Benduino Body Master for AstroComms Ultra

Software User Manual v1.0.0 - 10/2020

Welcome to this manual

This manual will show you how to use your Benduino Body Master board for AstroComms Ultra.

First connect a USB cable to your Benduino and open Arduino IDE (or any other app that supports a serial terminal using USB), but Arduino IDE it's highly recommended.

You will see the following welcome message on the app

```
Welcome to Benduino Body Master v1.0.0
Developed in 2020 by Walex and designed by Nitewing

You can enter commands directly on the terminal or type '/help' for more
information.
Do not forget to send CR after the command.

Do you need assistance?
Please visit www.astrocomms.net for more info
Much to learn you still have my old Padawan. This is just the beginning.
```

Type `/help` to get a list of available commands

```
You can enter the following commands on the terminal

/read all - read all servos configuration
/read xx - read servo configuration at a given index xx from 01 to 12.
Example: /read 5
/write xx parameter value - write servo parameter value at a given index
xx from 01 to 12
.        Example: /write 05 open 15 - trim open position to 15, default
value is 0 for full open, range 0 to 45
.        Example: /write 05 close 90 - trim close position to 75, default
value is 90 for full close, range 45 to 90
.        Example: /write 05 speed 250 - set servo speed to 250, default
value is 500 for high speed, range 10 to 500
.        Example: /write 05 reversed 1 - set servo reversed, default
value is 0 for normal and 1 for reversed
/reset - reset command table to factory
```

Connect your servos (do not attach shafts by now) and type `:OP00` and `:CL00` to test if servos moves correctly. 00 parameter means all servos, and 01 to 12 means servos 1 to 12

```
>>> Received from Serial :OP00
OK
```

OK indicates that order have been received successfully

If you bought a second hand board I suggest you to send a `/reset` command to set all parameters to factory values.

Installing servos

I suggest you to setup your servos for closing to an angle of 10 degrees. Making this if you need some extra movement in case that door does not closes properly you have some extra degrees to fix this, but this is only a suggestion.

Type `/read` all to read all servos parameters

```
>>> Received from Serial /read all

Servo 01: open: 0 close: 90 speed: 500 reversed: 1
Servo 02: open: 0 close: 90 speed: 500 reversed: 1
…
Servo 11: open: 0 close: 90 speed: 500 reversed: 1
Servo 12: open: 0 close: 90 speed: 500 reversed: 1
```

Attach one by one all servos testing open and close values and changing speed and configure your doors as you want.

NOTE: You need to send CR after any command, so please check if you need to change some parameter on your app to do so automatically when you press send button. For instance, on Arduino IDE you need to set Carriage Return

/write command

This command allows you to change parameters. A correct if formed like this

```
/write 05 open 15
```

Where write its the command, 05 its servo 4 open means open parameter and 15 means angle. Send the command to the device and test the new setting. This is the full list of commands.

```
/write 05 open 15 - trim open position to 15, default value is 0 for full
open, range 0 to 45

/write 05 close 90 - trim close position to 75, default value is 90 for
full close, range 45 to 90

/write 05 speed 500 - set servo speed to 500, default value is 250 for
high speed, range 10 to 500

/write 05 reversed 1 - set servo reversed, default value is 0 for normal
and 1 for reversed
```

DISCLAIMER

THE SOFTWARE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.