verfügbar.

1.2.1 Bauform-Matrix

Werkzeug max. Ø 38mm	Werkzeug max. Ø 70mm	Werkzeug max. Ø 18mm	Prismenschieber max. Ø 48mm
H-S14 	H-Q12 	H-T18 	P-P14
H-S24 	H-Q13 		
H-S25 	H-Q14 		

1.3 Technische Daten

Die technischen Daten der einzelnen MSTKN-Varianten finden Sie in den jeweiligen Datenblättern.

1.4 Schnittstellen

1.4.1 DB25M 24V I/O

Diese Schnittstelle ist nur auf den Hauptmodulen und den Gatewaymodulen verfügbar. Die Schnittstelle stellt standardkonforme 24V E/A Signale (13 Eingänge, 10 Ausgänge) zur Verfügung. Die Pinbelegung ist 1:1 kompatibel mit der Pinbelegung der IM24V-Schnittstellenkarte des Schraubsystems 350, so dass ein kostengünstiges DB25-Verlängerungskabel verwendet werden kann.

Für eine Beschreibung der Funktion der einzelnen Signale, siehe Funktionsbeschreibung unten.

DB25 male	MSTKN Eingangssignale					MSTKN Ausgangssignale				
	Pin	Signal	Beschreibung	IM24V		Pin	Signal	Beschreibung	IM24V	
	19	I0	Input bit 0	A1	B0.0	14	Q0	Output bit 0	E1	B0.0
	7	I1	Input bit 1	A2	B0.1	2	Q1	Output bit 1	E2	B0.1
	20	I2	Input bit 2	A3	B0.2	15	Q2	Output bit 2	E3	B0.2
	8	I3	Input bit 3	A4	B0.3	3	Q3	Output bit 3	E4	B0.3
	21	I4	Input bit 4	A5	B0.4	16	Q4	Output bit 4	E5	B0.4
	23	I5	Input bit 5	A9 (!)	B5.0	4	Q5	Output bit 5	E6	B0.5
	11	I6	Input bit 6	A10 (!)	B5.1	18	Q6	Output bit 6	E9 (!)	B1.0
	12	I7	Input bit 7	A12 (!)	B5.3	6	Q7	Output bit 7	E10 (!)	B1.1
	9	I8	Input bit 8	A6	B0.5	17	Q8	Output bit 8	E7	B0.6
	22	I9	Input bit 9	A7	B0.6	5	Q9	Output bit 9	E8	B0.7
	10	I10	Input bit 10	A8	B0.7					
Weitere Signale and Stromversorgung										
1	24V	24V Versorgung								
13	0V	Bezugspotential 0V								
24	LEDR	LED rot (24V=on)	A11(!)	B5.2						
25	LEDG	LED grün (24V=on)	A13	B5.4						

Wichtig: Die Zuordnung der logischen Signale zwischen IM24V und MSTKN ist aus Gründen der Kompatibilität zur ersten Hardwareversion der Stecknussköcher nicht 1:1 (bitte Spalte IM24V in der Tabelle oben beachten!).



Typische Konfiguration in BS350 (im Modus Direktbetrieb mit Kabelbrucherkennung):

The screenshot shows the configuration interface for the BS350 system. It is divided into 'Eingänge' (Inputs) and 'Ausgänge' (Outputs) sections. In the 'Eingänge' section, 'Prog6' is marked as 'Error flag (unused)'. In the 'Ausgänge' section, 'Ack5' is set to 'Mode 0-2', and 'Ack7' is set to 'LED red' and 'LED green'. A table on the right shows the hardware configuration for the modules.

Modul	E	A
IM24V	32	32
OP	48	24
ES	4	0
HVE	8	0
HVO	8	0
IL	32	32
XME	20	24

1.4.2 M12 Profibus

Das Profibus Hauptmodul verfügt über 2 M12 Feldbusanschlüsse (Bus IN, Bus OUT). Die Pinbelegung der Spoligen Buchse (Ausgang) und des Steckers (Eingang) entsprechen dem Profibus Standard.

M12 B-kodiert	Pin	Signal	Beschreibung/Funktion
<p>M12, B-kodiert</p>	1	+5V	Stromversorgung (nur an Bus OUT, max. 100mA)
	2	A	TX-/RX- Bus Signal
	3	0V	Stromversorgung Bezugspotenzial (0V)
	4	B	TX+/RX+ Bus Signal
	5	-	Not connected

1.4.3 M12 Ethernet

Der Anschluss der Ethernet-basierten Hauptmodule erfolgt über einen 4 poligen, D-kodiert M12 Steckverbinder mit Standardbelegung (für ProfiNet, EtherCAT und EtherNet/IP sind zwei Ports genutzt).

M12 D-kodiert, Buchse	Pin	Signal	Beschreibung/Funktion
<p>M12, D-kodiert</p>	1	Tx+	Sendeleitung +
	2	Rx+	Empfangsleitung +
	3	Tx-	Sendeleitung -
	4	Rx-	Empfangsleitung -

1.4.4 M12 Stromversorgung

Die Stromversorgung +24V erfolgt über einen 4-poligen, A-kodierten M12 Steckerbinder.

M12 A-kodiert, Stecker	Pin	Signal	Beschreibung/Funktion
<p>M12, A-kodiert</p>	1	+24V	Stromversorgung (Eingang)
	2		nicht genutzt
	3	0V	Stromversorgung Bezugspotenzial (0V)
	4		nicht genutzt

1.4.5 M12 Interne Busverbindung

Abhängig vom Gerätetyp sind ein oder zwei M12 Steckverbinder (A-kodiert, 5 Pins) für das interne Bussystem vorhanden. Für Haupt- und Gatewaymodule ist nur ein „Bus-out“ Steckverbinder vorhanden. Erweiterungsmodule besitzen zwei Steckverbinder („Bus-in“ und „Bus-out“). Die „Bus-out“-Schnittstelle (M12-female) stellt zusätzlich die Versorgungsspannung für Erweiterungsmodule bereit.

M12 A-kodiert	Pin	Signal	Beschreibung/Funktion
<p>male – bus in female – bus out</p>	1	+5V	Stromversorgung (bus in: input, bus out: output)
	2	A	TX-/RX- bus signal
	3	0V	Stromversorgung Bezugspotenzial (0V)
	4	B	TX+/RX+ bus signal
	5	-	Not connected

Die interne Busverbindung kann auch zur Diagnose und Parametrierung des Stecknussköcher-Systems verwendet werden. Weitere Informationen: siehe Kapitel 4.

1.4.6 Konfigurationsschnittstelle

Die Schnittstelle für die interne Busverbindung (siehe vorheriges Kapitel 1.4.5) kann auch zur Konfiguration und Diagnose des Stecknussköchersystems benutzt werden. Hierzu werden das MSTKN Programmierkabel und die zugehörige PC-Software benötigt.

Über die Konfigurationsschnittstelle können erweiterte Funktionen, wie z. B. die Nussgruppierung oder Nusserkennung aktiviert werden.

Weitere Informationen finden sich im Kapitel 4.

2 Funktionsbeschreibung

2.1 Allgemeines

2.1.1 EA24

Die Module der 24V E/A-Stecknussköcherfamilie bieten die folgenden Funktionen:

- Standardkonforme 24V E/A Schnittstelle
 - o 13 Eingänge (11 Steuersignale, 2 LEDs)
 - o 10 Ausgänge
- Drei LEDs pro Platz für Statusanzeige und Werkerführung
- DB25 E/A-Schnittstellensteckverbinder Pin-kompatibel zu System 350 IM24V
- Verschiedene Betriebsarten für flexiblen Einsatz (mit/ohne Werkerführung)
- Erhöhte Prozesssicherheit durch spezielle Betriebsarten („Direktbetrieb“) und Kabelbruchtest
- Zwei verschiedene Varianten für Kodierung der Nussentnahme: Ein E/A-Signal pro Platz für maximale Flexibilität (Option –OP) oder binär kodierte Platznummer (Standard) für Systeme mit bis zu 60 Nüsse

2.1.2 Profibus

Die Module der Profibus-Stecknussköcherfamilie bieten die folgenden Funktionen:

- Profibus-Slave
 - o Adresse 1-99 einstellbar über Drehschalter
 - o Dynamische Adressierung möglich
- Drei LEDs pro Platz für Statusanzeige und Werkerführung
- Standard M12 Profibus-Anschluss (Bus IN und Bus OUT)
- Verschiedene Betriebsarten für flexiblen Einsatz (mit/ohne Werkerführung)
- Erhöhte Prozesssicherheit durch spezielle Betriebsarten („Direktbetrieb“) und Kabelbruchtest
- Konfiguration über Module aus dem GSD File
 - o MSTKN intelligent mode
 - o MSTKN user mode

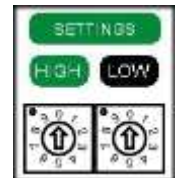
2.1.3 Ethernet

Die Module der Ethernet-Stecknussköcherfamilie (Profinet, OpenModbus, Ethernet/IP) bieten die folgenden Funktionen:

- IP-Einstellung
- feste IP-Adresse einstellbar über Drehschalter im Bereich 192.168.0.3 – 192.168.0.98 (dann Netmask 255.255.255.0 und Gateway 192.168.0.1)
 - o einstellbar über Softwaretool (HICP)
 - o dynamisch per DHCP
 - o dynamisch über feldbusspezifischen Mechanismus
- Drei LEDs pro Platz für Statusanzeige und Werkerführung

- Standard M12 Profibus-Anschluss (D-kodiert)
- Verschiedene Betriebsarten für flexiblen Einsatz (mit/ohne Werkerführung)
- Erhöhte Prozesssicherheit durch spezielle Betriebsarten („Direktbetrieb“) und Kabelbruchtest
- Konfiguration über „MODE“-Bit im Datentelegramm
 - o MSTKN intelligent mode
 - o MSTKN user mode
- 66 Byte I/O Datenaustausch

Betriebsart und IP-Adresse kann über Drehschalter auf der Rückseite des MSTKN-Hauptmoduls wie folgt eingestellt werden (siehe nebenstehendes Bild für den Aufdruck auf der Rückseite, bei neuen Geräten hinter Blindverschraubung bei M12-Steckern):



Schalter	Modus	Beschreibung
00	default	Standard-Konfiguration. Die zuletzt gespeicherte Konfiguration wird verwendet. Diese Variante wird üblicherweise im Produktivbetrieb verwendet und bietet die höchste Sicherheit, da alle optionalen Konfigurationsdienste inaktiv sind. Die feldbusspezifische Konfiguration bleibt trotzdem aktiv (z. B. Adress- und Namens-Vergabe über DCP bei Profinet).
01	HICP	Die zuletzt gespeicherte Konfiguration wird verwendet. Zusätzlich ist die HICP-Funktion aktiviert. Dies ermöglicht die Remote-Konfiguration der IP-Adresse mittels der AnyBus-IP-Config-Software.
02	DHCP	Die zuletzt gespeicherte Konfiguration wird verwendet. Zusätzlich wird beim Start versucht eine IP-Konfiguration von einem DHCP-Server zu empfangen. Diese wird dann auch ggf. gespeichert.
03-98	Fest	Fest über die Schalterstellung definierte IP-Adresse im Bereich 192.168.0.X (mit X = Schalterstellung zwischen 3 und 98). Netmask und Gatewayadresse werden fest vergeben: Netmask = 255.255.255.0 Gateway = 192.168.0.1
99	Admin	Nicht für Wifi! Zuletzt gespeicherte Konfiguration wird verwendet. Zusätzlich ist der Admin-Mode aktiv, d. h. alle im System verfügbaren Netzwerkdienste werden aktiviert und Änderungen der Systemkonfiguration über das Netzwerk sind erlaubt (z. B. Firmware-Update). Aktive Dienste sind dann (je nach tatsächlichem Ethernet-Feldbustyp): <ul style="list-style-type: none"> - HICP (zur IP-Konfiguration, s.o.) - DHCP (zur IP-Konfiguration, s.o.) - ARP Gleaming (zur IP-Konfiguration) - Web/FTP/Telnet Server (soweit in der Variante vorhanden), dann ggf. auch für Firmware-Updates.

2.2 Kalibrierung

Die Kalibrierung dient zur Maximierung der Sensorempfindlichkeit, zum Einlernen unbenutzter Plätze und zur Erkennung möglicher Erweiterungsmodule.



Wichtig: Es ist unbedingt darauf zu achten, dass alle Werkzeuge entnommen werden, bevor die Kalibrierung durchgeführt wird. Bei einer falsch durchgeführten Kalibrierung des Köchers ist dessen Funktion gestört, bzw. es werden ggf. einzelne Steckplätze deaktiviert!

2.2.1 Einzelne Stecknussköcher

Die Stecknussköcher werden bereits kalibriert ausgeliefert, sodass es normalerweise nicht nötig ist eine Kalibrierung durch den Anwender durchzuführen.

2.2.2 Mehrere Stecknussköcher (Erweiterungsbus)

Beim Einrichten einer Kette von über den Erweiterungsbus verbundener Stecknussköcher muss eine bestimmte Vorgehensweise eingehalten werden, um eine korrekte Konfiguration zu erhalten. Die Kette kann nicht auf einmal konfiguriert werden: Es muss ein Modul nach dem anderen hinzugefügt werden, beginnend mit dem ersten Modul, das die Steckplatznummern 1 ... 4 (oder 1 ... 8 bei einem 8-fach Modul) bekommen soll. Jedes danach hinzugefügte Modul wird die nächsten freien Steckplatznummern zugewiesen bekommen.

Falls ein Hauptmodul (WiFi, Ethernet, ProfiNET, Profibus, EtherCAT, 24V E/A) mit einem oder mehreren Erweiterungsmodulen verkettet werden soll, muss das Hauptmodul zuerst kalibriert werden. Wenn nur Erweiterungsmodule verwendet werden, kann jedes beliebige Gerät zuerst angeschlossen werden.

Bitte folgen Sie den nachfolgenden Schritten, um das System richtig zu konfigurieren:

1. Ersten Stecknussköcher verbinden (Haupt- oder Erweiterungsmodul).
2. Suchen Sie das "Kalibrieren" Signal in der Eingangssignalbeschreibung Ihres Hauptmoduls:
 - Kapitel 2.4 (Detaillierte Beschreibung der Betriebsarten EA24V)
 - Kapitel 2.5 (Aufbau der Ein- und Ausgangssignale Profibus/Profinet)
 - Kapitel 2.6 (Aufbau der Ein- und Ausgangssignale Ethernet/Wlan)
3. Senden Sie das Kalibriersignal zum ersten Stecknussköcher. Seine LEDs sollten gelb blinken.
4. Nehmen die das Kalibriersignal wieder weg. Der Stecknussköcher wird automatisch neu starten.
5. Fügen Sie den nächsten Stecknussköcher hinzu (Erweiterungsmodul).
6. Senden Sie das Kalibriersignal zum ersten Stecknussköcher. Seine LEDs sollten gelb blinken.
7. Nehmen die das Kalibriersignal wieder weg. Der Stecknussköcher wird automatisch neu starten.
8. Fügen Sie den nächsten Stecknussköcher hinzu (nächstes Erweiterungsmodul).
9.
10. Wiederholen Sie den Vorgang, bis die gesamte Kette konfiguriert ist.



Wichtig: Bei ProfiNet-Konfigurationen mit „User mode“ muss zusätzlich auf die Reihenfolge der Parametrierung der SPS geachtet werden, da das Kalibriersignal nur bei „korrekter“ Hardwarekonfiguration gesendet werden kann! Zum Hinzufügen eines Erweiterungsmoduls also z. B.:

1. Erweiterungsmodul anstecken (ohne SPS-Konfiguration zu ändern) und Kalibriersignal senden
2. SPS-Konfiguration ändern (neues Erweiterungsmodul einstecken)

Wird die SPS-Konfiguration geändert, ohne dass das Erweiterungsmodul „eingelernt“ wurde, wird die SPS einen Profinet-Fehler (Modul-Differenz) melden – damit kann dann das Kalibriersignal nicht mehr gesendet werden!



Hinweis: Wenn eine Kette mit Hauptmodul falsch konfiguriert wurde oder das Hauptmodul für einen anderen Einsatz (Einzelbetrieb oder in Reihe), können Sie die alte Konfiguration löschen, indem Sie das "Kalibrierung Zurücksetzen" Signal an das Hauptmodul senden (ohne angeschlossene Erweiterungsmodule). Das "Kalibrierung zurücksetzen Signal finden Sie ebenfalls in der Eingangssignalbeschreibung des Hauptmoduls.

2.2.3 Unbenutzte Steckplätze

Sollte ein Steckplatz ungenutzt bleiben, so gibt es zwei Möglichkeiten, diesen zu deaktivieren.

- 1. Möglichkeit: Bekleben Sie den Stopfen des nicht genutzten Steckplatzes mit dem mitgelieferten Kupferklebeband und verschrauben Sie den präparierten Stopfen mit dem Gehäuse. Die LED des Steckplatzes leuchtet dann dauerhaft gelb und das System meldet den Steckplatz als dauernd belegt. So hat dieser Steckplatz keinerlei Einfluss auf die Funktion des Geräts. Sollte dieser Steckplatz wieder benötigt werden, müssen Sie lediglich das Kupferband entfernen.
- 2. Möglichkeit: Bekleben Sie den nicht genutzten Stopfen mit einem Teil des mitgelieferten Kupferklebebands. Danach muss der Stopfen in den Stecknussköcher gesteckt und verschraubt werden, während alle anderen genutzten Steckplätze frei sein müssen. Ist dies der Fall, kann eine Kalibrierung durchgeführt werden. Die Software erkennt den gesteckten Stopfen mit Klebeband und deaktiviert diesen. Nach Ende der Kalibrierung wird die Leuchtdiodenanzeige des ungenutzten Steckplatzes deaktiviert und der Steckplatz kann nicht mehr verwendet werden. Um den Steckplatz wieder zu aktivieren, müssen der Stopfen mit Klebeband sowie alle anderen Werkzeuge entfernt und erneut eine Kalibrierung durchgeführt werden.

2.3 Betriebsarten

Der Stecknussköcher unterstützt folgende Betriebsarten:

- **Freie Auswahl:** Es steht dem Werker frei, welche Nuss er aus dem Köcher entnimmt. Die Steckplatznummer des entnommenen Werkzeugs wird über die Feldbus Ausgangssignale angezeigt. Entnimmt der Werker mehr als ein Werkzeug, wird über die Schnittstelle ein Fehler ausgegeben. Die Leuchtdioden am Stecknussköcher dienen zur Zustands- bzw. Fehleranzeige und zur Anzeige, ob die Entnahme korrekt ist (Fehleranzeige, falls z. B. mehr als ein Werkzeug entnommen wurde). Um den Köcher in dieser Betriebsart zu betreiben, müssen alle Eingänge des MSTKN „0“ sein.
- **Werkerführung:** In dieser Betriebsart gibt eine externe Steuerung (z.B. Schraubersteuerung oder SPS), über die Feldbus-Schnittstelle vor, welches Werkzeug entnommen werden soll. Der Stecknussköcher zeigt dies durch die Leuchtdioden der einzelnen Steckplätze an. Bei Entnahme eines Werkzeugs meldet der Köcher den Zustand an die externe Steuerung und setzt, bei korrekter Entnahme (Entnahme identisch mit Vorwahl), ein Freigabebit. Der Zustand der Leuchtdioden ändert sich bei korrekter Entnahme von grün blinkend zu einem grünen Dauerlicht. Bei Falschentnahme leuchtet die rote LED auf. In der Standardvariante kann die Steuerung nur ein Werkzeug vorwählen, in der Variante Open Protocol ist es möglich, den Werker eine Auswahl aus mehreren Werkzeugen zu erlauben („Bitmuster“).
- **Direktbetrieb:** Diese Betriebsart arbeitet ähnlich wie die Betriebsart „Freie Auswahl“, ist aber um eine Überwachungsfunktion erweitert, die zur Erkennung von Kurzschlüssen und nicht verbundenen Leitungen dient. Der Stecknussköcher sendet je nach Status sein Ausgangssignal an die Steuerung und erwartet dieses, innerhalb von ca. 150ms, als Eingangssignal zurück. Erst danach erfolgt das Setzen des Freigabebits durch den Stecknussköcher. Erhält der Stecknussköcher nicht den korrekten Wert zurück (Ausgangssignal \neq Eingangssignal), wird das Freigabebit nicht gesetzt und über die Leuchtdioden wird ein definiertes Leuchtmuster ausgegeben.
- **Diagnose- und Konfigurationsbetrieb:** Je nach eingesetztem Gerätetyp werden verschiedene Betriebsarten zu Diagnose- (z.B. Kabelbruch-Test) und Konfigurationszwecken (z.B. Sensorkalibrierung) unterstützt. Bitte entnehmen sie die detaillierte Beschreibung den nachfolgenden Erklärungen.
- **Intelligenter Betrieb:** Diese Betriebsart findet bei den Feldbus Varianten (Profibus, Modbus UDP etc.) ihren Einsatz. Mit dieser Einstellung übernimmt der Stecknussköcher sowohl die Steuerung der LEDs, als auch die Auswertung der Steckplatzzustände in Verbindung mit der getroffenen Vorwahl.

- **Benutzer Betrieb:** Auch diese Betriebsart ist nur bei den Feldbus Varianten von Bedeutung. In dieser Betriebsart übernimmt der Nutzer die vollständige Steuerung und Auswertung der Stecknussköchersignale. D.h. der Nutzer kann beliebige Leuchtdiodenmuster generieren und z.B. Mehrfachentnahmen erlauben.
- **Gruppierung:** In der Betriebsart „Intelligenter Betrieb“ können ab der Firmwareversion V1.6 Gruppen definiert werden. Die Gruppen verhalten sich auf der I/O-Schnittstelle wie eine einzelne Nuss, es müssen jedoch immer alle Nüsse einer Gruppe entnommen werden. Bei Vorwahl einer Nussgruppe blinken alle grünen LEDs der Gruppe bis alle Nüsse der Gruppe entnommen wurden – erst dann wird die Schrauberfreigabe erteilt. Die Gruppenfunktion ist eine erweiterte Funktion, die nur über das Programmierkabel konfiguriert werden kann.



Wichtig: Die Betriebsart „Freie Auswahl“ ist die Grundbetriebsart des Köchers und ist kompatibel zu einfachen Nussboxen. Es wird jedoch nicht empfohlen, den Köcher in dieser Betriebsart zu betreiben, da hier keine Überwachung für die Verkabelung vorhanden ist (Kabelbruch!). Um eine höhere Prozesssicherheit gewährleisten zu können, sollte stattdessen die Betriebsart „Direktbetrieb“ verwendet werden!

2.4 Aufbau der E/A-Signale EA24

2.4.1 Übersicht

Ein- und Ausgangssignale (siehe Kapitel 1.4.1) sind wie folgt aufgebaut:

24V-E/A Eingangssignale: Eingang I0...I10:

I	I	I	I	I	I	I	I	I	I	I		
10	9	8	7	6	5	4	3	2	1	0		
0...255 (0xFF)											Signale zur Vorwahl Steckplatznummer	8 Signale um die Steckplatzvorwahl zu treffen (Betriebsart Werkerführung)
0...7											Steuersignale	3 Signale um die Betriebsart auszuwählen und den Stecknussköcher zu steuern

24V-E/A Ausgangssignale: Ausgang Q0...Q9:

Q	Q	Q	Q	Q	Q	Q	Q	Q	Q			
9	8	7	6	5	4	3	2	1	0			
0...255 (0xFF)											Signale zur Anzeige der entnommenen Werkzeugs	8 Signale für die aktuelle Auswahl (also die Nummer oder den Platz des entnommenen Werkzeugs – für die Programmvorwahl)
0...3											Statussignale	2 Signale zur aktuellen Statusanzeige des Köchers

2.4.2 Geräteoptionen

Die MSTKN 24V-E/A Stecknussköcher sind in verschiedenen Softwarevarianten erhältlich, die sich in der Kodierung der Platznummern und in der LED-Anzeige unterscheiden.

2.4.2.1 Standard

In dieser Softwarevariante ist die Steckplatznummer binär kodiert. Über die 8 Signalleitungen können theoretisch 255 Steckplätze angewählt werden. Praktisch ist dieser Wert aber auf eine Anzahl von 63 Steckplätzen beschränkt („0“ = keine Vorwahl getroffen). Bitte beachten sie, dass in dieser Variante immer nur ein Werkzeug (oder Werkzeuggruppe) vorgegeben werden kann.

Binäre Auswahl der Steckplätze:

7	6	5	4	3	2	1	0	Signalnummer (Q0...Q7 or I0...I7)
0	0	0	0	0	0	0	1	Nummer = $1 \cdot 2^0 = 1$
0	0	0	0	0	0	1	0	Nummer = $1 \cdot 2^1 = 2$
0	0	0	0	0	0	1	1	Nummer = $1 \cdot 2^1 + 1 \cdot 2^0 = 2 + 1 = 3$
0	0	0	1	1	1	1	1	Nummer = $1 \cdot 2^4 + 1 \cdot 2^3 + 1 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0 = 16 + 8 + 4 + 2 + 1 = 31$
...								...

2.4.2.2 Open Protocol (OP)

In dieser Variante wird jeder Steckplatz mit einem separaten Signal angewählt. Da, wie oben beschrieben, nur 8 Auswahlssignale zur Verfügung stehen, ist die Anzahl der maximalen Steckplätze in der Variante OP auf 8 beschränkt. Bitte beachten sie, dass in dieser Variante mehrere Steckplätze vorgewählt werden können.

Anwahl bei Verwendung der Variante OP:

7	6	5	4	3	2	1	0	Signalnummer (Q0...Q7 or I0...I7)
0	0	0	0	0	0	0	1	Signal 0 → Steckplatz 1 ist ausgewählt
0	0	0	0	0	0	1	0	Signal 1 → Steckplatz 2 ist ausgewählt
0	0	0	0	0	0	1	1	Signal 0,1 → Steckplatz 1 and Steckplatz 2 pre-selected (Auswahl nur über die 8 Eingangsbits I0-I7)
...								...

2.4.2.3 XML

Die XML-Version ist eine leicht veränderte "Open Protocol" Version. Die Farbcodierung der LEDs und die Rückgabesignale des Stecknussköchers wurden verändert, um den internationalen Audi Produktionsstandards zu entsprechen. Die Signale sind genau wie bei "Open Protocol" binärcodiert (1 bit = 1 Steckplatz = 1 Signalleitung, Maximale Anzahl Steckplätze = 8).

Eine detailliertere Beschreibung der binärcodierten Signale finden Sie in Abschnitt 2.4.2.2.

Im Gegensatz zu den Standard- und OP-Versionen werden die Rückgabesignale für alle entnommenen Steckplätze ausgegeben, auch im Fehlerfall. Mit der XML-Version ist es auch möglich, eine Vorauswahl für mehrere Steckplätze zu treffen und es kann auch eine Freigabe erfolgen, wenn mehrere Werkzeuge gleichzeitig entnommen sind. Bei OP kann man zwar mehrere Steckplätze anwählen, aber immer nur ein Werkzeug entfernen.

Die folgende Tabelle zeigt die Farbcodierung:

- Vorwahl-Bit für Steckplatz gesetzt (Dieses Werkzeug soll für diesen Job benutzt werden)

Status	rote LED	gelbe LED	grüne LED
Nuss nicht entnommen	AN	AUS	AN
Nuss entnommen (und keine andere Nuss entnommen)	AN	AUS	AUS
Nuss entnommen (andere Nuss auch entnommen)	AN	Blinken	AUS

- Vorwahl-Bit für Steckplatz nicht gesetzt (Dieses Werkzeug soll für diesen Job nicht benutzt werden)

Status	rote LED	gelbe LED	grüne LED
Nuss nicht entnommen	AUS	AUS	AN
Nuss (fälschlicherweise) entnommen	AUS	Blinken	AUS

2.4.3 Betriebsarten

Die Auswahl der Betriebsarten erfolgt über die Steuersignale (I8...I10). Folgende Betriebsarten können ausgewählt werden.

MSTKN Eingangssignale: Eingänge IO...I10:

Byte 1			Byte 0								Betriebsart
I10	I9	I8	I7	I6	I5	I4	I3	I2	I1	I0	
0	0	0	0								Freie Auswahl
0	0	0	≠ 0								Werkerführung
0	0	1	0								Gesperrt (keine Entnahme erlaubt)
0	1	0	Rückgabewert								Direktbetrieb
1	0	0	0								Kalibrieren
1	0	1	0								Kalibrierung zurücksetzen
1	1	1	1	1	1	1	1	1	1	1	Kabelbruchtest

Alle anderen möglichen Kombinationen, die über die Steuersignale (I8...I10) erzeugt werden können, sind reserviert und dürfen nicht verwendet werden.

2.4.3.1 Freie Auswahl

Um diese Betriebsart auszuwählen, müssen sowohl Steuer- als auch Auswahlbits = 0 sein.

MSTKN Eingangssignale: Eingänge IO...I10:

Byte 1			Byte 0								Betriebsart
I10	I9	I8	I7	I6	I5	I4	I3	I2	I1	I0	
0	0	0	0								Freie Auswahl

In dieser Betriebsart zeigen die Statusbits folgende Zustände an:

- Q8 = Fehler: Dieses Signal wird aktiviert, wenn der Werker mehr als ein Werkzeug entnommen hat.
- Q9 = korrekte Entnahme: Dieses Signal ist aktiviert, wenn die Auswahl durch den Werker korrekt ist (nur ein Werkzeug entnommen).

MSTKN Ausgangssignale: Ausgänge Q0...Q9:

Byte 1		Byte 0								State	LEDs	Comment
Q9	Q8	Q7	Q6	Q5	Q4	Q3	Q2	Q1	Q0			
0	0	0								Idle	Alle gelben LEDs	Kein Werkzeug wurde entnommen
1	0	> 0								Ausgewählt	Entnommen Steckplatz grün, Rest gelb	Ein Werkzeug wurde entnommen, die Ausgangssignale entsprechen der entnommen Steckplatznummer (siehe Kapitel 2.4.2)

0	1	0	Fehler	Entnommen rot, Rest gelb	Mehr als ein Werkzeug entnommen. Die LEDs der entnommenen Steckplätze leuchten rot.
---	---	---	--------	-----------------------------	---

2.4.3.2 Werkerführung

Die Betriebsart Werkerführung wird ausgewählt, indem die Steuersignale "0" gesetzt werden und über die Vorwahlsignale eine Auswahl des Steckplatzes getroffen wird. Je nach Softwarevariante (Standard, Open Protocol oder XML) erfolgt die Auswahl des Steckplatzes binärkodiert (Standard) oder bitweise (OP und XML). (siehe Kapitel 2.4.2)

MSTKN Eingangssignale: Eingänge IO...I10:

Byte 1			Byte 0								Betriebsart
I10	I9	I8	I7	I6	I5	I4	I3	I2	I1	I0	
0	0	0	Steckplatzvorwahl (> 0)								Werkerführung

In dieser Betriebsart zeigen die Statusbits folgende Zustände an:

- Q8 = Fehler: Dieses Signal wird aktiviert, wenn der Werker mehr als ein Werkzeug entnommen hat.
- Q9 = korrekte Entnahme: Dieses Signal ist aktiviert, wenn die Auswahl durch den Werker korrekt ist (nur ein Werkzeug entnommen).

MSTKN Ausgangssignale: Ausgänge Q0...Q9:

Byte 1		Byte 0								State	LEDs	Comment
Q9	Q8	Q7	Q6	Q5	Q4	Q3	Q2	Q1	Q0			
0	0	0								Idle	Alle gelben LEDs	Kein Werkzeug wurde entnommen
1	0	> 0								Ausgewählt	Entnommen Steckplatz grün, Rest gelb	Ein Werkzeug wurde entnommen, die Ausgangssignale entsprechen der entnommenen Steckplatznummer (siehe Kapitel 2.4.2)
0	1	0								Fehler	Entnommen rot, Rest gelb	Mehr als ein Werkzeug entnommen. Die LEDs der entnommenen Steckplätze leuchten rot.

2.4.3.3 Direktbetrieb

Diese Betriebsart umfasst denselben Funktionsumfang wie die Betriebsart „Freie Auswahl“, ist aber um eine Überwachungsfunktion erweitert, die zur Überwachung von Kurzschlüssen und nicht verbundenen Leitungen dient. Der Stecknusskocher sendet je nach Status sein Ausgangssignal an die Steuerung und erwartet dieses, innerhalb von ca. 150ms, als Eingangssignal zurück. Erst danach erfolgt das Setzen des Freigabebits durch den Stecknusskocher. Erhält der Stecknusskocher nicht den korrekten Wert zurück (Ausgangssignal ≠ Eingangssignal), wird das Freigabebit nicht gesetzt und über die Leuchtdioden wird ein definiertes Leuchtmuster ausgegeben.

MSTKN Eingangssignale: Eingänge IO...I10:

Byte 1			Byte 0								Betriebsart
I10	I9	I8	I7	I6	I5	I4	I3	I2	I1	I0	
0	1	0	Eingangssignal = Ausgangssignal								Direktbetrieb

In dieser Betriebsart zeigen die Statusbits folgende Zustände an:

- Q8 = Fehler: Dieses Signal wird aktiviert, wenn der Werker mehr als ein Werkzeug entnommen hat.
- Q9 = korrekte Entnahme: Dieses Signal ist aktiviert, wenn die Auswahl durch den Werker korrekt ist (nur ein Werkzeug entnommen).

MSTKN Ausgangssignale: Ausgänge Q0...Q9:

Byte 1		Byte 0								State	LEDs	Comment
Q9	Q8	Q7	Q6	Q5	Q4	Q3	Q2	Q1	Q0			
0	0	0								Idle	Alle gelben LEDs	Kein Werkzeug wurde entnommen
1	0	> 0								Ausgewählt	Entnommen Steckplatz grün, Rest gelb	Ein Werkzeug wurde entnommen, die Ausgangssignale entsprechen der entnommen Steckplatznummer (siehe Kapitel 2.4.2)
0	1	0								Fehler	Entnommen rot, Rest gelb	Mehr als ein Werkzeug entnommen. Die LEDs der entnommenen Steckplätze leuchten rot.

Wenn der Rückgabewert der Steuerung nicht korrekt ist, wird ein definiertes Leuchtmuster ausgegeben, sodass der Stecknussköcher einen Verdrahtungsfehler am Gerät anzeigt.

2.4.3.4 Gesperrt

Die Betriebsart "keine Entnahme erlaubt" wird durch Setzen des Steuersignals I8 ausgewählt. Die Auswahlssignale müssen hierbei alle „0“ sein. In diesem Modus darf der Werker kein Werkzeug entnehmen. Wird dennoch ein Werkzeug entnommen, zeigt der Stecknussköcher über die Leuchtdioden und seine Ausgangssignale einen Fehler an.

MSTKN Eingangssignale: Eingänge I0...I10:

Byte 1			Byte 0								Betriebsart
I10	I9	I8	I7	I6	I5	I4	I3	I2	I1	I0	
0	0	1	0								Gesperrt (keine Entnahme erlaubt)

In dieser Betriebsart zeigen die Statusbits folgende Zustände an:

- Q8 = Fehler: Ein oder mehrere Werkzeuge entnommen.
- Q9 = 0 (immer Null)

MSTKN Ausgangssignale: Ausgänge Q0...Q9:

Byte 1		Byte 0								State	LEDs	Comment
Q9	Q8	Q7	Q6	Q5	Q4	Q3	Q2	Q1	Q0			
0	0	0								Idle	Alle gelben LEDs	Kein Werkzeug entnommen
0	1	0								Fehler	Entnommene rot, Rest gelb	Ein oder mehrere Werkzeuge wurden entnommen.

2.4.3.5 Kalibrierung

Der Modus Kalibrierung dient zur Maximierung der Sensorempfindlichkeit, der Konfiguration unbenutzter Steckplätze und zur Erkennung möglicher Erweiterungsmodule. Die Sensoren werden bereits kalibriert ausgeliefert, sodass es nicht nötig ist, dass eine Kalibrierung der Empfindlichkeit durch den Anwender durchgeführt werden muss. Eine Kalibrierung wird nur notwendig, wenn ein Hauptmodul nachträglich um ein oder mehrere Erweiterungsmodule erweitert wird oder ein Steckplatz des Köchers nicht verwendet werden soll.



Wichtig: Es ist unbedingt darauf zu achten, dass alle Werkzeuge entnommen werden, bevor die Kalibrierung durchgeführt wird. Bei einer falsch durchgeführten Kalibrierung des Köchers ist dessen Funktion gestört, bzw. es werden ggf. einzelne Steckplätze deaktiviert!

MSTKN Eingangssignale: Eingänge IO...I10:

Byte 1			Byte 0								Betriebsart	
I	I	I	I	I	I	I	I	I	I	I		
10	9	8	7	6	5	4	3	2	1	0		
1	0	0	0								Kalibrierung	Kalibrierung für max. Sensorempfindlichkeit und zur Erkennung von Erweiterungsmodulen
1	0	1	0								Kalibrierung rücksetzen	Löschen der Kalibrierung

In dieser Betriebsart zeigen die Statusbits folgende Zustände an:

MSTKN Ausgangssignale: Ausgänge Q0...Q9 während Kalibrierung läuft:

Byte 1		Byte 0								State	LEDs	Comment
Q	Q	Q	Q	Q	Q	Q	Q	Q	Q			
9	8	7	6	5	4	3	2	1	0			
0	0	1	1	1	1	0				Kalib. läuft	Alle gelben LEDs	Kalibrierung läuft

MSTKN Ausgangssignale: Ausgänge Q0...Q9 nachdem Kalibrierung beendet ist:

Byte 1		Byte 0								State	LEDs	Comment
Q	Q	Q	Q	Q	Q	Q	Q	Q	Q			
9	8	7	6	5	4	3	2	1	0			
0	0	1	0	0	0	# Steckplätze				Kalib. Ende	Gelbe LEDs blinken	Kalibrierung beendet, Q0-Q4 zeigen die Anzahl der Steckplätze an



Wichtig: Es ist wichtig darauf zu achten, dass die Steuerbits erst nach Anlegen des oben gezeigten Ausgangssignals (Kalibrierung beendet) zurückgesetzt werden und mit dem normalen Betrieb fortgesetzt wird.

2.4.3.6 Kabelbruchttest

Um die Verbindung zwischen Köcher und Steuerung zu überprüfen, beinhaltet der Funktionsumfang des Stecknussköchers die Möglichkeit, einen Kabelbruchttest durchzuführen. Um diesen auszuführen, müssen alle Eingangssignale „1“ sein, als Rückgabesignal setzt der Stecknussköcher alle seine Ausgangssignale auf „1“. Gleichzeitig wird der durchgeführte Test über die Leuchtdioden visualisiert. Deren Leuchtmuster wiederholt sich zyklisch wie folgt: alle roten LEDs -> alle gelben LEDs -> alle grünen LEDs

MSTKN Eingangssignale: Eingänge IO...I10:

Byte 1			Byte 0								Betriebsart	
I	I	I	I	I	I	I	I	I	I	I		
10	9	8	7	6	5	4	3	2	1	0		
1	1	1	1	1	1	1	1	1	1	1	Kabelbruchttest	

MSTKN Ausgangssignale: Ausgänge Q0...Q9:

Byte 1		Byte 0								State	LEDs	Comment
Q	Q	Q	Q	Q	Q	Q	Q	Q	Q			
9	8	7	6	5	4	3	2	1	0			
1	1	1	1	1	1	1				Checking	Alle gelben LEDs	Kabelbruchttest läuft

2.5 Aufbau der E/A-Signale Profibus/ProfiNet

Der Aufbau der E/A-Signale bei Profibus und ProfiNet ist abhängig davon, in welchem Modus (Intelligent, Benutzer) der Stecknussköcher betrieben werden soll.

2.5.1 Intelligenter Modus

Steuerungsbyte 0 (SPS → MSTKN) – Vorwahl des Steckplatzes:

7	6	5	4	3	2	1	0		
0..255(0xFF)								Signale zur Vorwahl Steckplatznummer	8 Signale um die Steckplatzvorwahl zu treffen (Betriebsart Werkerführung)

Steuerungsbyte 1 (SPS → MSTKN) – Auswahl Betriebsart:

7	6	5	4	3	2	1	0			
		(reserviert)			0	0	0	Betriebsart	Normalbetrieb	
						0	0		1	Keine Entnahme
						0	1		X	(reserviert)
						1	0		0	Kalibrieren
						1	0		1	Kalibrierung zurücksetzen
						1	1		X	(reserviert)
	X									Hauptmodul-LEDs
X									Grüne LED (OK Status)	



Statusbyte 0 (MSTKN → SPS) – Anzeige entnommener Steckplatz:

7	6	5	4	3	2	1	0		
0..255 (0xFF)								Steckplatz	Anzeige des aktuell entnommenen Steckplatzes

Statusbyte 1 (MSTKN → SPS) – Anzeige aktueller Status:

7	6	5	4	3	2	1	0			
		(reserviert)			0	0	0	Status	Keine Entnahme	
						0	0		1	Fehler: Werkzeug entnommen (nicht erlaubt)
						0	1		0	Mindestens ein Werkzeug entnommen
						0	1		1	Fehler: Sensorfehler an Steckplatz
						1	0		0	Kalibrierung wurde gestartet (noch nicht fertig)
						1	0		1	(reserviert)
						1	1		0	Kalibrierung fertig (OK)
						1	1		1	Kalibrierung fertig (mit Fehler)
X										Konfiguration



2.5.2 Benutzermodus (user mode)

Steuerungsbyte (SPS → MSTKN, Offset 0) – Vorwahl des Steckplatzes

7	6	5	4	3	2	1	0		
		(reserviert)			0	0	0	Betriebsart	Normalbetrieb
					0	0	1		Keine Entnahme
					0	1	X		(reserviert)
					1	0	0		Kalibrieren
					1	0	1		Kalibrierung zurücksetzen
					1	1	X		(reserviert)
	X							Hauptmodul-LEDs	Rote LED (NOK Status)
X									Grüne LED (OK Status)



LED-Anzeigebyte (SPS → MSTKN, Offset 1...N+1) – Achtung: 1 Byte je Steckplatz

7	6	5	4	3	2	1	0		
(reserviert)						0	0	LED Grün	Aus
						0	1		An
						1	X		Blinken
				0	0			LED Gelb	Aus
				0	1				An
				1	X				Blinken
		0	0					LED Rot	Aus
		0	1						An
		1	X						Blinken

Statusbyte (EINGANG SPS, Offset 0) – Anzeige entnommener Steckplatz

7	6	5	4	3	2	1	0		
	(reserviert)				0	0	0	Status	Keine Entnahme
					0	0	1		Fehler: Werkzeug entnommen (nicht erlaubt)
					0	1	0		Mindestens ein Werkzeug entnommen
					0	1	1		Fehler: Sensorfehler an Steckplatz
					1	0	0		Kalibrierung wurde gestartet (noch nicht fertig)
					1	0	1		(reserviert)
					1	1	0		Kalibrierung fertig (OK)
					1	1	1		Kalibrierung fertig (mit Fehler)
X									



Steckplatz Statusbyte (EINGANG SPS, Offset 1...N+1) – Status akt. Steckplatzstatus (1 Byte pro Platz)

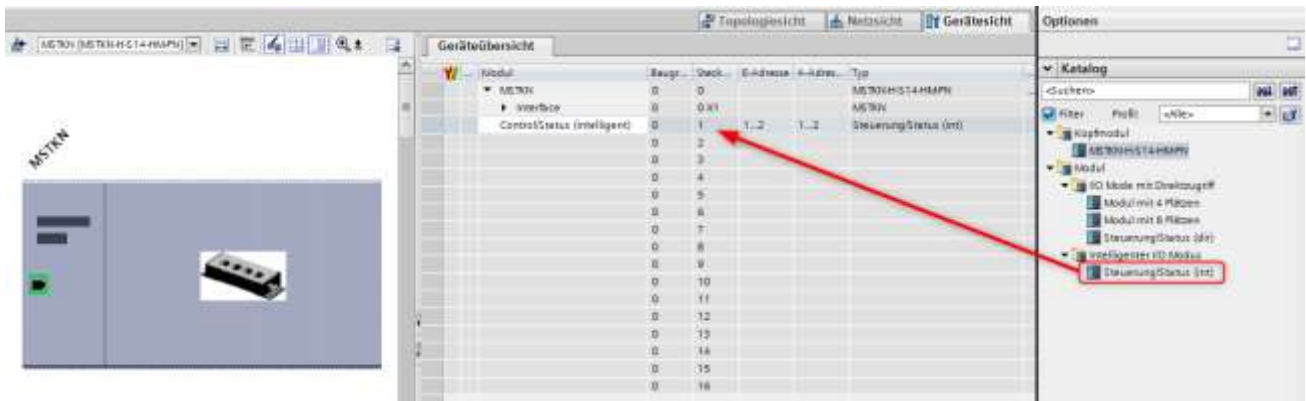
7	6	5	4	3	2	1	0		
	(reserviert)					0	0	Status	Werkzeug entnommen
						0	1		Werkzeug vorhanden (nicht entnommen)
						1	0		Fehler: Messwert zu groß (Teach mit Werkzeug)
						1	1		Fehler: Messwert zu klein (Werkzeuigerkennung)

2.5.3 Beispielkonfiguration Siemens SPS

2.5.3.1 Intelligenter Modus

Im intelligenten Modus werden 2 Bytes I/O für das Gerät benutzt. Es wird ein Byte für die Vorauswahl eines Köchers benutzt und gibt die Nummer des benutzten Werkzeugs (leerer Köcher) zurück. Im Allgemeinen sind die Signale kompatibel zu den 24V I/O-Modulen.

Hier ein Bildschirmausschnitt der Hardware-Konfiguration (TIA 5) – für die Konfiguration muss nur ein Modul (Stecknussköcher) angeschlossen sein:



Um die I/O-Signale in eine Applikation einzupflegen kann man einen Benutzerdatentyp benutzen und entsprechend den I/O-Adressen einfügen. Hier einige Beispiele für Benutzerdatentypen (UDTs) für Eingänge bzw. Ausgänge:

UDT_MSTKN_Int_I							
Name	Datentyp	Defaultwert	Erreichbar a...	Schreib...	Sichtbar i...	Einstellwert	Kommentar
1 Selected	Byte	1000	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Actually selected socket
2 state.0	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	State bit 0
3 state.1	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	State bit 1
4 state.2	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	State bit 2
5 bit3	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
6 bit4	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
7 bit5	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
8 bit6	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
9 config_error	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Configuration error (mismatch)

UDT_MSTKN_Int_Q							
Name	Datentyp	Defaultwert	Erreichbar a...	Schreib...	Sichtbar i...	Einstellwert	Kommentar
1 Preselect	Byte	1000	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Set socket number to blink to inform operator (0 = all)
2 mode.0	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Mode bit 0
3 mode.1	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Mode Bit 1
4 mode.2	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Mode Bit 2
5 bit3	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
6 bit4	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
7 bit5	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
8 led_red	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	LED red
9 led_gn	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	LED green

Achtung – Modus- und Status-Bits sind aktuell codierte Zahlen (siehe Kapitel 2.5.1 für Details). Diese werden über eine Variablen Tabelle in die aktuelle I/Os eingepflegt, sieh unten:

MSTKN									
Name	Datentyp	Adresse	Rema...	Erreic...	Schreib...	Sichtb...	Üb...	Kommentar	
1 MSTKN_Q	*UDT_MSTKN_Int_Q*	%Q1.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		MSTKN socket tray control signals (intelligent mode)	
2 MSTKN_I	*UDT_MSTKN_Int_I*	%I1.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		MSTKN socket tray status signals (intelligent mode)	

Um die Eingänge und Ausgänge anzuzeigen kann eine (Anzeige-)Tabelle angelegt werden, z. B. siehe unten:

	Name	Adresse	Anzeigeformat	Beobachtungswert	Steuerwert	...	Variablen-Kommentar
1	// Inputs						
2	"MSTKN_I".Selected	%IB1	Hex	16#03		<input type="checkbox"/>	Actually selected socket
3	"MSTKN_I"."state.0"	%I2.0	BOOL	<input type="checkbox"/> FALSE		<input type="checkbox"/>	State bit 0
4	"MSTKN_I"."state.1"	%I2.1	BOOL	<input checked="" type="checkbox"/> TRUE		<input type="checkbox"/>	State bit 1
5	"MSTKN_I"."state.2"	%I2.2	BOOL	<input type="checkbox"/> FALSE		<input type="checkbox"/>	State bit 2
6	"MSTKN_I".config_error	%I2.7	BOOL	<input type="checkbox"/> FALSE		<input type="checkbox"/>	Configuration error (mismatch)
7						<input type="checkbox"/>	
8	// Outputs						
9	"MSTKN_Q".Preselect	%QB1	Hex	16#00		<input type="checkbox"/>	Set socket number to blink to inform oper...
10	"MSTKN_Q"."mode.0"	%Q2.0	BOOL	<input type="checkbox"/> FALSE		<input type="checkbox"/>	Mode bit 0
11	"MSTKN_Q"."mode.1"	%Q2.1	BOOL	<input type="checkbox"/> FALSE		<input type="checkbox"/>	Mode Bit 1
12	"MSTKN_Q"."mode.2"	%Q2.2	BOOL	<input type="checkbox"/> FALSE		<input type="checkbox"/>	Mode Bit 2
13	"MSTKN_Q".led_gn	%Q2.7	BOOL	<input type="checkbox"/> FALSE		<input type="checkbox"/>	LED green
14	"MSTKN_Q".led_rd	%Q2.6	BOOL	<input type="checkbox"/> FALSE		<input type="checkbox"/>	LED red
15						<input type="checkbox"/>	

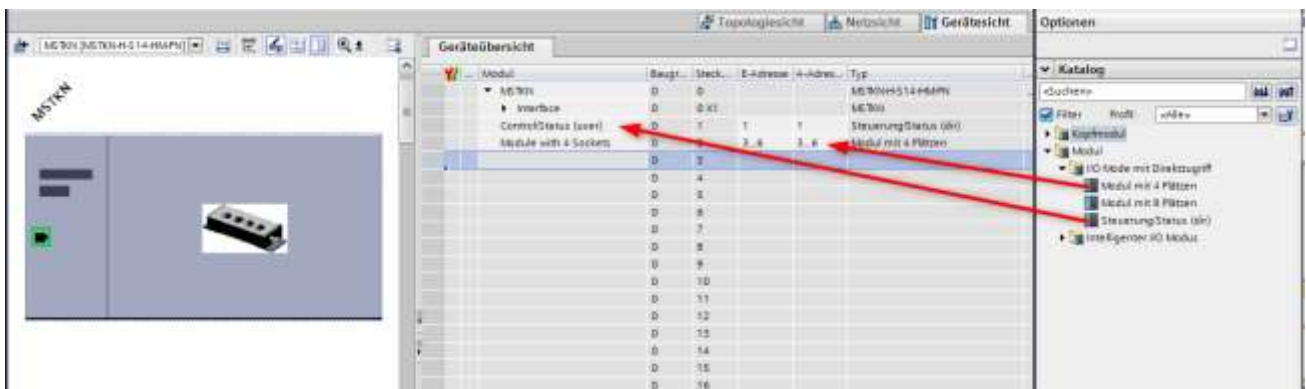
Um z. B. einen Kalibrier-Zyklus auszulösen wird der Modus auf 4 gesetzt – das heißt Bit 2 gesetzt, Modusbits 0, 1 sind NULL (ACHTUNG: bitte sicherstellen, dass alle Köcher leer sind/vorher alle Werkzeuge entfernen, bevor dies gestartet wird!):

	Name	Adresse	Anzeigeformat	Beobachtungswert	Steuerwert	...	Variablen-Kommentar
1	// Inputs						
2	"MSTKN_I".Selected	%IB1	Hex	16#04		<input type="checkbox"/>	Actually selected socket
3	"MSTKN_I"."state.0"	%I2.0	BOOL	<input checked="" type="checkbox"/> TRUE		<input type="checkbox"/>	State bit 0
4	"MSTKN_I"."state.1"	%I2.1	BOOL	<input checked="" type="checkbox"/> TRUE		<input type="checkbox"/>	State bit 1
5	"MSTKN_I"."state.2"	%I2.2	BOOL	<input type="checkbox"/> FALSE		<input type="checkbox"/>	State bit 2
6	"MSTKN_I".config_error	%I2.7	BOOL	<input type="checkbox"/> FALSE		<input type="checkbox"/>	Configuration error (mismatch)
7						<input type="checkbox"/>	
8	// Outputs						
9	"MSTKN_Q".Preselect	%QB1	Hex	16#00		<input type="checkbox"/>	Set socket number to blink to inform oper...
10	"MSTKN_Q"."mode.0"	%Q2.0	BOOL	<input type="checkbox"/> FALSE		<input type="checkbox"/>	Mode bit 0
11	"MSTKN_Q"."mode.1"	%Q2.1	BOOL	<input type="checkbox"/> FALSE		<input type="checkbox"/>	Mode Bit 1
12	"MSTKN_Q"."mode.2"	%Q2.2	BOOL	<input checked="" type="checkbox"/> TRUE	TRUE	<input checked="" type="checkbox"/> ⚠	Mode Bit 2
13	"MSTKN_Q".led_gn	%Q2.7	BOOL	<input type="checkbox"/> FALSE		<input type="checkbox"/>	LED green
14	"MSTKN_Q".led_rd	%Q2.6	BOOL	<input type="checkbox"/> FALSE		<input type="checkbox"/>	LED red
15						<input type="checkbox"/>	

Mit dieser Funktion werden alle aktuell leeren Köcher in „MSTKN_I“.Selected eingetragen. Um zurück in den Normalbetrieb zu gelangen den Modus 2 abwählen.

2.5.3.2 Benutzermodus

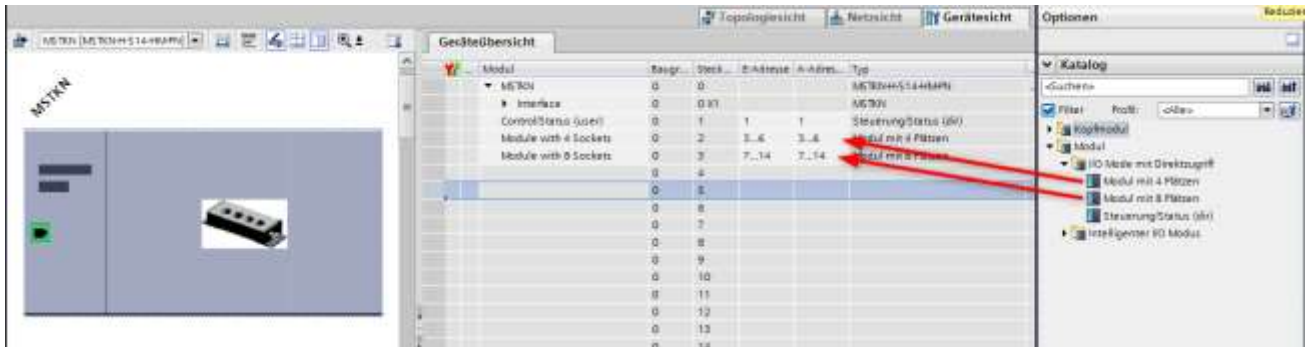
Benutzer Modus belegt 1 Byte I/O für das gesamte Gerät Kontrolle/Status plus ein Byte I/O für jeden Stecknussköcher. Somit hat die PLC volle Kontrolle über die LEDs und kann alle Sensor-Daten von jedem Stecknussköcher separat einlesen. Der Stecknussköcher fungiert als eine Art Remote-I/O-Modul, die PLC muss die Logik überwachen. Dies ermöglicht die Implementierung Benutzer-definierter Verhaltensweisen oder LED-Farbschema. Hier ein Bildschirmausschnitt der Hardware-Konfiguration (TIA 5) – um 1-Byte Status/Control Modul plus ein Modul pro Gerät (in diesem Fall muss ein 4-Fach Stecknussköcher mit 4 Bytes I/O angeschlossen sein):



Das Geräte I/O-Byte (status/control) arbeitet identisch zum "intelligenten Modus" status/control und benutzt das gleiche Bit-Mapping (siehe dazu das Beispiel für UDT-Definition). Verglichen zum "intelligenten Modus" liegt es auf I/O-Adresse mit Offset 0 (siehe Kapitel 2.5.2 für detaillierte Information über das Signal Layout).

Das Stecknussköcher I/O-Byte wird zur Kontrolle der drei LEDs für jeden Köcher benutzt und gibt Informationen über den Sensorstatus zurück. Bitte Kapitel 2.5.2 für weitere Details lesen.

Für Konstellationen mit einem Hauptmodul und Erweiterungsmodulen, die über den Erweiterungs-Bus verbunden sind, müssen in der PLC Hardware-Konfiguration weitere Module ergänzt werden. Die Module in der PLC müssen den physischen Gegebenheiten entsprechen, z. B. ein 4-fach Stecknussköcher Hauptmodul kombiniert mit einem 8-Fach Erweiterungsmodul würde wie siehe unten aussehen:



2.6 Aufbau der E/A-Signale Ethernet/WLAN

2.6.1 Modbus-Kommunikation

2.6.1.1 Übersicht

Die Kommunikation mit dem Stecknussköcher erfolgt über Modbus/UDP (bevorzugt) oder Modbus/TCP. Der Stecknussköcher arbeitet als Server und beantwortet die Anfragen der übergeordneten Steuerung.

Modbus arbeitet als Request/Response-Protokoll, d. h. der Stecknussköcher sendet nie spontan Daten, sondern immer nur als Antwort auf eine Anfrage der übergeordneten Steuerung. Um zu überwachen, ob eine „Verbindung“ mit der übergeordneten Steuerung aktiv ist, wird vom Stecknussköcher die Zeit zum letzten Empfang eines gültigen Modbus-Datenpaketes gemessen. Wird für eine vorgegebene Zeit (500ms) kein neuer Request empfangen, dann wechselt der Stecknussköcher in den Zustand „Verbindung inaktiv“ (LEDs blinken).

Der Modbus-Server im Stecknussköcher nutzt die folgenden Ports:

- 502/UDP und 502/TCP: Standard-Modbus-Protokoll
- 5003/UDP: Rückwärtskompatibler Modbus-Port (non-Standard Modbus)

Hinweise:

- Die UDP-Antwort wird an denselben Port zurückgesendet, der den Request ausgesendet hat. Auf der Seite der übergeordneten Steuerung ist damit nur ein Socket für die Kommunikation nötig.
- Zur Nutzung mit Windows muss bei UDP der Prozess freigeschaltet werden oder der Client muss eine feste Portnummer verwenden!
- Für single-cycle I/O können alle 33 I/O-Worte über FC23 in einer Transaktion geschrieben und gelesen werden. Eine feste Größe für das Telegramm ist für beide Betriebsarten (user/intelligent) möglich, da im Register 40001 die Betriebsart umgeschaltet wird.

ACHTUNG: Je nach Firmwareversion ist der Zugriff auf Teilbereiche der Register nicht möglich – um mit allen Varianten kompatibel zu bleiben empfiehlt sich der den FC23-Aufruf immer über den kompletten Bereich (33 Register ab Adresse 0 lesen und 33 Register ab Adresse 0 schreiben) durchzuführen!

ACHTUNG: Nicht alle Firmwarevarianten unterstützen TCP – aus Kompatibilitätsgründen empfiehlt sich die Nutzung von Modbus/UDP!

ACHTUNG: Bei Nutzung von TCP werden u. U. lange Timeouts/Retransmits vom Betriebssystem aktiv – um Verbindungsabbrüche schnell und zuverlässig zu erkennen sollte UDP bevorzugt eingesetzt werden (insbesondere bei WLAN)!

ACHTUNG: Multi-Master-Betrieb wird vom Stecknusskocher blockiert. Eine zweite Verbindung mit dem Stecknusskocher kann erst aufgebaut werden, wenn die bestehende Verbindung (z. B. über Timeout) getrennt wurde.

2.6.1.2 Unterstützte Modbus-Funktionscodes

Aktuell unterstützt der Stecknusskocher nur FC23.

Function Code (FC)	Name	Max. Message Size
23 (0x17)	Read and Write Registers	246 Bytes

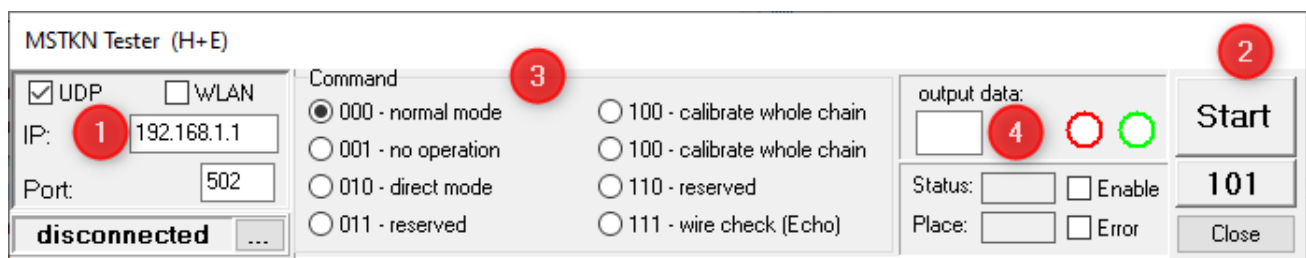
2.6.1.3 Modbus-Registermapping

Der Stecknusskocher stellt ausschließlich Register zur Verfügung (Register 4xxxx) – diese werden über FC23 angesprochen (über Adresse ab 0 ↔ Adresse = Registernummer - 40001). Den Registern sind in Schreib- und in Leserichtung unterschiedlichen Daten wie folgt zugeordnet:

Register	R/W	Name	Description
40001	R	Mode/Sockets	Mirrors actual mode, reports actual number of sockets
40002	R	Reserved	Always reads 0
40003	R	Select/Status	EA24 compatible status / selection (intelligent mode)
40004-40033	R	Sensor status	1 Byte per socket (user mode only)
40001	W	Mode/Sockets	Set mode (user/intelligent) and expected number of sockets
40002	W	Reserved	Always set to 0
40003	W	Preselect/Command	EA24 compatible commands / preselection (see below)
40004-40033	W	LED colors	1 Byte per socket LED colors (user mode only)

2.6.1.4 Test

Die Kommunikation über Modbus UDP kann mit Hilfe der Applikation „MSTKN Tester“ (MSTKN-Modbus-Master.exe) getestet werden:



Die folgenden Funktionen stehen über die Bedienoberfläche zur Verfügung:

- ① Einstellen der Kommunikationsparameter (UDP für Ethernet/WLAN)
- ② Start um die zyklische Kommunikation zu starten
- ③ Kommando (für intelligenten Modus)

- ④ Platzvorwahl, bzw. aktueller Status (Rückmeldung vom Köcher)

2.6.2 Intelligenter Modus

2.6.2.1 Ausgangsdaten (Modbus-Controller → MSTKN)

Adresse	0	1	2	3	4	5
Data	Modus (0 = „intelligent“)	Erwartete #Plätze	reserviert (=0)	reserviert (=0)	Kompatibel zu EA24	
					Vorwahl	Aktion

Byte 0: Auswahl Betriebsmodus „Intelligent“ = 0

Byte 1: Erwartete Anzahl Steckplätze. Bei einem Wert ungleich Null überprüft der Stecknussköcher die Anzahl der erwarteten Steckplätze mit der Anzahl der tatsächlich vorhandenen. Bei einem Wert von Null entfällt diese Überprüfung. Im Fehlerfall ist der Stecknussköcher nicht betriebsbereit und liefert eine entsprechende Fehlermeldung.

Byte 4: Vorwahl des Steckplatzes

7	6	5	4	3	2	1	0			
0									Keine Vorwahl	
1..60								Steckplatz		Vorwahl des gewünschten Steckplatzes
>60										reserviert

Byte 5: Steuerungsbyte (Aktion)

7	6	5	4	3	2	1	0		
		(reserviert)			0	0	0	Betriebsart	Normalbetrieb
					0	0	1		Keine Entnahme
					0	1	X		(reserviert)
					1	0	0		Kalibrieren
					1	0	1		Kalibrierung zurücksetzen
					1	1	X		(reserviert)
	X							Hauptmodul-LEDs	Rote LED (NOK Status)
X									Grüne LED (OK Status)

2.6.2.2 Eingangsdaten (MSTKN → Modbus-Controller)

Adresse	0	1	2	3	4	5
Data	Modus (0 = „intelligent“)	vorhandene #Plätze	reserviert (=0)	reserviert (=0)	Kompatibel zu EA24	
					Auswahl	Status

Byte 0: Spiegelt den Eingangswert der SPS (hier 0)

Byte 1: Vorhandene Steckplätze

7	6	5	4	3	2	1	0		
		0..60						#Steckplätze	Anzahl konfigurierter Steckplätze im System
	0							Reserviert	
X								Konfigurations-Fehler	= 0: Konfiguration ok oder Überprüfung inaktiv = 1: Konfigurationsfehler

Byte 4: Auswahl

7	6	5	4	3	2	1	0		
0	0	0..60						Steckplatz	Anzeige des aktuell entnommenen Steckplatzes

Byte 5: Status

7	6	5	4	3	2	1	0		
		(reserviert)			0	0	0	Status	Keine Entnahme
					0	0	1		Fehler: Werkzeug entnommen (nicht erlaubt)
					0	1	0		Mindestens ein Werkzeug entnommen
					0	1	1		Fehler: Sensorfehler an Steckplatz
					1	0	0		Kalibrierung wurde gestartet (noch nicht fertig)
					1	0	1		(reserviert)
					1	1	0		Kalibrierung fertig (OK)
					1	1	1		Kalibrierung fertig (mit Fehler)

2.6.3 Benutzermodus

2.6.3.1 Ausgangsdaten (Modbus-Controller → MSTKN)

Adresse	0	1	2	3	4	5
Data	Modus (1=user)	erwartete #Plätze	reserviert (=0)	reserviert (=0)	reserviert (=0)	Aktion
Adresse	6	7	64	65
Data	Platz #2*	Platz #1*	Platz #60*	Platz #59*

***ACHTUNG:** Bitte die Reihenfolge beachten!

Byte 0: Auswahl Betriebsmodus „user mode“ = 1

Byte 1: Erwartete Anzahl Steckplätze. Bei einem Wert ungleich Null überprüft der Stecknussköcher die Anzahl der erwarteten Steckplätze mit der Anzahl der tatsächlich vorhandenen. Bei einem Wert von Null entfällt diese Überprüfung. Im Fehlerfall ist der Stecknussköcher nicht betriebsbereit und liefert eine entsprechende Fehlermeldung.

Byte 5: Steuerungsbyte (Aktion)

7	6	5	4	3	2	1	0		
		(reserviert)			0	0	0	Betriebsart	Normalbetrieb
					0	0	1		Keine Entnahme
					0	1	X		(reserviert)
					1	0	0		Kalibrieren
					1	0	1		Kalibrierung zurücksetzen
					1	1	X		(reserviert)
	X								Hauptmodul-LEDs
X							Grüne LED (OK Status)		

Bytes 6 – 65: 1 Byte pro Steckplatz zur Ansteuerung der Leuchtdioden:

7	6	5	4	3	2	1	0		
(reserviert)					0	0		LED Grün	Aus
					0	1			An
					1	X			Blinken
				0	0			LED Gelb	Aus
				0	1				An
				1	X				Blinken
		0	0					LED Rot	Aus
		0	1						An

	1	X							Blinken
--	---	---	--	--	--	--	--	--	---------

2.6.3.2 Eingangsdaten (MSTKN → Modbus-Controller)

Adresse	0	1	2	3	4	5
Data	Modus (1=user)	vorhandene #Plätze	reserviert (=0)	reserviert (=0)	reserviert (=0)	Status
Adresse	6	7	64	65
Data	Platz #2*	Platz #1*	Platz #60*	Platz #59*

***ACHTUNG:** Bitte die Reihenfolge beachten!

Byte 0: Spiegelt den Eingangswert der SPS (hier 0)

Byte 1: Vorhandene Steckplätze

7	6	5	4	3	2	1	0			
								0..60	#Steckplätze	Anzahl konfigurierter Steckplätze im System
	0								Reserviert	
X									Konfigurations-Fehler	= 0: Konfiguration ok oder Überprüfung inaktiv = 1: Konfigurationsfehler

Byte 5: Status

7	6	5	4	3	2	1	0		
					0	0	0	Status	Keine Entnahme
					0	0	1		Fehler: Werkzeug entnommen (nicht erlaubt)
					0	1	0		Mindestens ein Werkzeug entnommen
					0	1	1		Fehler: Sensorfehler an Steckplatz
					1	0	0		Kalibrierung wurde gestartet (noch nicht fertig)
					1	0	1		(reserviert)
					1	1	0		Kalibrierung fertig (OK)
					1	1	1		Kalibrierung fertig (mit Fehler)

Bytes 6 – 65: 1 Byte pro Steckplatz mit Status des Steckplatzes:

7	6	5	4	3	2	1	0		
						0	0	Status	Keine Entnahme
						0	1		Werkzeug entnommen
						1	0		Fehler: Sensorwert zu hoch
						1	1		Fehler: Sensorwert zu niedrig

2.6.4 Python Beispielcode

Der folgende Code zeigt, wie man mit Hilfe von Python auf den Stecknusskocher zugreifen kann (Minimaler Code, ohne Fehlerüberprüfungen).

```
## NOTE: 'use pip install pymodbus' to install the python modbus library
from pymodbus.client.sync import ModbusUdpClient
import binascii
import time

client = ModbusUdpClient('10.10.2.64', 502)
arguments = {
    'read_address': 0, 'read_count': 33,
    'write_address': 0, 'write_registers': [
        0x0104, # hi = 1 (direct mode), lo = 4 (number of sockets)
```

```

0,          # reserved
0x0000,    # hi = 0 (socket preselect intelligent),
          # lo = 0 (mode select (intelligent mode only))
# Direct mode only; one byte per socket, two bits per LED,
# binary 00RRYYGG (RR=red, YY=yellow, GG=green,
# bits per LED: 00=off, 01=steady on, 10=blinking, 11=blinking)
# Sample:
# 01          Socket #1: LED off
#   02        Socket #2: LED green (steady)
#           03 Socket #3: LED green blinking
#           04 Socket #4: LED yellow
0x0001, 0x0204, 0x2010, 0x4030,
0, 0, 0, 0,
0, 0, 0, 0,
0, 0, 0, 0,
0, 0, 0, 0,
0, 0, 0, 0,
0, 0, 0, 0,
0, 0, 0, 0,
0, 0
]
}
while (True):
    result = client.readwrite_registers(**arguments)
    print(hex(result.registers[3]), hex(result.registers[4]),
hex(result.registers[5]))
    time.sleep(1.0)

client.close()

```

2.7 Aufbau der E/A-Signale EtherCAT

2.7.1 Übersicht

Der Aufbau der I/O-Daten der EtherCat-Schnittstelle entspricht der im Kapitel 2.6 beschriebenen Struktur der Ethernet/OpenModbus-Kommunikation.

Insgesamt werden über EtherCat 66 Bytes Eingangsdaten und 66 Bytes Ausgangsdaten zyklisch ausgetauscht, unabhängig von der Betriebsart „intelligenter Modus“ oder „Benutzermodus“. Die Umschaltung zwischen beiden Modi erfolgt wie bei Ethernet/OpenModbus über das erste Byte.

2.7.2 Eingangsdaten (MSTKN → SPS)

Die 66 Bytes der Eingangsdaten sind wie folgt aufgebaut (Details siehe Kapitel 2.6.2 und 2.6.3):

Byte-Adresse	Name (ESI-Datei)	Beschreibung
0	OperationMode	Current socket tray operation mode (0=intelligent, 1=usermode)
1	Sockets_Act	Actual number of sockets as configured in the trays internal config
2-3	Reserved_01 Reserved_02	
4	Selected	Current selected socket (if any, else = 0, intelligent mode only)
5	Status	Bitmask as described in chapter Fehler! Verweisquelle konnte nicht gefunden werden.
6..65	Socket_01 Socket_02 ... Socket_60	Socket state of socket #1 (as described in Fehler! Verweisquelle konnte nicht gefunden werden.) Socket #2 ...

Socket #60

2.7.3 Ausgangsdaten (MSTKN → SPS)

Die 66 Bytes der Ausgangsdaten sind wie folgt aufgebaut (Details siehe Kapitel 2.6.2 und 2.6.3):

Byte-Adresse	Name (ESI-Datei)	Beschreibung
0	OperationMode	Current socket tray operation mode (0=intelligent, 1=usermode)
1	Sockets_Exp	Expected number of sockets
2-3	Reserved_01 Reserved_02	
4	Preselect	To be selected socket (if any, else = 0, intelligent mode only)
5	Action	Bitmask as described in chapter Fehler! Verweisquelle konnte nicht gefunden werden.
6..65	LEDs_01 LEDs_02 ... LEDs_60	LED state for socket #1 (as described in Fehler! Verweisquelle konnte nicht gefunden werden. , only in usermode) LED state for socket #2 ... LED state for socket #60

2.7.4 Inbetriebnahme mit CtrlX

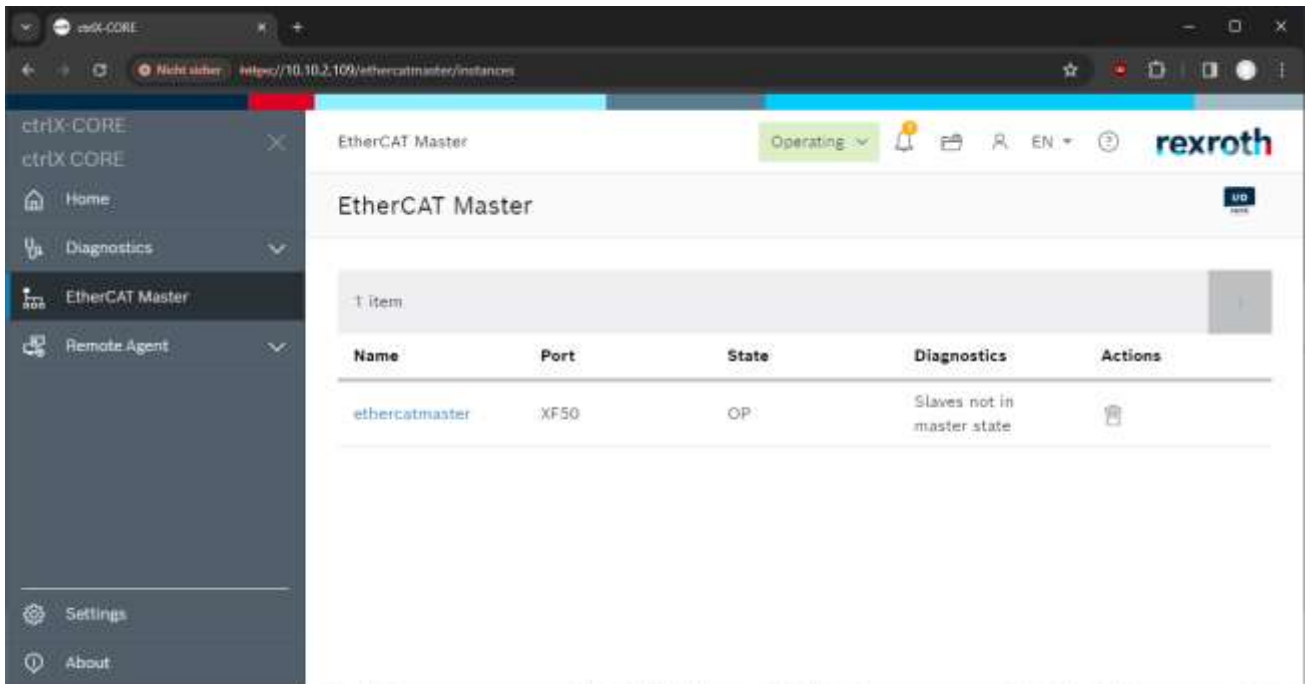
Die Inbetriebnahme des Stecknussköchers mit der CtrlX erfolgt wie in den folgenden Schritten beschrieben.

2.7.4.1 Öffnen der benötigten Software

Zur Konfiguration der CtrlX ist eine passende Version der „ctrlX I/O Engineering“ Software notwendig. Der PC mit der Software muss über die Schnittstelle XF10 der CtrlX verbunden sein. Der Stecknussköcher wird über die Schnittstelle XF50 an die CtrlX angebunden (EtherCat-Bus).

Durch Eingeben der IP-Adresse der Schittstelle XF10 im Webbrowser des PCs wird die CtrlX Device-Startseite geöffnet (falls die Adresse nicht bekannt ist, dann kann man 192.168.1.1 versuchen oder mit Hilfe der Software CtrlX WORKS das Netzwerk durchsuchen).

Nach der Anmeldung an der CtrlX Device-Startseite (z. B. <https://192.168.1.1>) wechselt man auf „EtherCAT Master“ Webapplikation:

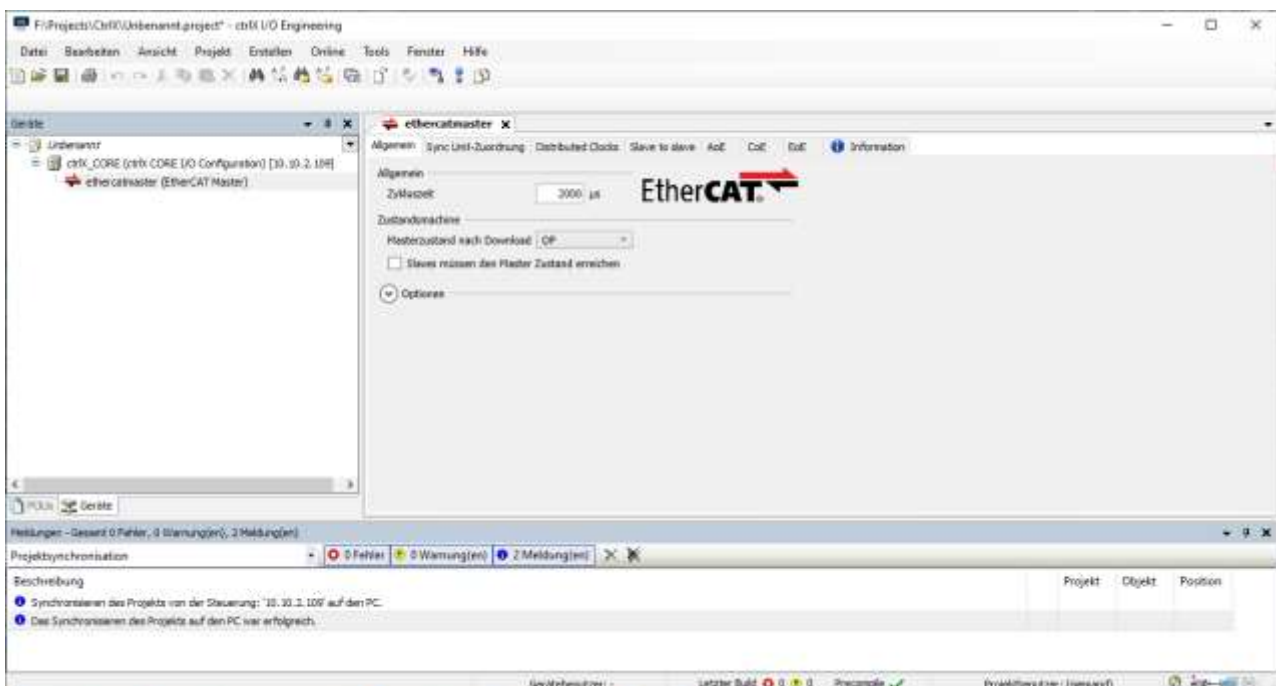


Ist diese nicht vorhanden, dann muss diese zuerst über den App-Store installiert werden (**ACHTUNG:** hierfür ist eine Lizenz nötig, das Registrieren am Lizenzportal benötigt ein paar Tage, zudem ist die Bestellung einer Lizenz nur offline über die Rexroth-Distributoren möglich – der Prozess dauert mindestens einige Tage!)

Zum Start des I/O-Engineering wird der kleine Button rechts oben mit dem „I/O“ logo geklickt:

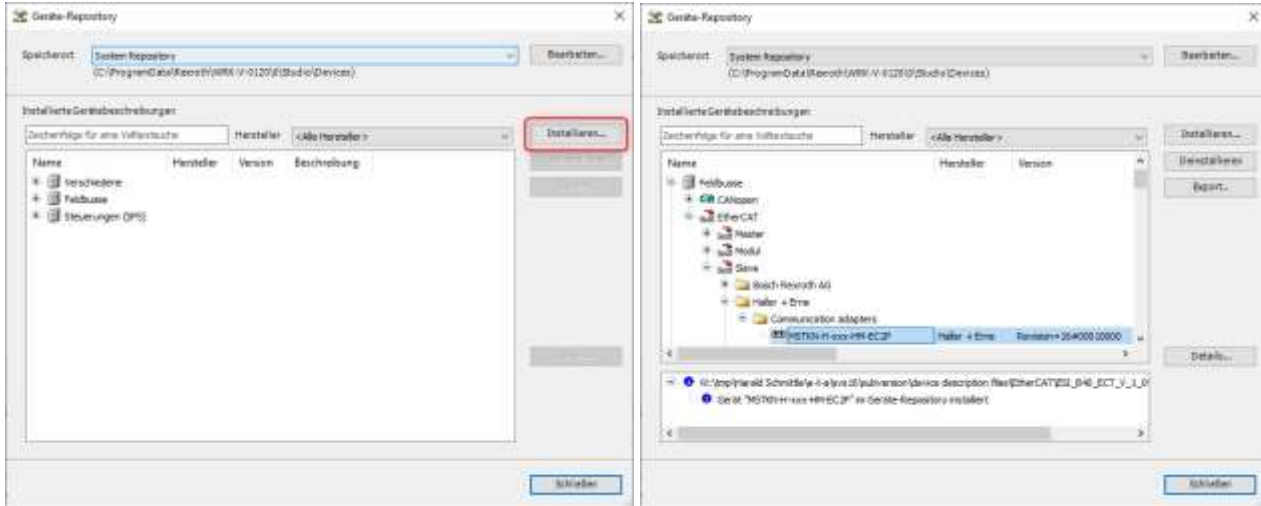


Danach öffnet sich dann das „CtrlX I/O Engineering“ (evtl. mit einem leeren Projekt):



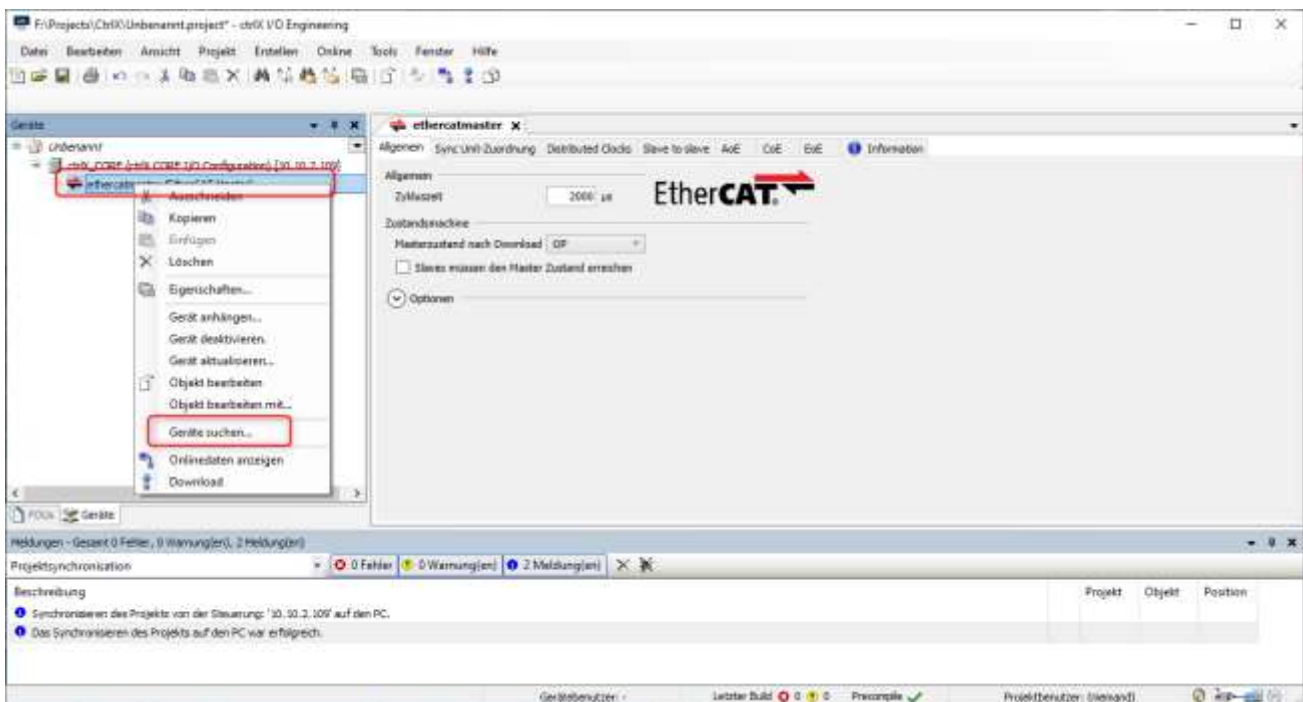
2.7.4.2 Installieren der ESI-Datei


Bevor das Gerät in die Konfiguration aufgenommen werden kann, muss die zum Stecknusskocher gehörige ESI-Datei in die Gerätedatenbank des I/O Engineering aufgenommen werden. Die Gerätedatenbank ist über das Hauptmenü über Tools → Geräterepository erreichbar. Dort kann dann über den Button „Installieren“ die ESI-Datei des Stecknusskochers (ESI_B40_ECT_V_1_09-MSTKN-R0002.xml) installiert werden:

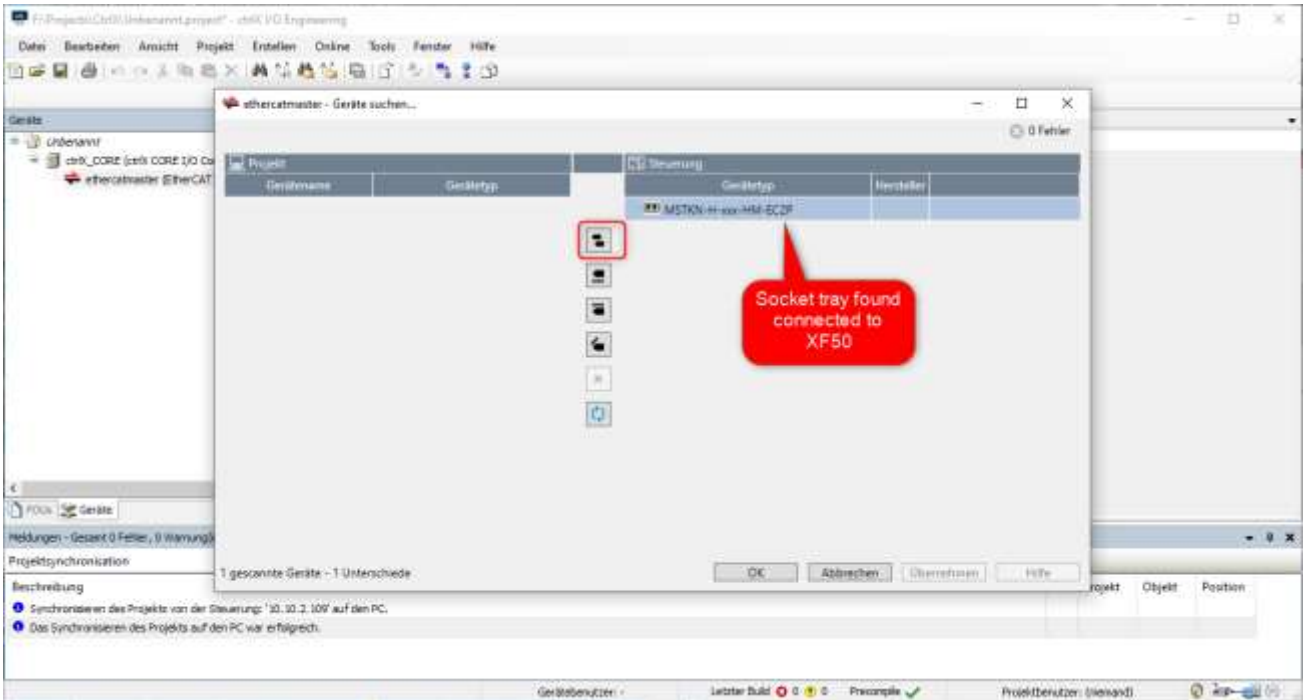


2.7.4.3 Stecknusskocher in I/O-Konfiguration einbinden

Die einfachste Vorgehensweise zum Einbinden des Stecknusskochers in das EtherCat-Bussystem ist nach dem (an XF50 angeschlossenen) Gerät zu suchen. Hierzu wird im I/O-Engineering das Kontextmenü des Masters (rechtsklick) aufgerufen und der Eintrag „Geräte suchen“ ausgewählt:

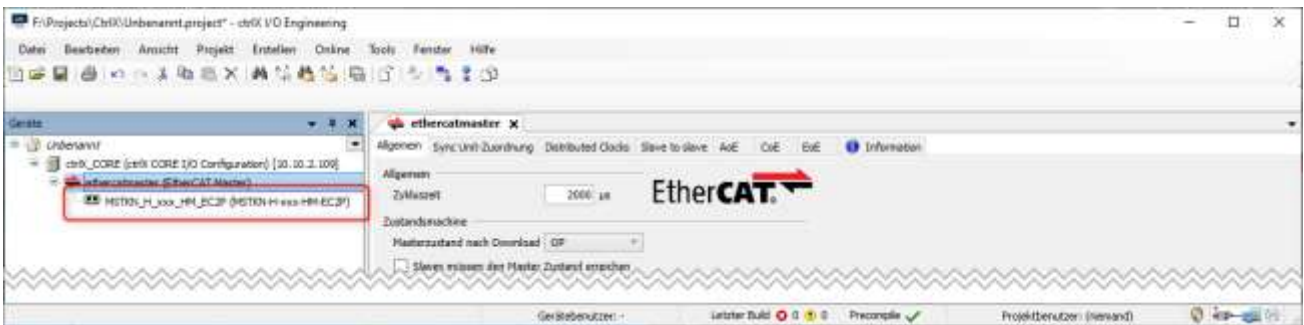


Der Stecknusskocher wird gefunden und kann über den Button  ins Projekt übernommen werden:



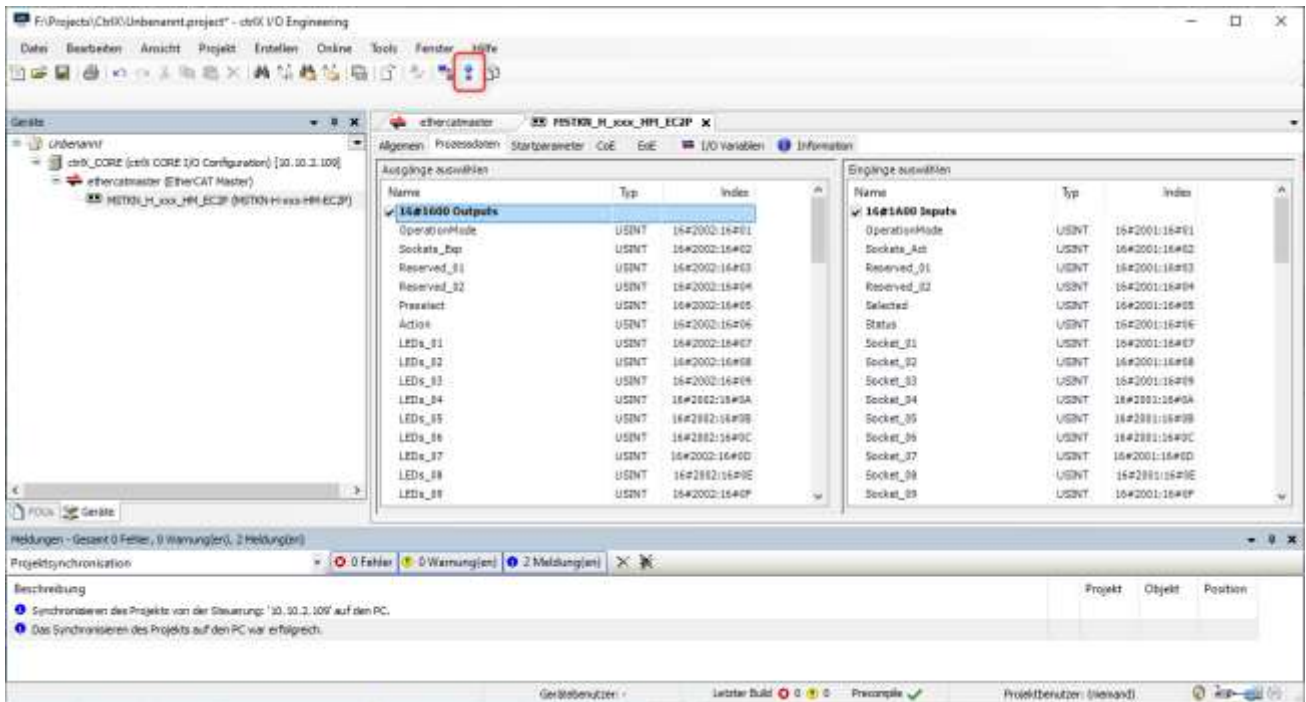
Nach der Übernahme kann ggf. noch der Gerätename (im Projekt) geändert werden.

Durch Klicken auf OK wird die geänderte Konfiguration dann ins Projekt übernommen:



2.7.4.4 Laden der I/O-Konfiguration in CtrlX

Nachdem die Konfiguration eingerichtet wurde, muss diese jetzt noch an die CtrlX übertragen werden. Hierzu wird der Button „Download“ aus der Menüzeile verwendet:

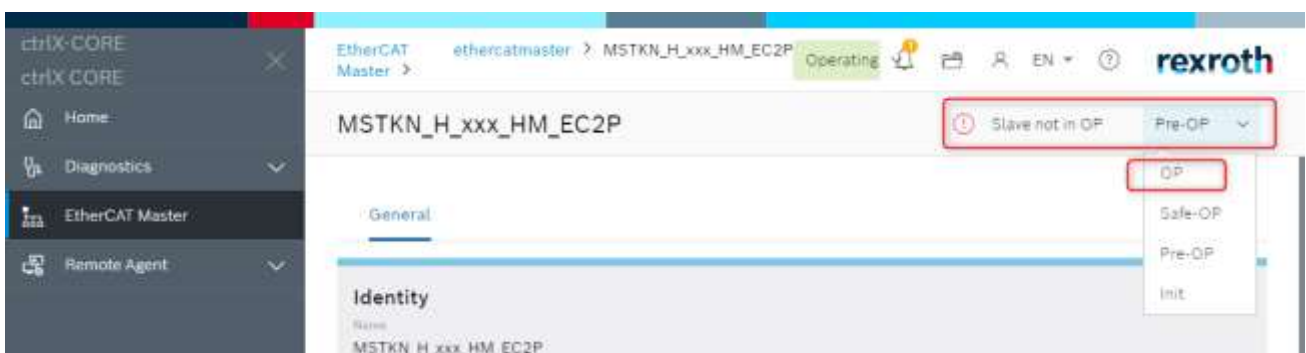


2.7.4.5 Starten der zyklischen Kommunikation

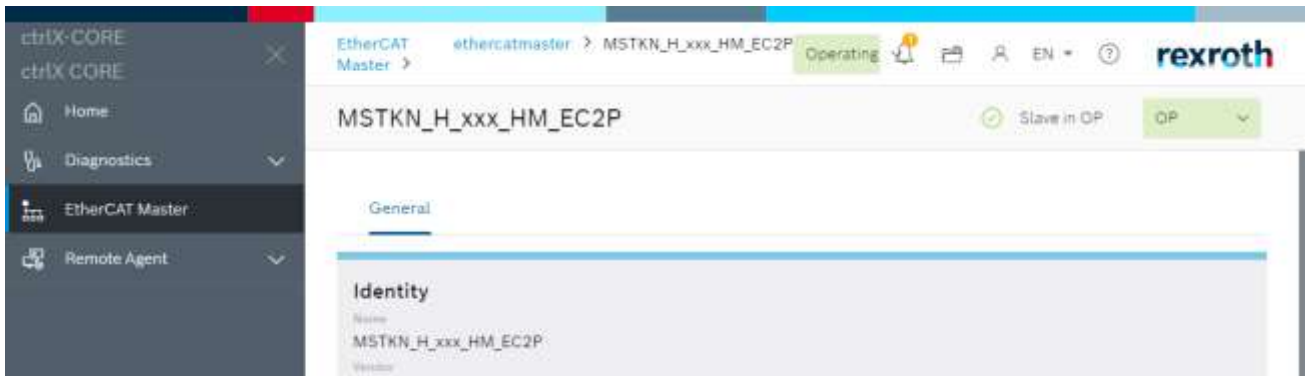
Standardmäßig stoppt der EtherCat-Master in der CtrlX die Kommunikation mit einem Gerät, wenn dessen Konfiguration geändert wurde – man kann dann entweder über die CtrlX Device-Website oder über das I/O-Engineering (Doppelklick auf den Master, dann „Online“ (aus der Menüzeile) und der Operation-State ändern) den Bus wieder starten. Hierzu auf der Webapplikation „EtherCAT Master“ den Namen des Interfaces anklicken, dann werden die angeschlossenen Geräte und deren Status angezeigt:



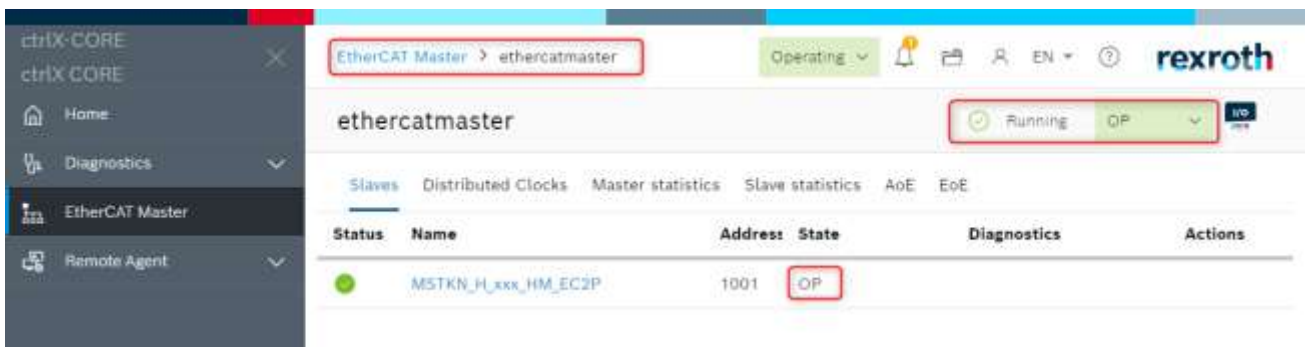
Durch einen Klick auf den Gerätenamen wird auf die Detail-Seite des Gerätes gewechselt:



Dort kann dann im Dropdown „OP“ ausgewählt werden, um die zyklische Kommunikation zu starten- der „Slave“-Zustand wechselt dann zu „OP“:



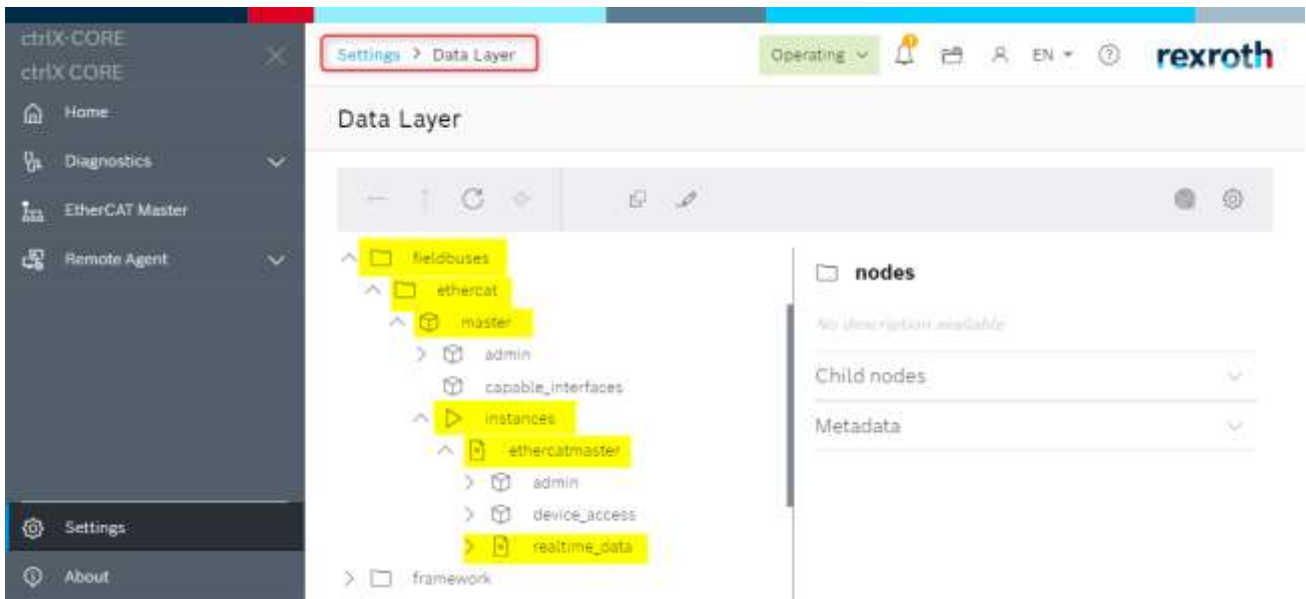
Auf der Master-Ebene wird dann ebenfalls alles als „OK“ angezeigt:



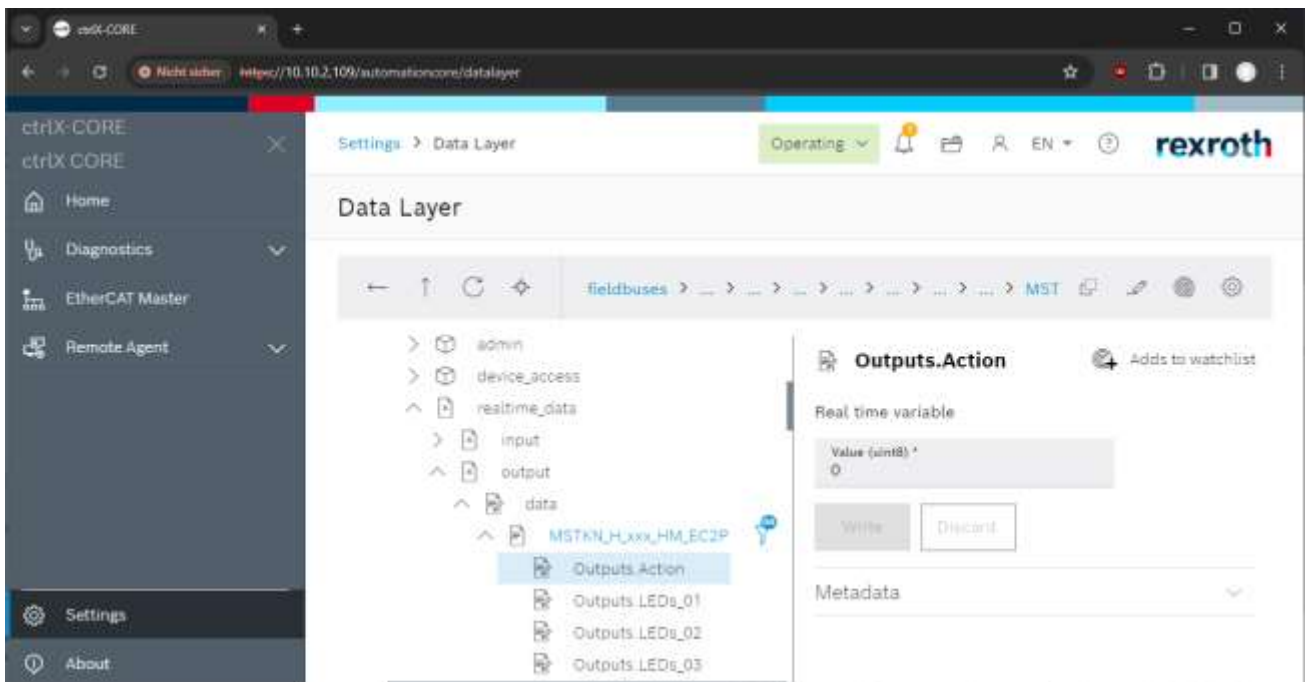
2.7.4.6 I/O-Diagnose und Test (DataLayer)

Die CtrIX bildet (theoretisch) alle zyklisch ausgetauschten Daten in den „Realtime“-Teil des Datalayers ab. Dort kann man den aktuellen Stand der Ein/Ausgänge sehen und diese auch manuell (zu Testzwecken) ändern (forcen).

Die Web-Applikation für den Datalayer ist über Settings → Datalayer erreichbar, die ausgetauschten Daten sind über ethercat → master → instances → ethercatmaster → realtime_data erreichbar:

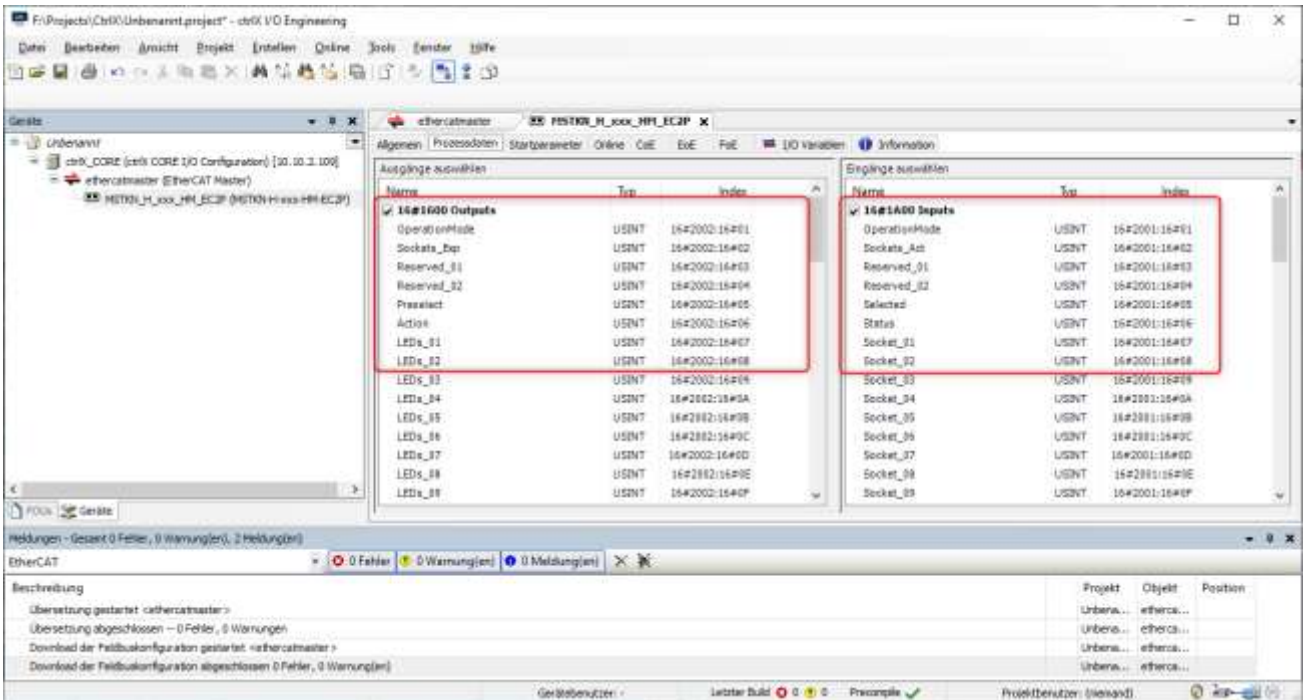


Dort können die Daten dann unter input/output → data → Gerätenamen eingesehen und auch ggf. gesteuert werden:

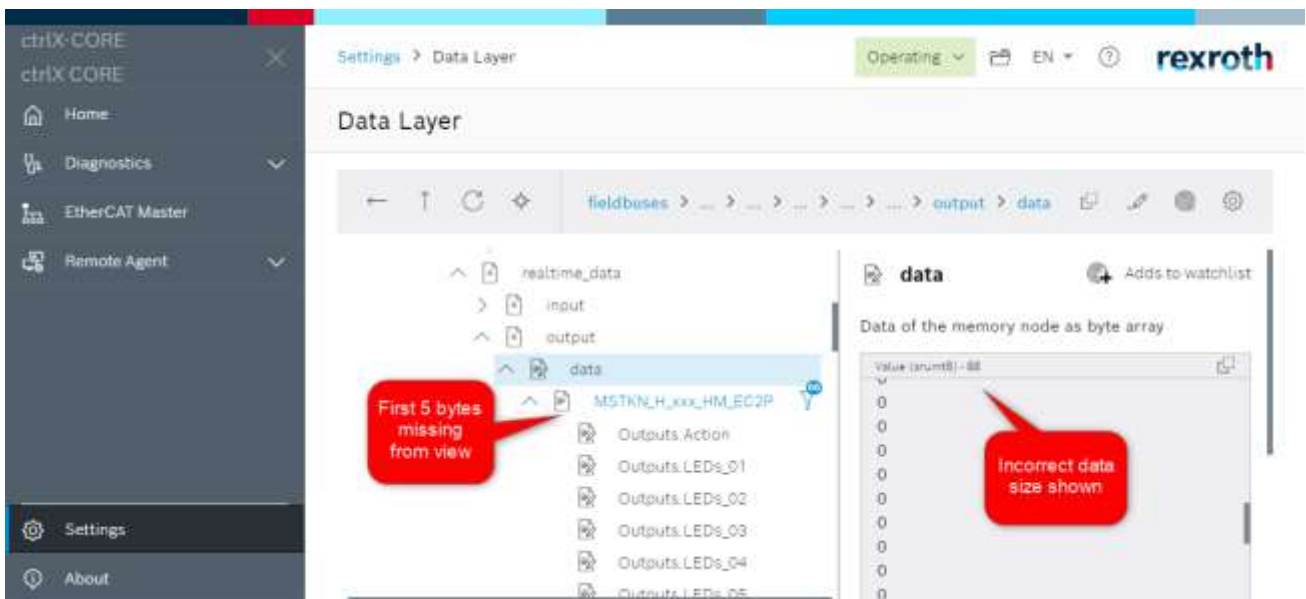


ACHTUNG: Die aktuelle Firmwareversion hat im Datalayer einige Bugs, die dazu führen, dass Variablen in der CtrlX Weboberfläche nicht korrekt angezeigt werden und gesteuert werden können. Dies betrifft die ersten 5

Variablen des „output“-Bereichs des Stecknussköchers – obwohl diese im I/O-Engineering korrekt dargestellt werden, fehlen die zugehörigen Datenfelder im Datalayer:



Obwohl wie im vorherigen Bild gezeigt, das I/O-Engineering die Datenfelder korrekt abbildet, fehlen in der Webansicht einige Datenfelder – die Ansicht startet mit „Action“ statt „OperationMode“:





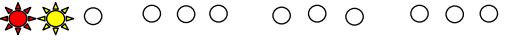
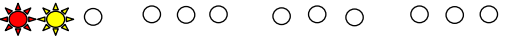


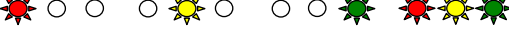

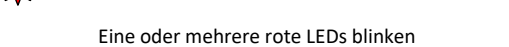

Ebenfalls fehlerhaft wird die Größe des I/O-Bereichs angezeigt und auch die letzten Bytes (Outputs.LEDs_50-60) fehlen – der Web-View scheint immer nur max. 50 Elemente anzuzeigen.



3 Anzeigen

Bei den MSTKN Stecknussköchern gibt es zwei Diagnosemöglichkeiten, die zur einfachen Fehlerbehebung beitragen. Zum einen werden auftretende Fehler über die Leuchtdioden der Steckplätze angezeigt, zudem sind je nach Gateway-Variante neben den Steckverbindern auf dem Seitenteil LEDs vorhanden.

3.1 Sensor-Anzeigemuster

Die nachfolgende Tabelle stellt alle möglichen Fehler sowie entsprechende Hinweise der Sensor-Anzeigen dar.

Fehlerbild	Fehlerursache	Fehlerbehebung
 Alle LEDs blinken	Interne MSTKN Kommunikation unterbrochen – Slave empfängt keine Daten, nur Spannungsversorgung	Gerät einschicken
 ROTE LED leuchtet dauernd, GELBE und GRÜNE LED(s) blinken abwechselnd	Es fehlen Köcher (Sensorplatinen)	Drehschalter auf den Einzel-Sensorplatinen kontrollieren – es darf keine gleiche Nummer eingestellt sein (die erste Platine hat immer die Nummer ,1') - bei Einzelsensoren wird auf der ersten Platine die Gesamtzahl der Kette eingestellt
 An einem oder mehreren Köchern blinkt ROT dauernd und GELB einmal	Sensorplatine empfängt keine BROADCAST-Telegramme vom MASTER, bei Einzelplatinen ist der MASTER die erste Platine	Bei Einzelsensoren interne Kabel prüfen Sonst Gerät einsenden
 An einem oder mehreren Köchern blinkt ROT dauernd und GELB zweimal	Sensorplatine empfängt keine Telegramme von Nachbarn	Drehschalter auf den Einzel-Sensorplatinen kontrollieren, insgesamt müssen die Nummern fortlaufend sein, ohne Lücke
 Alle roten und grünen LEDs blinken	Sensorplatine empfängt nur Telegramme die nicht an die eigene Modbus ID gerichtet sind – Modbus und Slave IDs prüfen und ggf. korrigieren	Slave-Adresse korrekt einstellen Stecknussköcher kalibrieren
 Alle roten und gelben LEDs blinken	Sensorplatine empfängt Modbus-Telegramme mit falscher Baudrate Sensorplatine empfängt Telegramme mit CRC Fehler	Kabellänge prüfen Abschlusswiderstand prüfen Gerät einschicken
 Falsche Rückmeldung im Direktbetrieb (EA24)	Verbindungsabbruch mit Feldbusmaster (Profibus, Modbus UDP etc.)	Feldbusmaster prüfen Kabelverbindungen überprüfen Feldbusadresse/IP prüfen Signaleingang von Steuerung auf Ausgänge spiegeln (EA24) Direktbetrieb deaktivieren (EA24V)
 Alle roten LEDs blinken	Master sendet ein Telegramm an einen Slave, erhält jedoch keine Rückmeldung	Prüfen, ob alle Erweiterungsmodule angeschlossen sind Stecknussköcher kalibrieren
 Eine oder mehrere rote LEDs blinken	Sensorfehler, Messwert größer als der Kalibrierwert	Stecknussköcher kalibrieren Gerät einschicken
 Lauflicht von links nach rechts	Kabelbruchtest aktiv	Kabelbruchtest deaktivieren

Fehlerbild	Fehlerursache	Fehlerbehebung
 Lauflicht (mit drei LEDs) von links nach rechts	Konfiguration defekt	Applikation über Bootloader neu programmieren (mit ‚clear config page‘ gesetzt wenn nicht möglich Bootloader neu programmieren, Gerät einschicken
 An einem Steckplatz leuchten alle LEDs	Interner Master hat einen Adressenkonflikt erkannt. Eine Slave-Adresse ist doppelt vergeben. Die drei LEDs der Steckplätze zeigen an, welche Adresse betroffen ist.	Slave-Adressen überprüfen Slave-Adressen aufsteigend und durchgängig vergeben

3.2 Hauptmodul-Anzeigen

3.2.1 Ok/NOK-LEDs

Die Hauptmodule der MSTKN Stecknussköcherfamilie verfügen über zwei zusätzliche Leuchtdioden (grün und rot), die vom Benutzer frei genutzt werden können, z.B. um eine „i.o.“ oder „n.i.O“ Verschraubung anzuzeigen

Ansteuerung bei EA24 (DSUB25):

MSTKN Eingangssignale			
Pin	Signal	Beschreibung	IM24V
PIN 24	LEDR	rote LED (24V=on)	A11
PIN 25	LEDG	grüne LED (24V=on)	A13

Ansteuerung bei Feldbussystemen:

MSTKN Eingangssignale								
Byte 1								
7	6	5	4	3	2	1	0	
	X							Hauptmodul-LEDs
								Rote LED (NOK Status)
X								Grüne LED (OK Status)

4 Diagnose und Konfiguration

4.1 Übersicht

Wie bereits in Kapitel 1.8 beschrieben, kann die Schnittstelle für die interne Busverbindung auch zur Konfiguration und Diagnose des Stecknussköchersystems eingesetzt werden. Hierzu wird ein USB ↔ M12 Adapterkabel verwendet, welches anstelle des Abschlusswiderstandes am Ende des Bus aufgesteckt wird.

Bitte beachten: Für einen störungsfreien Betrieb muss ein Hautmodul im Bus vorhanden und eingeschaltet sein! Der Betrieb des USB ↔ M12-Adapterkabels nur an einem Erweiterungsmodul ist nicht erlaubt.

Die Parametrier-/Diagnosesoftware kommuniziert dann über den Bus mit den verschiedenen Teilen des Stecknussköchersystems und stellt verschiedene Funktionen zur Parametrierung und Diagnose bereit.

Das USB ↔ M12-Adapterkabel sollte im spannungsfreien Zustand der Stecknussköcher aufgesteckt werden. Danach kann die Versorgungsspannung wieder angelegt werden.

4.2 Software-Installation

Zu Nutzung der Parametrier-/Diagnosesoftware muss ein virtueller COM-Port USB-Treiber für FTDI-Serielladapter installiert werden (siehe <https://ftdichip.com/drivers/vcp-drivers/>).

Die eigentliche Parametrier-/Diagnosesoftware erfordert keine Installation, es ist ausreichend, die Dateien in ein Verzeichnis auf der lokalen Festplatte zu kopieren und die Applikation zu starten.

4.3 Werkzeugerkennung parametrieren

Zum Aktivieren und Parametrieren der Werkzeugerkennung wird die Applikation STKN-SM-Config.exe verwendet.

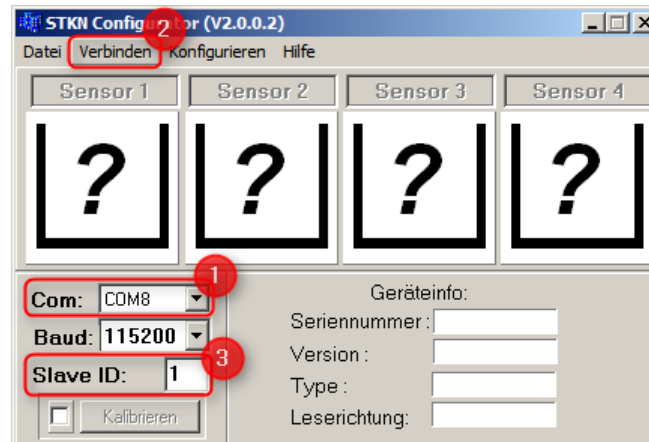
Die grundlegende Vorgehensweise ist wie folgt:

1. Verbindung mit dem Stecknussköcher über das USB ↔ M12-Kabel herstellen (siehe oben)
2. STKN-SM-Config.exe starten
3. Verbindung mit dem Stecknussköchersystem herstellen
4. Den Konfigurationsbildschirm zur Werkzeugerkennung öffnen
5. Die Werkzeugparameter anlernen
6. Die geänderten Parameter an den Stecknussköcher senden
7. Ggf. Testen

Die folgenden Kapitel zeigen die Vorgehensweise.

4.3.1 Verbindung herstellen

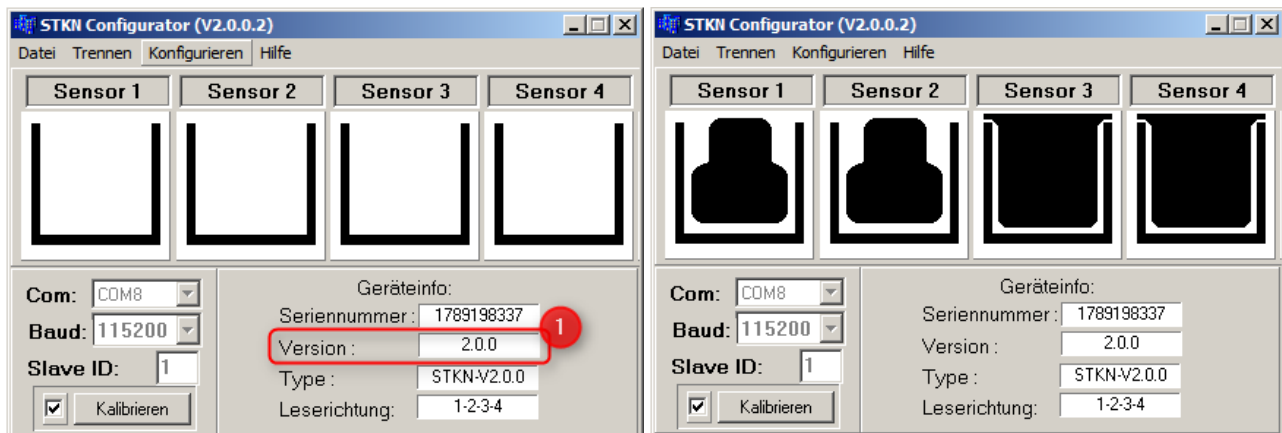
Der folgende Screenshot zeigt das Hauptfenster der STKN-SM-Config-Applikation nach dem Start:



Zum Herstellen einer Verbindung wie folgt vorgehen:

1. Auswählen der seriellen Schnittstelle (1). Falls die Nummer der seriellen Schnittstelle des Kabels nicht bekannt ist, dann kann diese über den Windows-Gerätemanager ausgelesen werden.
2. Auswahl der Slave-ID (3): Das Stecknussköchersystem adressiert die einzelnen Sensorelemente über eine eindeutige Adresse auf dem internen Bus. Die kleinste Nummer ist 1 und wählt das erste Sensormodul im System (die meisten Sensormodule enthalten 4 Sensoren, nur die MSTKN-**-T18-** enthalten 8 Sensoren).
3. Auswählen des Menüpunkts „Verbinden“ (2) zum Herstellen der Verbindung mit dem Stecknussköchersystem

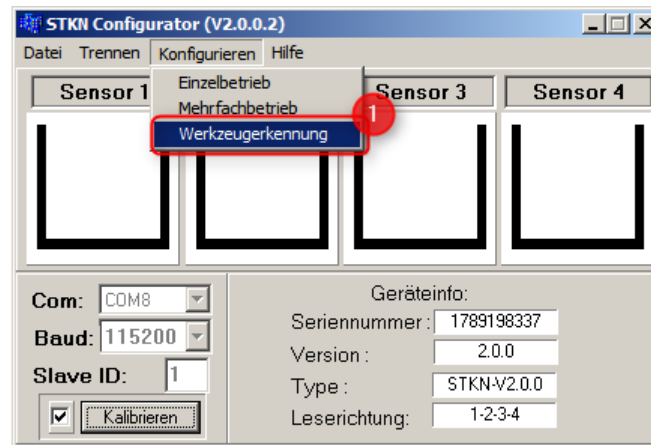
Nachdem eine Verbindung hergestellt ist, zeigt das Hauptfenster die aktuellen Informationen zum Stecknussköchersystem wie folgt (linkes Bild: keine Werkzeuge, rechtes Bild Werkzeuge in Platz 1/2, Platz 3/4 deaktiviert):



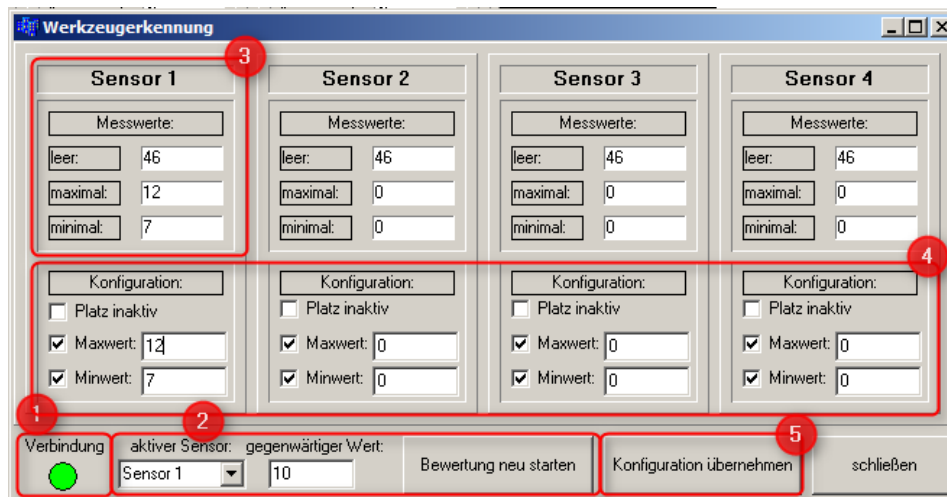
Wichtig: Werkzeugerkennung und Definieren der Werkzeugparameter sind erst ab der Firmware 2.0.0 verfügbar (siehe 1 im Screenshot oben)!

4.3.2 Übersicht Werkzeugparameter

Die Werkzeuigerkennung kann jetzt über das Menü Konfigurieren → Werkzeuigerkennung aufgerufen werden (① im Screenshot unten):



Nach Klicken des Menüpunktes wird das Fenster für die Werkzeugkonfiguration dargestellt:



Das Fenster zeigt die folgenden Informationen:

- Zustand der Verbindung mit dem Stecknussköchersystem (①): grün falls Verbindung OK, rot falls keine Verbindung vorhanden (dann muss das Fenster geschlossen werden und eine neue Verbindung aufgebaut werden, siehe oben)
- Aktiver Sensor (②): die Dropdown-Box erlaubt die Auswahl des aktiven Sensors für das Messen der Sensorparameter. Der aktuelle Messwert des Sensors wird im Feld „gegenwärtiger Wert“ angezeigt, die Maxima und Minima im Messwertefeld des ausgewählten Sensors (③). Der Maximal- und Minimalwert werden nur für den ausgewählten Sensor aktualisiert – zurückgesetzt werden diese durch Drücken des Buttons „Bewertung neu starten“.
- Eingabefelder für Werkzeuigerkennung (④): In diesem Bereich können die Grenzwerte für die Werkzeuigerkennung eingegeben werden. Die Maxima/Minima aus dem Messwertefeld (③) können als Eingabehilfe verwendet werden. Prinzipiell gibt es hier folgende Felder:
 - Platz inaktiv: Ist diese Checkbox gesetzt, dann wird der Platz als inaktiv markiert, d.h. die LEDs dieses Platzes werden ausgeschaltet und es wird erwartet, dass der Stopfen mit einem Kupferstreifen beklebt ist.
 - Maxwert/Minwert: Ist die zugehörige Checkbox aktiv, dann wird der im Eingabefeld eingegebene Wert verwendet, um zu prüfen, ob ein gültiges Werkzeug im Platz abgelegt ist.

- Sind beide Checkboxes bei Minwert und maxwert nicht aktiv, dann ist die Werkzeugprüfung abgeschaltet.
- Senden der Konfiguration (⑤): Durch Drücken des Buttons wird die Konfiguration (also alles, was im Eingabefeld für die Werkzeugerkennung (④) eingegeben ist) an den Stecknussköcher gesendet und dort im nichtflüchtigen Speicher abgelegt.



Wichtig: Bei der Werkzeugerkennung darf der Maxwert nicht größer als der Leerwert (Feld „leer“ Messwertefeld (③)) sein und darf nicht gleich groß wie der Minwert sein! Entsprechend darf der Minwert nicht größer als der Maxwert sein!

4.3.3 Parametrierung Werkzeugerkennung

Die Vorgehensweise zur Konfiguration der Werkzeugerkennung ist wie folgt:

1. Sensor auswählen: Mit Hilfe des Dropdowns „aktiver Sensor“ (②) wird der zu untersuchende Platz ausgewählt.
2. Einlegen des korrekten Werkzeugs in den zu konfigurierenden Platz.
3. Untersuchen der Werkzeugparameter:
 - a. Hierzu wird zuerst der Button „Bewertung neu starten“ (②) gedrückt – dies setzt die Min- und Maxwerte in der Messwertanzeige (③) zurück.
 - b. Dann wird das Werkzeug im Platz hin- und her bewegt, um die maximalen und minimalen Messwerte zu ermitteln. Die Messwerte werden in der Messwertanzeige (③) angezeigt (auf den korrekten Sensor bzw. Platz achten!).
4. Wiederholen der Untersuchung für alle weiteren Plätze/Sensoren: Bitte ab Schritt 1 (neuen Sensor wählen) den Prozess für alle weiteren Steckplätze wiederholen.
5. Bewerten der Ergebnisse: Nachdem alle Werkzeuge untersucht sind, muss geprüft werden, ob diese auch eindeutig erkannt werden können. Hierzu müssen die Min- und Maxwerte aller Werkzeuge (wie zuvor ermittelt) überlappungsfrei sein. Dies muss ggf. auch durch Wiederholen der Messung und Platzieren der Werkzeuge in den „falschen“ Plätzen gegengeprüft werden (da die Sensoren der verschiedenen Plätze u. U. leicht unterschiedliche Messwerte für das gleiche Werkzeug liefern – auf Grund der mechanischen Toleranzen der Bohrungen der Einsatzstopfen und der grundsätzlichen Abweichungen der Sensoren untereinander).



Wichtig: Falls sich die Werkzeuge nicht überlappungsfrei unterscheiden lassen, dann können die Werkzeuge nicht prozesssicher unterschieden werden – in diesem Fall muss die Anzahl der Werkzeuge reduziert werden oder durch mechanische Anpassungen (am Werkzeug oder dem Einsatzstopfen) eine überlappungsfreie Erkennung sichergestellt werden!

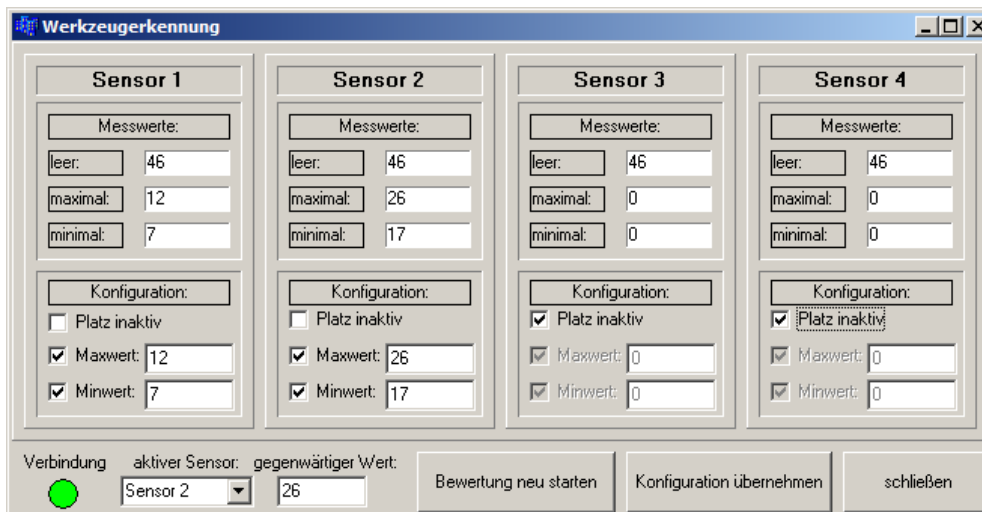
6. Übernehmen der Parameter: Nachdem die überlappungsfreien Bereiche für die Werkzeugerkennung festgelegt sind, können diese in die Eingabefelder der Werkzeugerkennung (④) übernommen werden. Um einen möglichst störungsfreien Betrieb zu erreichen, sollten die Bereiche gegenüber den gemessenen Max- und Minwerten etwas vergrößert werden (also z. B. konfigurierter Minwert zwei Punkte unterhalb des gemessenen Minwerts, ebenso zwei Punkte oberhalb für den Maxwert).



Wichtig: Damit die Werkzeugwerte für die Werkzeugüberwachung eingetragen werden können, müssen die Checkboxes vor den Eingabefeldern für Maxwert und Minwert gesetzt werden! Soll für einen Platz stattdessen keine Werkzeugüberprüfung durchgeführt werden, dann müssen die beiden Checkboxes abgeschaltet werden!

7. Senden der Konfigurierten Werte zu Stecknussköcher durch Drücken des Buttons „Konfiguration übernehmen“

Beispiel nach erfolgreicher Konfiguration:



Nachdem der Stecknussköcher für die Werkzeugererkennung konfiguriert ist, kann das Eingabefenster geschlossen werden (Button schließen).



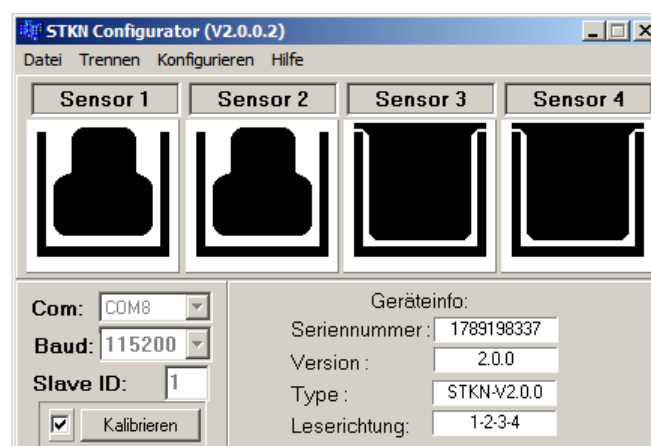
Wichtig: Die korrekte Funktion der Werkzeugererkennung muss erneut im Diagnosebetrieb (siehe 4.3.1) oder Normalbetrieb geprüft werden – bitte dabei auch prüfen, ob falsch zurückgelegte Werkzeuge auch korrekt als „falsch“ erkannt werden!

4.3.4 Prüfen der Werkzeugererkennung

Die korrekte Funktion der Werkzeugererkennung sollte im Diagnosebetrieb (siehe 4.3.1) oder Normalbetrieb geprüft werden – bitte sollte auch geprüft werden, ob falsch zurückgelegte Werkzeuge auch korrekt als „falsch“ erkannt werden!

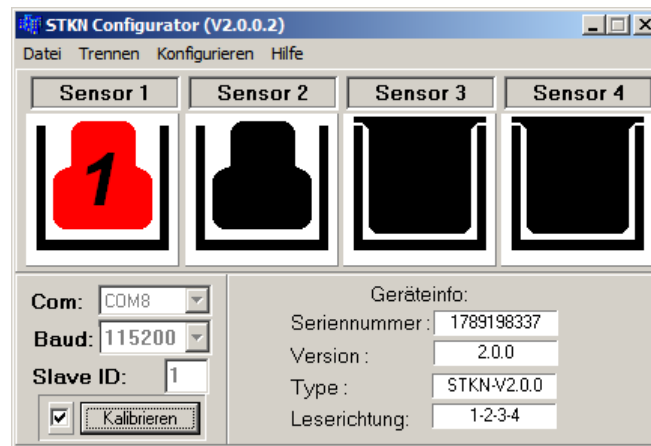
Im Normalbetrieb (Anschluss des Stecknussköchers an die Steuerung) kann die Funktion je nach Vorgabe des übergeordneten Systems geprüft werden.

Der Diagnosebetrieb (siehe 4.3.1) kann durch Start der STKN-SM-Config-Applikation gestartet werden. Die Anzeige sollte dann wie folgt aussehen (Beispiel):



Im gezeigten Fall sind die Werkzeuge im Platz 1 und 2 korrekt zurückgelegt, die Plätze 3 und 4 sind inaktiv. Bei Erkennung eines falschen Werkzeugs (außerhalb der definierten Bereiche für die Werkzeugererkennung) wird der Platz rot dargestellt.

Der folgende Screenshot zeigt, dass in Platz 1 ein falsches Werkzeug eingelegt ist:



4.4 Gruppierung parametrieren

Zur Aktivierung und Gruppierung mehrerer Steckplätze wird die Applikation MSTKN-HM-Config.exe verwendet.

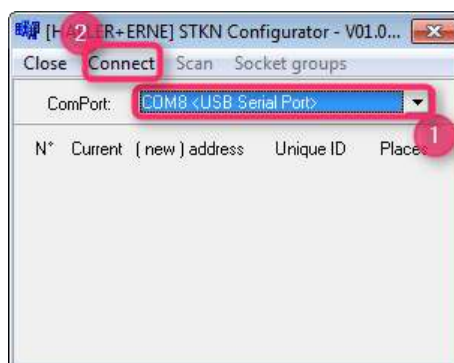
Die grundlegende Vorgehensweise ist wie folgt:

1. Verbindung mit dem Stecknussköcher über das USB M12-Kabel herstellen (siehe oben)
2. MSTKN-HM-Config.exe starten
3. Eine Verbindung mit dem Stecknussköchersystem herstellen
4. Den Konfigurationsbildschirm zum Gruppen-Konfigurator öffnen
5. Gruppierung parametrieren
6. Die geänderten Parameter an den Stecknussköcher senden
7. Ggf. Testen

Die folgenden Kapitel zeigen die Vorgehensweise.

4.4.1 Verbindung mit Stecknussköcher herstellen

Der folgende Screenshot zeigt das Hauptfenster der MSTKN-HM-Config-Applikation nach dem Start:



Zum Herstellen einer Verbindung wie folgt vorgehen:

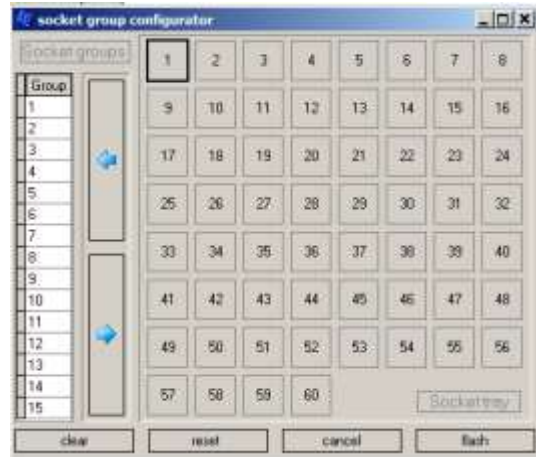
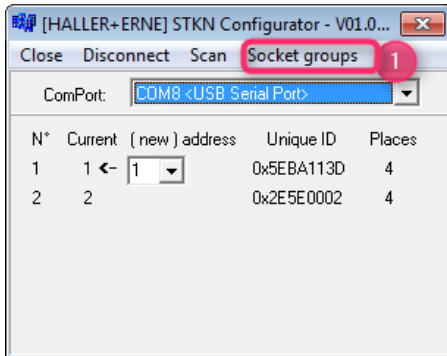
1. Auswählen der seriellen Schnittstelle ①. Falls die Nummer der seriellen Schnittstelle des Kabels nicht bekannt ist, kann diese über den Windows-Gerätmanager ausgelesen werden.
2. Auswählen des Menüpunkts „Connect/Verbinden“ (②) zum Herstellen der Verbindung mit dem Stecknussköchersystem. Nach Klicken der „Connect“-Taste wird die Verbindung hergestellt.



Wichtig: Werkzeugerkennung und Definieren der Werkzeugparameter sind erst ab der Firmware 1.6.7 verfügbar!

4.4.2 Gruppierung parametrieren

Die Gruppenparametrierung kann jetzt über das Menü „Socket groups“ aufgerufen werden (Siehe ❶ im Screenshot unten). Nach Anklicken des Menüpunktes wird das Fenster für die Gruppenparametrierung dargestellt (max. 60 Steckplätze):

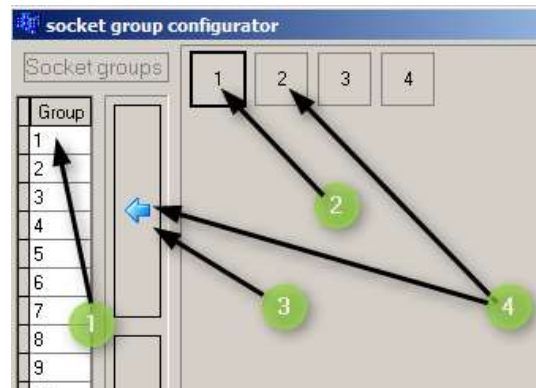


Die Vorgehensweise zum Herstellen der Standardkonfiguration (keine Gruppen) ist wie folgt:

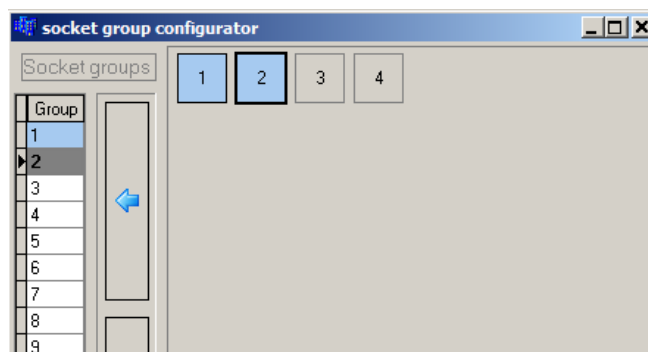
1. Mit der „clear“-Taste die vorherige Konfiguration löschen.
2. Mit der „flash“-Taste die Konfiguration an den Stecknussköcher senden.

Die Vorgehensweise zur Konfiguration ist wie folgt (siehe auch nebenstehenden Screenshot):

1. Gruppe auswählen (❶).
2. Steckplatz auswählen (❷).
3. Den ausgewählten Steckplatz zur ausgewählten Gruppe 1 zuzuordnen (❸).
4. Prozess für alle weiteren Steckplätze wiederholen (❹).
5. Senden der Konfigurierten Werte zu Stecknussköcher durch Drücken des Buttons „flash“



Das Bild unten zeigt den Stand, nachdem die erste Gruppe schon konfiguriert (Steckplätze 1 und 2) und die zweite Gruppe für die bevorstehende Konfiguration ausgewählt ist:



4.4.3 Gruppenparametrierung testen

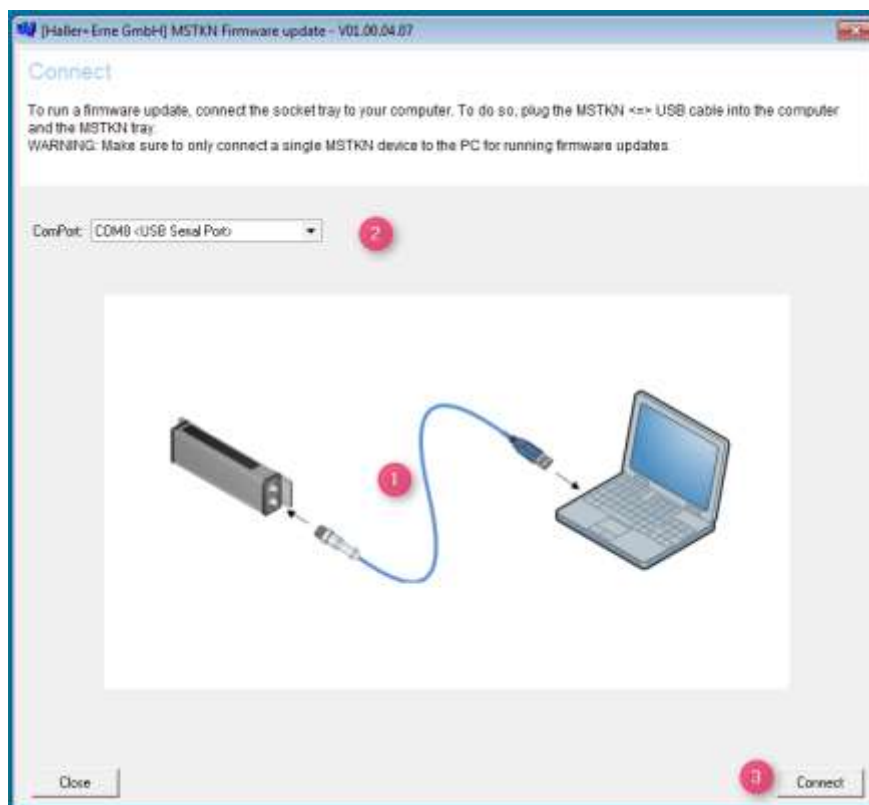
Die Gruppenparametrierung kann direkt wie folgt getestet werden:

- Wählen Sie mit der von Ihnen für die Stecknussköcher verwendeten Ansteuerung einen „Steckplatz“ an, also z.B. die Nummer 1. Dies wählt nun nicht mehr den Steckplatz „1“ an sondern die Gruppe „1“.
- Ist der Stecknussköcher wie oben im Beispiel parametrierung worden, müssten nun die Steckplätze 1 und 2 ausgewählt werden und anfangen zu blinken. Wählen sie „2“ an, sollten es die Plätze 3 und 4 sein.

4.5 Firmwareupdate durchführen

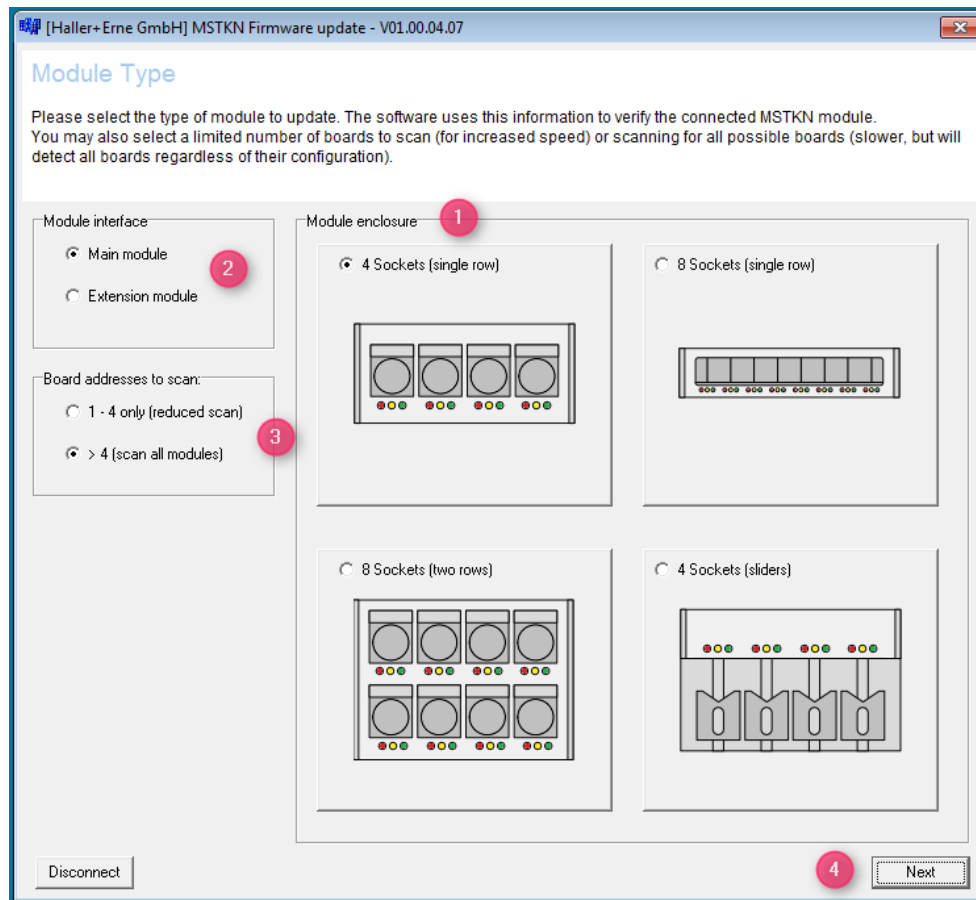
4.5.1 Verbindung mit dem Stecknussköchersystem herstellen

Bevor Sie anfangen können, starten Sie bitte die MSTKN Firmware Updater Software, die Sie mit Ihrem/Ihren Stecknussköcher(n) erhalten haben.



1. Verbindung mit dem (oder mehreren) Stecknussköcher(n) über das USB M12 Kabel herstellen.
2. Auswählen der seriellen Schnittstelle. Falls die Nummer der seriellen Schnittstelle des Kabels nicht bekannt ist, kann diese über den Windows-Gerätmanager ausgelesen werden.
3. Den Schalter „Connect“ drücken, um die Verbindung mit dem Firmware Updater herzustellen.

4.5.2 Stecknussköcherart wählen



Das obige Bild zeigt den Startbildschirm des Firmware Updaters, nachdem er mit dem/den MSTKN verbunden wurde. Wählen Sie nun den Typ den Stecknussköchers aus, den Sie aktualisieren möchten.

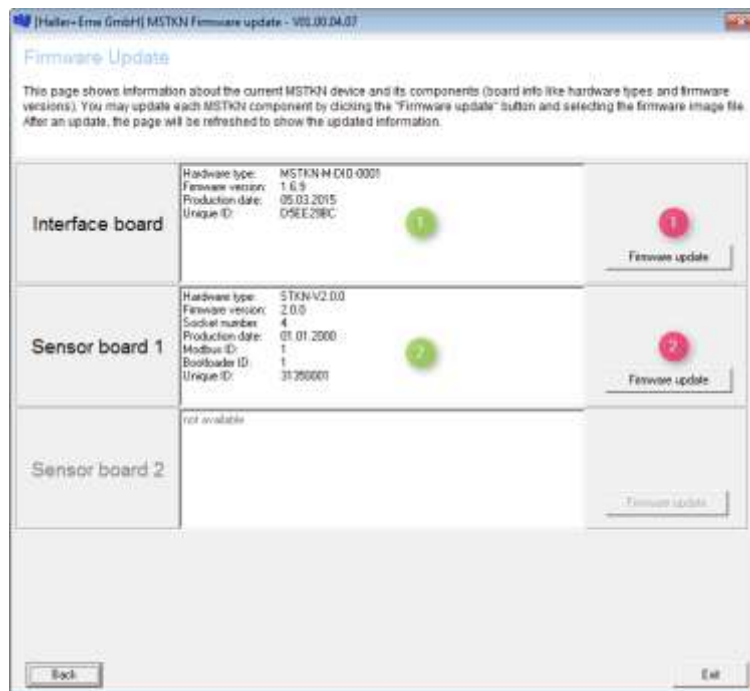
1. Wählen Sie die Gehäuseform aus.

4 Sockets (single row)	=	S14	(1 Sensorplatine)
8 Sockets (two rows)	=	S24	(2 Sensorplatinen)
8 Sockets (single row)	=	T18	(1 Sensorplatine)
4 Sockets (sliders)	=	P14	(1 Sensorplatine)

2. Wählen Sie, ob es sich um ein Hauptmodul (Main Module) oder ein Erweiterungsmodul (Extension Module) handelt.
3. Wählen Sie die Anzahl der Sensorplatinen aus. Hier empfiehlt es sich meistens, die Option „1 – 4 only (reduced scan)“ zu wählen, da die Suche nach allen Adressen sehr lange dauert. Wenn Sie z.B. ein Hauptmodul mit 2 Sensorplatinen, an das ein Erweiterungsmodul mit 2 Sensorplatinen angeschlossen ist, haben, haben die Sensorplatinen bei korrekter Parametrierung die Adressen 1, 2, 3 und 4, somit reicht noch ein „reduced scan“ aus, um alle verbundenen Platinen zu finden. Die Anzahl der Interface-Boards ist hierbei uninteressant, es geht nur um die Anzahl der Sensor-Boards.
4. Wenn Sie alle Optionen richtig ausgewählt haben, klicken Sie auf „Next“, um den Scan zu starten.

4.5.3 Firmwarepaket auswählen und Update durchführen

Der Firmware Updater hat nun den/die verbundenen MSTKN gescannt und alle Platinen mit der dazugehörigen, aktuell aufgespielten Software-Version aufgelistet.



Im Bild oben sieht man ein Beispiel, bei dem ein EA24 MSTKN mit einer Sensorplatine (4 Steckplätze) angeschlossen wurde.

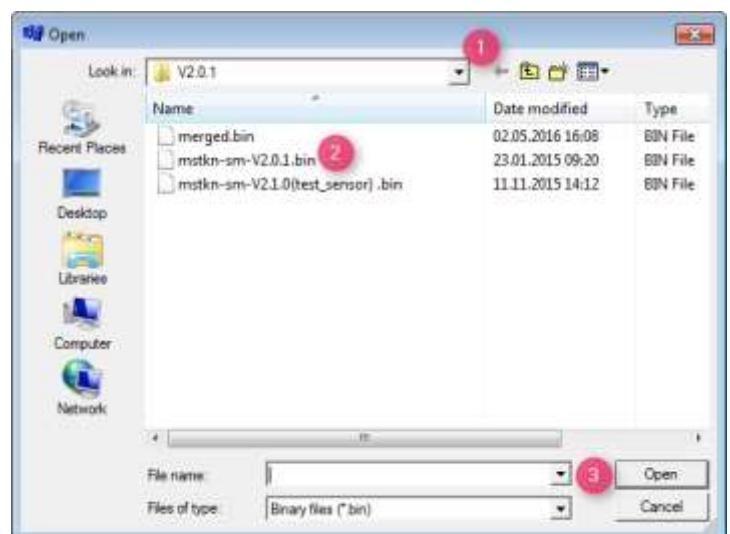
1. Zum Update des Gatewayelements („Interface board“) wählt man nun den Button „Firmware update“ **1** an.
2. Zum Update des Sensorelements („Sensor board 1“) wählt man nun den Button „Firmware update“ **2** an.

Für dieses Beispiel wird nun das Sensorelement aktualisiert.

Nun öffnet sich das Fenster, das nebenstehend dargestellt ist. Hier wird nun das benötigte Firmwarepaket ausgewählt, mit der das Update durchgeführt werden soll:

1. Im Dropdown-Menü wird der gewünschte Ordner mit den Firmwaredateien ausgewählt.
2. Firmwarepaket auswählen
1. Abschließend bestätigen Sie die richtige Datei mit „Open“

Der Firmware Updater beginnt nun mit dem Updateprozess.



Während das Update aktiv ist, wird eine Fortschrittsanzeige dargestellt. Ist das Firmwareupdate abgeschlossen, dann wird erneut der Status wie folgt dargestellt:

The screenshot shows a web browser window titled "[Haller+Erne GmbH] MSTKN Firmware update - V01.00.04.07". The page content is as follows:

Firmware Update

This page shows information about the current MSTKN device and its components (board info like hardware types and firmware versions). You may update each MSTKN component by clicking the "Firmware update" button and selecting the firmware image file. After an update, the page will be refreshed to show the updated information.

Interface board	Hardware type: MSTKN-M-DIO-0001 Firmware version: 1.6.9 Production date: 05.03.2015 Unique ID: D5EE29BC	2 Firmware update
Sensor board 1	Hardware type: STKN-V2.0.0 Firmware version: 2.0.1 Socket number: 4 Production date: 01.01.2000 Modbus ID: 1 Bootloader ID: 1 Unique ID: 31350001	✓ Firmware update
Sensor board 2	not available	Firmware update

At the bottom of the page, there are three buttons: "Back" (marked with a red circle containing the number 3), "Exit" (marked with a red circle containing the number 1), and a "Firmware update" button (marked with a red circle containing the number 2).

5 Mechanische Daten

5.1 Aufnahmeblöcke und Stopfen



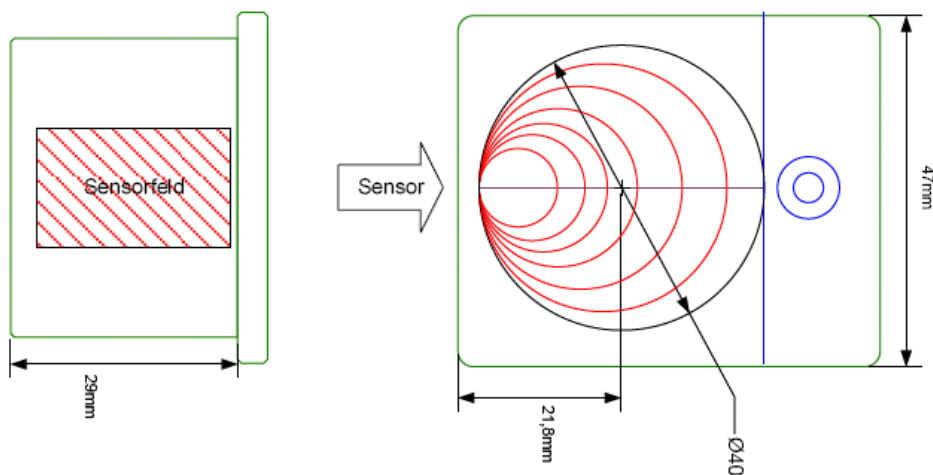
Wichtig: Um eine größtmögliche Empfindlichkeit der Sensoren zu gewährleisten, muss sichergestellt werden, dass die Bohrungen für die Werkzeugaufnahmen so nahe wie möglich am Sensor platziert werden. Einfräsungen auf der Oberseite der Stopfen bzw. Aufnahmeblöcke markieren die Mitte des Sensors und auch den optimalen Abstand zum Sensor.

Typische Maße:

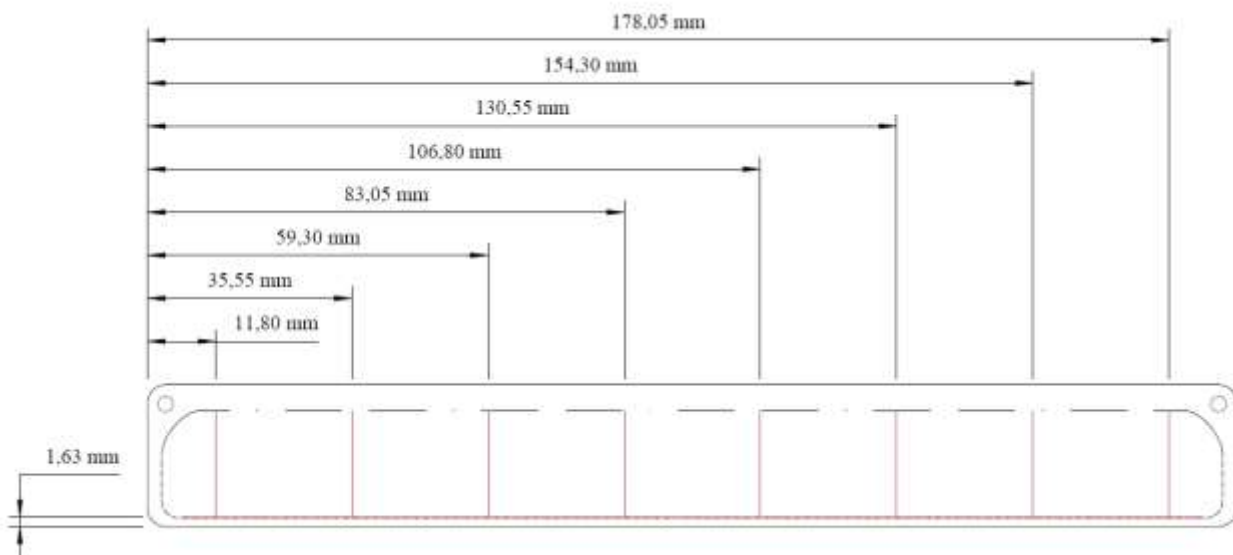
Bohrtiefe:	min. 12mm - max. 30mm (ab Oberkante Stopfen bzw. Block)
Sensorabstand:	max. 2mm

Sollte ein Werkzeug nicht sicher erkannt werden, kann dies durch einen kleineren Abstand zum Sensor und/oder eine tiefere Bohrung korrigiert werden.

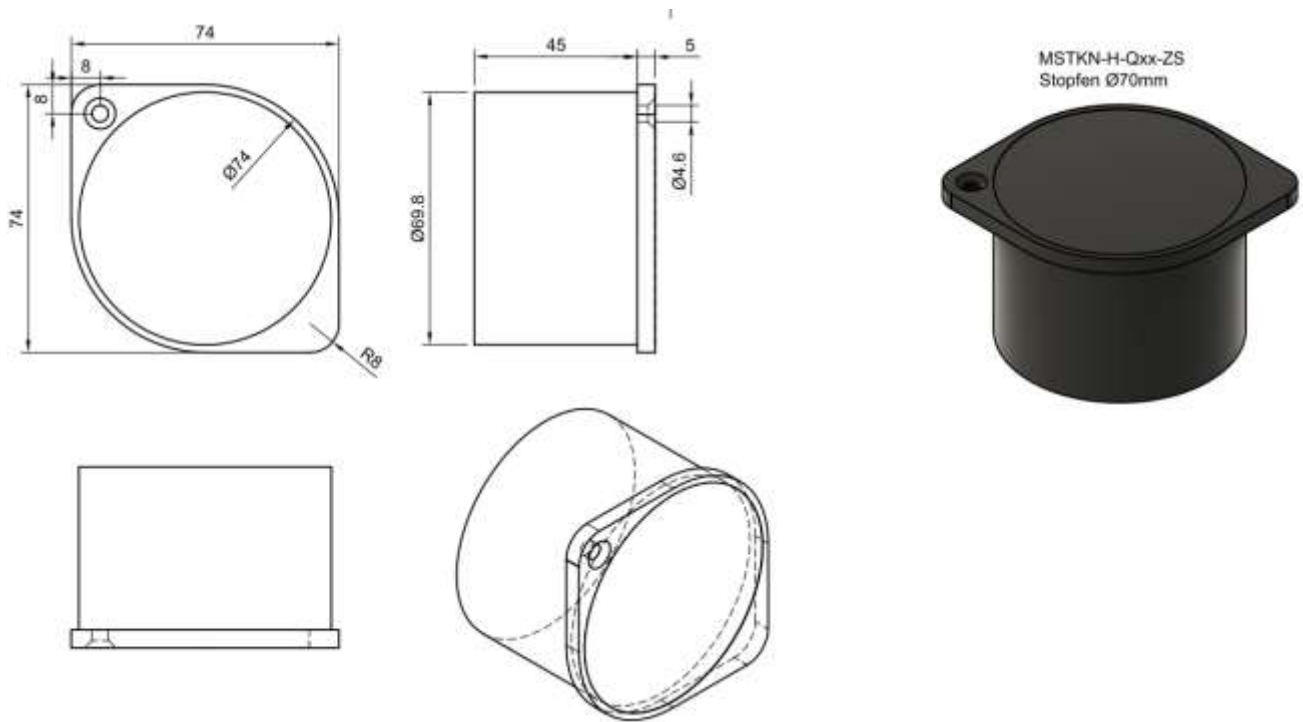
5.1.1 Maße für Stopfen S14 und S25



5.1.2 Maße für T18 Aufnahmeblock



5.1.3 Maße für Stopfen Qxx (70mm)



5.2 Gehäuseabmessungen

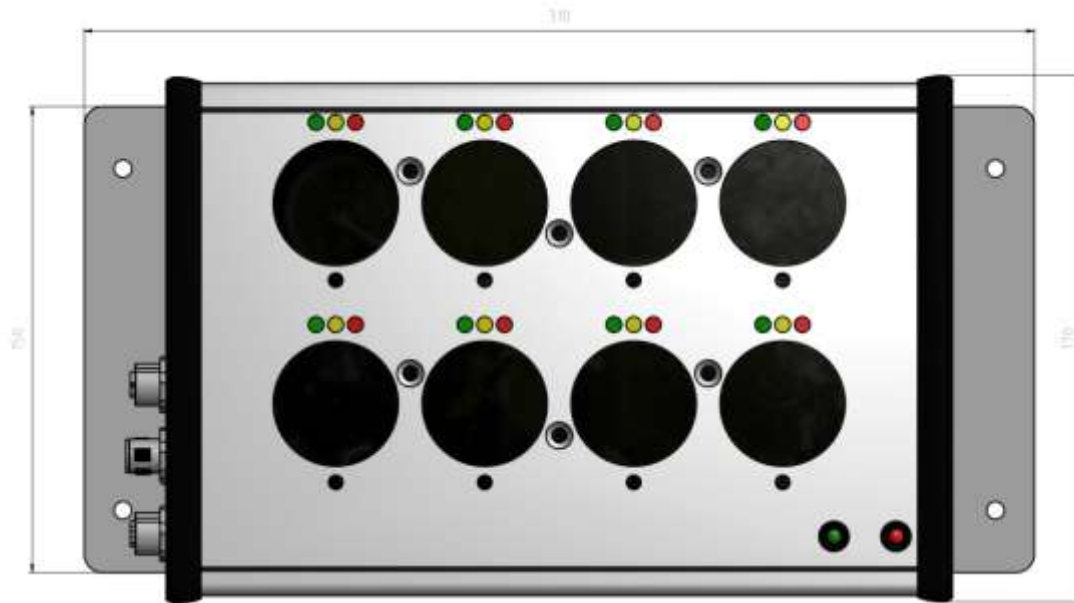
H-T18 Gehäuse



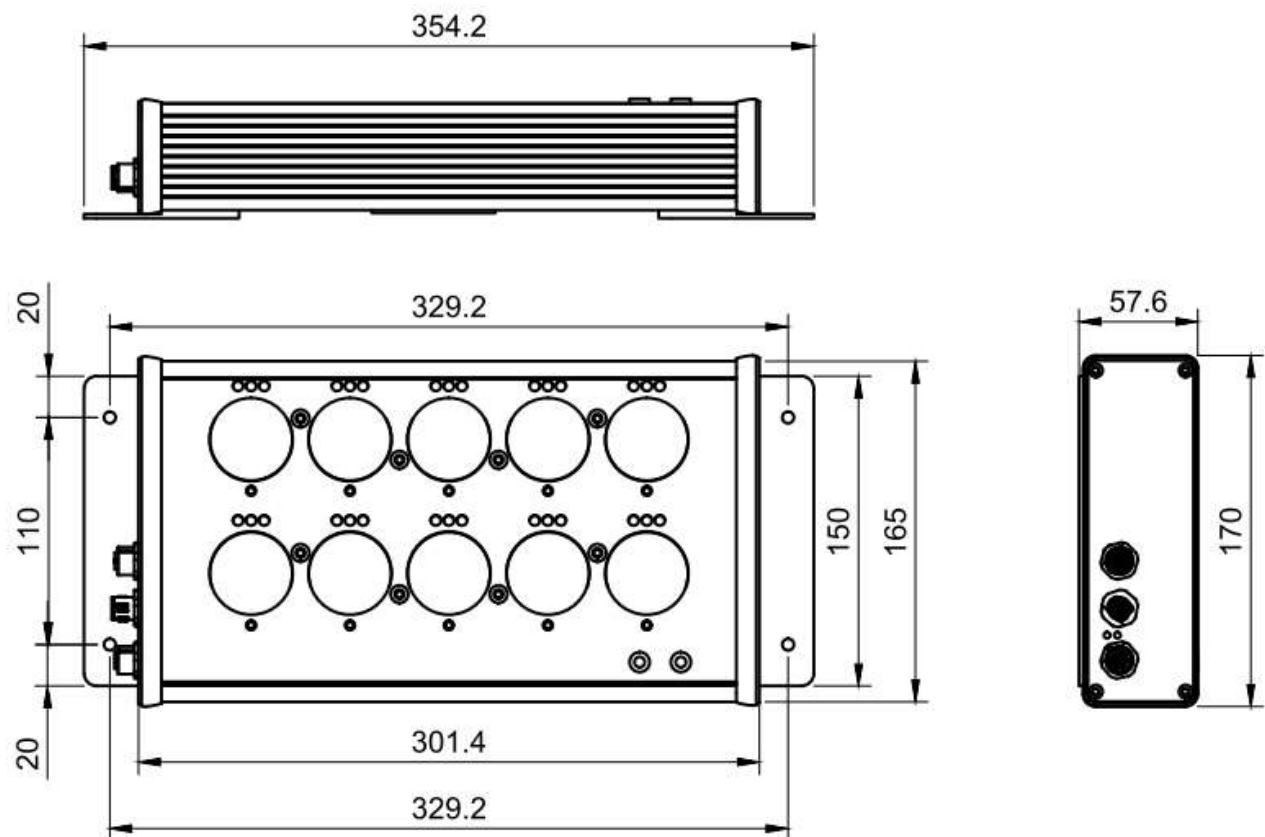
H-S14 Gehäuse



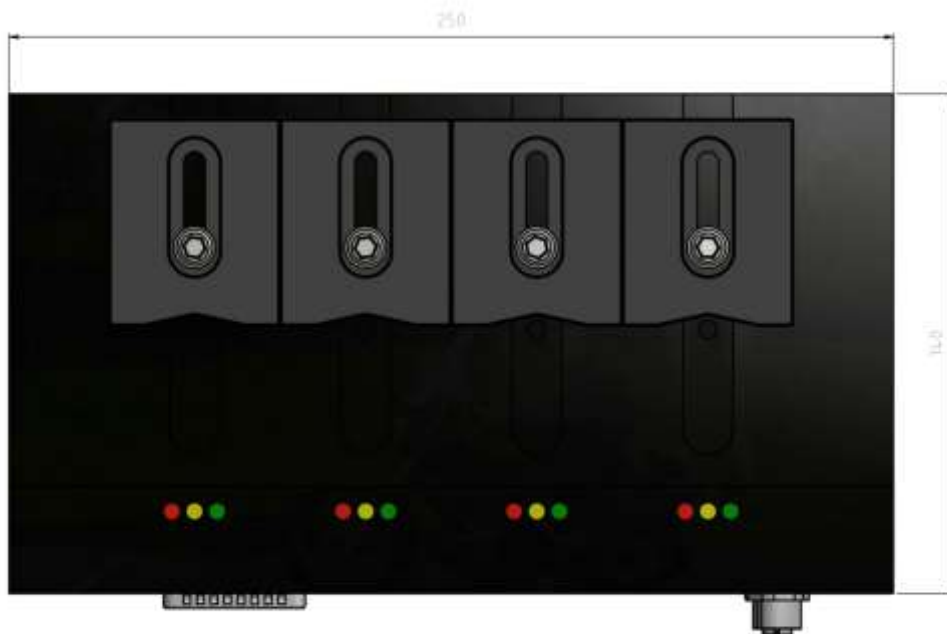
H-S24 Gehäuse:



H-S25 Gehäuse:



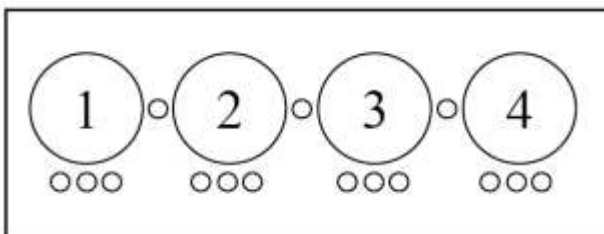
P-P14 Gehäuse:



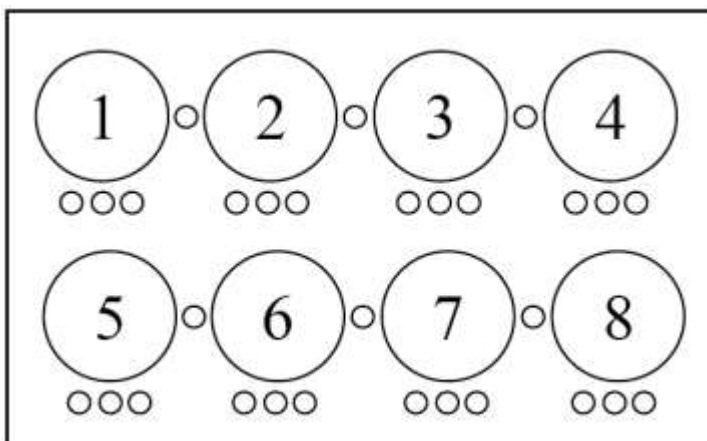
5.3 Nummerierung der Steckplätze

Die folgenden Darstellungen zeigen die Standard-Steckplatznummern – die Nummerierung kann per Software geändert werden (siehe Kapitel 4.4):

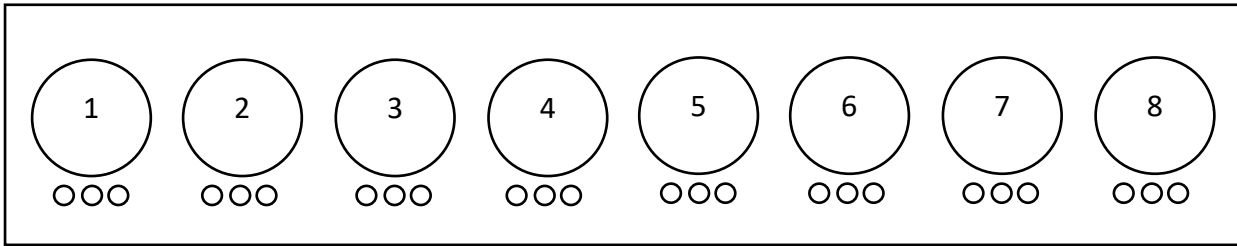
H-S14 / P-P14:



H-S24:



H-T18:



6 Konformitätserklärung

CE-Konformitätserklärung

CE-Conformity Declaration

Déclaration de conformité CE

Der Unterzeichner erklärt, dass die in der beiliegenden Artikelliste aufgeführten Stecknussköcher der EMV-Richtlinie 2014/30/EU und der Niederspannungsrichtlinie 2014/35/EU entsprechen

The undersigned declares, that the socket trays listed in the article list are conform to EMV guide-line 2014/30/EU and low-voltage-directive 2014/35/EU

Le signataire déclare que les boîtes à douilles énumérées sur le liste sont en la directive EMV 2014/30/EU et en la directive sur la basse tension 2014/35/EU

Produkttypen	Product types	Types de produits
MSTKN-H-S14-HM-DB25-M12		MSTKN-H-S24-HM-DB25-M12
MSTKN-H-S14-HM-DB25-M12-OP		MSTKN-H-S24-HM-DB25-M12-OP
MSTKN-H-S14-HM-DB25-M12-XML		MSTKN-H-S24-HM-DB25-M12-XML
MSTKN-H-S14-HM-PB-M12		MSTKN-H-S24-HM-PB-M12
MSTKN-H-S14-HM-PN-M12		MSTKN-H-S24-HM-PN-M12
MSTKN-H-S14-HM-PN2P-M12		MSTKN-H-S24-HM-PN2P-M12
MSTKN-H-S14-HM-ETH-M12		MSTKN-H-S24-HM-ETH-M12
MSTKN-H-S14-HM-WLAN-GBA		MSTKN-H-S24-HM-WLAN-GBA
MSTKN-H-S14-HM-WLAN-24V		MSTKN-H-S24-HM-WLAN-24V
MSTKN-H-S14-EM-M12		MSTKN-H-S24-EM-M12
MSTKN-H-S25-HM-DB25-M12		MSTKN-H-S25-HM-PN
MSTKN-H-S25-EM-M12		MSTKN-H-S25-HM-PN2P
MSTKN-H-Q13-ETH-M12		MSTKN-H-Q13-HM-WLAN-24V
MSTKN-H-Q12-ETH-M12		MSTKN-H-Q13-EM-M12
MSTKN-H-Q14-ETH-M12		MSTKN-H-Q12-EM-M12
MSTKN-H-Q15-ETH-M12		MSTKN-H-Q14-EM-M12
MSTKN-P-P14-HM-DB25-M12		MSTKN-H-Q15-EM-M12
MSTKN-P-P14-HM-PN-M12		MSTKN-P-P14-HM-DB25-M12-OP
MSTKN-P-P14-HM-PN2P-M12		MSTKN-P-P14-HM-ETH-M12
MSTKN-P-P14-EM-M12		MSTKN-H-T18-HM-DB25-M12
MSTKN-H-T18-HM-DB25-M12-OP		MSTKN-H-T18-HM-DB25-M12-XML
MSTKN-H-T18-HM-PB-M12		MSTKN-H-T18-HM-PN-M12
MSTKN-H-T18-HM-ETH-M12		MSTKN-H-T18-HM-PN2P-M12
		MSTKN-H-T18-EM-M12

Angewandte Normen:	Applied harmonized standards:	Normes appliquées en particulier:
EN 61000-6-4	EN 61000-6-2	EN 61000-4-3 2001-12
EN 61000-4-2: 1995 + A1:1998 + A2: 2001	EN 61000-4-3:2006 + A1: 2008	EN 60240-1

Nordheim, 01.01.2024



Holger Erne, Geschäftsführer

Haller + Erne GmbH

im Scholmental 4
D 74226 Nordheim
Telefon 07133 961230
www.haller-erne.de