UM1040
User manual

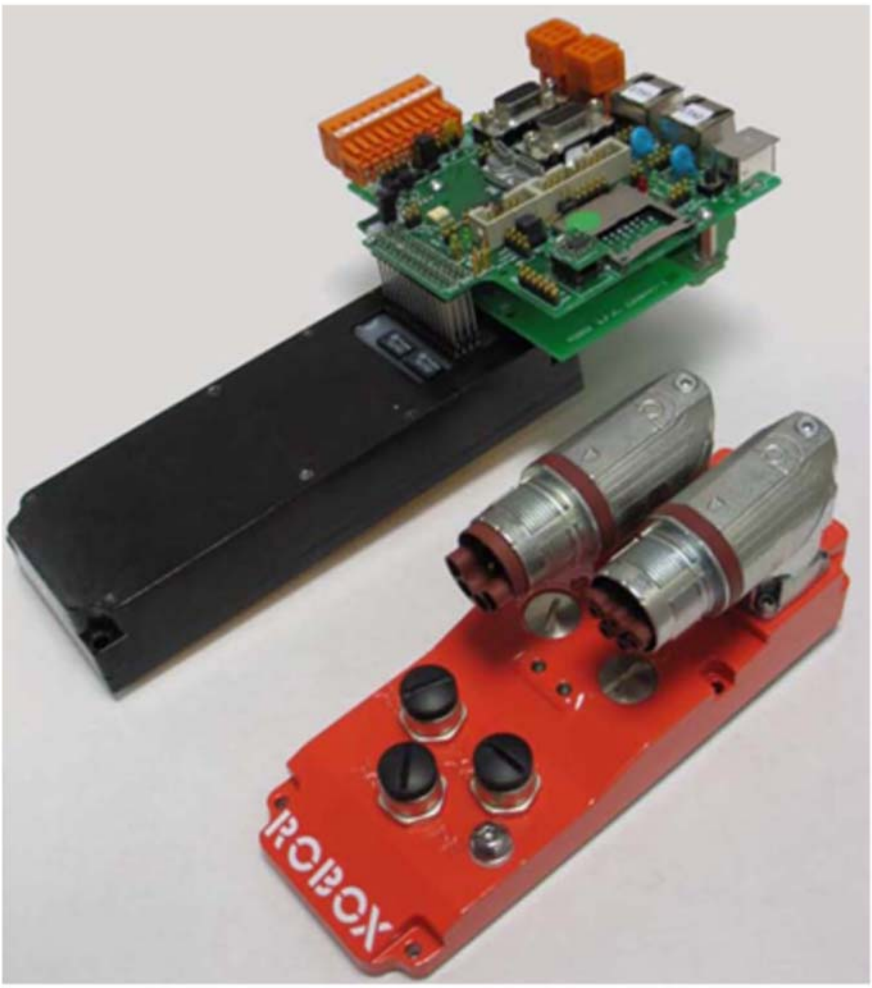
STEVAL-SPIMD20 demonstration kit based on the SPIMD20 integrated motor drive

Introduction

The STEVAL-SPIMD20 demonstration kit provides a complete demonstration platform for evaluating the functionality and the performance of the SPIMD20 power drive system designed by STMicroelectronics in cooperation with ROBOX S.p.A.

The SPIMD20 is an integrated motor drive with real-time connectivity enabling brushless motor manufacturers to create a proprietary motion control system based on a general purpose brick.

Figure 1. STEVAL-SPIMD20 demonstration kit



Contents

1 STEVAL-SPIMD20 kit contents

The STEVAL-SPIMD20 demonstration kit contains the following items:

- 2 kW shuttle module (SPIMD20)
- Software package
- Connection board adapter (08447.501)
- Junction tower for mounting on motor (AS3036.001)
- Cables and connectors
- User manuals

This manual refers to the use of the SPIMD20 with the junction tower (AS3036.001) and the Robox RDE programming suite (supplied with this kit). All these operations can also be carried out using the connection adapter board (refer to the other documentation in the kit) which outputs all the communication interfaces of the SPIMD20.

Figure 2. STEVAL-SPIMD20 demonstration kit package

2 Connection and initialization

To employ the SPIMD20 module with the software supplied in the kit, the following steps are required:

- Electrical connection of the SPIMD20 to the motor and the junction tower AS3036.00 (see *Section Appendix A: Connectors diagrams*)
- SPIMD20 initialization
- Software loading
- Initialization of the SPIMD20/motor base parameters
- Execution of the autotuning and parameter saving procedures
- Execution of the motor control tests.

2.1 Electrical connection of the SPIMD20 to the motor and the junction tower AS3036.001

1. Connect the SPIMD20 to the motor following the indications from the “Pinout description” section of the SPIMD20 datasheet, and install it on the motor.
2. Install the junction tower (code AS3036.001) on the SPIMD203.
3. Connect the cable CAVO2SPIMD/2.5/2 to the auxiliary supply (24 VDC) and to the DC bus bar (up to 800 VDC)

2.2 SPIMD20 initialization

1. Connect the serial cable CAVO1SPIMD/RS232/1.5 to the appropriate connector of the junction tower (code AS3036.001) and to the PC
2. Turn on the auxiliary voltage (18/48 VDC) on the SPIMD20 by pressing and holding the PB button of the serial cable. This sets the SPIMD20 to BOOT mode (the 2 LEDs - red and green - on the junction tower blink simultaneously)
3. Open a HyperTerminal connection on the PC serial port with the settings: 115200,8,n,1, no flow control
4. Press the “ENTER” key to display the operations menu
5. Press key “2” to format the Flash memory and remove the older software versions and parameter files (if present)

2.3 Loading the software

In the HyperTerminal menu, press key “3” to enter the software loading mode and start the loading session:

1. Right-click in the HyperTerminal menu > send file > SPIMD20_LOCAL_vnn_nn_nnnn.rbx
2. When the loading session is terminated, press “0” to reset the SPIMD20 (or switch the SPIMD20 off and on) and close the HyperTerminal window
3. The BOOT software tests the presence of application software and updates the system before running it (after a few seconds the two LEDS on the junction tower blink again)

The basic software provides three interface options:

– SPIMD20_LOCAL_vnn_nn_nnnn.rbx (Local interface): allows control of the SPIMD20 from the PC via the RS232 serial connection, and to parameterize the SPIMD20 and perform the current, speed, and autotuning tests.
– SPIMD20_CANOP_vnn_nn_nnnn.rbx (CANopen interface): allows the operation of the SPIMD20 as a slave of a CANopen master with the DS301/DSP402 protocols. This software version also contains the Local interface, which is selectable with the dedicated command in the shell partition. In this way the CANopen interface is completely bypassed and the SPIMD20 is controlled only via the serial channel.
– SPIMD20_ECAT_CoE_vnn_nn_nnnn.rbx (interface Can Over EtherCAT): allows the use of the SPIMD20 as a slave of an EtherCAT master with the CoE DSP402 protocol.This software version also contains the Local interface, which is selectable with the dedicated command in the shell partition. In this way the CANopen interface is completely bypassed and the SPIMD20 is controlled only via the serial channel.

2.4 Initialization of the SPIMD20/motor base parameters

1. Install the Robox RDE programming suite and copy the licence file into the installation folder
2. Start RDE
3. Open the workspace “SPIMD20_Local.rws”
4. If connecting for the first time, set the communication port in the Workspace window: Menu Display > WorkSpace Windows, to open the window
5. Double-click Connection > BCC/31 on RS232 > mycomm, and set the COM port of the PC (speed=autodetection, handshake=none)
6. Save the workspace to avoid repeating the procedure at the next startup
7. Enter the command “sysinfo” into the shell command bar and press Enter to check the answer, to verify that communication is running correctly
8. Activate the IMD configuration tool by pressing the relevant key in the toolbar and enter “Live” mode
9. Set the NMT machine state to “Pre Operational” mode by pressing the relevant key (some parameters can be set only in this mode)

Note: The dictionary tooltip is available on each configurator item. Refer to the IMD Local Interface SW Manual for each entry description.

10. Set the SPIMD20 basic parameters in the following menus:

- Configuration > Transducer > Transducer Type:

Select the transducer type and fill in the corresponding parameter menu:

for resolver

– number of poles
– frequency of the driving signal (4/8 KHz)
– disactivate the resolver fault alarm

for rotary EnDat

– configure the number of bits per rotation for linear EnDAT

- configure the number of bits per polar step

for sincos read through the resolver input

- disactivate the sincos fault alarm

- Configuration > Motor
  - number of poles
  - resistance and inductance phase-to-phase
  - torque constant
  - nominal and max. torque (used only in the autotuning session)
  - motor max. speed
  - temperature alarm threshold which must be enabled if the motor installs the correct type of probe for the SPIMD20 (see the SPIMD20 datasheet)
- Configuration > Holding Brake:

Parameters for a holding brake connected to the SPIMD20 (for the power supply and the brake max. absorption, see the SPIMD20 datasheet).

- Configuration > Drive:

“Max current” alarm enable:

- configure the thresholds for the used DCBUS and enable the relevant diagnosis
- set the required thresholds and enable the temperature alarm for the heatsink and the control board

- Configuration > Advanced Settings:
  - –Confirm the parameters’ default configuration
  - Configuration > Digital I/O:
  - I/O 1 and I/O 2 are digital inputs
  - I/O 3 and I/O 4 are digital outputs
  - Loop > Speed Reference > torque limit and current filters. Configure the required torque limits

Note: For the auto-tuning it is recommended to set the motor nominal torque value in all three fields

- Save the parameters into the SPIMD20 Flash through the dedicated key in the configurator toolbar
- At the end of the parameter save session, restart the SPIMD20 by switching off and on the auxiliary supply or by entering the command “SWRESET” in the shell partition (some parameters are detected only at startup)

2.5 Execution of the autotuning and parameter saving procedures

If connecting, for the first time, motors installing EnDat type transducers, it is necessary to temporarily save the EnDat phase into the retentive memory of the transducer, in order to reset the relevant alarm.

- enter the "Live" mode
- select the "Pre Operational" state on the NMT state machine
- select the menu Configuration > Transducer > EnDat (Rotary Encoders)
- press StorePhase

When the autotuning test "Calculate and Store EnDat Phase" is executed, this temporary value is overwritten with the correct value.

1. 1.Supply the DC-BUS to the SPIMD20 (ensure that the bulk capacitors are correctly connected)
2. Enter "Live" mode and select the "Operational" state on the NMT state machine
3. Select the autotuning menu depending on the motor transducer type: UTILITIES > AUTOTUNING > Motor with xxx
4. 4. Select the tests to be run and press "Start Test". To abort a selected test when already running press "Disable Drive".

At the end of the tests remember to save the calculated parameters into the Flash memory.

The following autotuning tests are available (following their internal execution sequence if more than one is selected):

- Calculate Theoretical C_LOOP gains

  Calculates the current loop theoretical gains following the values of the parameters "Motor Resistance", "Motor Inductance" and "Current Loop Frequency".
- Calculate C_LOOP gains

  Calculates the current loop gains by positioning a current vector modulated with a square wave on the "U" phase, with torque value changing from 10% to 50% of the nominal torque parameter.

  A very low proportional gain is used at the beginning with integral gain = 0.

  The proportional gain is then doubled every time, until the current loop oscillates.It is then progressively reduced by 10% of the oscillation value, until the oscillation disappears. The correct proportional gain is therefore obtained. This operation is repeated for the integral gain.
- Tune Resolver Driving Wave

  Refers to motors installing BRX resolvers.If it is activated with other types of transducers, an autotuning alarm indicates that the transducer is wrong.

  A fixed current vector is positioned on the "U" phase. The sinusoidal driving wave is tuned in such a way that phase is synchronized to the current loop, the maximum width is adjusted according to requirements and the max. and min. thresholds, to output the resolver fault alarm, are also set. The resolver alarm is then enabled.
- Calculate Resolver Phase

  Refers to motors installing BRX resolvers.

If it is activated with other types of transducers, an autotuning alarm indicates that the transducer is wrong. A fix current vector is positioned on the “U” phase. The offset angle between the transducer and the motor (Resolver phase) is calculated.

- Calculate and Store EnDat Phase

  Refers to motors installing EnDat resolvers.

  If it is activated with other types of transducers, an autotuning alarm indicates that the transducer is wrong.

  A fixed current vector is positioned on the “U” phase. The offset angle between the transducer and the motor (EnDat phase) is calculated and then saved in the EnDat retentive memory. The SPIMD20 transducer is then automatically initialized.

- Calculate SinCos Phase

  Refers to motors installing SinCos resolvers read through the resolver input of the SPIMD20.

  If it is activated with other types of transducers, an autotuning alarm indicates that the transducer is wrong.

  A fixed current vector is positioned on the “U” phase. The offset angle between the transducer and the motor (Resolver phase) is calculated.

- Check Motor Wiring

  Checks that the motor phases sequence and the transducer wiring are correct, so that the speed loop is correctly closed.

  It checks that the ratio between the number of the motor poles and those of the transducer is correct.

  A driving sequence of the current vector is executed. The vector is firstly positioned on the “U” phase, then on the “V” phase, and in the end, on the “W” phase.

  Note that he correct phase sequence must ensure that the motor moves clockwise (view from the shaft side) when the current tests are activated with an adequate

  reference and the current vector is positioned in sequence on the U, V, and W phases. If not, the motor phases wiring must be reversed.

  With this sequence of the current vector the position read by the position transducer must increase in a positive direction by the correct number of bits, depending on the motor/transducer poles ratio.

  If this test results in a “Delta Angle” alarm, it means that the torque value is too small to execute the test.

  If the polar sequence and the transducer direction are not correct, it results in a “Motor Wiring” alarm.

  With motors installing resolvers the transducer direction is inverted, for instance, reversing the signals cos+ and cos-.

  The software has been designed following the regulation of the BRX resolvers with the following wiring colors:

  – Primary Exc+ (red/white) Exc- (yellow/white or black/white)
  – Secondary Sin+ (yellow) Sin- (blue)
  – Secondary Cos+ (red) Cos- (black)

  If a motor with rotary EnDat is installed, the parameter “EnDat Rotation Sign” in the menu Configuration > Transducer > EnDat (Rotary Encoders) must be reversed (it can be modified only in “Pre Operational” mode, remember to save it to the Flash memory then return to “operational” mode).

- Calculate S_LOOP gains

  To be used only on rotary motors and if the “Check Motor Wiring” test was terminated without errors.

  It calculates the speed loop gains by generating a speed square wave with the reference value changing from 0% and 10% of the max. speed parameter.

  A very low proportional gain is entered at first, the integral gain = 0 and the speed filter “Number of Speed Samples” = 1.

  The proportional gain is doubled from time to time until the speed loop oscillates.

  The proportional gain is progressively reduced by 10% of the oscillation value, until the oscillation disappears.

  The correct value for the proportional gain is therefore obtained.

  This operation is repeated for the integral gain.In the case of a too high speed ripple, the number of speed samples is automatically increased and the test is repeated.
- Execution of the motor tests

  Supply the DC-BUS to the SPIMD20 (ensure that the bulk capacitors are correctly wired to the connector).

  Enter the “Live” mode and set the NMT state machine in “Operational” mode.

  Select the current or speed test menu:

  UTILITIES > TORQUE TEST or UTILITIES > SPEED TEST

  When the test is activated, its parameters are re-initialized with their default values and their setpoints are zeroed
- Torque test

  It is possible to generate a torque reference using a wave form synthesizer capable of generating a constant value, a sinusoidal wave, a square wave, or any combination of these.

  If a square wave is selected, the generator returns a value +1 (if high state) or -1 (if low state) which is multiplied by the reference. The frequency and duty cycle are programmable.

  If a sinusoidal wave is selected, the generator returns a sinusoidal wave whose value ranges from +1 to -1 and which is multiplied by the reference. The frequency is programmable.

  If a combination of the two waves is selected, the square wave is added to the sinusoidal wave.

  Finally, a constant value can be added.

  If working for a “forced” electrical angle, the value read by the position transducer is not used. Instead, the angle value is read from the “Requested Electrical Angle”.

  Value 30 corresponds to the vector positioned on the U phase, value 150 corresponds to the vector positioned on the V phase, and value 270 corresponds to the vector positioned on the W phase.

  If you do not force the electrical angle, the motor will follow the transducer read-out and will start running at the max possible speed.

  It is also possible to generate a turning current vector turning the forced angle.

  The motor will then rotate. To get such behaviour, select a "Speed for forced electrical Angle" different from 0 and check that the flag "Requested speed on forced Electrical Angle".
- Speed test

It is possible to generate a speed reference using a wave form synthetizer able to generate a constant value, a sinusoidal wave, a square wave, or any combination of them.

If a constant value is selected an acceleration can be programmed in order to ramp at the selected speed.

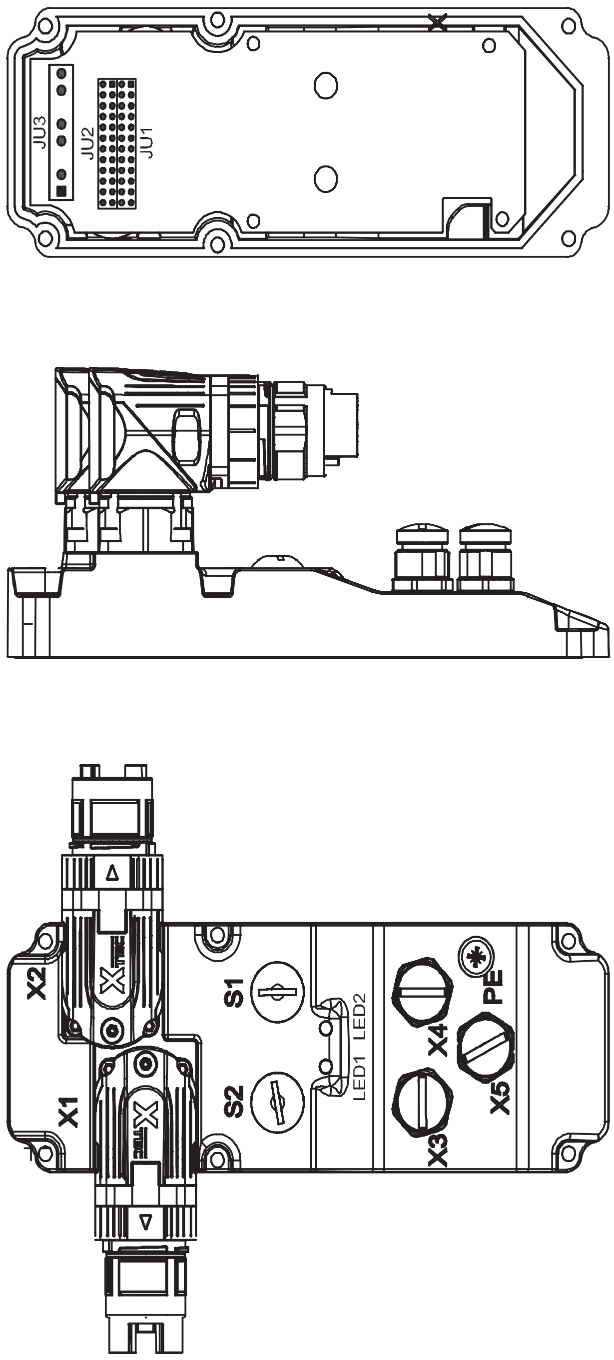
If a square wave is selected the generator returns a value +1 (if high state) or -1 (if low state) which is multiplied by the reference. The frequence and duty-cycle are programmable.

If a sinusoidal wave is selected the generator returns a sinusoidal wave whose value ranges from +1 to -1 and which is multiplied by the reference. The frequence is programmable.

Any combination of the above can be programmed. In this case the ramp generator is excluded.

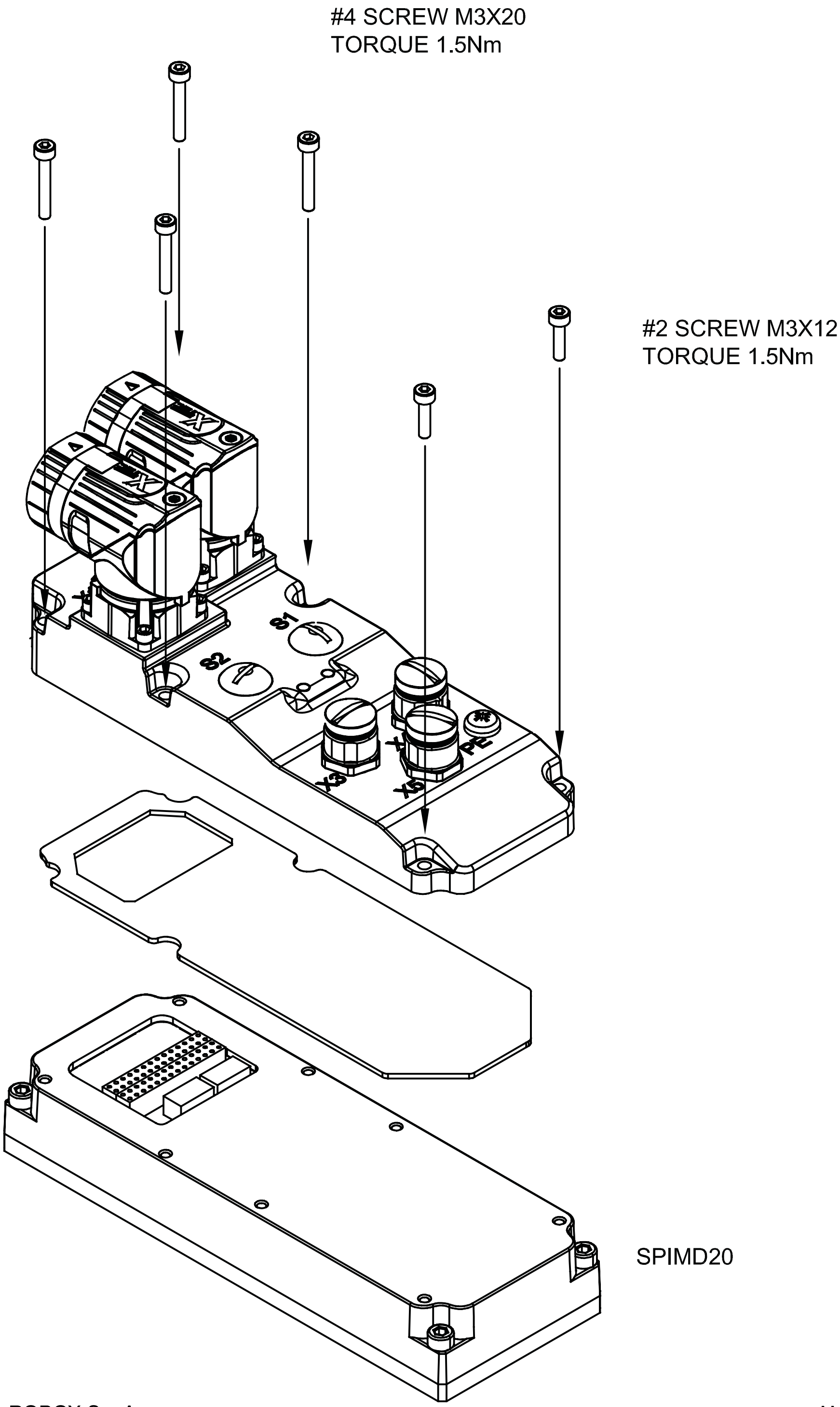
Appendix A Connectors diagrams

Figure 3. AS3036.001 junction tower layout



Drawing by ROBOX S.p.A. AM00196v1

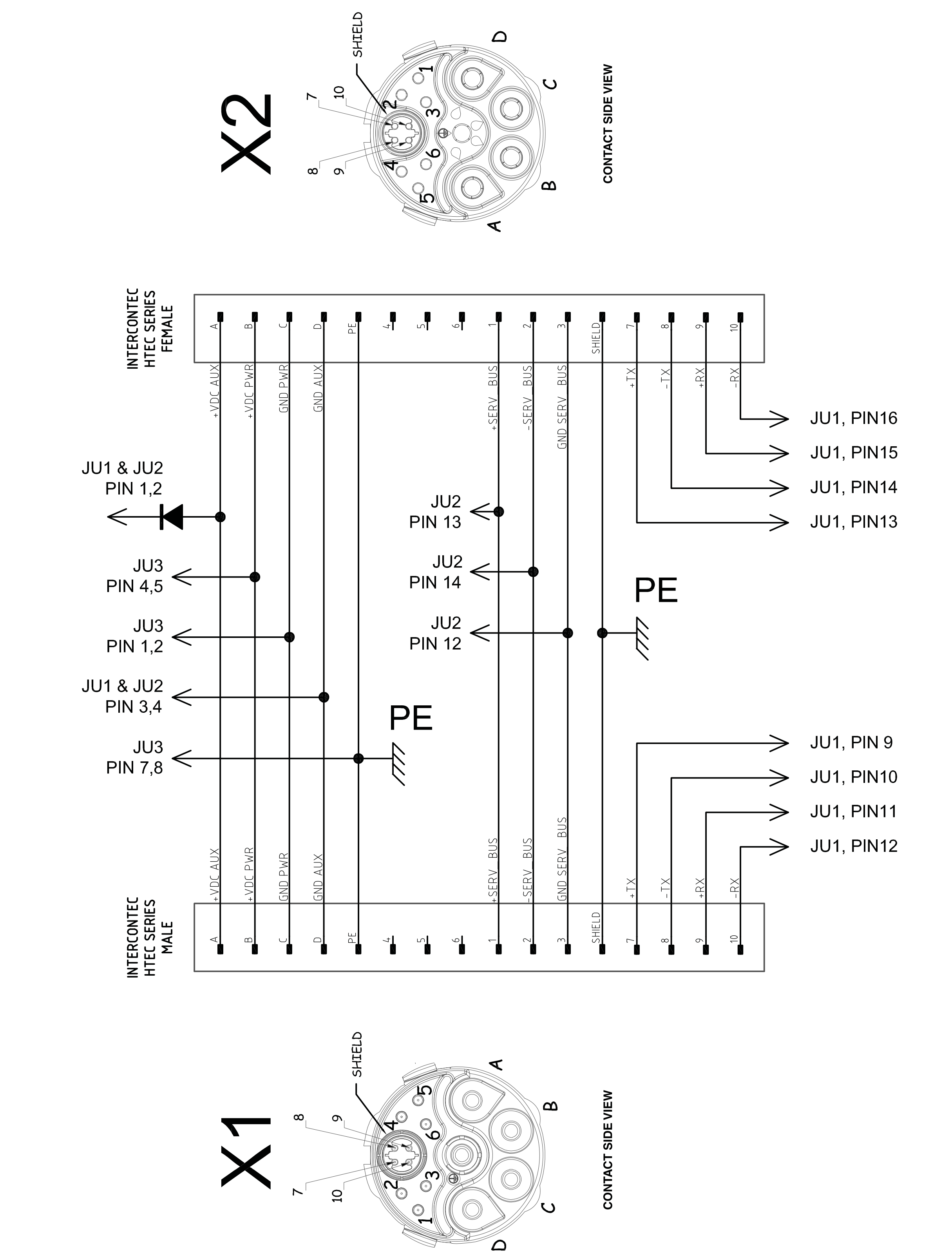
Figure 4. AS3036.001 junction tower mounting diagram



Drawing by ROBOX S.p.A.

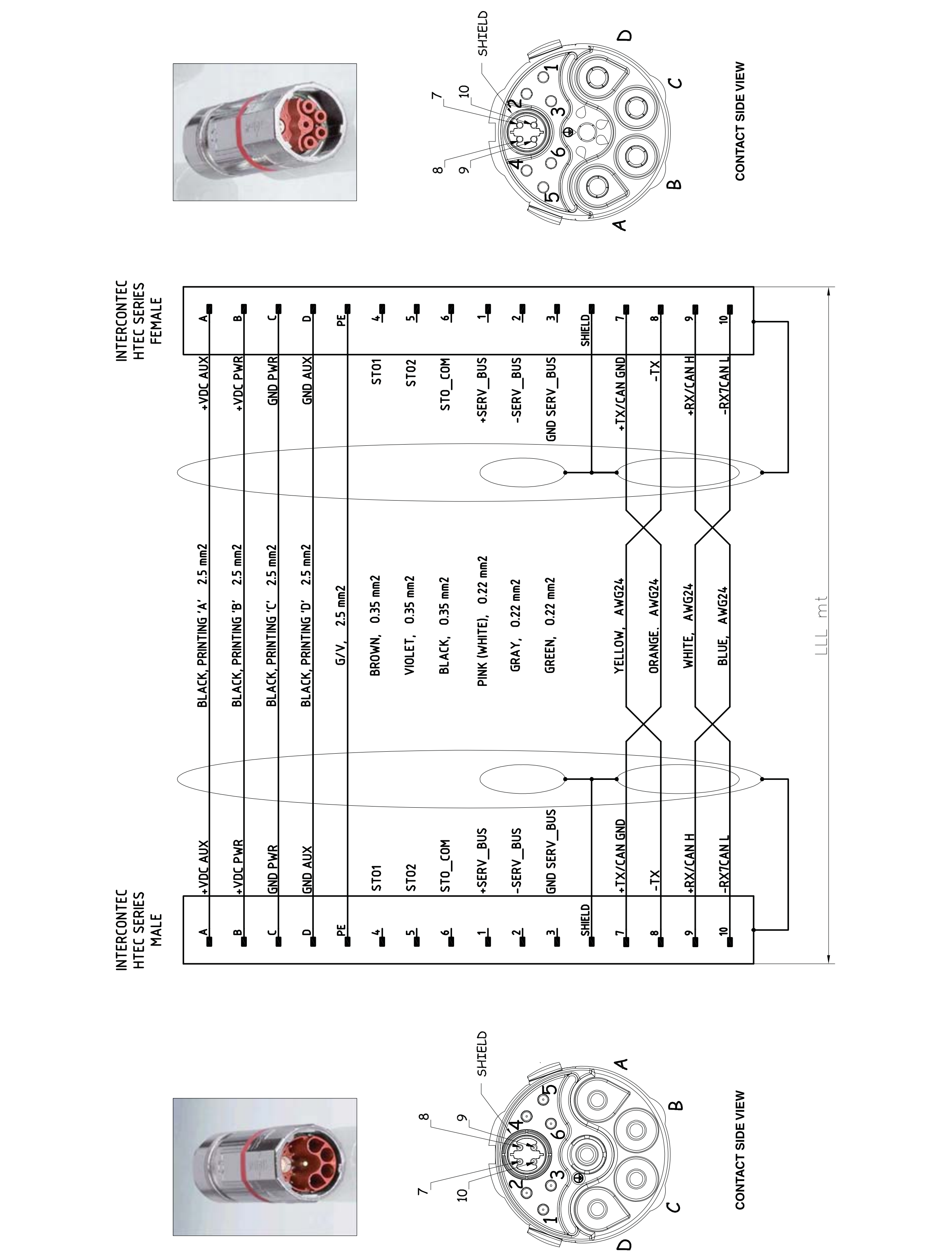
AM00197v1

Figure 5. AS3036.001 junction tower connector



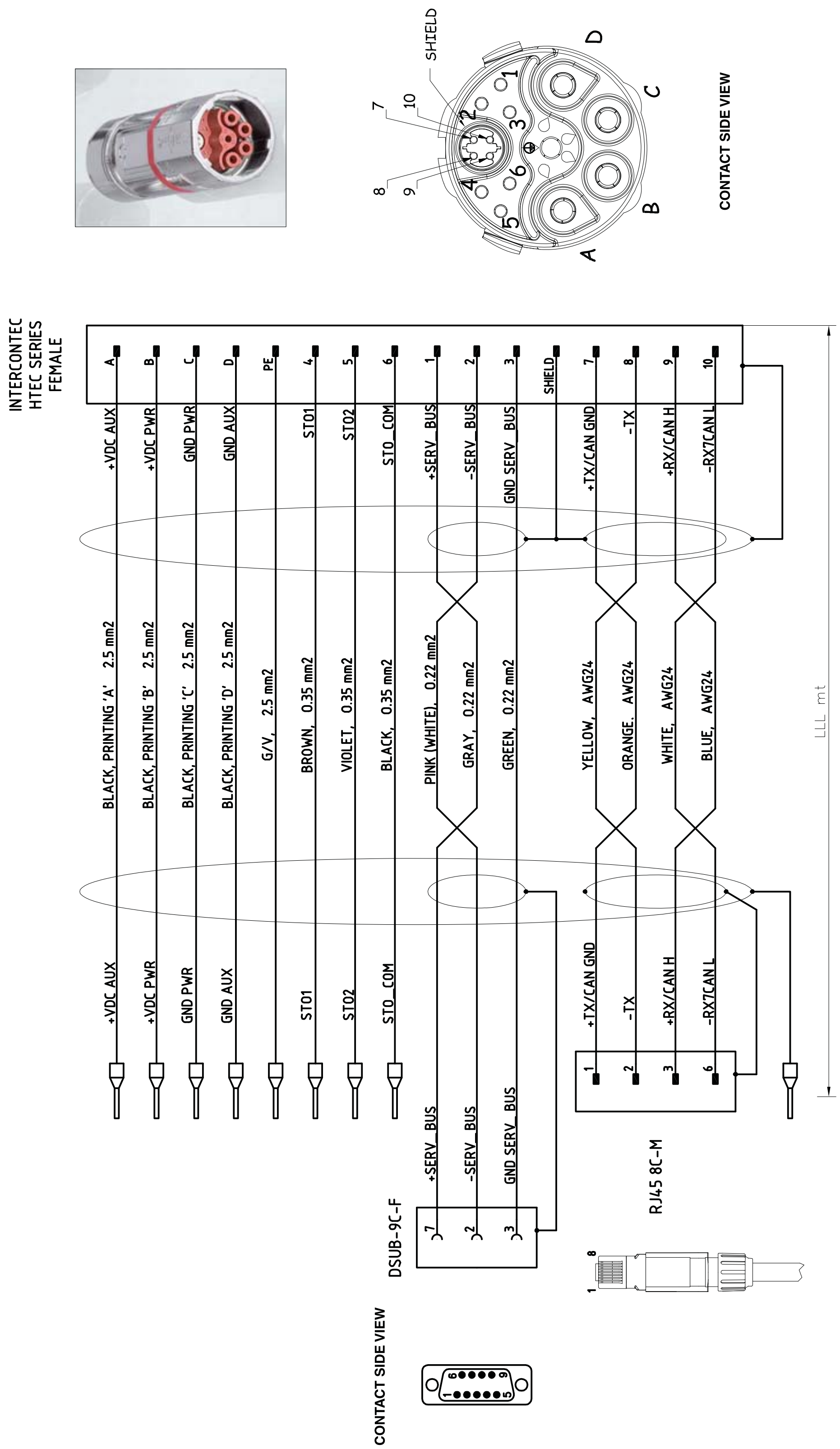
Drawing by ROBOX S.p.A. AM00198v1

Figure 6. CAVO1SPIMD/2.5 connector



Drawing by ROBOX S.p.A. AM00199v1

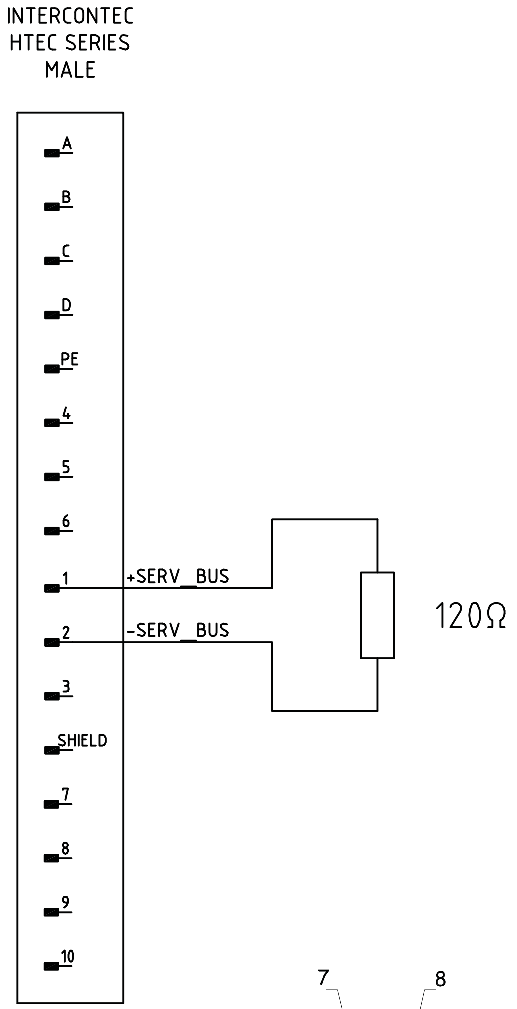
Figure 7. CAVO1SPIMD/2.5/2 connector

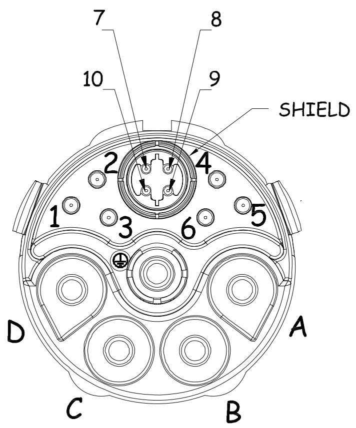


Drawing by ROBOX S.p.A.

AM00200v1

Figure 8. CAVO1SPIMD terminal

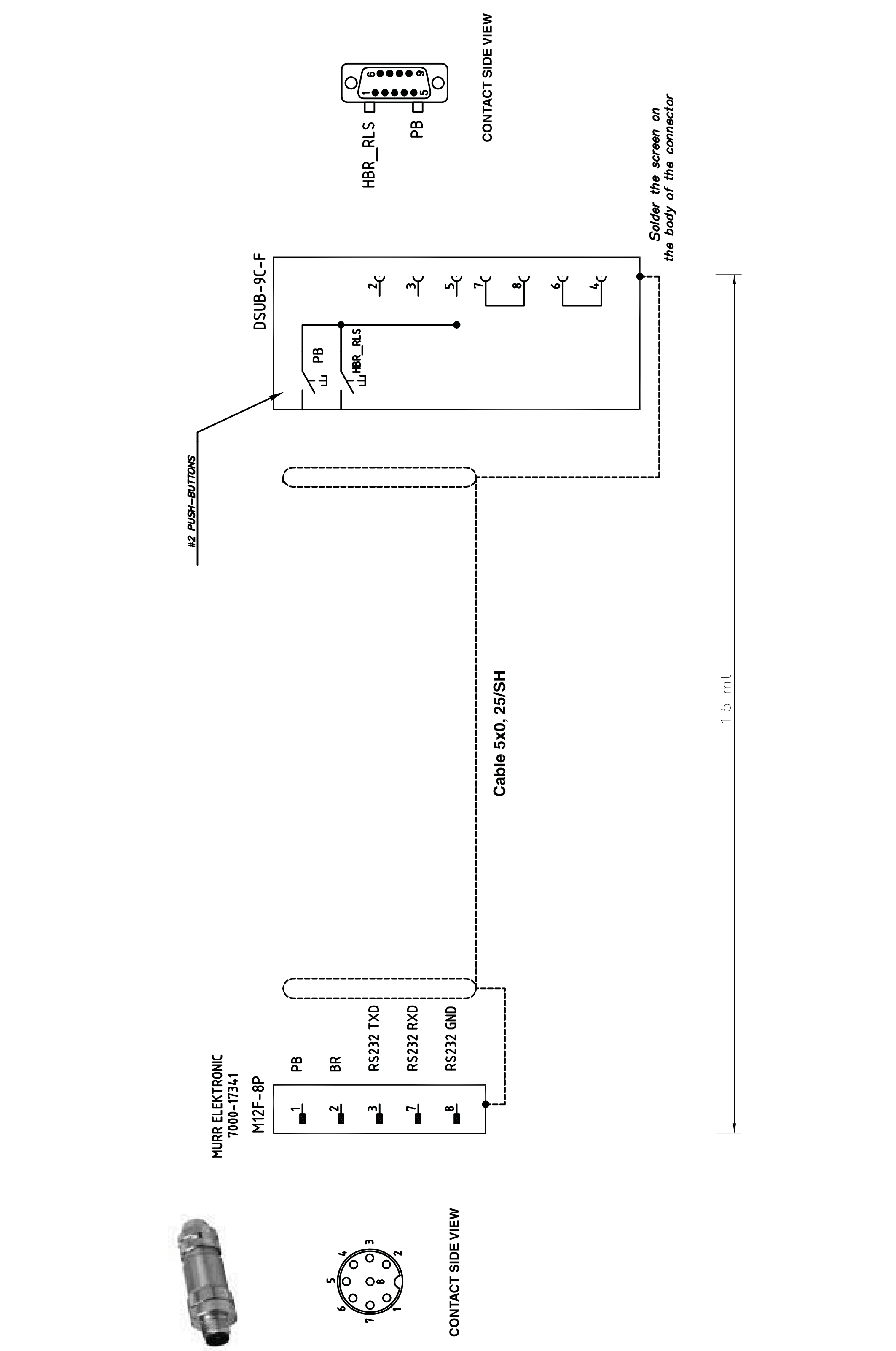




CONTACT SIDE VIEW

Drawing by ROBOX S.p.A. AM00201v1

Figure 9. CAVO1SPIMD/RS232/1.5 connector diagram



Drawing by ROBOX S.p.A. AM00202v1

3 Revision history

Table 1. Document revision history

Date	Revision	Changes
17-Jan-2011	1	Initial release.
19-Jan-2011	2	Corrected the operational state to be selected from “Pre Operational” to “Operational” in list item 2 of *Section 2.5: Execution of the autotuning and parameter saving procedures on page 7*.
21-Dec-2011	3	Updated *Section l: Torque test* and added section *Section l: Speed test on page 9*