Manual control software

Introduction

This document aims at presenting the basis of the manual control software and how to operate it. It will present the required hardware and how to connect everything together. It also presents the software requirements and how to install them.
The positioner is controlled by the motor control PCB. Two interface PCBs shrink the power delivered to the 3mm motors. The motor control PCB communicates with the computer using a CAN protocol through a CAN-USB transceiver. Each of the 4 PCB provided is configured with a unique ID from 1 to 4 that is noted on the bottom of the support. A specific control software issues the commands to the positioner.

Hardware

Inventory

The hardware required to operate one positioner consists of the following items

- 1 power supply capable of delivering 12V, 2A
- 1 Motor control PCB
- 1 daisy chain interface cable
- 1 power supply cable
- 1 CAN-USB transceiver with USB cable
- 1 computer operating under windows 10

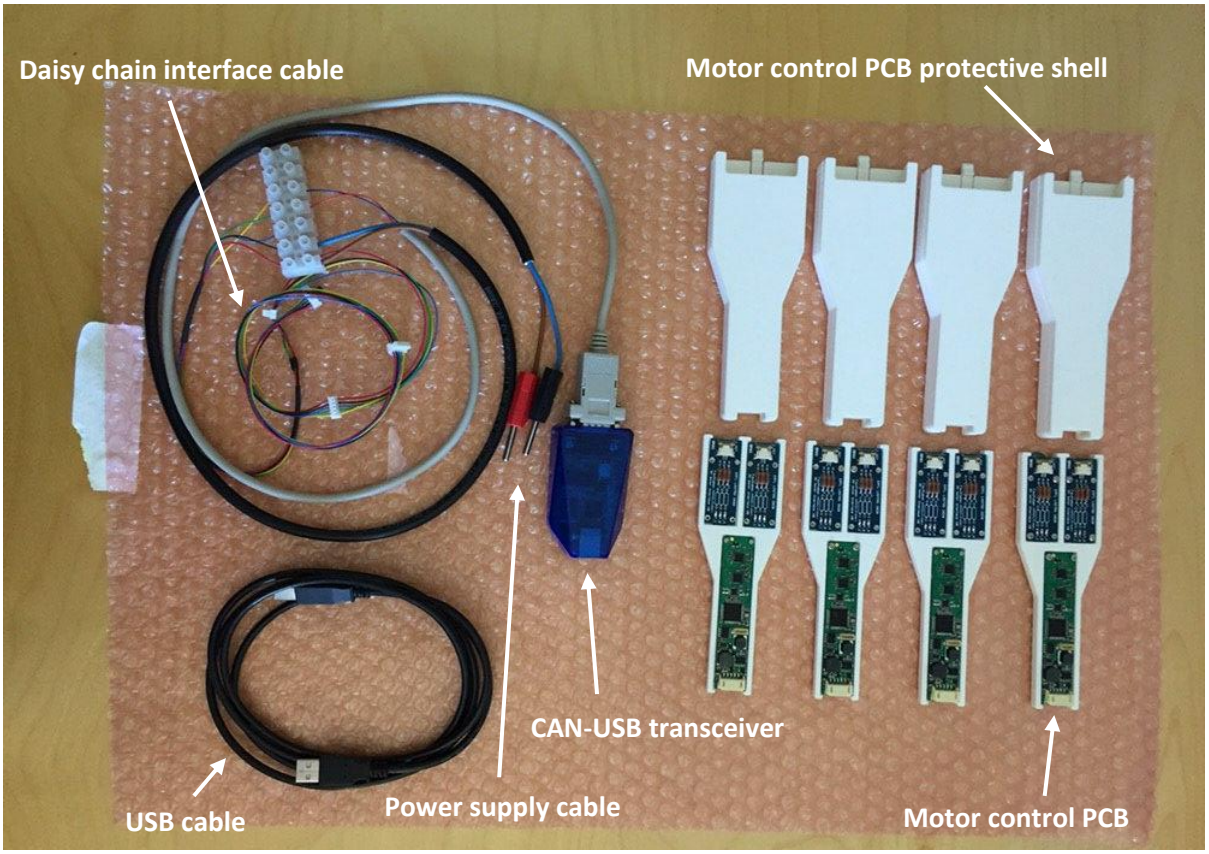

Figure 1: Box content

Installation

Power supply configuration

The motor control PCB accepts input voltages from 8V to 24V and can draw up to 5W power. It is recommended to configure the power supply to deliver 12V and at least 0.5A. In case more positioners are used on the daisy chain, the current shall be increased accordingly.
The black power wire shall be connected to the negative pole of the power supply and the red one to the positive pole. To avoid any tension spikes, the power supply must be turned on and the output enabled before plugging in the power wires. The negative (black) wire shall be connected before the positive (red) wire.

CAN-USB configuration

The CAN-USB transceiver must first be configured before it can be used.

1. Plug the CAN-USB in the computer using the USB cable
2. Wait for the default windows driver to install
3. Open the device manager ("Control panel"→"Device manager")
4. Open the device ("Universal Serial Bus Controller"→"USB Serial Converter")
5. Check the manufacturer of the device is FTDI. If not, repeat point 4 with the next "USB Serial Converter" until the correct device is found
6. Open the advanced properties ("Advanced" tab)
7. Enable "Load VCP" and click "Ok"
8. Unplug the CAN-USB converter, wait 5 seconds and plug it back in
9. Check if a new COM port appeared under "Ports (COM and LPT)". Right click on it and verify the manufacturer is *FTDI.*

Hardware preparation

The motor control PCB is in a protective shell and can be removed if needed. There are 3 connections to make:

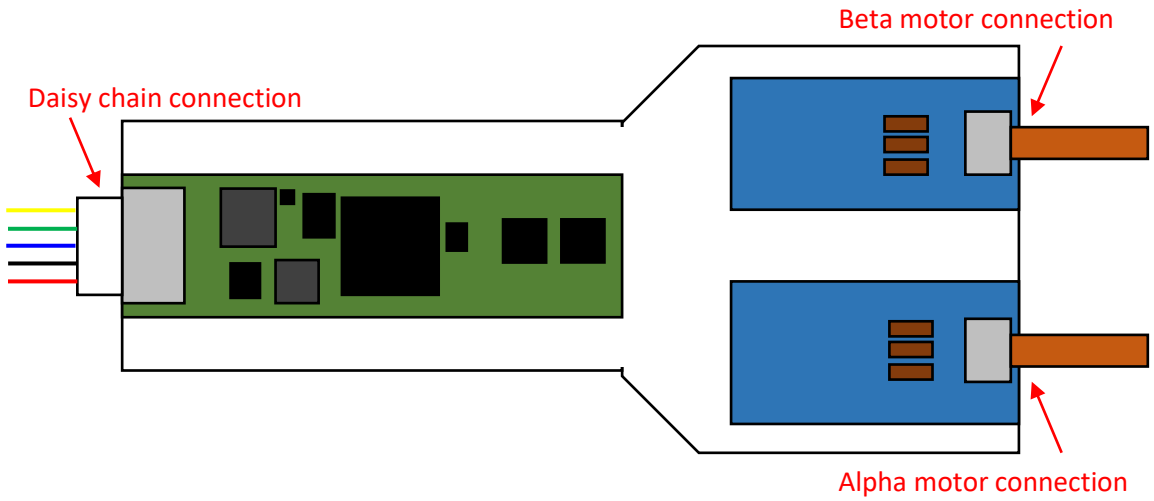

Figure 2: Top view of the Motor interface PCB with the protective shell removed

The following scheme shows how to connect all the devices together:

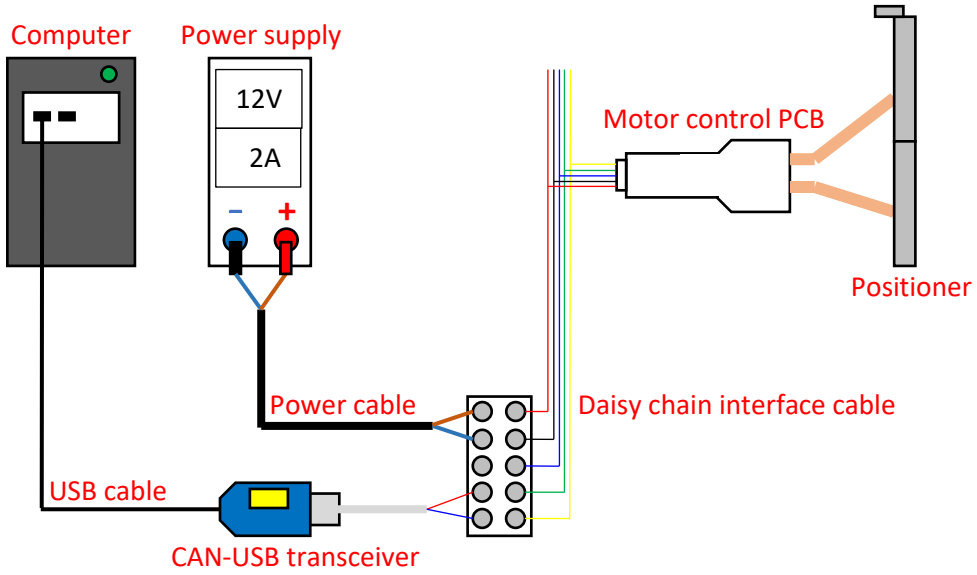

Figure 3: Global connection scheme

Software

Installation

1. Install python (version 3.7.0): https://www.python.org/downloads/release/python-370/
    a. Open the executable
    b. Install the following optional features (at least):
        i. pip
        ii. tcl/tk and IDLE
        iii. py launcher
    c. Check the following advanced options (at least):
        i. Associate files with python (requires the py launcher)
        ii. Add Python to environment variables
    d. Click install
2. Install the complementary python modules
    a. Open the Windows command prompt
    b. Type the following command:
    python -m pip install scipy pyserial pysimplegui

Operation

Open the manual control software program by double-clicking on *GUI.pyw*. A window should open:

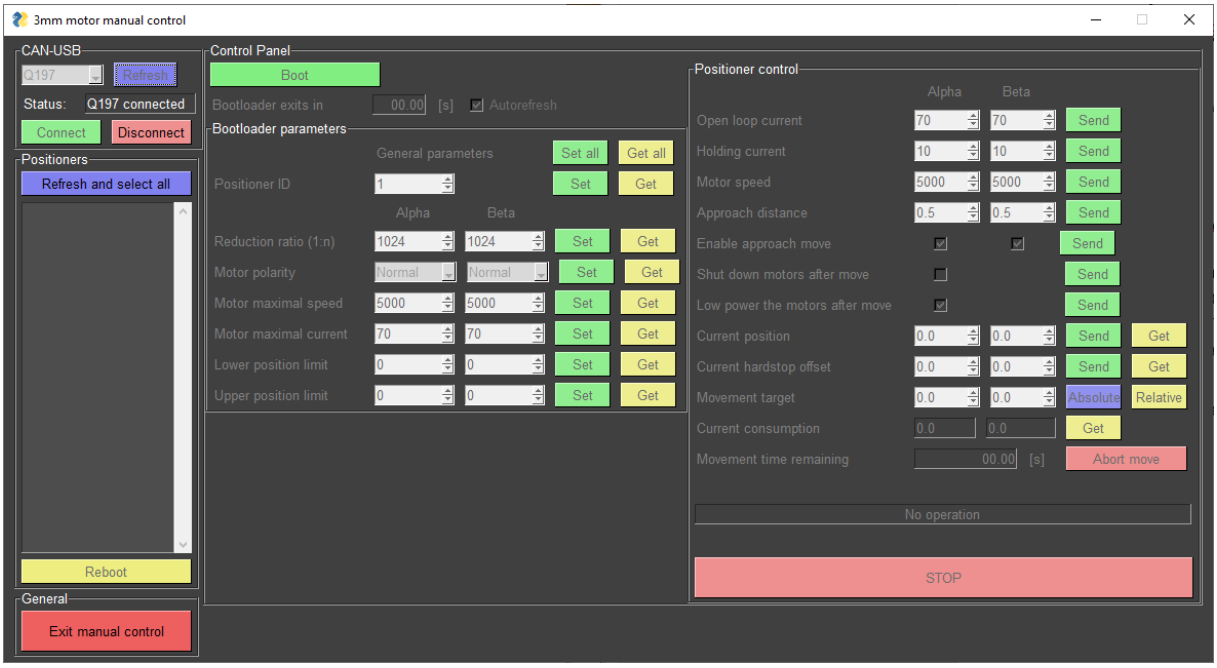

Figure 4: Manual control window

This window is divided in several subsections. Each element has a tooltip that describes what it does. The general operation of each frame will be described in the following sections.

CAN-USB frame

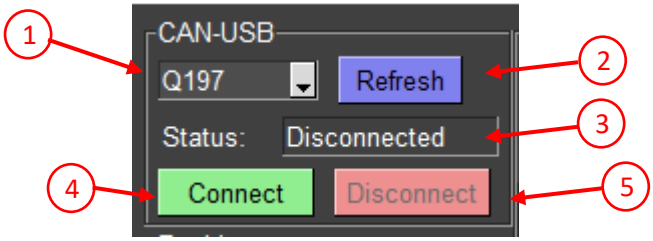

Figure 5: CAN-USB connection frame

1. CAN-USB serial number selection. If multiple CAN-USB devices are available, select the correct one by checking its serial number (X123 on the yellow sticker). This item is disabled when a CAN-USB device is connected.
2. Refresh button. Click to refresh the list of CAN-USB serials, for example if a new device was connected. This item is disabled when a CAN-USB device is connected.
3. Connection status. It displays the current state of the connection with the CAN-USB device.
4. Connect button. When clicked, it will connect to the selected CAN-USB. This item is disabled if no CAN-USB serial is available.
5. Disconnect button. When clicked, it will disconnect the connected CAN-USB. This item is disabled if no CAN-USB device is connected.

Positioners frame

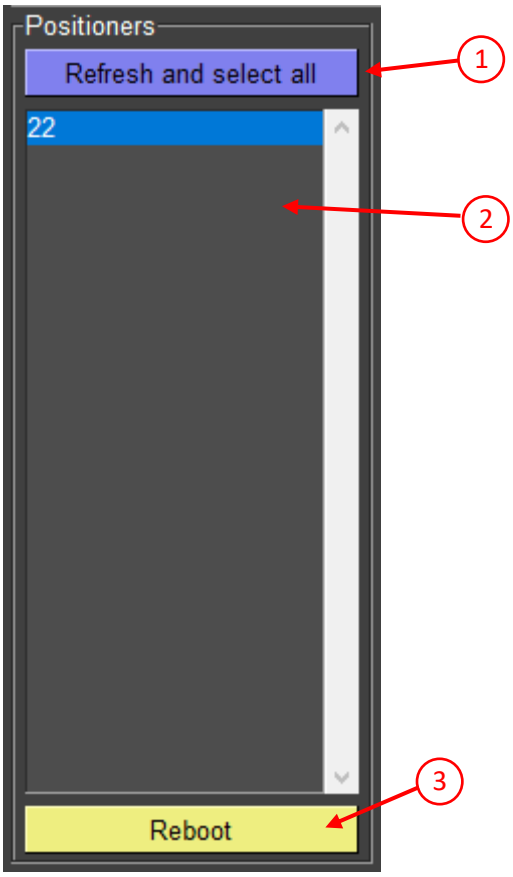

Figure 6: Positioners selection frame

The whole frame is disabled if no CAN-USB device is connected.

1. Refresh and select button. It will send a general identification poll on the CAN bus. All positioners that answer to this poll will be added in (2) and selected.
2. Available positioners list. It lists all the positioners on the CAN bus. The selected positioners are shown in blue. The commands will only be sent to the selected positioners and the unselected will be ignored.
3. Reboot button. It sends a reboot command to all the selected positioners. This item is disabled if no positioner is selected.

General frame

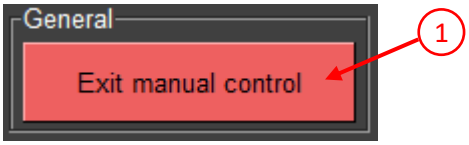

Figure 7: General frame

1. Exit button. This will close the manual control window and terminate all connections. It has the same effect as clicking on the closing cross on the top right of the window shown in Figure 4

Control panel frame

The control panel consists of 2 separate frames, accessible depending on the positioner's operating mode. The positioner can have 2 modes:

1. Bootloader mode. This is the mode it enters directly after powering up. It allows to change the operational parameters and limits of the positioner. These settings are related to the physical construct of the positioner and should be handled with care. It will exit automatically after 10 seconds without communication.
2. Normal operation mode. Once the bootloader exited, the positioner will enter normal operation mode and will be able to receive and execute movement commands.

Bootloader parameters frame

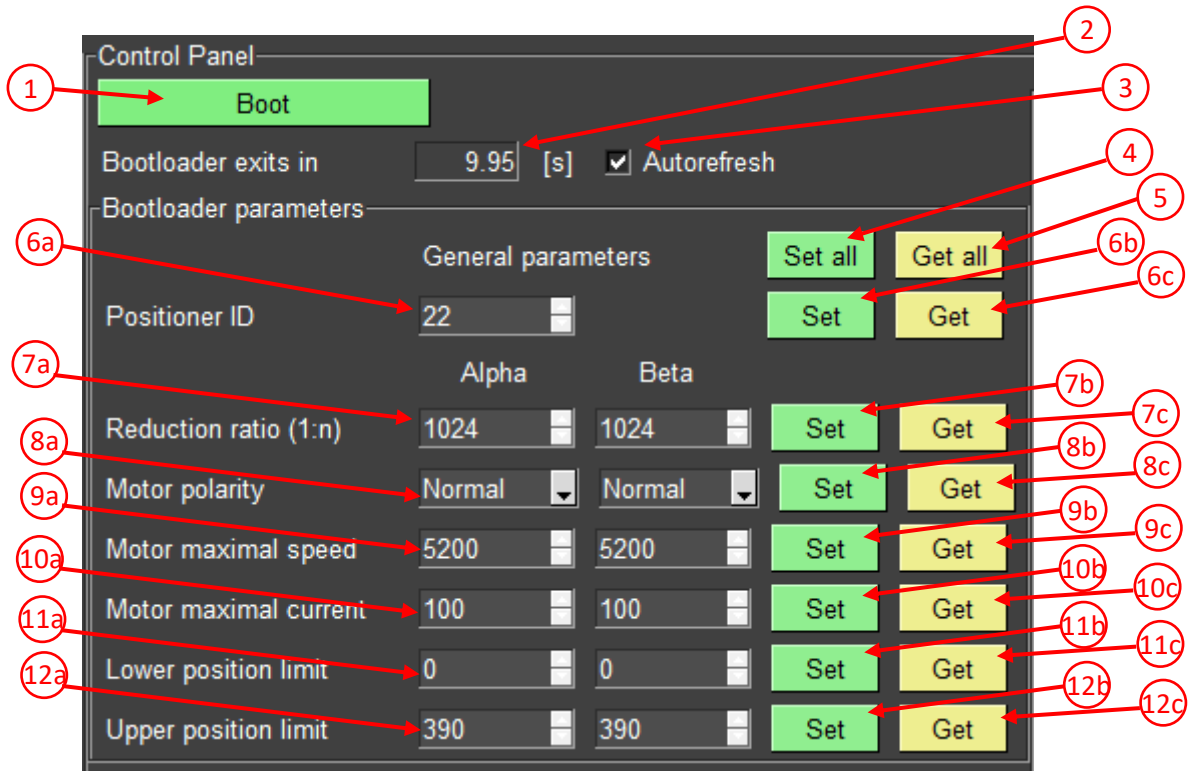

Figure 8: Bootloader parameters frame

The whole frame is disabled if the first selected positioner is not in bootloader mode.

1. Boot button. Sends a boot command to the positioners, making them leave the bootloader mode immediately.
2. Bootloader countdown timer. This is the remaining time before the positioners boot due to timeout.
3. Autorefresh checkbox. When checked, it will periodically issue a status request command to the positioners, what makes them stay in the bootloader mode.
4. Set all button. Sends the current set of parameters to all the selected positioners, except for the *Positioner ID*. The change becomes effective once the positioners have booted.
5. Get all button. Requests all the bootloader parameters values of the first selected positioner and updates the displayed values.
6. Positioner ID
    a. Positioner ID value. It must be an integer between 1 and 2047.
    b. Sets the positioner ID. Sends the new ID to the first positioner. If a positioner with the same ID is already connected, the command will not be issued. The change becomes effective once the positioner has booted.

    c. Retrieves the first positioner's ID value.
7. Motor reduction ratio. It represents the number of motor turns needed to perform one turn at the output.
    a. Reduction ratio value for the alpha motor (left) and the beta motor (right). It must be an integer between 1 and 9999.
    b. Set the motor reduction ratios. Sends the new motor reduction ratios to all the selected positioners. The change becomes effective once the positioners have booted.
    c. Retrieves the first positioner's motor reduction values.
8. Motor polarity. It represents rotation direction of the motors.
    a. Motor polarity for the alpha motor (left) and the beta motor (right).
    b. Set the motor polarities. Sends the new motor polarity to all the selected positioners. The change becomes effective once the positioners have booted.
    c. Retrieves the first positioner's motor polarity values.
9. Motors maximal speed. It represents the maximal RPM of the motors. The maximal input speed of an eventual reduction gear must be considered.
    a. Maximal speed value for the alpha motor (left) and the beta motor (right). It must be an integer between 1 and 9999.
    b. Set the motors maximal speed. Sends the new motors maximal speed to all the selected positioners. The change becomes effective once the positioners have booted.
    c. Retrieves the first positioner's motors maximal speed.
10. Motors maximal open loop current. It represents the maximal duty cycle of the PWM controller.
    a. Maximal open loop current value for the alpha motor (left) and the beta motor (right). It must be an integer between 1 and 100.
    b. Set the motors maximal open loop current. Sends the new motors maximal open loop current to all the selected positioners. The change becomes effective once the positioners have booted.
    c. Retrieves the first positioner's motors maximal open loop current.
11. Lower position limit. It represents the smallest reachable angle for each positioner arm.
    a. Lower position limit for the alpha motor (left) and the beta motor (right). It must be an integer between -720 and 719, and smaller than the upper position limit (12).
    b. Set the lower position limits. Sends the new lower position limits to all the selected positioners. The change becomes effective once the positioners have booted.
    c. Retrieves the first positioner's lower position limits.
12. Upper position limit. It represents the smallest reachable angle for each positioner arm.
    a. Lower position limit for the alpha motor (left) and the beta motor (right). It must be an integer between -720 and 719, and greater than the lower position limit (11).
    b. Set the upper position limits. Sends the new upper position limits to all the selected positioners. The change becomes effective once the positioners have booted.
    c. Retrieves the first positioner's upper position limits.

Positioner control frame

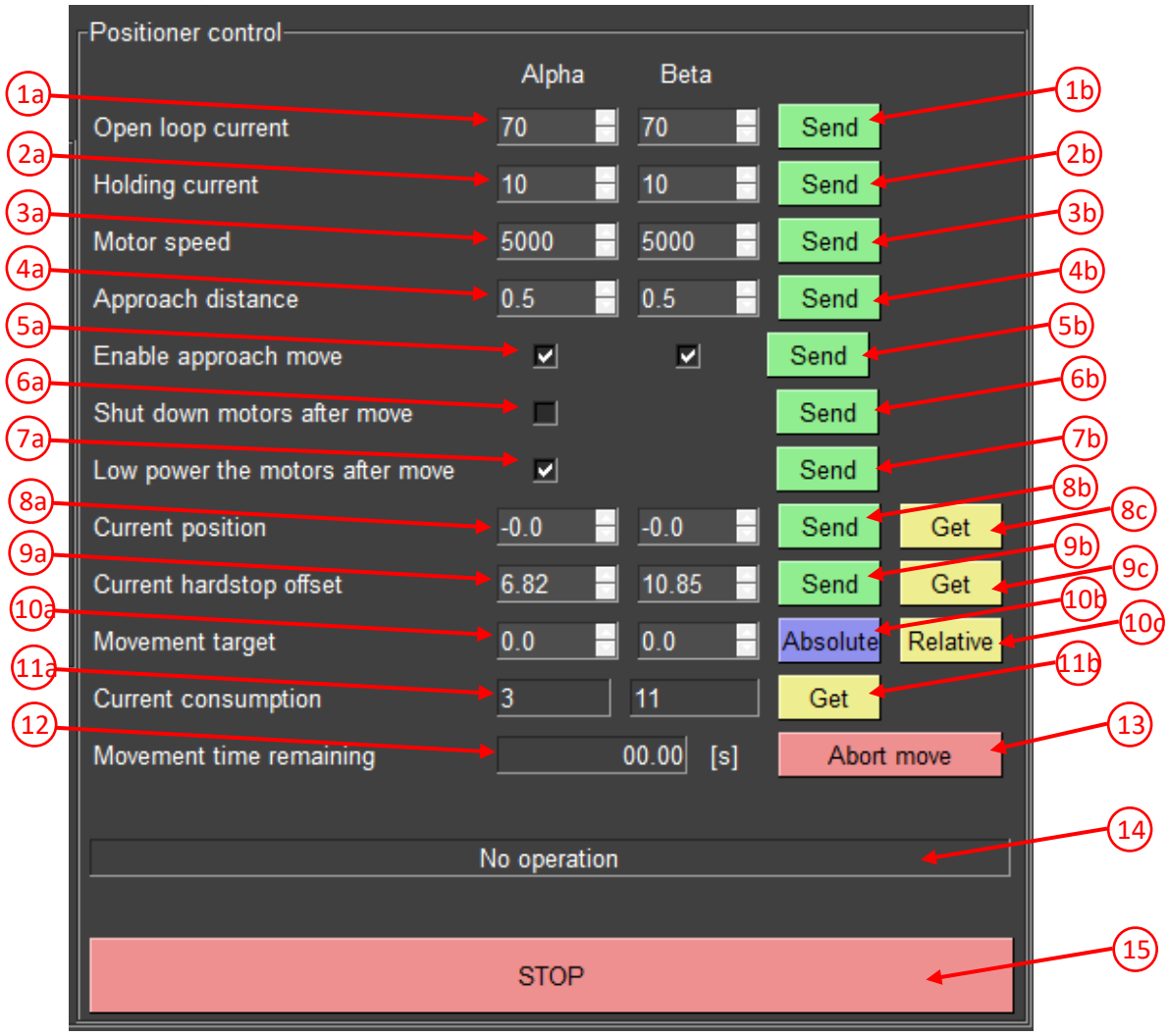

Figure 9: Positioner control frame

The whole frame is disabled if the first selected positioner is not in normal operation mode.

1. Open loop current.
    a. Open loop current values. This is the PWM duty cycle fed to the alpha motor (left) and the beta motor (right) when they are moving. It must be an integer between 0 and 100 (maximal value depending on the bootloader configuration). If set too low, the motors may slip and not end in the desired position. If set too high, the motor may overheat, specially if “Shut down motors after move” and “Low power the motors after move” are both disabled.
    b. Set the open loop current. Sends the open loop current values to all selected positioners.
2. Holding current.
    a. The holding current values. It must be an integer between 0 and 100 (maximal value depending on the bootloader configuration). If “Shut down motors after move” is disabled and “Low power the motors after move” is enabled, this is the PWM duty cycle fed to the alpha motor (left) and the beta motor (right) when they are not moving. Else this parameter is not used.
    b. Set the holding current. Sends the holding current values to all selected positioners.
3. Motor speed.

   a. The motor speed values. This is the alpha motor (left) and beta motor (right) rotation speed when performing a move. It must be an integer between 1 and 9999 (maximal value depending on the bootloader configuration).
   b. Set the motor speed. Sends the motor speed values to all selected positioners.
4. Approach distance.
   a. The approach distance values. This is the alpha motor (left) and beta motor (right) approach distance when performing a move. It must be a float between -5 and 5. When going to a target position, the positioner will proceed in two steps: first it will go to *targetPosition + approachDistance*, and then go to *targetPosition*. This ensures the positioner always approaches from the same direction and with a minimal displacement that should be big enough to compensate any hysteresis. This parameter does nothing if the motor's "Enable approach move" checkbox is disabled.
   b. Set the approach distance. Sends the approach distances values to all selected positioners.
5. Enable approach move.
   a. Enable approach move checkboxes. If the corresponding checkbox is checked, the alpha motor (left) and beta motor (right) approach move are enabled. Else, the corresponding approach move is disabled.
   b. Set the approach move status. Sends the approach move status to all selected positioners.
6. Shut down motors after move.
   a. Shut down motors after move checkbox. If checked, it will stop the H-bridges after the move is complete, cutting all power to the motors. Else, the H-bridges will stay enabled.
   b. Set the motor power status. Sends the motor power status to all selected positioners.
7. Low power the motors after move.
   a. Enable low power the motors after move checkbox. If "Shut down motors after move" is checked, this does nothing. Else, it will enable the motor low power mode if checked. Low power mode will use the holding current value instead of the open loop current value when the positioner is not moving.
   b. Set the motor low power status. Sends the motor low power status to all selected positioners.
8. Current position. These values are automatically updated while moving based on the first selected positioner.
   a. The positioner's current alpha (left) and beta (right) positions. This must be a float between -720 and 719.
   b. Set the current alpha (left) and beta (right) positions.
   c. Retrieves the first positioner's current alpha (left) and beta (right) positions.
9. Current hardstop position. These values are automatically updated while moving based on the first selected positioner.
   a. The positioner's current alpha (left) and beta (right) hardstop positions. This is the angle from the hardstop (lower mechanical limit) to the 0 point. This must be a float between -720 and 719.
   b. Set the current alpha (left) and beta (right) hardstop positions.
   c. Retrieves the first positioner's current alpha (left) and beta (right) hardstop positions.
10. Movement target.
   a. The positioner's alpha (left) and beta (right) target positions for the next move. This must be a float between -720 and 719.

   b. Absolute move button. Starts an absolute move from the current position to the target position. The absolute move will remain in the position limits defined in the bootloader and will not accept any command that would bring the positioner outside this range.
   c. Relative move button. Starts a relative move from the current position, with angles equal to the target angles. This command can go outside the defined position limits. If the relative moves go below -720° or above 719°, an overflow will occur in the positioner's firmware and its position will not be reliable anymore. Either modifying the *Current position* values or moving with the exact same inverse relative moves is recommended if this happens.
11. Current consumption. These values are automatically updated while moving based on the first selected positioner.
   a. The alpha motor (left) and beta motor (right) current consumption. These values are read only.
   b. Retrieves the current of the first selected positioner.
12. Movement time remaining. This value is automatically updated while moving based on the slowest selected positioner.
13. Abort move button. If pressed, the positioners will immediately stop their current move.
14. Positioner current operation status bar.
15. Stop button. If pressed, the positioners will immediately stop their current move.