



## **USBoard-USS4**

**Neobotix GmbH**

**29.04.2024**

|          |   |          |
|----------|---|----------|
| <b>1</b> | <b>Einleitung</b>                                   | <b>1</b> |
| 1.1      | Zu dieser Dokumentation . . . . .                   | 1        |
| 1.1.1    | Symbole und Konventionen . . . . .                  | 1        |
| 1.1.2    | Weiterführende Informationen . . . . .              | 2        |
| 1.2      | Rechtliche Anmerkungen . . . . .                    | 2        |
| 1.2.1    | Versionsangabe . . . . .                            | 2        |
| 1.2.2    | Haftung . . . . .                                   | 2        |
| 1.2.3    | Downloads und weitergehende Informationen . . . . . | 2        |
| <b>2</b> | <b>USBoard-USS4</b>                                 | <b>3</b> |
| 2.1      | Einleitung . . . . .                                | 4        |
| 2.1.1    | Bestimmungsgemäße Verwendung . . . . .              | 4        |
| 2.1.2    | Unzulässige Verwendung . . . . .                    | 4        |
| 2.1.3    | Sachkundiges Personal . . . . .                     | 4        |
| 2.2      | Funktionsweise . . . . .                            | 4        |
| 2.3      | Technische Daten . . . . .                          | 5        |
| 2.3.1    | USBoard-USS4 . . . . .                              | 5        |
| 2.3.2    | Ultraschallsensor Bosch Parkpilot USS4 . . . . .    | 5        |
| 2.4      | Gehäuse . . . . .                                   | 6        |
| 2.5      | Inbetriebnahme . . . . .                            | 6        |
| 2.6      | Parametersatz . . . . .                             | 6        |
| 2.7      | Befehlssatz . . . . .                               | 7        |
| 2.8      | CAN-Kommunikation . . . . .                         | 7        |
| 2.8.1    | Adressen . . . . .                                  | 7        |
| 2.8.2    | Befehle . . . . .                                   | 8        |
| 2.9      | RS-232-Kommunikation . . . . .                      | 11       |
| 2.10     | ROS-Node . . . . .                                  | 12       |
| 2.11     | Grafische Bedienoberfläche . . . . .                | 12       |
| 2.11.1   | Einleitung . . . . .                                | 12       |
| 2.11.2   | Erste Schritte . . . . .                            | 13       |
| 2.11.3   | Messwerte empfangen . . . . .                       | 13       |
| 2.11.4   | Verbindung beenden . . . . .                        | 13       |
| 2.11.5   | Das USBoard-USS4 konfigurieren . . . . .            | 14       |
| 2.12     | Abmessungen und Steckerbelegung . . . . .           | 17       |
| 2.12.1   | Abmessungen . . . . .                               | 17       |
| 2.12.2   | Steckerbelegung . . . . .                           | 18       |

|          |                                    |           |
|----------|------------------------------------|-----------|
| 2.13     | Zukaufteile . . . . .              | 21        |
| 2.13.1   | Steckverbinder . . . . .           | 21        |
| 2.13.2   | Konfigurationskabel . . . . .      | 21        |
| 2.13.3   | Sensorkabelsätze . . . . .         | 22        |
| 2.13.4   | Ultraschallsensor USS4 . . . . .   | 23        |
| 2.13.5   | Montagesets für USS4 . . . . .     | 24        |
| 2.14     | Rechtliche Anmerkungen . . . . .   | 24        |
| 2.14.1   | EG-Konformitätserklärung . . . . . | 24        |
| 2.14.2   | Informationen zu RoHS . . . . .    | 25        |
| <b>3</b> | <b>Steckverbinder</b>              | <b>26</b> |
| 3.1      | TE Connectivity - HE14 . . . . .   | 26        |
| 3.2      | Würth Elektronik - MPC4 . . . . .  | 27        |
| 3.3      | Würth Elektronik - MPC3 . . . . .  | 28        |
| <b>4</b> | <b>Sachkundiges Personal</b>       | <b>30</b> |

## 1.1 Zu dieser Dokumentation

### 1.1.1 Symbole und Konventionen

In dieser Dokumentation werden die folgenden Symbole und Hervorhebungen verwendet:

**Gefahr:** Weist auf eine Gefahrenstelle oder unmittelbar gefährliche Situation hin, die zu schweren Verletzungen bis hin zum Tod führen kann.

**Warnung:** Weist auf eine Gefahrenstelle oder potentiell gefährliche Situation hin, die zu schweren Verletzungen oder Schäden führen kann.

**Achtung:** Weist auf Gefahren oder Situationen hin, die zu leichten Verletzungen, Schäden oder anderen negativen Auswirkungen führen kann.

---

**Bemerkung:** Kennzeichnet wichtige Informationen, die für den sicheren Betrieb beachtet werden müssen.

---

---

**Tipp:** Gibt hilfreiche Hinweise, die die Arbeit mit dem mobilen Roboter einfacher und effizienter machen.

---

- Aufzählungen enthalten mehrere Informationen zum gleichen Thema.
- Sie sind, soweit möglich und sinnvoll, nach Priorität geordnet, mit dem jeweils wichtigsten Eintrag ganz oben.
- Aufzählungen erheben keinen Anspruch auf Vollständigkeit, sofern nichts anderes angegeben wurde.

1. Handlungsanleitungen sind nummeriert.
2. Nummerierte Anleitungen müssen in der gegebenen Reihenfolge befolgt werden.

### 1.1.2 Weiterführende Informationen

Weitere Informationen, insbesondere zu kundenspezifischen Konfigurationen und Systemen, erhalten Sie bei Auslieferung zusammen mit Ihrem Roboter oder [auf Anfrage](#)<sup>1</sup>. In den meisten Fällen stehen Ihnen alle Unterlagen zu Ihrem individuellen Roboter auch [im Kundenbereich](#)<sup>2</sup> unserer Website zu Verfügung.

## 1.2 Rechtliche Anmerkungen

### 1.2.1 Versionsangabe

Der deutsche Teil dieser Online-Dokumentation ist das Original.

### 1.2.2 Haftung

Dieses Dokument wurde mit größtmöglicher Sorgfalt verfasst und repräsentiert den Stand der Technik zum Zeitpunkt seiner Erstellung. Fehler und Irrtümer sind jedoch nicht auszuschließen. Bitte informieren Sie Neobotix, sollten Sie solche im Dokument bemerken.

Die Neobotix GmbH ist nicht haftbar für technische oder schriftliche Fehler in diesem Dokument und behält sich das Recht vor, Änderungen seines Inhalts vorzunehmen, ohne diese vorher anzukündigen. Neobotix übernimmt keinerlei Garantie für die in diesem Dokument beschriebenen Produkteigenschaften. Insbesondere ergibt sich aus dem Inhalt kein Anspruch jedweder Art, weder auf Eigenschaften des Produkts noch auf seine Eignung für spezielle Anwendungsfälle. Die Neobotix GmbH kann nicht für Schäden haftbar gemacht werden, die aus der unsachgemäßen Nutzung eines oder mehrerer der beschriebenen Produkte resultieren.

### 1.2.3 Downloads und weitergehende Informationen

Weitergehende Informationen, Datenblätter und Dokumentationen, auch von weiteren Neobotix-Produkten, finden Sie im Downloadbereich unserer Website: <https://www.neobotix-roboter.de/service/downloads>.

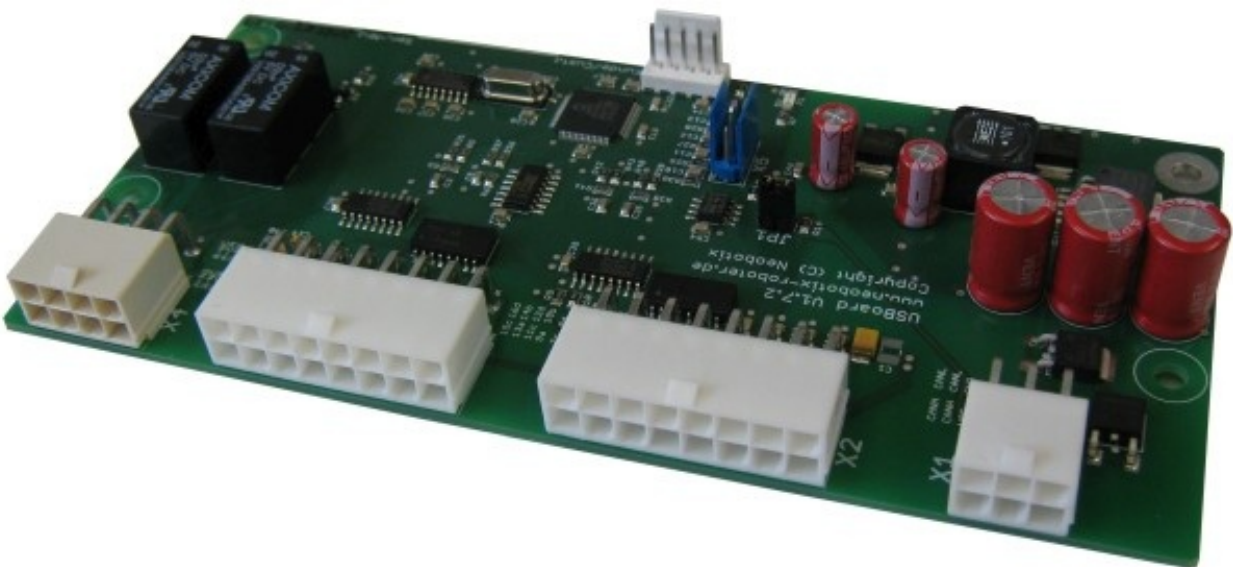
---

<sup>1</sup> <https://www.neobotix-roboter.de/kontakt/daten>

<sup>2</sup> <https://www.neobotix-roboter.de/login>

### USBoard-USS4

↓ Als PDF herunterladen<sup>3</sup>



Ultraschallsensoren messen die Entfernung zu Objekten, die sich innerhalb des Detektionsbereichs des Sensors befinden. Das physikalische Messprinzip beruht auf der Laufzeitmessung des Schalls, der vom Sensor ausgesandt, vom Hindernis reflektiert und vom Sensor wieder aufgefangen wird.

Das USBoard-USS4 ist für den Anschluss von bis zu 16 Ultraschallsensoren vom Typ Bosch Parkpilot USS4 (auch als URF6 oder URF7 bekannt) ausgelegt. Es eignet sich beispielsweise für die Kollisionsabsicherung von großen Fahrzeugen wie Bussen, Landmaschinen oder Baumaschinen. Das USBoard-USS4 bietet einfache Inbetriebnahme, komfortable Parametrierung und vielfältige Anwenderschnittstellen inklusive der Definition von Warn- und Alarmbereichen, die eine einfache Überwachung der Kollisionsgefahr ermöglichen.

<sup>3</sup> <https://neobotix-docs.de/hardware/de/USBoard-USS4.pdf>

Weiterhin stehen auf dem Board vier Analogeingänge zur Verfügung, die beispielsweise für den Anschluss von zusätzlichen Sensoren genutzt werden können.

---

**Tipp:** Die graphische Benutzeroberfläche für das USBoard-USS4 finden Sie unter *Grafische Bedienoberfläche* (Seite 12).

---

## 2.1 Einleitung

### 2.1.1 Bestimmungsgemäße Verwendung

Das USBoard-USS4 wurde für den Einsatz in mobilen Robotern und ähnlichen Geräten und Maschinen entwickelt, beispielsweise fahrenden Drohnen oder fahrerlosen Transportfahrzeugen. Es kann dort zur Erkennung von Hindernissen und zur Kollisionsvermeidung eingesetzt werden. Darüber hinaus kann das USBoard-USS4 auch in stationären Aufbauten oder Maschinen eingesetzt werden, um etwa Messungen vorzunehmen.

Das USBoard-USS4 ist ausschließlich für die Bereitstellung nicht-sicherer Daten und Informationen entwickelt worden.

### 2.1.2 Unzulässige Verwendung

Das USBoard-USS4 ist explizit **kein Sicherheitsbauteil** und darf nicht als solches verwendet werden. Insbesondere darf es nicht zur Absicherung von Gefahrenstellen oder als alleiniges System zur Kollisionsvermeidung eingesetzt werden.

Das USBoard-USS4 ist nur zur Verwendung innerhalb der angegebenen *Umgebungsbedingungen* (Seite 5) freigegeben. Ein Einsatz außerhalb dieser Bedingungen kann zu fehlerhaften Messungen oder zum Versagen des Produkts führen und ist deshalb unzulässig.

Ein Einsatz des USBoard-USS4 ist nicht empfehlenswert in Anwendungen, die sehr hohe Anforderungen an die Messgenauigkeit, an die Zuverlässigkeit der Messungen oder an die Geschwindigkeit der Messungen stellen. Dies sind zum Beispiel Anwendungen im Bereich Bewegungsregelung (Parken, Rangieren etc.).

### 2.1.3 Sachkundiges Personal

Das USBoard-USS4 darf nur von *sachkundigem Personal* (Seite 30) montiert, angeschlossen und in Betrieb genommen werden.

## 2.2 Funktionsweise

Das USBoard-USS4 misst die Laufzeit des Schalls vom Sensor zum nächstgelegenen Hindernis. Aus der Laufzeit wird dann die Entfernung zum Hindernis berechnet.

Die Sensoren werden nacheinander zyklisch abgefeuert und ausgewertet, um Fehlmessungen durch Kreuzechos zu minimieren. Dabei ist die Reihenfolge 1, 5, 9, 13, 2, 6, 10, 14, 3, 7, 11, 15, 4, 8, 12, 16.

Die gemessenen Entfernungen können wahlweise zyklisch oder asynchron über die CAN- und / oder RS-232-Schnittstelle abgefragt werden.

Zur einfachen Verwendung als Kollisionsschutz können für jeden Sensor ein Warnbereich und ein Alarmbereich definiert werden. Wird bei mindestens einem Sensor die Warndistanz unterschritten, werden die entsprechende LED auf

dem Board und ein Relaisausgang aktiviert. Eine zweite LED und ein weiterer Relaisausgang ermöglichen die Reaktion auf das Unterschreiten der Alarmdistanz. Warn- und Alarmbereiche werden komfortabel über den graphischen Parameter-Editor definiert und als Teil des Parametersatzes übertragen.

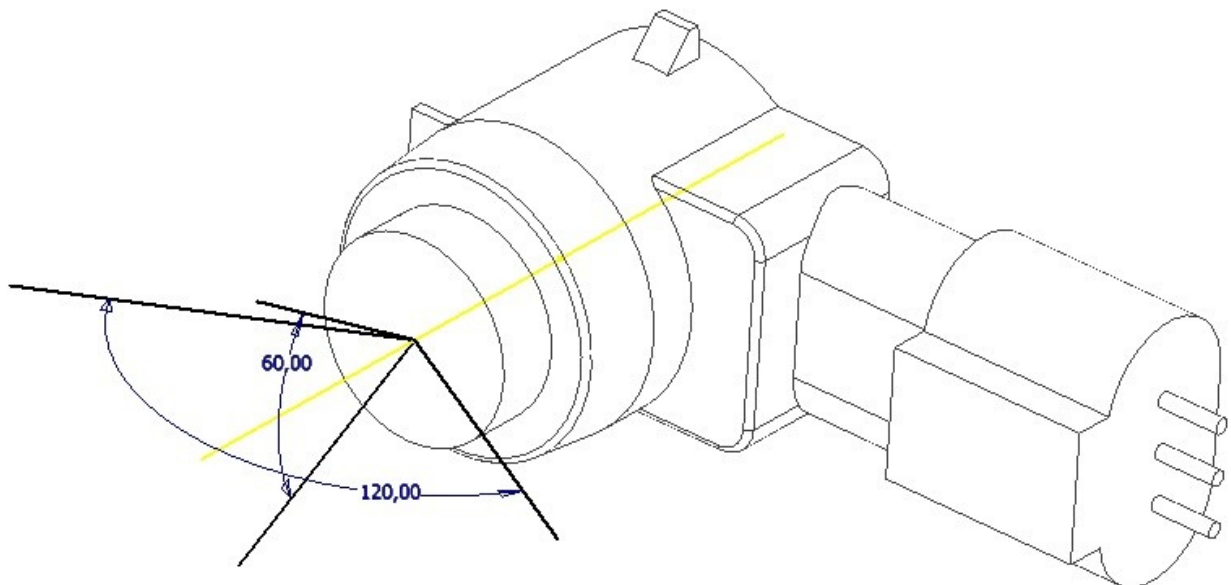
**Warnung:** Das USBoard-USS4 ist jedoch kein Sicherheitssystem und kann ausschließlich unterstützende, nicht sichere Daten liefern. Verwenden Sie das USBoard-USS4 niemals zur Absicherung von Gefahrenstellen.

## 2.3 Technische Daten

### 2.3.1 USBoard-USS4

- Versorgungsspannung von +9 VDC bis +60 VDC, max. 5 W, verpolsicher
- Digitale Kommunikationsschnittstellen CAN und RS-232
- Optische Anzeige einer Warn- / Alarmbereichsverletzung
- Relaisausgänge zur Abfrage von Warn- / Alarmbereichsverletzungen
- Vier Analogeingänge (0 V – 5 V)
- Gewicht: 93 g
- Lagertemperaturbereich: -40 °C – 85 °C
- Betriebstemperaturbereich: 0 °C – 70 °C
- Zolltarifnummer: 9031 9000
- Bestellnummer: X100

### 2.3.2 Ultraschallsensor Bosch Parkpilot USS4



- Messfrequenz: 42 kHz – 45 kHz



- Messentfernung: 0.15 m bis 1.5 m
- Zollltarifnummer: 9031 8020
- Bestellnummer: X200

## 2.4 Gehäuse

Standardmäßig wird das USBoard-USS4 als OEM-Komponente ohne Gehäuse geliefert. Bei Bedarf kann ein entsprechendes Gehäuse geliefert werden.

## 2.5 Inbetriebnahme

Das USBoard-USS4 wird mit den unter *Parametersatz* (Seite 6) zu findenden Voreinstellungen ausgeliefert und ist sofort betriebsbereit. Die individuelle Konfiguration kann ab Werk nach Absprache mit dem Kunden oder vom Kunden selbst mit dem mitgelieferten Parameter-Editor über die RS-232-Schnittstelle erfolgen. Nach der Übertragung des Parametersatz vom Editor zum USBoard-USS4 werden die neuen Einstellungen sofort aktiviert und wahlweise in den nicht-flüchtigen EEPROM-Speicher geschrieben.

Das USBoard-USS4 kann auf mehrere Arten verwendet werden:

1. Die LED-Anzeigen auf dem Board werden für die Überwachung von Warn- und Alarmbereichen verwendet.
2. Die Relaisausgänge von Warn- und Alarmbereich werden an geeignete Signalgeber (Licht, Tongeber) angeschlossen und für die Überwachung der Felder verwendet.
3. Über die CAN- oder RS-232-Schnittstelle werden die Messwerte abgefragt und in einem externen Rechner ausgewertet.
4. Eine Kombination der beschriebenen Möglichkeiten.

## 2.6 Parametersatz

Das USBoard-USS4 besitzt einen Parametersatz, mit dem eine individuelle Konfiguration möglich ist. Der Parametersatz kann komfortabel mit dem mitgelieferten Parameter-Editor editiert, auf der Festplatte des Konfigurationsrechners gespeichert, in das EEPROM des USBoard-USS4 geschrieben und aus dem EEPROM gelesen werden.

Die folgende Tabelle zeigt den Aufbau des Parametersatzes, der aus 54 Bytes besteht.

**Byte 1** CAN-Datenrate

|   |                               |
|---|-------------------------------|
| 0 | 1000 kBaud (Werkseinstellung) |
| 1 | 500 kBaud                     |
| 2 | 250 kBaud                     |
| 3 | 125 kBaud                     |
| 4 | 100 kBaud                     |
| 5 | 50 kBaud                      |

**Bytes 2-5** CAN Basisadresse (standardmäßig 0x400), zu berechnen wie folgt:

```
(byte_5 << 24) | (byte_4 << 16) | (byte_3 << 8) | byte_2
```

**Byte 6** 1 falls CAN extended ID genutzt wird, 0 wenn nicht (Standard)

**Byte 7** Sendemodus für Messwerte

|   |  |
|---|--|
| 0 | Senden der Messwerte nur auf Anforderung (Werkseinstellung)  |
| 1 | kontinuierliches Senden ohne Anforderung über CAN            |
| 2 | kontinuierliches Senden ohne Anforderung über RS-232         |
| 3 | kontinuierliches Senden ohne Anforderung über CAN und RS-232 |

**Byte 8** Zeitintervall für kontinuierliches Senden

|   |       |
|---|-------|
| 0 | 0.5 s |
| 1 | 1.0 s |
| 2 | 2.0 s |
| 3 | 0.2 s |

**Bytes 9-10** Zeigt aktive Sensoren an (ein Bit pro Sensor, 1 für aktiv, 0 für inaktiv), standardmäßig sind alle Sensoren aktiv

**Bytes 11-26** Warndistanz Sensoren 1 – 16, in cm; Werkseinstellung: 100 cm

**Bytes 27-42** Alarmdistanz Sensoren 1– 16, in cm; Werkseinstellung: 30 cm

**Bytes 43-51** Beliebiger Wert, nicht mehr verwendet

**Bytes 52-54** Seriennummer, nur lesen

## 2.7 Befehlssatz

Die folgende Tabelle listet die vom USBoard-USS4 unterstützten Befehle auf.

| Kommando                    | Wert | Beschreibung  |
|-----------------------------|------|---|
| CMD_CONNECT                 | 0    | Verbindungstest ausführen                           |
| CMD_SET_CHANNEL_ACTIVE      | 1    | Kanal für Senden/ Empfangen aktivieren              |
| CMD_GET_DATA_1TO8           | 2    | Messdaten Sensoren 1 bis 8 anfordern                |
| CMD_GET_DATA_9TO16          | 3    | Messdaten Sensoren 9 bis 16 anfordern               |
| CMD_WRITE_PARASET           | 4    | Parametersatz zum Board übertragen (flüchtig)       |
| CMD_WRITE_PARASET_TO_EEPROM | 5    | Parametersatz ins EEPROM schreiben (nicht-flüchtig) |
| CMD_READ_PARASET            | 6    | Parametersatz vom Board lesen                       |
| CMD_GET_ANALOGIN            | 7    | Werte der 4 Analogeingänge lesen                    |
| CMD_SET_DEBUG_PARA          | 8    | Nur für Testzwecke                                  |
| CMD_GET_DEBUG_PARA          | 9    | Nur für Testzwecke                                  |
| CMD_UNKNOWN                 | 10   | Nur für Testzwecke                                  |

## 2.8 CAN-Kommunikation

Den Befehlssatz finden Sie unter *Befehlssatz* (Seite 7).

### 2.8.1 Adressen

Die Werksvoreinstellung für die Basisadresse lautet 0x400. Diese Basisadresse kann bei Bedarf innerhalb des Parametersatzes geändert werden.

Die vom USBoard-USS4 benutzten Adressen werden von der Basisadresse berechnet, indem jeweils die folgenden Offsets addiert werden.

| Offset zur Basisadresse | Nachricht                               |
|-------------------------|---|
| +0                      | Empfangen von Befehlen                  |
| +1                      | Antwort auf CMD_CONNECT                 |
| +2                      | Erste Antwort auf CMD_GET_DATA_1TO8     |
| +3                      | Zweite Antwort auf CMD_GET_DATA_1TO8    |
| +4                      | Erste Antwort auf CMD_GET_DATA_9TO16    |
| +5                      | Zweite Antwort auf CMD_GET_DATA_9TO16   |
| +6                      | Antwort auf CMD_READ_PARASET            |
| +7                      | Antwort auf CMD_GET_ANALOGIN            |
| +8                      | Antwort auf CMD_WRITE_PARASET           |
| +9                      | Antwort auf CMD_WRITE_PARASET_TO_EEPROM |

## 2.8.2 Befehle

Im Folgenden werden die CAN IDs als Offset zur Basisadresse angegeben, d.h. +3 bedeutet Basisadresse plus 3.

### 2.8.2.1 CMD\_CONNECT

Mit diesem Kommando kann die Kommunikationsverbindung zum Board hergestellt und getestet werden.

**Kommando** ID: +0

|             |   |   |   |   |   |   |   |
|-------------|---|---|---|---|---|---|---|
| CMD_CONNECT | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
|-------------|---|---|---|---|---|---|---|

**Antwort** ID: +1

|             |   |   |   |   |   |   |   |
|-------------|---|---|---|---|---|---|---|
| CMD_CONNECT | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|-------------|---|---|---|---|---|---|---|

### 2.8.2.2 CMD\_SET\_CHANNEL\_ACTIVE

Mit diesem Kommando werden aus den 16 verfügbaren Kanälen die Kanäle ausgewählt, auf denen gesendet und empfangen werden soll. Zwei Bytes D1 und D2 für die Kanäle 1 bis 8 und 9 bis 16 enthalten die Information, ob der Kanal entsprechend der Bitnummer aktiv sein soll. Die Bytes sind Bit-codiert, wobei eine 1 einen aktiven Kanal markiert. Der Befehl kann verwendet werden, wenn Sensoren aktiv oder passiv geschaltet werden sollen, ohne den vollständigen Parametersatz zu übertragen.

Beispiel:  $0 \times 1F$  als erstes Byte bedeutet Sensoren 1 bis 5 sind aktiv, Sensoren 6 bis 8 sind deaktiviert.

**Kommando** ID: +0

|                        |                    |                     |   |   |   |   |   |
|------------------------|--------------------|---------------------|---|---|---|---|---|
| CMD_SET_CHANNEL_ACTIVE | (Sensoren 1 bis 8) | (Sensoren 9 bis 16) | 0 | 0 | 0 | 0 | 0 |
|------------------------|--------------------|---------------------|---|---|---|---|---|

**Antwort** Keine Antwort.

### 2.8.2.3 CMD\_GET\_DATA\_1TO8

Mit diesem Kommando werden die Messwerte der Sensoren 1 bis 8 abgefragt.

**Kommando** ID: +0

|                   |   |   |   |   |   |   |   |
|-------------------|---|---|---|---|---|---|---|
| CMD_GET_DATA_1TO8 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
|-------------------|---|---|---|---|---|---|---|

**Antwort (zwei Teile)** IDs: +2, +3

|                   |   |   |   |              |
|-------------------|---|---|---|--------------|
| CMD_GET_DATA_1TO8 | 0 | Werte für Sensoren 1-4 (jeweils ein Byte) | 0 | (reserviert) |
| CMD_GET_DATA_1TO8 | 1 | Werte für Sensoren 5-8 (jeweils ein Byte) | 0 | (reserviert) |

Die Messwerte sind in cm.

### 2.8.2.4 CMD\_GET\_DATA\_9TO16

Mit diesem Kommando werden die Messwerte der Sensoren 9 bis 16 abgefragt.

**Kommando** ID: +0

|                    |   |   |   |   |   |   |   |
|--------------------|---|---|---|---|---|---|---|
| CMD_GET_DATA_9TO16 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
|--------------------|---|---|---|---|---|---|---|

**Antwort (zwei Teile)** IDs: +4, +5

|                    |   |   |   |              |
|--------------------|---|---|---|--------------|
| CMD_GET_DATA_9TO16 | 0 | Werte für Sensoren 9-12 (jeweils ein Byte)  | 0 | (reserviert) |
| CMD_GET_DATA_9TO16 | 1 | Werte für Sensoren 13-16 (jeweils ein Byte) | 0 | (reserviert) |

Die Messwerte sind in cm.

### 2.8.2.5 CMD\_WRITE\_PARASET

Mit diesem Kommando wird ein vollständiger Parametersatz an das USBoard-USS4 übermittelt (flüchtig, nach dem Ausschalten verloren). Der Parametersatz kann komfortabel mit dem mitgelieferten Parameter-Editor erstellt und übertragen werden.

Die neuen Parameter werden nach dem Schreiben sofort aktiviert.

**Kommando** ID: +0

Der Befehl besteht aus neun Nachrichten, die nacheinander gesendet werden, wobei jede Nachricht ein paar Bytes des Parametersatzes enthält.

|                   |   |                 |
|-------------------|---|-----------------|
| CMD_WRITE_PARASET | 0 | Bytes 1 bis 6   |
| CMD_WRITE_PARASET | 1 | Bytes 7 bis 12  |
| CMD_WRITE_PARASET | 2 | Bytes 13 bis 18 |
| CMD_WRITE_PARASET | 3 | Bytes 19 bis 24 |
| CMD_WRITE_PARASET | 4 | Bytes 25 bis 30 |
| CMD_WRITE_PARASET | 5 | Bytes 31 bis 36 |
| CMD_WRITE_PARASET | 6 | Bytes 37 bis 42 |
| CMD_WRITE_PARASET | 7 | Bytes 43 bis 48 |
| CMD_WRITE_PARASET | 8 | Bytes 49 bis 54 |

**Antwort** ID: +8

Jede Nachricht wird einzeln beantwortet, wobei die ersten 8 Antworten identisch sind

|                   |   |   |   |   |   |   |   |
|-------------------|---|---|---|---|---|---|---|
| CMD_WRITE_PARASET | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
|-------------------|---|---|---|---|---|---|---|

und die letzte Antwort die Summe aller Bytes des Parametersatzes enthält.

|                   |          |           |   |   |   |   |   |
|-------------------|----------|-----------|---|---|---|---|---|
| CMD_WRITE_PARASET | low byte | high byte | 0 | 0 | 0 | 0 | 0 |
|-------------------|----------|-----------|---|---|---|---|---|

### 2.8.2.6 CMD\_WRITE\_PARASET\_TO\_EEPROM

Mit diesem Kommando wird ein vollständiger Parametersatz in das EEPROM des USBoard-USS4 geschrieben (nicht flüchtig). Ansonsten ist der Befehl mit CMD\_WRITE\_PARASET identisch.

Der Parametersatz wird nach dem Schreiben sofort übernommen.

**Kommando** ID: +0

Dieser Befehl verhält sich identisch zu *CMD\_WRITE\_PARASET* (Seite 9), nur dass er das Befehls-Byte CMD\_WRITE\_PARASET\_TO\_EEPROM benutzt.

**Antwort** ID: +9

Die Antwort ist identisch zu den Antworten auf *CMD\_WRITE\_PARASET* (Seite 9), nur dass sie das Befehls-Byte CMD\_WRITE\_PARASETE\_TO\_EEPROM benutzt.

### 2.8.2.7 CMD\_READ\_PARASET

Mit diesem Kommando wird der aktuelle, vollständige Parametersatz vom USBoard-USS4 gelesen. Die Antwort besteht aus neun Nachrichten, die hintereinander gesendet werden.

**Kommando** ID: +0

|                  |   |   |   |   |   |   |   |
|------------------|---|---|---|---|---|---|---|
| CMD_READ_PARASET | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
|------------------|---|---|---|---|---|---|---|

**Antwort** ID: +6

Die Antwort besteht aus neun Nachrichten, die nacheinander gesendet werden, wobei jede Nachricht ein paar Bytes des Parametersatzes enthält.

|                  |   |                 |
|------------------|---|-----------------|
| CMD_READ_PARASET | 0 | Bytes 1 bis 6   |
| CMD_READ_PARASET | 1 | Bytes 7 bis 12  |
| CMD_READ_PARASET | 2 | Bytes 13 bis 18 |
| CMD_READ_PARASET | 3 | Bytes 19 bis 24 |
| CMD_READ_PARASET | 4 | Bytes 25 bis 30 |
| CMD_READ_PARASET | 5 | Bytes 31 bis 36 |
| CMD_READ_PARASET | 6 | Bytes 37 bis 42 |
| CMD_READ_PARASET | 7 | Bytes 43 bis 48 |
| CMD_READ_PARASET | 8 | Bytes 49 bis 54 |

### 2.8.2.8 CMD\_GET\_ANALOGIN

Mit diesem Kommando werden die Messwerte der vier Analogeingänge abgefragt.

**Kommando ID:** +0

|                  |   |   |   |   |   |   |   |
|------------------|---|---|---|---|---|---|---|
| CMD_GET_ANALOGIN | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
|------------------|---|---|---|---|---|---|---|

**Antwort ID:** +7

Die Auflösung beträgt 12 Bit, so dass innerhalb der CAN-Antwort zunächst das untere Byte des entsprechenden Kanals übertragen wird. In den Bytes 5 und 6 der CAN-Nachricht sind die oberen vier Bits der Kanäle 1 – 4 enthalten.

|                  |                       |                       |                       |                       |                                    |                                    |   |
|------------------|-----------------------|-----------------------|-----------------------|-----------------------|------------------------------------|------------------------------------|---|
| CMD_GET_ANALOGIN | low byte<br>channel 1 | low byte<br>channel 2 | low byte<br>channel 3 | low byte<br>channel 4 | high bits of<br>channel 2 and<br>1 | high bits of<br>channel 4 and<br>3 | 0 |
|------------------|-----------------------|-----------------------|-----------------------|-----------------------|------------------------------------|------------------------------------|---|

## 2.9 RS-232-Kommunikation

**Bemerkung:** Bitte beachten Sie, dass die RS-232-Schnittstelle eine gemeinsame Masseverbindung zwischen USBoard-USS4 und Rechner erfordert.

Die RS-232-Schnittstelle wird mit 19.200 Baud betrieben. Das Protokoll besitzt das gleiche Format wie bei der *CAN-Kommunikation* (Seite 7), jedoch mit den folgenden Erweiterungen:

Jede Nachricht vom **USBoard-USS4** beginnt mit einem Startbyte, das den Wert 0xFF hat. Abschließend zu jeder aus 8 Datenbytes bestehenden Nachricht wird eine 16 Bit Checksumme gesendet. Die vollständige Nachricht hat somit 11 Bytes:

| Byte 1 | Bytes 2-9  | Byte 10            | Byte 11           |
|--------|------------|--------------------|-------------------|
| 0xFF   | Datenbytes | checksum high byte | checksum low byte |

Die Checksumme berechnet sich nach CRC-CCITT über alle 8 Datenbytes.

Die zum **USBoard-USS4** verschickten Nachrichten enthalten lediglich die Datenbytes 1-8.

Implementierung der Checksummenberechnung (C-Code):

```
unsigned int getChecksum(unsigned char *c, size_t iNumBytes) {
    unsigned int uCrc16;
    unsigned char ucData[2];
    size_t i;

    uCrc16 = 0;
    ucData[0] = 0;
    for(i=0; i<iNumBytes; i++){
        ucData[1] = ucData[0];
        ucData[0] = c[i];

        if(uCrc16 & 0x8000){
            uCrc16 = (uCrc16 & 0x7fff) << 1;
        }
    }
}
```

(Fortsetzung auf der nächsten Seite)

(Fortsetzung der vorherigen Seite)

```
        uCrc16 ^= 0x1021;
    }else{
        uCrc16 <= 1;
    }
    uCrc16 ^= ((unsigned int)(ucData[0]) | ((unsigned int)(ucData[1]) << 8));
}

return uCrc16;
}
```

## 2.10 ROS-Node

Hier geht es zu unserem ROS Node für das USBoard-USS4 auf GitHub: [https://github.com/neobotix/neo\\_usboard](https://github.com/neobotix/neo_usboard)

## 2.11 Grafische Bedienoberfläche

Laden Sie die GUI hier herunter: [32bit<sup>4</sup>](#), [64bit<sup>5</sup>](#)

---

**Bemerkung:** Die 64-Bit-Version ist leider nicht mehr mit der aktuellsten JRE verwendbar!

---

### 2.11.1 Einleitung

Das USBoard-USS4 wird zusammen mit einer grafischen Bedienoberfläche, der USBoard-USS4 configuration GUI, ausgeliefert. Mit diesem Programm lassen sich die Einstellungen des USBoard-USS4 komfortabel ändern sowie die aktuellen Messwerte auslesen und anzeigen.

Die USBoard-USS4-GUI benötigt einen PC mit:

- einer seriellen RS-232-Schnittstelle (COM-Port) und
- Java® Laufzeitumgebung mindestens in Version 1.5.

Die Verbindung zwischen COM-Port und USBoard-USS4 wird über ein spezielles Konfigurationskabel hergestellt, das von Neobotix bezogen oder mit Hilfe *der Steckerbelegung* (Seite 17) selbst gefertigt werden kann. Das Konfigurationskabel stellt über einen gesonderten Anschluss auch *die Stromversorgung* (Seite 5) sicher, welche durch eine Status-LED an der Längsseite des USBoard-USS4 signalisiert wird.

Die Bedienoberfläche kann ohne Installation durch einen Doppelklick auf die Datei `start.bat` gestartet werden. Achten Sie darauf, die für Ihr Betriebssystem passende Datei zu verwenden.

Die Bedienoberfläche kann ebenfalls unter Linux und Mac OS verwendet werden. Dazu muss jedoch gewährleistet sein, dass die RXTX-Bibliothek von der Java-Laufzeitumgebung gefunden werden kann und dass der Benutzer die nötigen Berechtigungen besitzt, um die serielle Schnittstelle zu benutzen. Unter Ubuntu Linux genügt es, das Paket `librxtx-java` zu installieren und den Benutzer der Gruppe `dialout` hinzuzufügen.

---

<sup>4</sup> [https://neobotix-docs.de/files/USBoardGUI\\_v5802\\_32Bit.zip](https://neobotix-docs.de/files/USBoardGUI_v5802_32Bit.zip)

<sup>5</sup> [https://neobotix-docs.de/files/USBoardGUI\\_v5802\\_64Bit.zip](https://neobotix-docs.de/files/USBoardGUI_v5802_64Bit.zip)

## 2.11.2 Erste Schritte

### 2.11.2.1 Verbindung aufbauen

Stellen Sie bitte sicher, dass vor dem Starten der Verbindung per USBoard-USS4-GUI folgende Bedingungen erfüllt sind

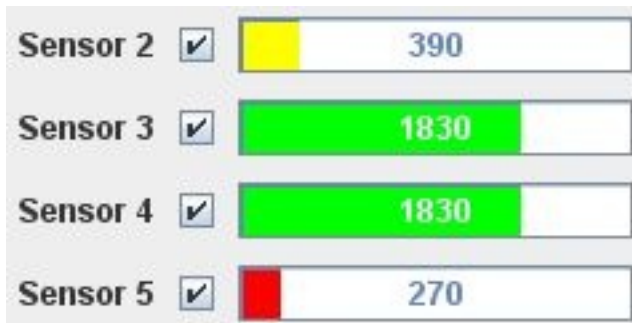
- Das USBoard-USS4 ist mit dem PC über eine serielle RS-232-Schnittstelle verbunden
- Es ist an eine geeignete Stromquelle angeschlossen (Status-LED leuchtet)

Wählen Sie nun im Programm die Schnittstelle (COM-Port) aus, über die sich das Programm verbinden soll. Klicken Sie anschließend auf die Schaltfläche „Connect“, um die Verbindung herzustellen.



### 2.11.3 Messwerte empfangen

Nach dem Verbindungsvorgang sollten Ihnen die aktuellen Mess- und Einstellungswerte angezeigt werden. Abhängig von den Parametern aktualisieren sich die Messwerte in einem bestimmten Intervall automatisch oder nur auf Anfrage per Klick auf die Request Schaltflächen. Unterschreitet der Messwert eines Sensors die Warndistanz wird der Messwert gelb und beim Unterschreiten der Alarmedistanz rot dargestellt. Gleichzeitig wird die entsprechende gelbe bzw. rote LED samt Relais auf dem USBoard-USS4 aktiviert.



Die Messwerte der Analogeingänge (Spannungsbereich 0 V – 5 V) werden prozentual unter denen der Sensoren angezeigt.

### 2.11.4 Verbindung beenden

Um die Verbindung der Anwendung zum USBoard-USS4 zu trennen, klicken Sie auf „Disconnect“.



## 2.11.5 Das USBoard-USS4 konfigurieren

### 2.11.5.1 Kommunikationseinstellungen

In der linken Spalte können die Parameter für die Kommunikation mit dem USBoard-USS4 festgelegt werden. Im Feld CAN können die Basis-ID und die Übertragungsrate, die bei der Kommunikation via CAN-Bus verwendet werden sollen, eingestellt werden.

Das generelle Übertragungsverhalten kann im Feld Transmission Mode eingestellt werden. Die möglichen Verhaltensweisen sind in Tabelle 4 aufgelistet. Die unter Interval wählbare Zykluszeit ist nur für kontinuierliche Übertragungsmodi relevant.

| Übertragungsmodus | Verhalten                                 |
|-------------------|---|
| Request           | Sendet nur auf Anfrage                    |
| CAN cont.         | Sendet kontinuierlich über CAN            |
| RS232 cont.       | Sendet kontinuierlich über RS-232         |
| RS232 + CAN cont. | Sendet kontinuierlich über RS-232 und CAN |

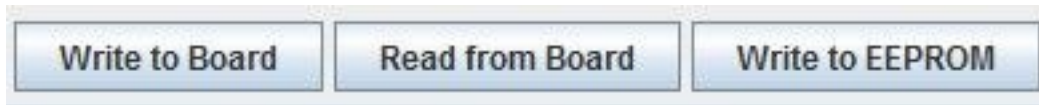
### 2.11.5.2 Sensoren konfigurieren

Rechts können verschiedene Parameter für die Sensoren festgelegt werden: In den Kontrollkästchen neben den Bezeichnungen der Sensoren kann man diese jeweils aktivieren oder deaktivieren (Häkchen gesetzt bzw. nicht gesetzt). Deaktivierte Sensoren senden keinen Echo-Impuls und keine Messwerte. Rechts neben der Anzeige des aktuellen Entfernungsmesswertes können die Warn- und Alarmbereiche festgelegt werden. Die ID des USBoard-USS4 und die Analogeingänge sind nicht konfigurierbar.

**Warnung:** Das USBoard-USS4 ist trotz Warn- und Alarmbereich kein Sicherheitssystem und kann ausschließlich unterstützende, nicht sichere Daten liefern. Verwenden Sie das USBoard-USS4 niemals zur Absicherung von Gefahrenstellen.

### 2.11.5.3 Konfigurationen auf das Board überspielen

Die geänderten Konfigurationsparameter müssen manuell auf das USBoard-USS4 übertragen werden, um in Funktion zu treten. Die USBoard-USS4-GUI bietet dafür zwei Optionen an:



- Write to Board schreibt die Parameter in den flüchtigen Speicher des USBoard-USS4, d.h. nach dem nächsten Hardware-Neustart werden wieder die vorherigen Parameter genutzt.
- Write to EEPROM schreibt die Parameter in den nichtflüchtigen EEPROM-Speicher des USBoard-USS4, d.h. die Parameter bleiben auch nach einem Hardware-Neustart erhalten und werden genutzt, bis sie überschrieben werden.

Um den jeweils aktuellen Parametersatz aus dem Speicher des USBoard-USS4 neu auszulesen, drücken Sie „Read from Board“. Änderungen in der Eingabemaske gehen dabei jedoch verloren.

### 2.11.5.4 Konfigurationen speichern und laden

Mit Save to File und Load from File lassen sich die aktuellen Konfigurationsdaten in einer Datei speichern bzw. aus dieser laden. Geben Sie dazu einfach in dem nach Knopfdruck erscheinendem Dialog den gewünschten Dateinamen ein bzw. wählen Sie die entsprechende Datei aus und bestätigen mit „Ok“.

### 2.11.5.5 Logging

Die USBoard-USS4-GUI schreibt Log-Dateien auf die Festplatte, um die Einrichtung des Systems und die Fehlerbehebung zu unterstützen. Die Anzahl der zu schreibenden Meldungen kann über den Level Wert zwischen Severe (nur kritische Probleme) und Finest (alle Meldungen) gewählt werden. Die Log-Dateien befinden sich im Unterordner log des Verzeichnisses, in dem auch die GUI selbst liegt.

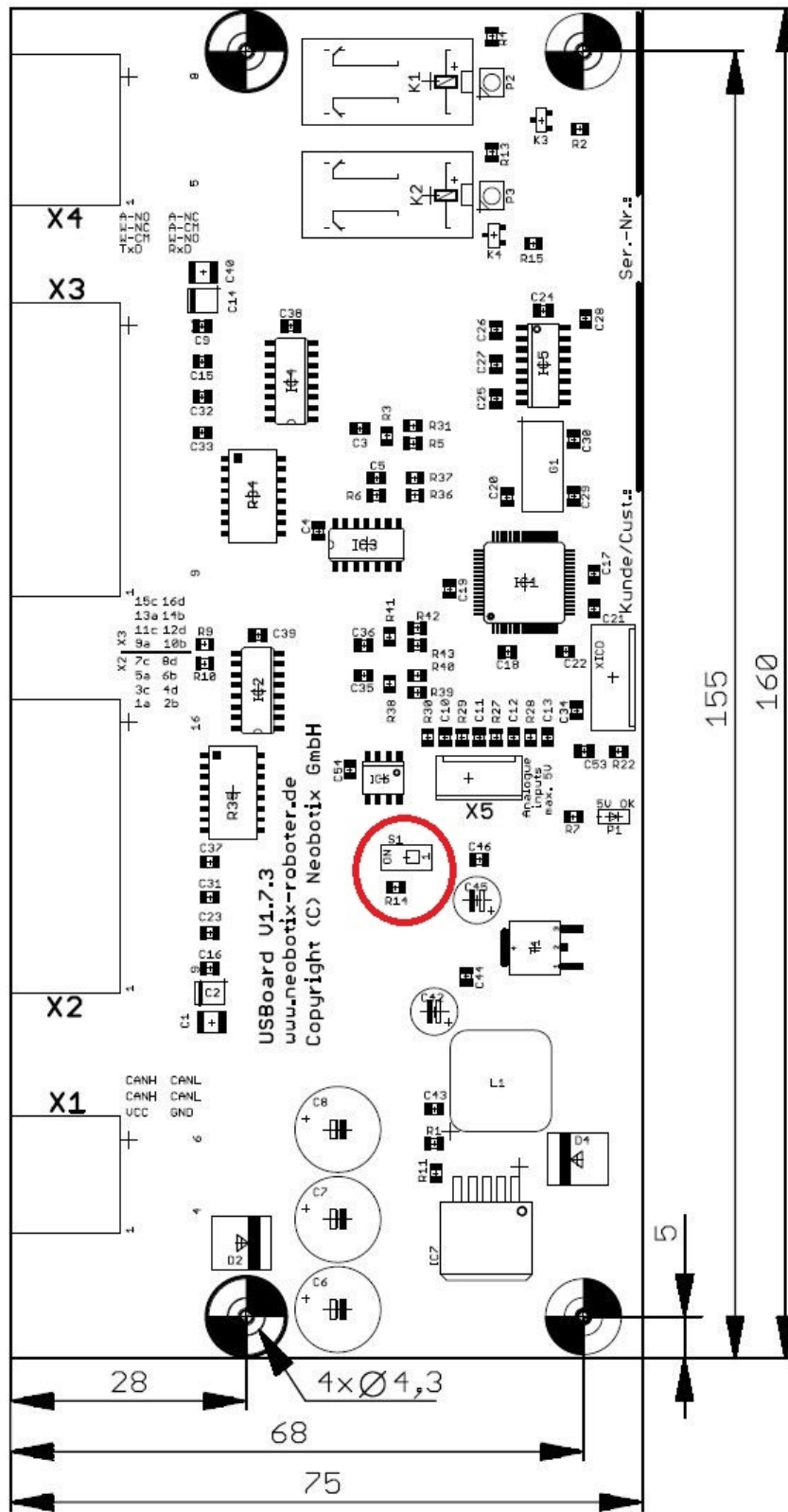
Eine Echtzeitausgabe des aktuellen Logs kann über die Schaltfläche „Show / Hide“ angezeigt werden.





## 2.12 Abmessungen und Steckerbelegung

### 2.12.1 Abmessungen



Schieben Sie S1 auf Position ON, um den CAN-Abschlusswiderstand zu aktivieren.

Die Befestigungsbohrung unten rechts ist elektrisch mit der Massefläche des USBoard-USS4 verbunden. Bitte isolieren Sie diesen Befestigungspunkt, wenn Ihre Installation es erfordert.

Das USBoard-USS4 kann sowohl waagrecht als auch an Wänden montiert werden. Bei Überkopfmontage kann es zu Fehlfunktionen der Relais kommen, die Datenausgabe ist dadurch jedoch nicht beeinträchtigt.

## 2.12.2 Steckerbelegung

---

**Tipp:** Weitere Informationen zu den verwendeten Steckverbindern finden Sie unter [Steckverbinder](#) (Seite 26).

---

### 2.12.2.1 Stecker X1 – WE MPC4, 6-polig

| Pin | Beschreibung                             |
|-----|--|
| 1   | Versorgungsspannung                      |
| 2   | CAN High                                 |
| 3   | CAN High                                 |
| 4   | Masse (zu Versorgungsspannung und RS232) |
| 5   | CAN Low                                  |
| 6   | CAN Low                                  |

Pins 2 und 3 bzw. 5 und 6 sind jeweils verbunden. Dadurch können zwei CAN-Leitungen (CAN-in und CAN-out) komfortabel angeschlossen werden.

### 2.12.2.2 Stecker X2 – WE MPC4, 16-polig

| Pin  | Beschreibung                      |
|------|-----------------------------------|
| 1-4  | Versorgungsspannung Sensoren 0 V  |
| 5    | Sensor 1                          |
| 6    | Sensor 3                          |
| 7    | Sensor 5                          |
| 8    | Sensor 7                          |
| 9-12 | Versorgungsspannung Sensoren +8 V |
| 13   | Sensor 2                          |
| 14   | Sensor 4                          |
| 15   | Sensor 6                          |
| 16   | Sensor 8                          |

**2.12.2.3 Stecker X3 – WE MPC4, 16-polig**

| Pin   | Beschreibung                      |
|-------|-----------------------------------|
| 1     | Sensor 9                          |
| 2     | Sensor 11                         |
| 3     | Sensor 13                         |
| 4     | Sensor 15                         |
| 5-8   | Versorgungsspannung Sensoren 0 V  |
| 9     | Sensor 10                         |
| 10    | Sensor 12                         |
| 11    | Sensor 14                         |
| 12    | Sensor 16                         |
| 13-16 | Versorgungsspannung Sensoren +8 V |

**2.12.2.4 Stecker X4 – WE MPC4, 8-polig**

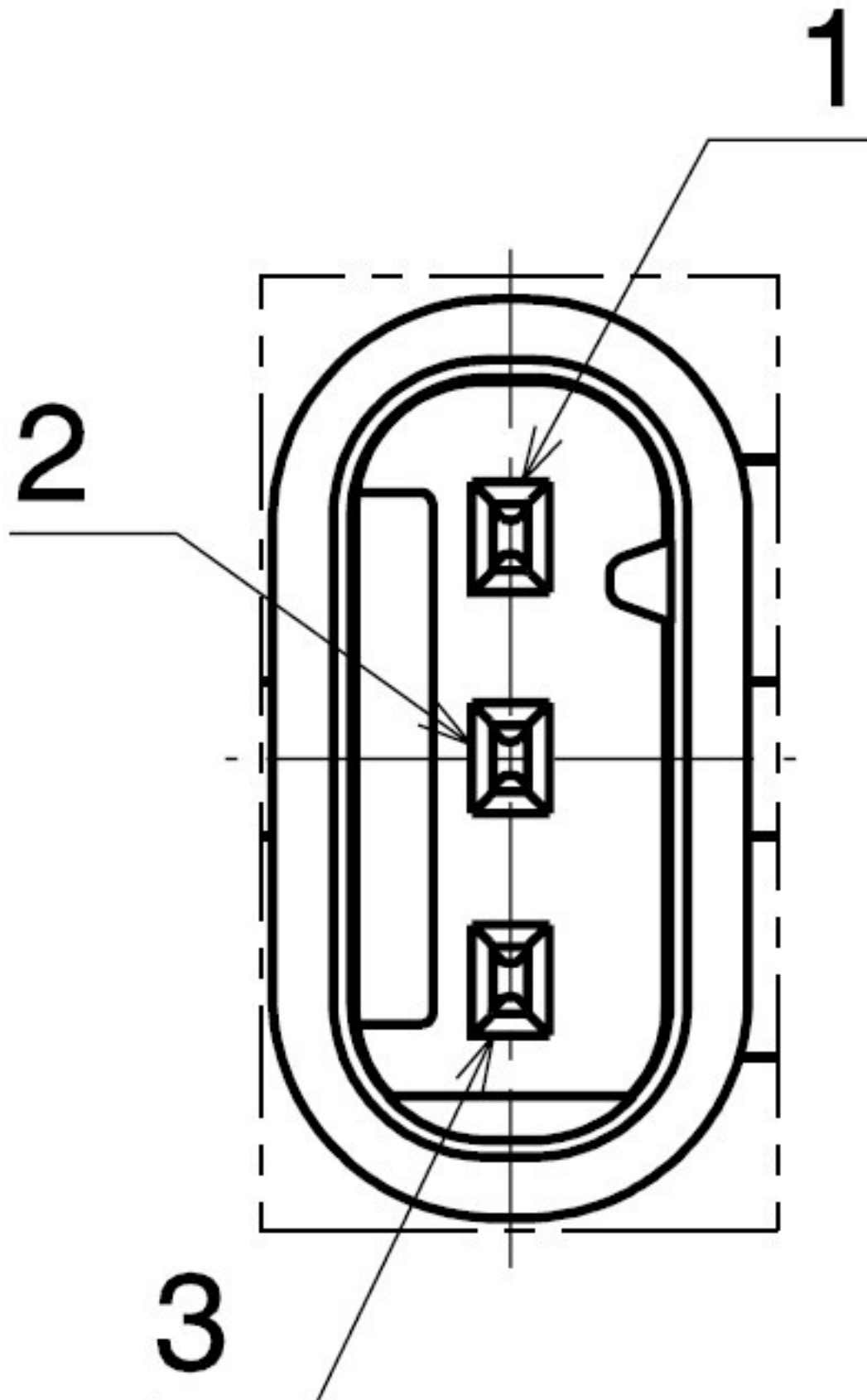
| Pin | Beschreibung                              |
|-----|---|
| 1   | RS-232 Transmit                           |
| 2   | Relais „Warnbereich“ gemeinsamer Kontakt  |
| 3   | Relais „Warnbereich“ Öffnerkontakt        |
| 4   | Relais „Alarmbereich“ gemeinsamer Kontakt |
| 5   | RS-232 Receive                            |
| 6   | Relais „Warnbereich“ gemeinsamer Kontakt  |
| 7   | Relais „Alarmbereich“ gemeinsamer Kontakt |
| 8   | Relais „Alarmbereich“ Öffnerkontakt       |

**2.12.2.5 Analogeingänge – TE Connectivity HE14, 4-polig**

| Pin | Beschreibung    |
|-----|-----------------|
| 1   | Analogeingang 1 |
| 2   | Analogeingang 2 |
| 3   | Analogeingang 3 |
| 4   | Analogeingang 4 |

**2.12.2.6 Sensoren – Gehäuse: TE 1-1718644-1, Kontakte: TE 1452668-1**

| Pin | Beschreibung                      |
|-----|-----------------------------------|
| 1   | Senden / Empfangen                |
| 2   | Versorgungsspannung Sensoren 0 V  |
| 3   | Versorgungsspannung Sensoren +8 V |



## 2.13 Zukaufteile

### 2.13.1 Steckverbinder

Eine Übersicht der verwendeten Steckverbinder finden Sie [hier](#) (Seite 26).

### 2.13.2 Konfigurationskabel

Das Konfigurationskabel (Best.-Nr. X201) ermöglicht eine schnelle und einfache Inbetriebnahme und erste Tests des USBoard-USS4. Mit den oben aufgeführten Steckverbindern kann es auch ohne Weiteres selbst hergestellt werden. Achten Sie in diesem Fall auf die Masseverbindung der RS-232-Schnittstelle.

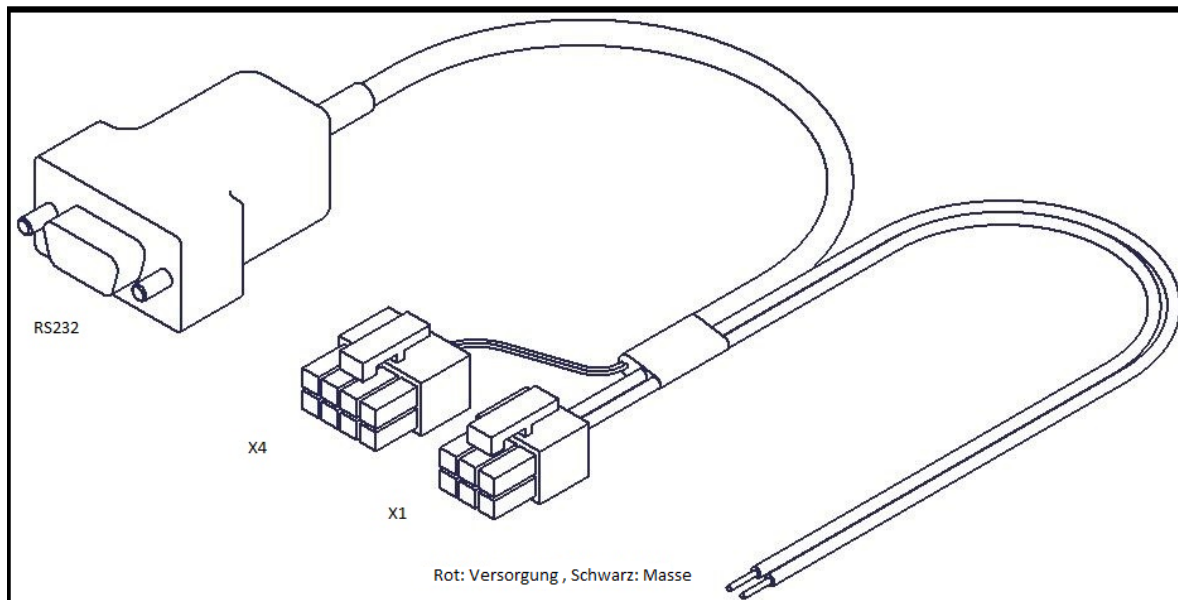


Abb. 1: USBoard-USS4-Konfigurationskabel (X201)



### 2.13.3 Sensorkabelsätze

Für einzelne Testaufbauten können Kabelsätze für den Anschluss von je acht Sensoren geliefert werden. Diese unterscheiden sich in ihrer Belegung und sind für den Anschluss an Stecker X2 (Best.-Nr. X204) oder Stecker X3 (Best.-Nr. X205) erhältlich. Die Kabel sind jeweils 2 m lang und geschirmt.

Bitte fertigen Sie für den Einsatz in Ihrer tatsächlichen Anwendung die Kabel selbst entsprechend der jeweiligen Anforderungen.



Abb. 2: Kabelsatz für USS4 Sensoren (X204/X205)

Bei geringen Stückzahlen können die Steckverbinder für das USBoard-USS4 (Best.-Nr. X202) und einzelne USS4 Ultraschallsensoren (Best.-Nr. X203) ebenfalls geliefert werden. Die Sets enthalten jeweils die Steckergehäuse sowie die lose beiliegenden Crimpkontakte in passender Anzahl.

Falls Sie keine passenden Werkzeuge besitzen und die Stecker nicht mit Universalcrimpzangen verarbeiten möchten, wenden Sie sich bitte an Neobotix.

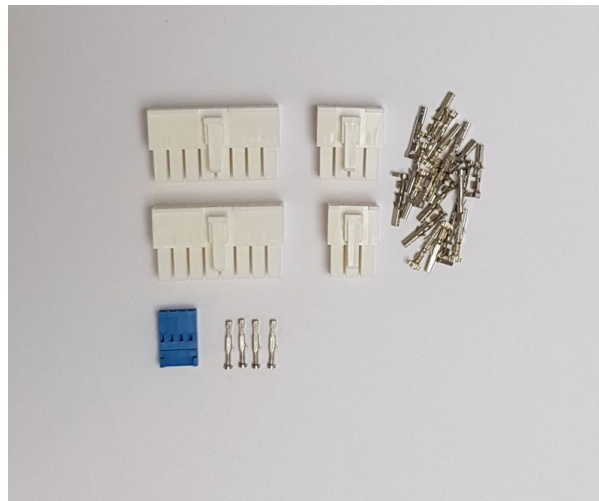


Abb. 3: Steckersatz für USBoard-USS4 (X202)

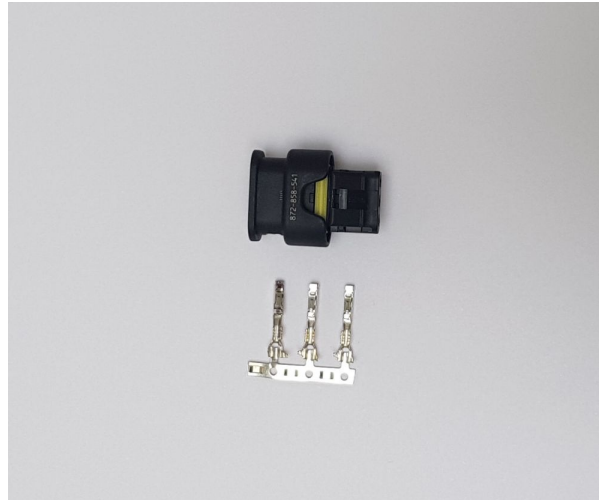


Abb. 4: Steckersatz für USS4 (X203)

### 2.13.4 Ultraschallsensor USS4

Die Ultraschallsensoren Bosch ParkPilot USS4 (Bosch-Teilenummer 0 263 009 525) sind unter der Bestellnummer X200 erhältlich.

---

**Bemerkung:** Die blanken Sensoren heißen USS (Ultra Sonic Sensor), momentan in der vierten Generation. Die fünfte Generation wird von unserem unserem USBoard-USS5 unterstützt.

Bosch bietet für Privatkunden das URF-System an. Dabei handelt es sich um acht Sensoren mit Kabelsatz und einer kleinen Auswerteeinheit für den Anbau am Armaturenbrett von PKW. Dieses System zeigt lediglich über einige LEDs und durch ein akustisches Signal an, dass Sie gerade dicht an ein Hindernis heranfahren. Eine Datenausgabe bietet es nicht. Das URF gibt es in zwei Versionen (Gen. 6 und 7), die sich aber nur in der Firmware unterscheiden und beide die USS4 verwenden.

---



Abb. 5: Ultraschallsensor Bosch USS4 (X200)

### 2.13.5 Montagesets für USS4

Mit dem *Montageset für USS4, gerade* (X206) wird der Sensor an einer Frontplatte befestigt.

Für eine optimale Sensorposition sind bei dem *Montageset für USS4, 0°, 5°, 10°, 15°* (X207) vier Frontringe für unterschiedliche Montagewinkel enthalten.



Abb. 6: Montageset für USS4 gerade (X206)



Abb. 7: Montageset für USS4 0°, 5°, 10°, 15° (X207)

## 2.14 Rechtliche Anmerkungen

Die allgemeinen rechtlichen Anmerkungen finden Sie unter *Rechtliche Anmerkungen* (Seite 2).

### 2.14.1 EG-Konformitätserklärung



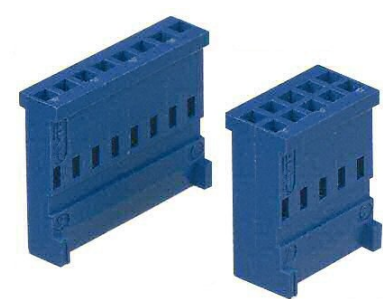
Hiermit bestätigt Neobotix, dass das beschriebene Produkt die relevanten EU-Richtlinien erfüllt.

### 2.14.2 Informationen zu RoHS



Hiermit bestätigt Neobotix, dass das beschriebene Produkt die RoHS-Richtlinien 2011/65/EU (RoHS 2) und 2015/863/EU des Europäischen Parlaments und des Rates zur Beschränkung der Verwendung gefährlicher Stoffe in Elektro- und Elektronikgeräten erfüllt.

3.1 TE Connectivity - HE14

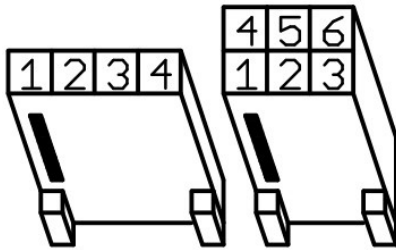


| Pole               | TE Connectivity | Farnell | RS Components |
|--------------------|-----------------|---------|---------------|
| 3-polig, 1-reihig  | 281838-3        | 429582  | 532-333       |
| 4-polig, 1-reihig  | 281838-4        | 429594  | 532-349       |
| 5-polig, 1-reihig  | 281838-5        | 429600  | 532-355       |
| 6-polig, 2-reihig  | 281839-3        | 429650  | 532-406       |
| 8-polig, 2-reihig  | 281839-4        | 429661  | 532-412       |
| 10-polig, 2-reihig | 281839-5        | 429673  | 532-428       |
| 12-polig, 2-reihig | 281839-6        | 429685  | 532-434       |



| Crimpkontakte | TE Connectivity | Farnell | RS Components |
|---------------|-----------------|---------|---------------|
| AWG 28-24     | 182734-2        | 429715  | 532-456       |

Bei Neobotix-Produkten ist die Pinbelegung der HE14-Stecker wie unten dargestellt.



## 3.2 Würth Elektronik - MPC4

Für nähere Informationen zum MPC4<sup>6</sup> konsultieren Sie bitte den Würth Elektronik Onlinekatalog<sup>7</sup>.



| Pole (2-reihig) | Würth Elektronik |
|-----------------|------------------|
| 2               | 649002113322     |
| 4               | 649004113322     |
| 6               | 649006113322     |
| 8               | 649008113322     |
| 10              | 649010113322     |
| 16              | 649016113322     |

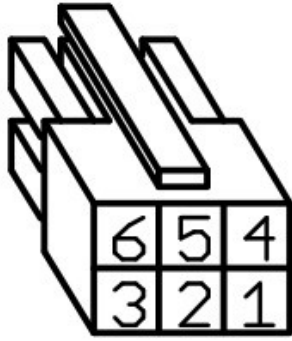


| Crimpkontakte | Würth Elektronik |
|---------------|------------------|
| AWG 24-18     | 64900613722      |

Bei Neobotix-Produkten ist die Pinbelegung der MPC4-Stecker wie unten dargestellt.

<sup>6</sup> [https://www.we-online.de/katalog/de/em/connectors/wire-to-board/wr\\_mpc4/](https://www.we-online.de/katalog/de/em/connectors/wire-to-board/wr_mpc4/)

<sup>7</sup> <https://www.we-online.com/de/produkte/bauelemente/uebersicht>



### 3.3 Würth Elektronik - MPC3

Für nähere Informationen zum MPC3<sup>8</sup> konsultieren Sie bitte den [Würth Elektronik Onlinekatalog](#)<sup>9</sup>.



| Pole (2-reihig) | Würth Elektronik |
|-----------------|------------------|
| 4               | 662004113322     |
| 6               | 662006113322     |
| 12              | 662012113322     |

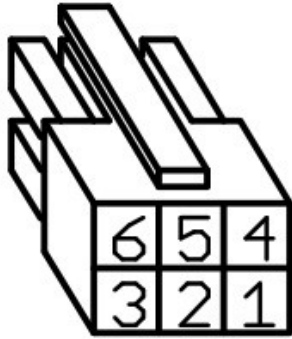


| Crimpkontakte | Würth Elektronik |
|---------------|------------------|
| AWG 24-20     | 66200113722      |

Bei Neobotix-Produkten ist die Pinbelegung der MPC3-Stecker wie unten dargestellt.

<sup>8</sup> [https://www.we-online.de/katalog/de/em/connectors/wire-to-board/wr\\_mpc3/](https://www.we-online.de/katalog/de/em/connectors/wire-to-board/wr_mpc3/)

<sup>9</sup> <https://www.we-online.com/de/produkte/bauelemente/uebersicht>





---

### Sachkundiges Personal

---

Dieses Produkt darf nur von sachkundigem Personal umgebaut, in Betrieb genommen und betrieben werden. Sachkundig ist, wer

- aufgrund seiner fachlichen Ausbildung und Erfahrung ausreichende Kenntnisse auf den für die jeweilige Arbeit erforderlichen Gebieten der Mechatronik und Robotik hat,
- vom Betreiber der Maschine in der Bedienung und den gültigen Sicherheitsrichtlinien unterwiesen wurde,
- mit den einschlägigen staatlichen Arbeitsschutzvorschriften, Unfallverhütungsvorschriften, Richtlinien und allgemein anerkannten Regeln der Technik (z. B. DIN-Normen, VDE-Bestimmungen, technische Regeln) so weit vertraut ist, dass er den arbeitssicheren Zustand des Produkts beurteilen kann und
- Zugriff auf diese Unterlagen und hat und diese gelesen hat.

Als nicht-sachkundig gelten im Allgemeinen, aber nicht ausschließlich:

- Nicht mit dem Produkt vertraute Praktikanten oder Mitarbeiter,
- Besucher und Gäste,
- alle Mitarbeiter anderer Abteilungen des Unternehmens oder der Einrichtung, in der das Produkt betrieben wird.

Diese Liste ist nicht abschließend.