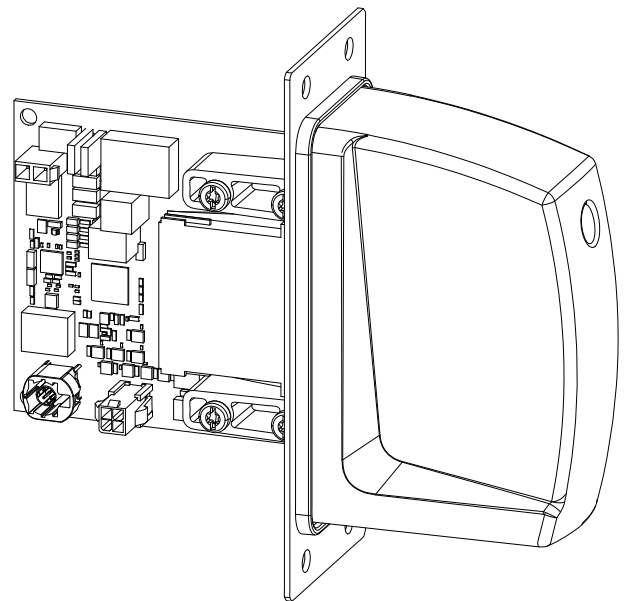


Operating Instructions

RI FB/i CRC 1.0

RI MOD/i CC Modbus TCP-2P



EN-US | Operating instructions



42,0426,0453,EA

002-23012025

Table of contents

| | |
|--------------------------------------------------------|----|
| General..... | 4 |
| Safety | 4 |
| Device Concept..... | 4 |
| Block diagram..... | 5 |
| Scope of supply..... | 5 |
| Required Tools and Materials..... | 5 |
| Installation Requirements..... | 5 |
| Connections and Indicators..... | 6 |
| Connections on the Robot Interface | 6 |
| LEDs on Robot Interface PCB..... | 6 |
| LEDs for Power Supply Diagnosis..... | 7 |
| LEDs for Network Connection Diagnosis..... | 8 |
| Connections and indicators on the RJ 45 module | 8 |
| Technical data..... | 10 |
| Environmental Conditions..... | 10 |
| Robot Interface Technical Data | 10 |
| Data Transfer Properties..... | 10 |
| Configuration parameters | 10 |
| Configuration of robot interface..... | 12 |
| General..... | 12 |
| Setting the process image..... | 12 |
| Configuring the Robot Interface | 12 |
| Installing the Robot Interface..... | 13 |
| Safety | 13 |
| Preparation..... | 13 |
| Routing the Data Cable..... | 14 |
| Installing the Robot Interface..... | 15 |
| Final Tasks..... | 15 |
| Installing the Bus Module..... | 16 |
| Safety | 16 |
| Installing the Bus Module..... | 16 |
| Input and output signals..... | 17 |
| Data types | 17 |
| Input signals | 17 |
| Value range Process line selection | 21 |
| Value Range for TWIN Mode..... | 21 |
| Value Range for Documentation mode..... | 22 |
| Value range for Working mode..... | 22 |
| Value Range for Command value selection..... | 22 |
| Value range for Motor Type..... | 22 |
| Output signals..... | 22 |
| Value range for welding process and process image..... | 26 |
| Assignment of Sensor Statuses 1–4 | 26 |
| Value range Safety status..... | 26 |
| TAG table..... | 27 |
| Cooling unit mode | 29 |
| Language..... | 29 |
| Unit..... | 30 |
| Welding standard..... | 30 |
| Modbus – General Information..... | 31 |
| Protocol Description..... | 31 |
| Data Coding..... | 31 |
| Application Data Unit (ADU)..... | 31 |
| Modbus Functions..... | 33 |
| 03 (03) Read Holding Register..... | 33 |
| 06 (06) Write Single Register | 34 |
| 16 (10) Write Multiple Register..... | 35 |
| 23 (17) Read/Write Multiple Register..... | 36 |
| 103 (67) Read Holding Register Float..... | 38 |
| 104 (68) Write Single Register Float..... | 39 |

General

Safety



WARNING!

Danger from incorrect operation and work that is not carried out properly.

This can result in serious personal injury and damage to property.

- ▶ All the work and functions described in this document must only be carried out by technically trained and qualified personnel.
- ▶ Read and understand this document in full.
- ▶ Read and understand all safety rules and user documentation for this equipment and all system components.



WARNING!

Danger from electrical current.

This can result in serious personal injury and damage to property.

- ▶ Before starting work, switch off all the devices and components involved and disconnect them from the grid.
- ▶ Secure all devices and components involved so they cannot be switched back on.



WARNING!

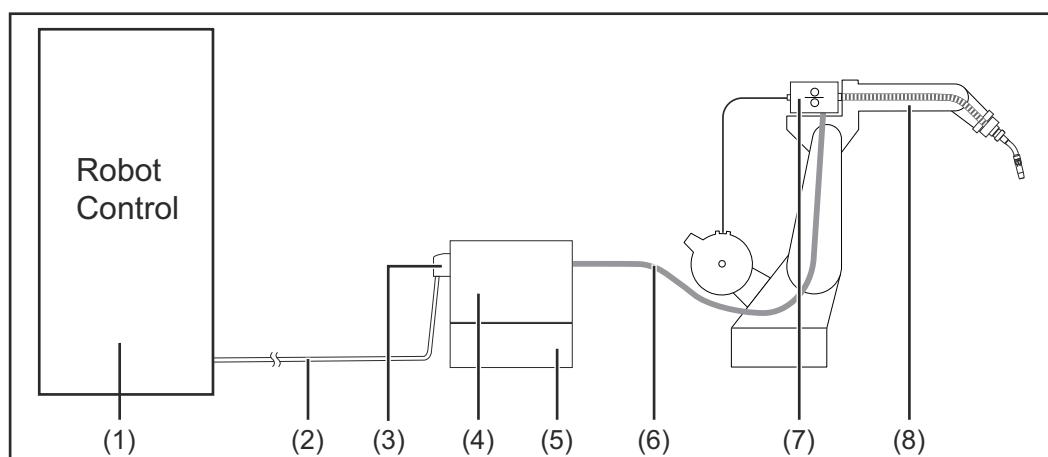
Danger from unplanned signal transmission.

This can result in serious personal injury and damage to property.

- ▶ Do not transfer safety signals via the interface.

Device Concept

The robot interface serves as an interface between the welding machine and standardized bus modules supporting a wide range of communication protocols. Fronius may factory-fit the robot interface in the welding machine but it can also be retrofitted by appropriately trained and qualified personnel.



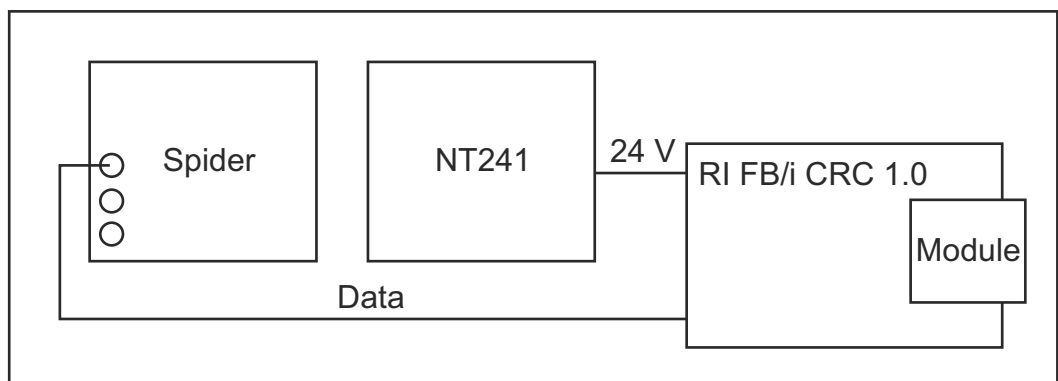
(1) Robot control system

(2) SpeedNet data cable

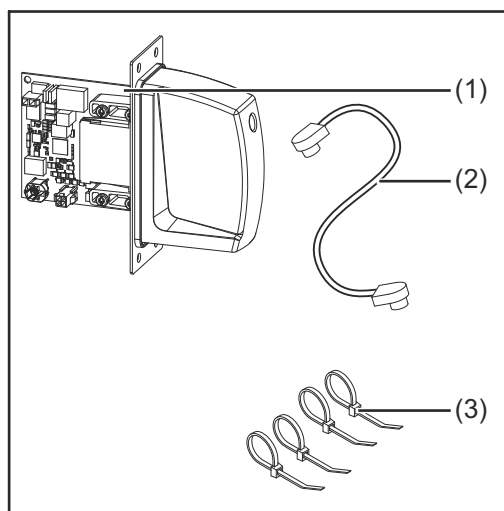
(3) Robot interface

| | |
|-----|--------------------------|
| (4) | Welding machine |
| (5) | Cooling unit |
| (6) | Interconnecting hosepack |
| (7) | Wirefeeder |
| (8) | Robot |

Block diagram



Scope of supply



| | |
|-----|---------------------------------|
| (1) | RI FB/i CRC 1.0 |
| (2) | Data cable 4-pin |
| (3) | Cable ties |
| (4) | This document (not pictured) |

Required Tools and Materials

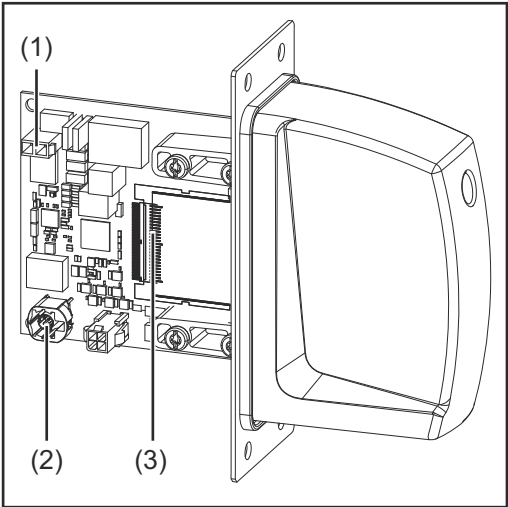
- Screwdriver TX8
- Screwdriver TX20
- Screwdriver TX25
- Diagonal cutting pliers

Installation Requirements

The robot interface may only be installed in the designated opening on the rear of the welding machine.

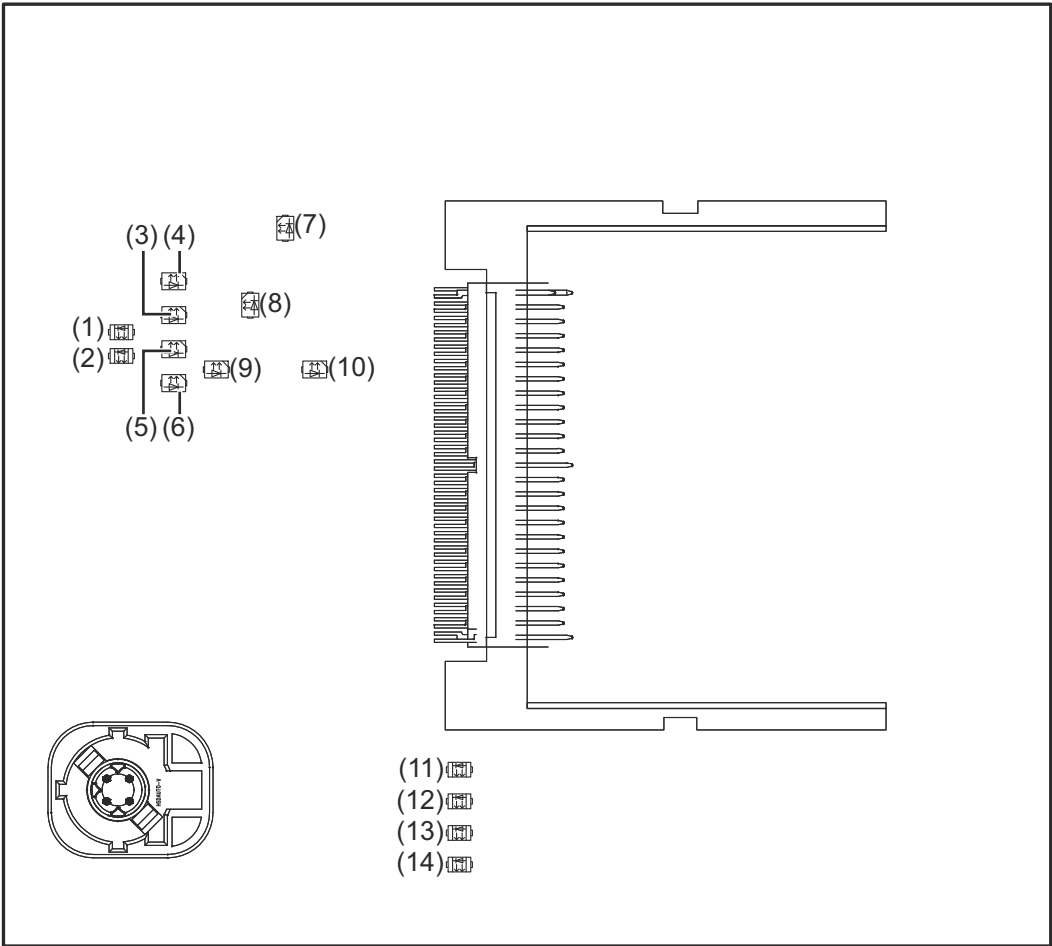
Connections and Indicators

Connections on the Robot Interface



- (1) Power supply connection 2-pin
- (2) SpeedNet data cable connection 4-pin
- (3) Bus module connection

LEDs on Robot Interface PCB



| | | | |
|-----|----------|--------|-------------------------------------------------------------------------------------------------------------------------|
| (1) | ETH1 LED | Green | For diagnosing the network connection. For details, see section below titled "LEDs for Network Connection Diagnosis" |
| (2) | ETH2 LED | Orange | |

| | | | |
|------|---------------|-------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| (3) | LED 3 | Green | No function |
| (4) | LED 4 | Green | |
| (5) | LED 5 | Green | <ul style="list-style-type: none"> - Flashes at 4 Hz = No SpeedNet connection - Flashes at 20 Hz = Establishing SpeedNet connection - Flashes at 1 Hz = SpeedNet connection established |
| (6) | LED 6 | Red | Lights up when an internal error occurs. Remedy: Restart the robot interface. If this does not resolve the issue, inform the service team. |
| (7) | +3V3 LED | Green | For diagnosing the power supply. For details, see section below titled "LEDs for Power Supply Diagnosis" |
| (8) | +24V LED | Green | |
| (9) | DIG OUT 2 LED | Green | Digital output 2. LED lights up when active |
| (10) | DIG OUT 1 LED | Green | Digital output 1. LED lights up when active |
| (11) | LED 11 | Green | No function |
| (12) | LED 12 | Green | |
| (13) | LED 13 | Green | |
| (14) | LED 14 | Green | |

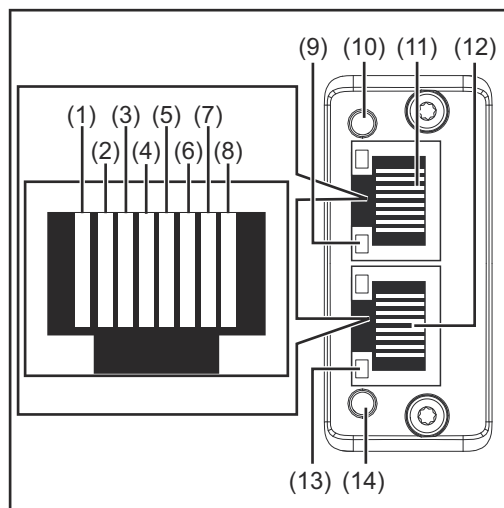
LEDs for Power Supply Diagnosis

| LED | Indicator | Meaning | Cause |
|------|-----------|----------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------|
| +24V | Off | No supply voltage available for interface | <ul style="list-style-type: none"> - Robot interface power supply not established - Power supply cable faulty |
| | Lights up | 24 VDC supply voltage present on robot interface | |
| +3V3 | Off | No operating voltage present on robot interface | <ul style="list-style-type: none"> - 24 VDC supply voltage not present - Robot interface power supply unit is faulty |
| | Lights up | 3 VDC operating voltage present on robot interface | |

LEDs for Network Connection Diagnosis

| LED | Indicator | Meaning | Cause |
|------|-----------|--------------------------------|-------------------------------------------------------------------------------------------------------------------------------------|
| ETH1 | Off | No network connection | <ul style="list-style-type: none"> - No network connection established for interface - Network cable faulty |
| | Lights up | Network connection established | |
| | Flashes | Data transfer in progress | |
| ETH2 | Off | Transmission speed 10 Mbit/s | |
| | Lights up | Transmission speed 100 Mbit/s | |

Connections and indicators on the RJ 45 module



| | |
|------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| (1) | TX+ |
| (2) | TX- |
| (3) | RX+ |
| (6) | RX- |
| (4) | Not normally used; to ensure signal completeness, these pins must be interconnected and, after passing through a filter circuit, must terminate at the ground conductor (PE). |
| (5) | |
| (7) | |
| (8) | |
| (9) | Connection/Activity 2 LED |
| (10) | Module status LED |

| | |
|------|-----------------------------|
| (11) | RJ 45 Ethernet connection 2 |
| (12) | RJ 45 Ethernet connection 1 |
| (13) | Connection/Activity 1 LED |
| (14) | Network status LED |

| Network status LED: | |
|---------------------|---------------------------------------------------------------------------------------------------------------|
| Status | Meaning |
| Off | No IP address or exception state |
| Lights up green | At least one Modbus message received |
| Flashes green | Waiting for first Modbus message |
| Lights up red | IP address conflict, serious error |
| Flashes red | Connection timeout. No Modbus message was received within the period defined for the "Process active timeout" |

| Module status LED: | |
|----------------------------------|----------------------------------------------------|
| Status | Meaning |
| Off | No supply voltage |
| Lights up green | Normal operation |
| Lights up red | Major error (exception state, serious fault, etc.) |
| Flashes red | Minor error |
| Alternates between red and green | Firmware update in progress |

| Connection/Activity LED: | |
|--------------------------|-------------------------------------|
| Status | Meaning |
| Off | No connection, no activity |
| Lights up green | Connection established (100 Mbit/s) |
| Flickers green | Activity (100 Mbit/s) |
| Lights up yellow | Connection established (10 Mbit/s) |
| Flickers yellow | Activity (10 Mbit/s) |

Technical data

Environmental Conditions



CAUTION!

A risk is posed by prohibited environmental conditions.

This can result in severe damage to equipment.

- Only store and operate the device under the following environmental conditions.

Temperature range of ambient air:

- During operation: -10 °C to +40 °C (14 °F to 104 °F)
- During transport and storage: -20 °C to +55 °C (-4 °F to 131 °F)

Relative humidity:

- Up to 50% at 40 °C (104 °F)
- Up to 90% at 20 °C (68 °F)

Ambient air: free of dust, acids, corrosive gases or substances, etc.

Altitude above sea level: up to 2000 m (6500 ft).

Robot Interface Technical Data

| | |
|----------------------|-----------------|
| Power supply | Internal (24 V) |
| Degree of protection | IP 23 |

Data Transfer Properties

RJ-45 Connection

Transmission technology:

Ethernet

Medium (4 x 2 twisted-pair copper cable):

Category 5 (100 Mbit/s) or higher

Transmission speed:

10 Mbit/s or 100 Mbit/s

Bus connection:

Ethernet RJ-45

Configuration parameters

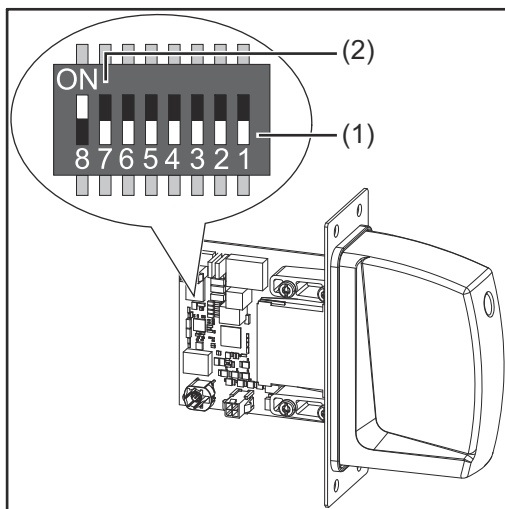
In some robot control systems, it may be necessary to state the configuration parameters described here so that the bus module can communicate with the robot.

| Parameter | Value |
|----------------------|------------------------------------------------------|
| Vendor name | Fronius International GmbH |
| Product code | 0304 _{hex} (772 _{dec}) |
| Major/minor revision | V1.00 |
| Vendor URL | www.fronius.com |

| Parameter | Value |
|-----------------------|------------------------------------------------------------------------|
| Product name | fronius-fb-pro-modbus-2p |
| Model name | Fronius FB PRO/i Modbus -TCP-2-Port |
| User application name | Fronius welding controller for the series TPS/i with Modbus-TCP-2-Port |

Configuration of robot interface

General



The DIP switch on the robot interface is used to configure:

- The process image (standard image, retrofit image)
- The IP address

Default setting for process image:
Positions 7 and 8 of DIP switch set to OFF (1) = standard image = Weldcom V2.0

Default setting for IP address = 192.168.255.210:

- Positions 6, 5, 3, and 1 of DIP switch set to OFF (1)
- Positions 2 and 4 of DIP switch set to ON (2)

Setting the process image

| Dip switch | | | | | | | | Configuration |
|------------|-----|---|---|---|---|---|---|--------------------------|
| 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | |
| OFF | OFF | - | - | - | - | - | - | Standard image (CRC 1.0) |
| OFF | ON | - | - | - | - | - | - | Not used |
| ON | OFF | - | - | - | - | - | - | Not used |
| ON | ON | - | - | - | - | - | - | Not used |

The process image defines the volume of data transferred and the system compatibility.

Configuring the Robot Interface

- 1 Set the DIP switch in accordance with the desired configuration

NOTE!

Risk due to invalid DIP switch settings.

This may result in malfunctions.

- Whenever changes are made to the DIP switch settings, the interface must be restarted. This is the only way for the changes to take effect.
- Restart the interface = interrupting and restoring the power supply or executing the relevant function on the website of the welding machine (SmartManager).

Installing the Robot Interface

Safety

WARNING!

Danger from electrical current.

Serious personal injuries may result.

- ▶ Before starting work, switch off all the devices and components involved and disconnect them from the grid.
- ▶ Secure all devices and components involved so they cannot be switched back on.
- ▶ After opening the device, use a suitable measuring instrument to check that electrically charged components (such as capacitors) have been discharged.

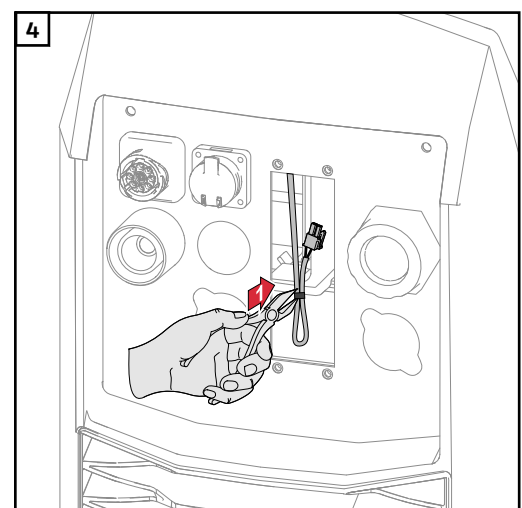
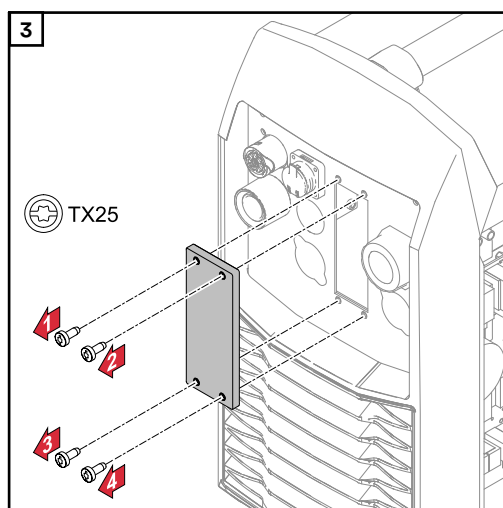
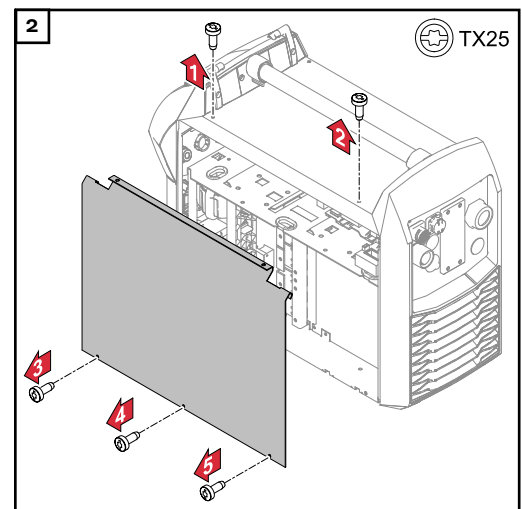
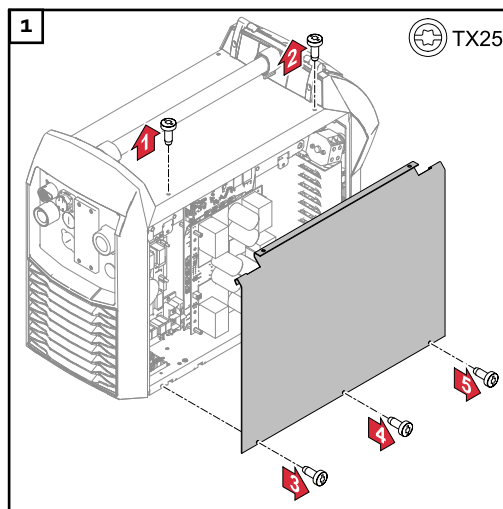
WARNING!

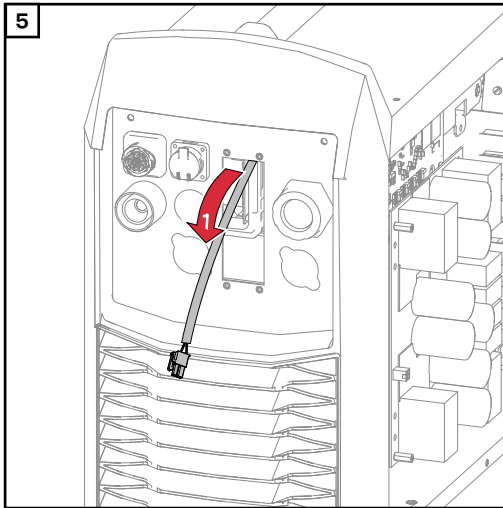
Danger from electrical current due to inadequate ground conductor connection.

This can result in serious personal injury and damage to property.

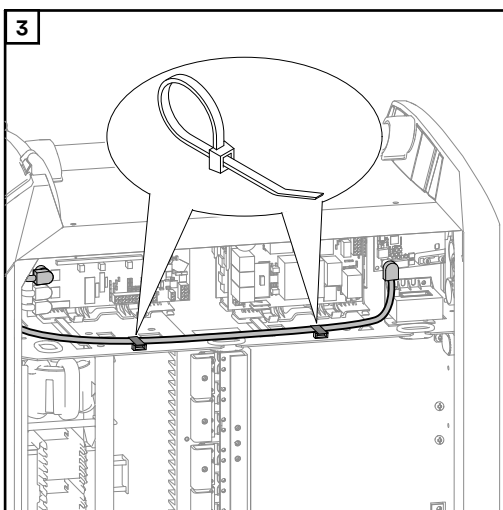
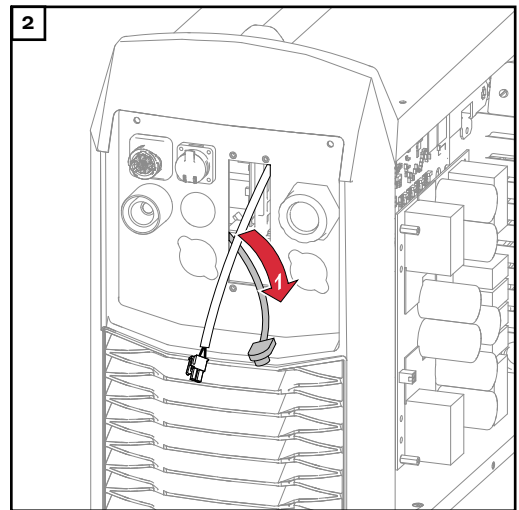
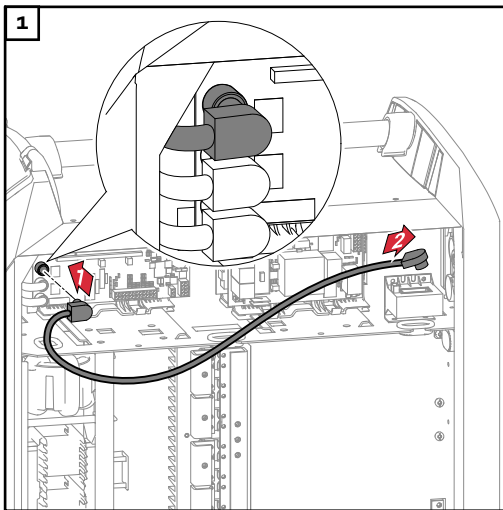
- ▶ Always use the original housing screws in the quantity initially supplied.

Preparation

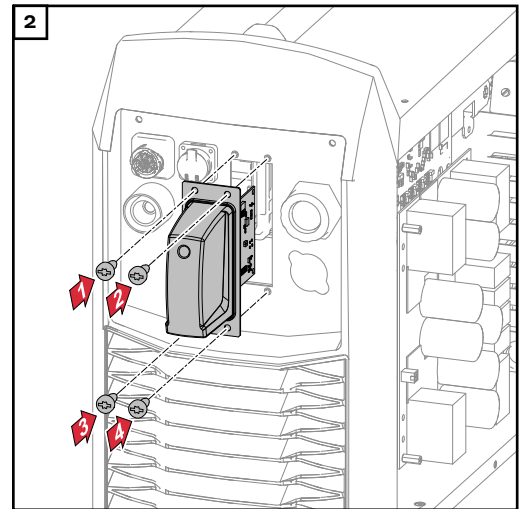
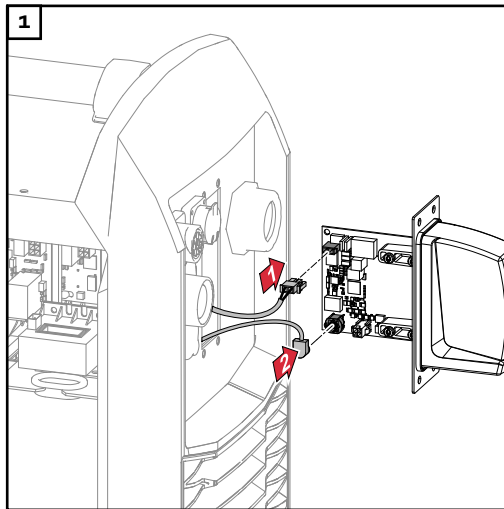




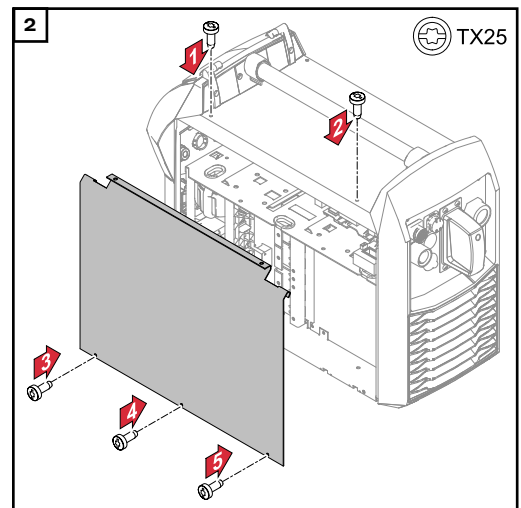
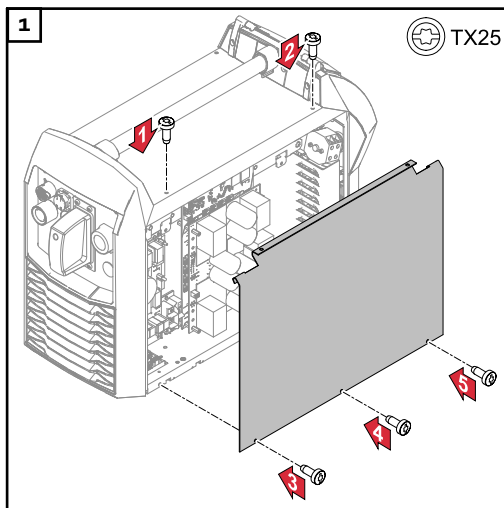
Routing the Data Cable



Installing the Robot Interface



Final Tasks



Installing the Bus Module

Safety



WARNING!

Danger from electrical current.

Serious injuries or death may result.

- Before starting work, switch off all devices and components involved, and disconnect them from the grid.
- Secure all devices and components involved so that they cannot be switched back on.



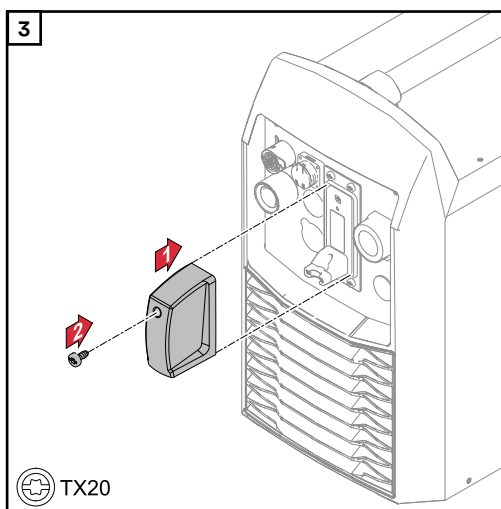
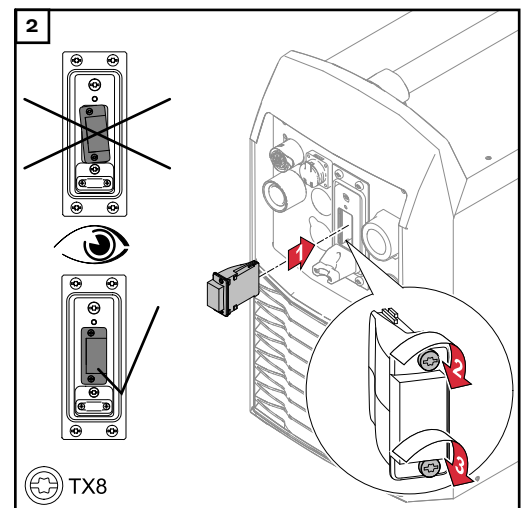
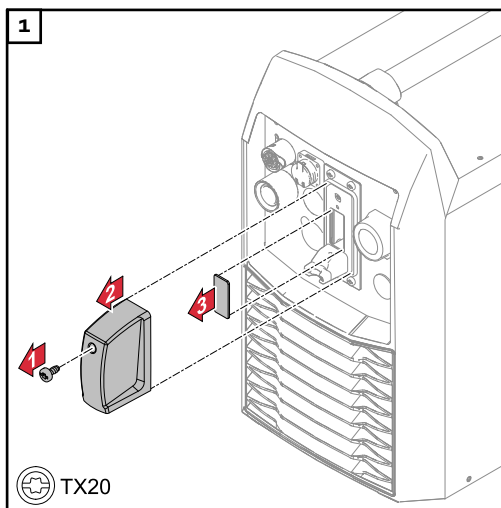
WARNING!

Danger from electrical current due to inadequate ground conductor connection.

Serious personal injury and property damage may result.

- Always use the original housing screws in the quantity initially supplied.

Installing the Bus Module



Input and output signals

Data types

The following data types are used:

- **UINT16** (Unsigned Integer)
Whole number in the range from 0 to 65535
- **SINT16** (Signed Integer)
Whole number in the range from -32768 to 32767

Conversion examples:

- for a positive value (SINT16)
e.g. desired wire speed x factor
 $12.3 \text{ m/min} \times 100 = 1230_{\text{dec}} = 04\text{CE}_{\text{hex}}$
- for a negative value (SINT16)
e.g. arc correction x factor
 $-6.4 \times 10 = -64_{\text{dec}} = \text{FFC0}_{\text{hex}}$

Input signals

From robot to welding machine

Applicable to firmware V4.1.0 and higher

| HEX address | Signal | | Data type Activity | Unit/range | Factor |
|-------------|----------------------|------------------------|--------------------|------------|--------|
| F000 | Control Flag Group 1 | | | | |
| | Bits 0 to 7 | Process active timeout | Byte | ms | 10 |
| | Bits 8 to 15 | Reserved | | | |

| HEX address | Signal | | Data type Activity | Unit/range | Factor |
|-------------|----------------------|--------------------|--------------------|------------|--------|
| F001 | Control Flag Group 2 | | | | |
| | Bit 0 | Welding start | Rising Edge | | |
| | Bit 1 | Robot ready | High | | |
| | Bit 2 | Source error reset | High | | |
| | Bit 3 | Gas on | Rising Edge | | |
| | Bit 4 | Wire inching | Rising Edge | | |
| | Bit 5 | Wire retract | Rising Edge | | |
| | Bit 6 | Torch blow out | Rising Edge | | |
| | Bit 7 | Welding simulation | High | | |
| | Bit 8 | Touch sensing | Rising Edge | | |
| | Bit 9 | Booster manual | High | | |
| | Bit 10 | SFI ON | High | | |
| | Bit 11 | Synchro pulse on | High | | |
| | Bit 12 | WireBrake | High | | |
| | Bit 13 | Torch XChange | High | | |
| | Bit 14 | Teach mode | High | | |
| | Bit 15 | Reserved | | | |

| HEX address | Signal | | Data type Activity | Unit/range | Factor |
|-------------|----------------------|------------------------------|--------------------|-------------------------------------------------------------------|--------|
| F002 | Control Flag Group 3 | | | | |
| | Bit 0 | Process line selection Bit 0 | High | See Value range Process line selection on page 21 | |
| | Bit 1 | Process line selection Bit 1 | High | | |
| | Bit 2 | TWIN mode Bit 0 | High | See Value Range for TWIN Mode on page 21 | |
| | Bit 3 | TWIN mode Bit 1 | High | | |
| | Bits 4 to 10 | Reserved | | | |
| | Bit 11 | Wire sense start | Rising Edge | | |
| | Bit 12 | Wire sense break | Rising Edge | | |
| | Bits 13 to 15 | Reserved | | | |
| F003 | Control Flag Group 4 | | | | |
| | Bit 0 | Documentation mode | High | See Value Range for Documentation mode on page 22 | |
| | Bits 1 to 4 | Reserved | | | |
| | Bit 5 | Motor type Bit 0 | High | See Value range for Motor Type on page 22 | |
| | Bit 6 | Motor type Bit 1 | High | | |
| | Bit 7 | Motor type Bit 2 | High | | |
| | Bits 8 to 15 | Reserved | | | |
| F004 | Control Flag Group 5 | | | | |
| | Bits 0 to 15 | Reserved | | | |
| F005 | Control Flag Group 6 | | | | |
| | Bits 0 to 15 | Reserved | | | |

| HEX address | Signal | | Data type Activity | Unit/range | Factor |
|-------------|----------------------|----------------------------------|--------------------|------------------------------------------------------------------------|--------|
| F006 | Control Flag Group 7 | | | | |
| | Bits 0 to 9 | Reserved | | | |
| | Bit 10 | Enable Start-End-Parameter | High | | |
| | Bit 11 | Enable components setup | High | | |
| | Bit 12 | Enable Unit / Standard | High | | |
| | Bits 13 to 15 | Reserved | | | |
| F007 | Control Flag Group 8 | | | | |
| | Bit 0 | ExtInput1 => OPT_Output 1 | High | | |
| | Bit 1 | ExtInput2 => OPT_Output 2 | High | | |
| | Bit 2 | ExtInput3 => OPT_Output 3 | High | | |
| | Bit 3 | ExtInput4 => OPT_Output 4 | High | | |
| | Bit 4 | ExtInput5 => OPT_Output 5 | High | | |
| | Bit 5 | ExtInput6 => OPT_Output 6 | High | | |
| | Bit 6 | ExtInput7 => OPT_Output 7 | High | | |
| | Bit 7 | ExtInput8 => OPT_Output 8 | High | | |
| | Bits 8 to 15 | Reserved | | | |
| F008 | Working mode | | | | |
| | Bit 0 | Working Mode Bit 0 | | See Value range for Working mode on page 22 | |
| | Bit 1 | Working Mode Bit 1 | | | |
| | Bit 2 | Working Mode Bit 2 | | | |
| | Bit 3 | Working Mode Bit 3 | | | |
| | Bit 4 | Working Mode Bit 4 | | | |
| | Bits 5 to 13 | Reserved | | | |
| | Bit 14 | Command value selection Bit 0 | High | See Value Range for Command value selection on page 22 | |
| | Bit 15 | Reserved | | | |
| F009 | Bits 0 to 15 | Job number | UINT16 | 0 to 1000 | |
| F00A | Bits 0 to 15 | Characteristic number (xml-file) | UINT16 | 0 to 65,535 | |
| F00B | Bits 0 to 15 | Feeder command value | SINT16 | -327.68 to 327.67 m/min | 100 |

| HEX address | Signal | | Data type Activity | Unit/range | Factor |
|---------------|--------------|--------------------------------|--------------------|------------------------------------|--------|
| F00C | Bits 0 to 15 | Arc length correction | SINT16 | -10 to +10 | 10 |
| F00D | Bit 0-15 | Pulse/Dynamic correction | SINT16 | -10 to +10 | 10 |
| F00E | Bits 0 to 15 | Wire retract | SINT16 | 0 to +10 | 10 |
| F00F | Bits 0 to 15 | Welding speed | UINT16 | 0 to 65,535 (0 to 6553.5 m/min) | 10 |
| F010 | Bit 0-15 | Penetration stabilizer | SINT16 | 0 to +10 | 10 |
| F011 | Bit 0-15 | Arc length stabilizer | UINT16 | 0 to +10 | 10 |
| F012- F019 | Bit 0-15 | Reserved | | | |
| F01A | Bits 0 to 15 | Wire forward / backward length | UINT16 | OFF/1 to 65,535 mm | 1 |
| F01B | Bits 0 to 15 | Wire sense edge detection | UINT16 | OFF/0.5 to 20.0 mm | 10 |
| F01C | Bit 0-15 | Reserved | | | |
| F01D | Bit 0-15 | Seam number | UINT16 | 0 to 65,535 | 1 |
| F01E- F031 | Bit 0-15 | Reserved | | | |

Value range Process line selection

| Bit 1 | Bit 0 | Description |
|-------|-------|--------------------------|
| 0 | 0 | Process line 1 (default) |
| 0 | 1 | Process line 2 |
| 1 | 0 | Process line 3 |
| 1 | 1 | Reserved |

Value range for process line selection

Value Range for TWIN Mode

| Bit 1 | Bit 0 | Description |
|-------|-------|------------------|
| 0 | 0 | TWIN Single mode |
| 0 | 1 | TWIN Lead mode |
| 1 | 0 | TWIN Trail mode |
| 1 | 1 | Reserved |

Value range for TWIN mode

Value Range for Documentation mode

| Bit 0 | Description |
|-------|-------------------------------------------|
| 0 | Seam number of welding machine (internal) |
| 1 | Seam number of robot |

Value range for documentation mode

Value range for Working mode

| Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Description |
|-------|-------|-------|-------|-------|--------------------------------------|
| 0 | 0 | 0 | 0 | 0 | Internal welding parameter selection |
| 0 | 0 | 0 | 0 | 1 | Special 2-step mode characteristics |
| 0 | 0 | 0 | 1 | 0 | Job mode |
| 0 | 1 | 0 | 0 | 0 | 2-step mode characteristics |
| 0 | 1 | 0 | 0 | 1 | 2-Step manual mode |
| 1 | 0 | 0 | 0 | 1 | Stop cooling pump |

Value range for operating mode

Value Range for Command value selection

| Bit 14 | Description |
|--------|---------------------------|
| 0 | Wirefeeder set value |
| 1 | Welding current set value |

Value range for set value

Value range for Motor Type

| Bit 2 | Bit 1 | Bit 0 | Description |
|-------|-------|-------|--------------------|
| 0 | 0 | 0 | Fronius wirefeeder |
| 0 | 0 | 1 | M500 |
| 0 | 1 | 0 | P-600Z |
| 0 | 1 | 1 | Reserved |
| 1 | 0 | 0 | Reserved |
| 1 | 0 | 1 | Reserved |
| 1 | 1 | 0 | Reserved |
| 1 | 1 | 1 | Reserved |

Output signals

From welding machine to robot
Applicable to firmware V4.1.0 and higher

| HEX address | Signal | | Data type Activity | Unit/range | Factor |
|-------------|---------------------|--------------------------------|--------------------|----------------------------------------------------------------------------------|--------|
| F100 | Status Flag Group 1 | | | | |
| | Bits 0 to 7 | Reserved | | | |
| F101 | Status Flag Group 2 | | | | |
| | Bit 0 | Heartbeat Powersource | Tppgle | 0.5 Hz | |
| | Bit 1 | Power source ready | High | | |
| | Bit 2 | Arc stable / Touch signal | High | | |
| | Bit 3 | Current flow signal | High | | |
| | Bit 4 | Main current signal | High | | |
| | Bit 5 | Collision protection | Low | Low = Collision | |
| | Bits 6 to 7 | Reserved | | | |
| | Bit 8 | Touch signal | High | | |
| | Bit 9 | Torchbody connected | High | | |
| | Bit 10 | Command value out of range | High | | |
| | Bit 11 | Correction out of range | High | | |
| | Bit 12 | Process active | High | | |
| | Bit 13 | Robot Motion Release | High | | |
| | Bit 14 | Wire stick workpiece | High | | |
| | Bit 15 | Reserved | | | |
| F102 | Status Flag Group 3 | | | | |
| | Bit 0 | Welding Mode Bit 0 | High | See Value range for welding process and process image on page 26 | |
| | Bit 1 | Welding Mode Bit 1 | High | | |
| | Bit 2 | Welding Mode Bit 2 | High | | |
| | Bit 3 | Welding Mode Bit 3 | High | | |
| | Bit 4 | Welding Mode Bit 4 | High | | |
| | Bits 5 to 7 | Reserved | | | |
| | Bit 8 | Parameter selection internally | High | | |
| | Bit 9 | Characteristic number valid | High | | |
| | Bit 10 | Reserved | | | |
| | Bit 11 | Process run | High | | |
| | Bits 12 to 13 | Reserved | | | |
| | Bit 14 | Process image Bit 0 | High | See Value range for welding process and process image on page 26 | |
| | Bit 15 | Process image Bit 1 | High | | |

| HEX address | Signal | | Data type Activity | Unit/range | Factor |
|-------------|---------------------|-----------------------------|--------------------|-------------------------------------------------------------------|--------|
| F103 | Status Flag Group 4 | | | | |
| | Bit 0 | Penetration stabilizier | High | | |
| | Bit 1 | Arclength stabilizier | High | | |
| | Bits 2 to 4 | Reserved | | | |
| | Bit 5 | Motor type Bit 0 | High | See Value range for Motor Type on page 22 | |
| | Bit 6 | Motor type Bit 1 | High | | |
| | Bit 7 | Motor type Bit 2 | High | | |
| | Bits 8 to 13 | Reserved | | | |
| | Bit 14 | Short circuit contact tip | High | | |
| | Bit 15 | Gas nozzle touched | High | | |
| F104 | Status Flag Group 5 | | | | |
| | Bit 0 | Sensor status 1 | High | See Assignment of Sensor Statuses 1–4 on page 26 | |
| | Bit 1 | Sensor status 2 | High | | |
| | Bit 2 | Sensor status 3 | High | | |
| | Bit 4 | Sensor status 4 | High | | |
| | Bits 4 to 10 | Reserved | | | |
| | Bit 11 | Safety status Bit 0 | High | See Value range Safety status on page 26 | |
| | Bit 12 | Safety status Bit 1 | High | | |
| | Bit 13 | Reserved | | | |
| | Bit 14 | Notification | High | | |
| | Bit 15 | System not ready | High | | |
| F105 | Status Flag Group 6 | | | | |
| | Bit 0 | Limit Signal | High | | |
| | Bits 1 to 8 | Reserved | | | |
| | Bit 9 | TWIN synchronization active | High | | |
| | Bit 10 | Main supply status | High | | |
| | Bit 11 | Standby active | High | | |
| | Bit 12 | Active process line bit 0 | High | See Value range Process line selection on page 21 | |
| | Bit 13 | Active process line bit 1 | High | | |
| | Bit 14 | Warning | High | | |
| | Bit 15 | Reserved | | | |

| HEX address | Signal | | Data type Activity | Unit/range | Factor |
|-------------|---------------------|-------------------------------|--------------------|---------------------------|--------|
| F106 | Status Flag Group 7 | | | | |
| | Bits 0 to 15 | Reserved | | | |
| F107 | Status Flag Group 8 | | | | |
| | Bit 0 | ExtOutput1 <= OPT_Input1 | High | | |
| | Bit 1 | ExtOutput2 <= OPT_Input2 | High | | |
| | Bit 2 | ExtOutput3 <= OPT_Input3 | High | | |
| | Bit 3 | ExtOutput4 <= OPT_Input4 | High | | |
| | Bit 4 | ExtOutput5 <= OPT_Input5 | High | | |
| | Bit 5 | ExtOutput6 <= OPT_Input6 | High | | |
| | Bit 6 | ExtOutput7 <= OPT_Input7 | High | | |
| | Bit 7 | ExtOutput8 <= OPT_Input8 | High | | |
| | Bits 8 to 15 | Reserved | | | |
| F108 | Bit 0-15 | Main error number | UINT16 | 0 to 65,535 | |
| F109 | Bit 0-15 | Warning number | UINT16 | 0 to 65,535 | 1 |
| F10A | Bit 0-15 | Welding voltage actual value | UINT16 | 0.0 to 327.67 volts | 100 |
| F10B | Bit 0-15 | Welding current actual value | UINT16 | 0.0 to 3276.7 amperes | 10 |
| F10C | Bit 0-15 | Motor current actual value M1 | SINT16 | -327.68 to 327.67 amperes | 100 |
| F10D | Bit 0-15 | Motor current actual value M2 | SINT16 | -327.68 to 327.67 amperes | 100 |
| F10E | Bit 0-15 | Motor current actual value M3 | SINT16 | -327.68 to 327.67 amperes | 100 |
| F10F | Bits 0 to 15 | Reserved | | | |
| F110 | Bit 0-15 | Wire speed actual value | SINT16 | -327.68 to 327.67 m/min | 100 |
| F111 | Bit 0-15 | Seam tracking actual value | UINT16 | 0 to 6.5535 | 10,000 |
| F112 | Bits 0 to 15 | Real energy actual value | UINT16 | 0 to 6553.5 kilo-joules | 10 |
| F113 | Bits 0 to 15 | Wire position | SINT16 | -327.68 to 327.67 mm | 100 |
| F114-F11F | Bit 0-15 | Reserved | | | |
| F120 | Bit 0-15 | External feeder command | SINT16 | -327.68 to 327.67 m/min | 100 |

| HEX address | Signal | | Data type Activity | Unit/range | Factor |
|-------------|----------|-----------------------------|--------------------|-----------------------|--------|
| F121 | Bit 0-15 | External feeder slope value | UINT16 | 0 to 6553.5 m/min/sec | 10 |
| F122-F126 | Bit 0-15 | Reserved | | | |

Value range for welding process and process image

| Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Description |
|-------|-------|-------|-------|-------|---------------------------|
| 0 | 0 | 0 | 0 | 0 | Internal mode selection |
| 0 | 0 | 0 | 0 | 1 | MIG/MAG pulsed synergic |
| 0 | 0 | 0 | 1 | 0 | MIG/MAG standard synergic |
| 0 | 0 | 0 | 1 | 1 | MIG/MAG PMC |
| 0 | 0 | 1 | 0 | 0 | MIG/MAG LSC |
| 0 | 0 | 1 | 0 | 1 | MIG/MAG standard manual |
| 0 | 0 | 1 | 1 | 0 | Electrode |
| 0 | 0 | 1 | 1 | 1 | TIG |
| 0 | 1 | 0 | 0 | 0 | CMT |

Value range for welding process

| Bit 15 | Bit 14 | Description |
|--------|--------|--------------------------|
| 0 | 0 | Standard image (CRC 1.0) |

Value range for process image

Assignment of Sensor Statuses 1–4

| Signal | Description |
|-----------------|---------------------------------------|
| Sensor status 1 | OPT/i WF R wire end (4,100,869) |
| Sensor status 2 | OPT/i WF R wire drum (4,100,879) |
| Sensor status 3 | OPT/i WF R ring sensor (4,100,878) |
| Sensor status 4 | Wire buffer set CMT TPS/i (4,001,763) |

Assignment of sensor statuses

Value range Safety status

| Bit 1 | Bit 0 | Description |
|-------|-------|-------------|
| 0 | 0 | Reserve |
| 0 | 1 | Hold |
| 1 | 0 | Stop |

| Bit 1 | Bit 0 | Description |
|-------|-------|------------------------|
| 1 | 1 | Not installed / active |

Value range Safety status

TAG table

- To read the following TAGs, use the mode function 03dec (03hex) - see section [103dec \(67hex\)](#) [Read Holding Register Float](#) from page 38
- To edit the following TAGs, use the mode function 06dec (06hex) - see section [104dec \(68hex\)](#) [Write Single Register Float](#) from page 39

| HEX address | Signal | Access | Type | Range | Unit | Step size |
|-------------|----------------------------------------|-------------------|-------|---------------|-------|-----------|
| D000 / E064 | Gas preflow [Gpr] | Reading & writing | FLOAT | 0.0 to 9.9 | s | 0.1 |
| D001 / E065 | Gas postflow [Gpo] | Reading & writing | FLOAT | 0.0 to 9.9 | s | 0.1 |
| D002 / E0A3 | Inching speed [Fdi] | Reading & writing | FLOAT | 0.5 to vD-max | m/min | 0.1 |
| D003 / E032 | SynchroPulse DeltaWireFeed | Reading & writing | FLOAT | 0.1 to 6.0 | m/min | 10 |
| D004 / E031 | SynchroPulse Frequency | Reading & writing | FLOAT | 0.5 to 10.0 | Hz | 10 |
| D005 / E033 | SynchroPulse DutyCycle | Reading & writing | FLOAT | 10 to 90 | % | 1 |
| D006 / E034 | SynchroPulse ArcLength Correction High | Reading & writing | FLOAT | -10.0 to 10.0 | | 10 |
| D007 / E035 | SynchroPulse ArcLength Correction Low | Reading & writing | FLOAT | -10.0 to 10.0 | | 10 |
| D008 / E06A | Starting current [I-S] | Reading & writing | FLOAT | 0.0 to 200.0 | % | 1 |
| D009 / E011 | Start Arclength Correction | Reading & writing | FLOAT | -10.0 to 10.0 | | 0.1 |
| D00A / E056 | Starting Current Time [t-S] | Reading & writing | FLOAT | 0.0 to 10.0 | s | 0.1 |
| D00B / E06B | Slope 1 | Reading & writing | FLOAT | 0.0 to 9.9 | s | 0.1 |
| D00C / E06C | Slope 2 | Reading & writing | FLOAT | 0.0 to 9.9 | s | 0.1 |

| HEX address | Signal | Access | Type | Range | Unit | Step size |
|-------------|---------------------------|-------------------|-------|--------------------------------------------------------|-------|-----------|
| D00D / E06D | End current [I-E] | Reading & writing | FLOAT | 0.0 to 200.0 | % | 1 |
| D00E / E012 | End arc length correction | Reading & writing | FLOAT | -10.0 to 10.0 | | 0.1 |
| D00F / E057 | End Current Time [t-e] | Reading & writing | FLOAT | 0.0 to 10.0 | s | 0.1 |
| D010 / E02E | SFI HotStart | Reading & writing | FLOAT | 0.01 to 2.00 | s | 0.01 |
| D011 / E0BF | Ignition time out | Reading & writing | FLOAT | 5 to 100 | mm | 1 |
| D012 / E09E | Cooling unit mode | Reading & writing | FLOAT | See table Cooling unit mode on page 29 | | |
| D013 | Cooler filter time | Reading & writing | FLOAT | 5 to 25 | s | 5 |
| D014 | Cooler flow warning level | Reading & writing | FLOAT | 0.75 to 0.95 | l/min | 0.01 |
| D015 | Touch sensitivity | Reading & writing | FLOAT | 0 to 10 | | 1 |
| D016 / E06F | Language | Reading & writing | FLOAT | See table Language on page 29 | | |
| D017 | Units | Reading & writing | FLOAT | See table Unit on page 30 | | |
| D018 | Welding standard | Reading & writing | FLOAT | See table Welding standard on page 30 | | |
| D100 / F10B | Error number | Reading | FLOAT | 0 to 65,535 | | 1 |
| D101 / E062 | Min. feeder value | Reading | FLOAT | 0.0 to 100.0 | m/min | 0.1 |
| D102 / E063 | Max. feeder value | Reading | FLOAT | 0.0 to 100.0 | m/min | 0.1 |
| D103 / E0A6 | Hourmeter Current flow | Reading | FLOAT | 0.0 to 1,000,000 | h | 0.1 |
| D104 / E0A7 | Hourmeter Power on | Reading | FLOAT | 0.0 to 1,000,000 | h | 0.1 |
| D105 / E0AA | Power value | Reading | FLOAT | 0.1 to 1,000,000 | kW | 0.1 |
| D106 / E0AB | Real energy value | Reading | FLOAT | 0.1 to 1,000,000 | kJ | 0.1 |

| HEX address | Signal | Access | Type | Range | Unit | Step size |
|-------------|-------------------|---------|-------|--------------|-------|-----------|
| D107 / E0BB | Coolertemperature | Reading | FLOAT | -100 to 200 | °C | 0.1 |
| D108 / E0BC | Coolerflow | Reading | FLOAT | -100 bis 100 | l/min | 0.01 |

Cooling unit mode

| Value | Description |
|-------|-------------|
| 20e34 | Eco |
| 13e34 | Auto |
| 11e34 | On |
| 12e34 | Off |

Language

| Value | Description |
|-------|-------------|
| 8e34 | English |
| 9e34 | German |
| 58e34 | Japanese |
| 10e34 | Chinese |
| 23e34 | Spanish |
| 24e34 | French |
| 25e34 | Czech |
| 26e34 | Hungarian |
| 27e34 | Italian |
| 28e34 | Norwegian |
| 29e34 | Polish |
| 30e34 | Portuguese |
| 31e34 | Slovak |
| 32e34 | Turkish |
| 33e34 | Russian |
| 34e34 | Swedish |
| 35e34 | Estonian |
| 36e34 | Finnish |
| 39e34 | Lithuanian |
| 40e34 | Latvian |
| 41e34 | Dutch |

| Value | Description |
|--------|-------------|
| 42e34 | Slovenian |
| 43e34 | Romanian |
| 44e34 | Croatian |
| 59e34 | Ukrainian |
| 61e34 | Korean |
| 66e34 | Icelandic |
| 67e34 | Vietnamese |
| 70e34 | Thai |
| 71e34 | Indonesian |
| 75e34 | Serbian |
| 76e34 | Hindi |
| 130e34 | Tamil |
| 151e34 | Danish |
| 156e34 | Bulgarian |

Unit

| Value | Description |
|-------|-------------|
| 37e34 | Metrisch |
| 38e34 | Imperial |

Welding standard

| Value | Description |
|-------|-------------|
| 49e34 | AWS |
| 57e34 | CEN |

Modbus – General Information

Protocol Description

The MODBUS ADU is constructed by the client that initiates the MODBUS transaction. The function tells the server which action is to be performed. The MODBUS application protocol defines the format of a client-initiated request.

The function code field of a MODBUS data unit is coded in one byte. Valid codes are in the range of 1 ... 255 decimal (the range 128-255 is reserved for exception responses). When the server receives a message from a client, the function code field tells the server which action to perform.

If several actions are to be performed, subfunction codes are added to some function codes. When messages are sent to servers by a client, the data field in the message contains additional information that the server uses to perform the action defined by the function code. This can include elements such as discrete addresses, register addresses, the quantity to be handled, or the number of actual data bytes contained within the field.

With certain types of request, there might not be a data field (length: zero). In this case, the server does not require any additional information because the action is specified by the function code alone.

If a MODBUS ADU is correctly received without any errors occurring in connection with the requested MODBUS function, the requested data will be included in the data field when a server responds to a client. If an error does occur in connection with the requested MODBUS function, the field will contain an exception code that the server application can use to determine what action to perform next.

For instance, a client can read the ON/OFF statuses of a group of discrete inputs or outputs, or it can read/write the data contents of a group of registers.

When sending a response to the client, the server uses the function code field either to indicate that the response is normal (free of errors) or that an error has occurred (this kind of response is called an "exception response"). In the case of a normal response, the server simply echoes the original function code.

Data Coding

For addresses and data elements, MODBUS uses a big-endian format. When a number larger than a single byte is transmitted, this means that the most significant byte is sent first.

| Register Size | Value |
|------------------------------|------------------------------------------------------------------------|
| 16 bits, 1234 _{hex} | 12 _{hex} is sent as the first byte and then 34 _{hex} |

Application Data Unit (ADU)

This section describes the encapsulation method used for a MODBUS request or response when it is transmitted over a MODBUS TCP network.

| | | |
|-------------|---------------|------|
| MPAP header | Function code | Data |
|-------------|---------------|------|

| Description of MPAP header: | |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------|
| Transaction Identifier Used to allocate the transaction. The MODBUS server copies the Transaction Identifier of the request into the response. | |
| Transaction Identifier This is used for transaction pairing. The MODBUS server copies the transaction identifier from the request into the response. | |
| Length: | 2 bytes |
| Description: | For identifying a MODBUS request/response transaction |
| Client: | Initialized by the client |
| Server: | Copied back by the server from the request received |
| Protocol Identifier This is used for multiplexing within the system. The MODBUS protocol is identified by the value 0. | |
| Length: | 2 bytes |
| Description: | 0 = Modbus protocol |
| Client: | Initialized by the client |
| Server: | Copied back by the server from the request received |
| Length This field is used to specify the number of bytes in the field to follow, including the unit identifier, function code, and data field. | |
| Length: | 2 bytes |
| Description: | Number of bytes to follow |
| Client: | Initialized by the client |
| Server: | - |
| Unit Identifier This field is used for routing within the system. It is usually used for communication with a serially connected MODBUS- or MODBUS+ slave where communication takes place via a gateway between an Ethernet network and a serial MODBUS line. The field value is set in the request by the MODBUS client and must be replicated exactly in the response from the server. | |
| Length: | 1 byte |
| Description: | For identifying a remote slave that is connected via a serial line or other type of bus. |
| Client: | Initialized by the client |

All MODBUS/TCP ADUs are sent via TCP on registered port 502.

Modbus Functions

03_{dec} (03_{hex}) Read Holding Register

This code is used to read the contents of a contiguous block of holding registers in a remote device. The request PDU determines the starting register address and the number of registers.

The registers are addressed in the PDU starting at zero. This means registers numbered 1-16 will be addressed using 0-15.

The register data in the response message is packed as two bytes per register, with the binary contents precisely aligned/justified within each byte. Within the individual registers, the first byte contains the high-order bits and the second byte the low-order bits.

| Request | | |
|---------------------|---------|--------------------------------------------|
| Function code | 1 byte | 03 _{hex} |
| Start address | 2 bytes | 0000 _{hex} to FFFF _{hex} |
| Number of registers | 2 bytes | 1 to 125 (7D _{hex}) |

| Response | | |
|--------------------------|--------------|-------------------|
| Function code | 1 byte | 03 _{hex} |
| Number of bytes | 2 bytes | 2 x N* |
| Register value | N* x 2 bytes | - |
| N* = Number of registers | | |

| Errors | | |
|----------------|--------|----------------------|
| Error code | 1 byte | 83 _{hex} |
| Exception code | 1 byte | 01 or 02 or 03 or 04 |

| Example Example of a read request for register F009 (job number). | | | |
|----------------------------------------------------------------------|-----|---------------------------|-----|
| Request | | Response | |
| Field name | Hex | Field name | Hex |
| Transaction Identifier Hi | 00 | Transaction Identifier Hi | 00 |
| Transaction Identifier Lo | 01 | Transaction Identifier Lo | 01 |
| Protocol Identifier Hi | 00 | Protocol Identifier Hi | 00 |
| Protocol Identifier Lo | 00 | Protocol Identifier Lo | 00 |
| Length Hi | 00 | Length Hi | 00 |
| Length Lo | 06 | Length Lo | 05 |
| Unit Identifier | 00 | Unit Identifier | 00 |
| Function code | 03 | Function code | 03 |
| Starting Address Hi | F0 | Byte Count | 02 |
| Starting Address Lo | F9 | Register value Hi (108) | 02 |
| No. of Registers Hi | 00 | Register value Lo (108) | 37 |

| Example Example of a read request for register F009 (job number). | | | |
|----------------------------------------------------------------------|-----|------------|-----|
| Request | | Response | |
| Field name | Hex | Field name | Hex |
| No. of Registers Lo | 01 | | |

The contents of register F009 (job number) are displayed in the form of the two-byte values 237_{hex} or 567_{dec}.

06_{dec} (06_{hex}) Write Single Register

This function code is used to write a single holding register in a remote device. The request PDU specifies the address of the register to be written. Registers are addressed starting at zero. This means that the register that has been numbered as 1 will be addressed using 0. The normal response is an echo of the request, which is returned after the register contents are written.

| Request | | |
|------------------|---------|--------------------------------------------|
| Function code | 1 byte | 06 _{hex} |
| Register address | 2 bytes | 0000 _{hex} to FFFF _{hex} |
| Register value | 2 bytes | 0000 _{hex} or FFFF _{hex} |

| Response | | |
|------------------|---------|--------------------------------------------|
| Function code | 1 byte | 06 _{hex} |
| Register address | 2 bytes | 0000 _{hex} to FFFF _{hex} |
| Register value | 2 bytes | 0000 _{hex} or FFFF _{hex} |

| Errors | | |
|----------------|--------|----------------------|
| Error code | 1 byte | 86 _{hex} |
| Exception code | 1 byte | 01 or 02 or 03 or 04 |

| Example Example request for writing the value 237 _{hex} (567 _{dec}) to register F009 (job number). | | | |
|--------------------------------------------------------------------------------------------------------------------------|-----|---------------------------|-----|
| Request | | Response | |
| Field name | Hex | Field name | Hex |
| Transaction Identifier Hi | 00 | Transaction Identifier Hi | 00 |
| Transaction Identifier Lo | 01 | Transaction Identifier Lo | 01 |
| Protocol Identifier Hi | 00 | Protocol Identifier Hi | 00 |
| Protocol Identifier Lo | 00 | Protocol Identifier Lo | 00 |
| Length Hi | 00 | Length Hi | 00 |
| Length Lo | 06 | Length Lo | 06 |
| Unit Identifier | 00 | Unit Identifier | 00 |
| Function code | 06 | Function code | 06 |
| Register Address Hi | F0 | Register Address Hi | F0 |

Example
Example request for writing the value 237_{hex} (567_{dec}) to register F009 (job number).

| Request | | Response | |
|---------------------|-----|---------------------|-----|
| Field name | Hex | Field name | Hex |
| Register Address Lo | 09 | Register Address Lo | 09 |
| Register Value Hi | 02 | Register Value Hi | 02 |
| Register Value Lo | 37 | Register Value Lo | 37 |

16_{dec} (10_{hex}) **Write Multiple Register**

This function code is used to write a block of contiguous registers in a remote device. The requested written values are specified in the request data field. Data is packed as two bytes per register. The normal response returns the function code, the starting address, and the number of registers written.

| Request | | |
|----------------------------------------|--------------|--------------------------------------------|
| Function code | 1 byte | 10 _{hex} |
| Starting address | 2 bytes | 0000 _{hex} to FFFF _{hex} |
| Number of registers | 2 bytes | 0001 _{hex} or 0078 _{hex} |
| Number of bytes | 1 byte | 2 x N* |
| Register values | N* x 2 bytes | Value |
| N* = number of registers to be written | | |

| Response | | |
|---------------------|---------|--------------------------------------------|
| Function code | 1 byte | 10 _{hex} |
| Starting address | 2 bytes | 0000 _{hex} to FFFF _{hex} |
| Number of registers | 2 bytes | 1 to 123 (7B _{hex}) |

| Errors | | |
|----------------|--------|----------------------|
| Error code | 1 byte | 90 _{hex} |
| Exception code | 1 byte | 01 or 02 or 03 or 04 |

Example
Example request for writing two registers (F00B_{hex} – F00C_{hex}).

| Request | | Response | |
|---------------------------|-----|---------------------------|-----|
| Field name | Hex | Field name | Hex |
| Transaction Identifier Hi | 00 | Transaction Identifier Hi | 00 |
| Transaction Identifier Lo | 01 | Transaction Identifier Lo | 01 |
| Protocol Identifier Hi | 00 | Protocol Identifier Hi | 00 |
| Protocol Identifier Lo | 00 | Protocol Identifier Lo | 00 |
| Length Hi | 00 | Length Hi | 00 |
| Length Lo | 11 | Length Lo | 11 |

| Example Example request for writing two registers (FOOB _{hex} – FOOC _{hex}). | | | |
|----------------------------------------------------------------------------------------------------|-----|--------------------------|-----|
| Request | | Response | |
| Field name | Hex | Field name | Hex |
| Unit Identifier | 00 | Unit Identifier | 00 |
| Function code | 10 | Function code | 10 |
| Starting Address Hi | F0 | Starting Address Hi | F0 |
| Starting Address Lo | 0B | Starting Address Lo | 0B |
| Quantity of Registers Hi | 00 | Quantity of Registers Hi | 00 |
| Quantity of Registers Lo | 02 | Quantity of Registers Lo | 02 |
| Byte Count | 04 | | |
| Register Value Hi | 04 | | |
| Register Value Lo | CE | | |
| Register Value Hi | FF | | |
| Register Value Lo | C0 | | |

23_{dec} (17_{hex}) Read/Write Multiple Register

This function code performs a combination of one read operation and one write operation in a single MODBUS transaction. The write operation is performed before the read operation.

Holding registers are addressed starting at zero. This means that holding registers 1-16 will be addressed in the PDU using 0-15.

The request PDU specifies:

- The starting address and number of holding registers to be read
- The starting address, number of holding registers, and data for the write operation.

The byte count field specifies the number of bytes to follow in the write data field.

The normal response contains the data from the group of registers read. The byte count field specifies the number of bytes to follow in the read data field.

| Request | | |
|------------------------------|--------------|----------------------------------------------------|
| Function code | 1 byte | 17 _{hex} |
| Read starting address | 2 bytes | 0000 _{hex} to FFFF _{hex} |
| Number of registers to read | 2 bytes | 0001 _{hex} to approx. 0076 _{hex} |
| Write starting address | 2 bytes | 0000 _{hex} to FFFF _{hex} |
| Number of registers to write | 2 bytes | 0001 _{hex} to approx. 0076 _{hex} |
| Write number of bytes | 1 byte | 2 x N* |
| Write register values | N* x 2 bytes | |

| Request | | |
|----------------------------------------|--|--|
| N* = number of registers to be written | | |

| Response | | |
|-------------------------------------|--------------|-------------------|
| Function code | 1 byte | 17 _{hex} |
| Number of bytes | 1 byte | 2 x N* |
| Write register values | N* x 2 bytes | |
| N* = number of registers to be read | | |

| Errors | | |
|----------------|--------|----------------------|
| Error code | 1 byte | 97 _{hex} |
| Exception code | 1 byte | 01 or 02 or 03 or 04 |

| Example Example request for reading 2 registers and writing 2 registers. | | | |
|-----------------------------------------------------------------------------|-----|---------------------------|-----|
| Request | | Response | |
| Field name | Hex | Field name | Hex |
| Transaction Identifier Hi | 00 | Transaction Identifier Hi | 00 |
| Transaction Identifier Lo | 01 | Transaction Identifier Lo | 01 |
| Protocol Identifier Hi | 00 | Protocol Identifier Hi | 00 |
| Protocol Identifier Lo | 00 | Protocol Identifier Lo | 00 |
| Length Hi | 00 | Length Hi | 00 |
| Length Lo | 11 | Length Lo | 7 |
| Unit Identifier | 00 | Unit Identifier | 00 |
| Function code | 17 | Function code | 17 |
| Read Starting Address Hi | F1 | Byte Count | 2 |
| Read Starting Address Lo | 0A | Read Registers Value Hi | 04 |
| Quantity to Read Hi | 00 | Read Registers Value Lo | 08 |
| Quantity to Read Lo | 2 | Read Registers Value Hi | 0A |
| Write Starting Address Hi | F0 | Read Registers Value Lo | C8 |
| Write Starting Address Lo | 0B | | |
| Quantity to Write Hi | 00 | | |
| Quantity to Write Lo | 04 | | |
| Write Byte Count | 2 | | |
| Write Registers Value Hi | 04 | | |
| Write Registers Value Lo | CE | | |
| Write Registers Value Hi | FF | | |
| Write Registers Value Lo | C0 | | |
| Transaction Identifier Hi | 00 | | |

103_{dec} (67_{hex})
Read Holding
Register Float

This function is used to read the contents of a contiguous block of registers in the TAG tables contained in this document. The register uses floating-point format (32 bits). The request PDU determines the starting register address and the number of registers.

The registers are addressed in the PDU starting at zero. This means registers numbered 1-16 will be addressed using 0-15.

The register data in the response message is packed as two bytes per register, with the binary contents precisely aligned/justified within each byte. Within the individual registers, the first byte contains the high-order bits and the second byte the low-order bits.

| Requirement | | |
|---------------------|---------|--------------------------------------------|
| Function code | 1 byte | XX _{hex} |
| Starting address | 2 bytes | XXXX _{hex} to XXXX _{hex} |
| Number of registers | 2 bytes | 1 to 125 (7D _{hex}) |

| Response | | |
|--------------------------|--------------|-------------------|
| Function code | 1 byte | 03 _{hex} |
| Number of bytes | 2 bytes | 2 x N* |
| Register value | N* x 2 bytes | - |
| N* = number of registers | | |

| Error | | |
|----------------|---------|----------------------|
| Error code | 1 bytes | 83 _{hex} |
| Exception code | 1 byte | 01 or 02 or 03 or 04 |

| Example Example read request for register E064 _{hex} (gas pre-flow): | | | |
|----------------------------------------------------------------------------------|-----|---------------------------|-----|
| Requirement | | Response | |
| Field Name | Hex | Field Name | Hex |
| Transaction Identifier Hi | 00 | Transaction Identifier Hi | 00 |
| Transaction Identifier Lo | 01 | Transaction Identifier Lo | 01 |
| Protocol Identifier Hi | 00 | Protocol Identifier Hi | 00 |
| Protocol Identifier Lo | 00 | Protocol Identifier Lo | 00 |
| Length Hi | 00 | Length Hi | 00 |
| Length Lo | 06 | Length Lo | 05 |
| Unit Identifier | 00 | Unit Identifier | 00 |
| Function code | 67 | Function code | 67 |
| Starting Address Hi | E0 | Byte Count | 02 |
| Starting Address Lo | 64 | Register Value High Hi | 3F |
| No. of Registers Hi | 00 | Register Value High Lo | C0 |
| No. of Registers Lo | 01 | Register Value Low Hi | 00 |
| | | Register Value Low Lo | 00 |

The contents of register E064_{hex} (gas pre-flow) are displayed in the form of the two-byte values 3FC00000 or 1.5_{dec}.

104_{dec} (68_{hex}) Write Single Register Float

This function is used to edit registers in the TAG tables contained in this document. The register uses floating-point format (32 bits). The request PDU specifies the address of the register to be written. Registers are addressed starting at zero. This means that the register that has been numbered as 1 will be addressed using 0.

The normal response is an echo of the request, which is returned after the register contents are written.

| Requirement | | |
|------------------|---------|------------------------------------------------|
| Function code | 1 byte | 68 _{hex} |
| Register address | 2 bytes | E000 _{hex} to Exxx _{hex} |
| Register value | 2 bytes | 0000 _{hex} or FFFFFFFF _{hex} |

| Response | | |
|------------------|---------|------------------------------------------------|
| Function code | 1 byte | 68 _{hex} |
| Register address | 2 bytes | E000 _{hex} to Exxx _{hex} |
| Register value | 2 bytes | 0000 _{hex} or FFFFFFFF _{hex} |

| Error | | |
|----------------|---------|-------------------|
| Error code | 1 bytes | E8 _{hex} |
| Exception code | 1 byte | 01 or 02 or 03 |

Example
Example request for writing the value 3FC00000_{hex} (1.5_{dec}) to register E064_{hex} (gas pre-flow):

| Requirement | | Response | |
|---------------------------|-----|---------------------------|-----|
| Field Name | Hex | Field Name | Hex |
| Transaction Identifier Hi | 00 | Transaction Identifier Hi | 00 |
| Transaction Identifier Lo | 01 | Transaction Identifier Lo | 01 |
| Protocol Identifier Hi | 00 | Protocol Identifier Hi | 00 |
| Protocol Identifier Lo | 00 | Protocol Identifier Lo | 00 |
| Length Hi | 00 | Length Hi | 00 |
| Length Lo | 08 | Length Lo | 08 |
| Unit Identifier | 00 | Unit Identifier | 00 |
| Function code | 68 | Function code | 68 |
| Register Address Hi | E0 | Register Address Hi | E0 |
| Register Address Lo | 64 | Register Address Lo | 64 |
| Register Value High Hi | 3F | Register Value Hi | 45 |
| Register Value High Lo | C0 | Register Value Lo | 09 |
| Register Value Low Hi | 00 | Register Value Hi | 80 |

Example
Example request for writing the value 3FC00000_{hex} (1.5_{dec}) to register E064_{hex} (gas pre-flow):

| Requirement | | Response | |
|-----------------------|-----|-------------------|-----|
| Field Name | Hex | Field Name | Hex |
| Register Value Low Lo | 00 | Register Value Lo | 00 |



Fronius International GmbH

Froniusstraße 1
4643 Pettenbach
Austria
contact@fronius.com
www.fronius.com

At www.fronius.com/contact you will find the contact details
of all Fronius subsidiaries and Sales & Service Partners.