

ZUMUNGO

WiFi Stepper motor control (up to 4 motors)

Software for ESP32 and ESP8266

Smart – Programmable – Controllable

Control by HTTP

Reports by HTTP and UDP

USER GUIDE

rev. 3.1.3 Jan., 2020

WARNING !!!

This is not a consumer software !

Basic skills and understanding of electrical circuits are required to properly install the software and operate the device!

Improper handling can result in property damage, bodily injury or death !

Table of Contents

1. Specifications

WiFi /controller module	ESP8266 or ESP32
AdHoc WiFi network SSID	Zumungo XXXXXX where XXXXXX is an unique board ID, needed for licensing (see zumungo.com)
Factory set password for the AdHoc network	Zumungo123 (do not forget to change it to your own password for protection. Go to “WiFi settings” at the bottom of the main web page.)
IP address within the AdHoc WiFi network	192.168.4.1
WiFi SSID	User setup in the Installer (see zumungo.com) to download the free installer
WiFi password	User setup in the Installer (see zumungo.com) to download the free installer
Modes of operation	3 modes – by time, by # steps, by destination (up to 99 positions in memory)
Setup	<ul style="list-style-type: none"> - Steps per rev for each motor (200 or 300 typically) - GPIO pins (3 per motor, SLEEP*, DIR and STEP) - password for the operations - up to 99 pre-set positions - web server – port <p>by JSON HTTP POST requires the header - Authorization: "password". Default password is "Zumungo"</p>
Control	<ul style="list-style-type: none"> - speed in rpm - start/stop - by time to run - by # steps (overrides time to run) - by destination (up to 99 positions could be set in memory) - web server – port <p>by JSON HTTP POST requires the header - Authorization: "password". Default password is "Zumungo"</p>
Reports	<ul style="list-style-type: none"> - status (running or stopped) - time to destination - #steps to destination - all settings (see setup above) - IP address <p>a) Reports are generated as a response to any control JSON b) by HTTP GET requires the header - Authorization: "password". Default password is "Zumungo"</p>
URL for POST and GET	Example: 192.168.1.100/stepper Replace the IP address with the IP address of your board. Get the IP address of

	your board by UDP "CallAll" or in your router
IP address of the board	Send message "CallAll" to UDP port 60560 (default) or look in your router tables. Another way to find out what IP address is assigned to the board is to use "Serial Monitor" tool of your Arduino IDE software when you flash your Zumungo installer to your board.
Default UDP address	60560 can be changed at the device's web page
Board's "location"	Default is blank. In case you have multiple boards you can add location information for each board to easily identify them. "Location" is reported in the response to the UDP "CallAll" command. Nowhere else is the location used or reported.

2. How to install the software in your device

- 2.1 Copy the free "Zumungo Installer" from zumungo.com "get started" page. Customize 3 lines in the code per instructions. Project must be Project_S. Add your own SSID and WiFi password in the code as instructed. Flash it and run it on your board.
- 2.2 Find your board's 6 characters board ID. Go to "settings"-->"WiFi" of your smart phone and look for WiFi network Zumungo_XXXXXX, where XXXXXX is your 6 characters ID
- 2.3 Go to zumungo.com and purchase the licenses that you need. Free trial is available. No credit card required for a free trial.
- 2.4 Restart the board and wait about 30 sec. for the board to download the software automatically. Congratulations !!! Your board is operational now.
- 2.5 Find your board's local IP address by one of the following methods
 - a) looking in your router's tables. The aforementioned "board ID" is the last 6 digits of the board's MAC address, that shows in the router tables.
 - b) sending "CallAll" message to UDP port 60560. Many free TCP/UDP clients are available for that. PacketSender is one such free program for Windows and Linux. The board will report it's name, software version and IP address.
 - c) connect to the AdHoc Access Point that your device creates (see p. 3 below)
- 2.6 Go to the IP address of your board, setup your board, setup your motors and start using it.

3. Connect to AdHoc WiFi Network (optional)

Please note that this step is optional and can be skipped as the board will connect to your WiFi network and will become operational if you provided the proper credentials as instructed in 2.1 above. Use this procedure if you change WiFi networks or move your board to another location.

ZUMUNGO creates its own WiFi network with SSID

ZUMUNGO_XXXXXX where XXXXXX is unique board ID.
--

Use any WiFi capable device such as laptop, tablet, smart phone etc and find the ZUMUNGO network there and connect using the default password

123456789

Once connected to ZUMUNGO network go to this default IP address (not subject to change)

192.168.4.1

Use the default password for the stepper motors board

Zumungo

to open the web interface and setup the board.

Please do not forget to change the default password to your own password for your own protection. Go to “WiFi settings” at the bottom of the main web page to do that.

You can use the board as it is at this moment, however you have to always have a device (computer, laptop, smart phone) connected to ZUMUNGO network. This could be inconvenient. See next chapter for how to connect ZUMUNGO powered device to your home WiFi.

4. Connect to home WiFi network manually (optional).

Important ! Connection to your WiFi network happens automatically if you follow the instructions in p. 2. above.

Use the steps below only if you follow the optional p 3. above.

Place ZUMUNGO powered device in an area where you have good coverage of your home WiFi network. This applies also to the final permanent destination of your ZUMUNGO powered device.

IMPORTANT !

Never place your ZUMUNGO powered device in a metal enclosure if you want to control it over WiFi.

Take a note of your board ID listed at the webpage or part of the ZUMUNGO AdHoc WiFi.

Looks like this : Board ID: 6AB41A (example only, your will be different)

After you have taken the Board ID on a piece of paper than go to the bottom of the page and click the button "WiFi settings" at the bottom.

Select the name of your home WiFi network from the drop-down box, enter the password for your home WiFi network and click "connect".

After few seconds your ZUMUNGO powered device will connect to your home WiFi network.

How do you find it is another question. To do so you need to go to your router's DHCP tables and look for the MAC address (last 6 characters of the MAC are your board ID mentioned above) you just took note of. You will find the IP address next to it that looks like this

192.168.1.150 for example your digits will be different

Go to any browser on your home WiFi network and enter the IP address of the board and you are in business. While at the router you may consider turning the dynamic IP address assigned by the router into a static one so that you always find your board at the same IP address.

Alternatively you can issue "CallAll" command by UDP to port 60560. Multiple free software TCP/UDP clients are available for desktops, tablets and smart phones. One such software is PacketSender for desktops.

5. Setup name and password for the board

Default name is Zumungo

Default password is: Zumungo

Go to home web page and log-in with the default password.

Scroll down to the bottom and click on the “Board Setup” button.

At the top of the “Board Setup” page choose new name and location for your board and click “save”.

Also at the top enter the existing password (default Zumungo but enter your previously selected password if there is one).

Pick a new password and enter it twice. Also pick a password hint to remind you of your new password.

Click “Save password” and wait 10-20 sec as the board reboots after this change.

6. Setting up HTML and TCP ports

Default port is 80 for the HTML commands to the board and for the web pages.

Default port is 8000 for the TCP server residing on the board where you will be sending commands to if using TCP.

Please enter the appropriate port numbers and IP address and click “Save” button.

Useful hints.

1. You will lose your web pages if you switch the HTTP port to anything different than 80. To get them back you need to use web address with suffix :xx where xx is your new port number. If you have chosen port 82 for example and your board is at this address for example 192.168.1.150 than your main web page will be at 192.168.1.150:82 (just an example. Use your own settings)
2. Most likely port 80 is already taken by something on your home WiFi network. Use another port say 82 for your board. Than go to your router and forward port 82 to the IP address of the board. Now you can access your board from anywhere in the world by myhomenetwork.com:82 where “myhomenetwork.com” is the global IP address of your home network (see your internet provider or Google “My IP address” from your home network).

7. Setting up location, date and time

Choose your location to setup board real time clock and sync it with the Internet.

Your board must be connected to your home WiFi network that is connected to the Internet.

If there is no Internet connection than your only option is to setup date and time manually.

8. GPIO configuration

Go to the "stepper setting" page. Here you select what GPIOs of your board will control each of the 4 stepper motors. Please note that your free trial license allows full configuration of up to 4 motors. Once it expires than you'll control as many motors as you buy license for.

The setting "steps per rev." must match the value provided by your motor vendor. Default is 200. Typical values are 200, 300 or 400. Check the specs of your motor and set accordingly.

ATTENTION !

You can easily blow and damage your WiFi board if you do not observe important rules and restrictions.

1. Make sure that there is hardware compatibility between your board and the stepper motor driver.
2. The outputs have limited voltage (3.3V and current 20 ma capabilities). Make sure that your stepper motor driver is in compliance. Most of them are.
3. If your stepper motor driver can't be driven by the power of the GPIO outputs (unlikely) than consider using simple opto-coupler such as this 4 channel opto-coupler LTV-847

10. Control your motors by POST JSON

10.1 What is POST

POST is HTTP method for communication between computers. There are multiple free software packages for desktops and portable devices that can execute POST commands. "Postman" is one excellent choice and it is free.

JSON is a method of formatting data. See below for details.

10.2 POST Parameters

IP address – You must know the IP address of your board. See p. 2.5 above for ways to find it out.

Important POST to IP address/stepper.

Example: 192.168.1.150/stepper

Headers: Your POST must include the following headers

Authorization:password

Content-type:application/json

where "password" is the password you've selected for your board at the web pages (see p.5 above).
Default password is "Zumungo".

10.3 POST JSON

Here is a sample JSON file for driving all 4 motors.

```
{
  "motor1": {
    "rpm": 3600,
    "time_sec": 15,
    "steps": 200,
    "direction": 0,
    "stopped": 1
  },
  "motor2": {
    "rpm": 600,
    "time_sec": 15,
    "steps": 0,
    "direction": 0,
```

```

        "stopped": 1
    },
    "motor3": {
        "rpm": 400,
        "time_sec": 0,
        "steps": 200,
        "direction": 0,
        "stopped": 1
    },
    "motor4": {
        "rpm": 60,
        "time_sec": 10,
        "steps": 200,
        "goto": "position 5",
        "stopped": 1
    }
}

```

Where

rpm – revolutions per minute (check the setting "steps per rev." in the GPIO settings if actual rpm are different than what you expect).

time_sec – time to run in seconds (unlimited if t=0) (ignored if steps >0)

steps – number of steps (if steps=0 than time t rules, otherwise time t is ignored)

position x – is one of the memorized up to 100 positions. If present than it overrules time, steps and direction.

direction – 0 -clockwise, 1 – counter clock wise

stopped – stop the motor if 0, run per instructions if 1

Not all of the fields are necessary.

Example:

```

{
    "motor1": {
        "time_sec": 15,
        "stopped": 1
    }
}

```

This short command will put motor1 in motion for 15 sec. using last used values for "rpm" and "direction". The GET report will list all values used.

10.4 JSON validation

Some of the available software products for POST will do automatic validation of your JSON input for syntax errors. "postman" is an example of such software that we highly recommend. If not than we

recommend that you use one of the JSON validators available online so that you do not get nasty surprises just due to JSON syntax errors.

JSON validators will not catch misspelling of keywords. For example if you put "rpn" instead of "rpm" than the Zumungo software will produce an error.

10.5 Positions

Position is defined as number of steps (with a sign) from "position 0".

Use setup by current position (see below) in order to setup the starting position – "position 0". Resetting "position 0" does not change any other memorized positions. It only resets the current position to 0.

Example: This command will reset the "current position" to 0

```
{ "motor3": {  
    "position 0": 0,  
  }  
}
```

Zumungo memorizes up to 100 positions in the flash memory. Positions are named "positionxxx" where xxx is an integer number from 1 to 100.

Examples "position23" "position 23" "position023" "position 023" are all permissible.

Setup position by "current position". Use "0" in the command in order to specify "current position".

Example

```
{ "motor3": {  
    "position 23": 0,  
  }  
}
```

Setup position by absolute number of steps. Number of steps is an integer number with sign.
Example.

```
{ "motor3": {  
    "position 23": -25432,  
    "position 25": 25673  
  }  
}
```

Copy all memorized positions from one motor to another. Use the names of the motors to copy to or the name "all" to copy to all. Example

```
{ "motor3": {  
    "copyto": "motor2",  
  }  
}
```

```
        "copyto": "motor1"
    }
}
```

This command will copy all of the positions memorized for motor 3 to motor2 and motor1.

Example

```
{ "motor3": {
    "copyto": "all"
  }
}
```

this command will copy all of the positions memorized for motor 3 to all other motors in the system.

Go to position. Example:

```
{ "motor3": {
    "stopped":1,
    "goto": position 25
  }
}
```

"stopped":1 is necessary to put the motor in motion.

10.6 GET reports

You can get a report of the status of all motors by HTTP GET with following parameters

URL - IP Address/stepper

Example: 192.168.1.150/stepper

Headers: Your GET must include the following headers

Authorization:password

Content-type:application/json

where "password" is the password you've selected for your board at the web pages (see p.5 above).
Default password is "Zumungo".

The report will cover ALL motors regardless of usage and activity. The report is in JSON format for

easy parsing by computer programs

Here is a sample report (fractional for one motor only)

```
"motor3": {
  "rpm": 3600,
  "time_sec": 15,
  "steps": 200000,
  "direction": 0,
  "stopped": 1,
  "target": "position 22",
  "positions": [{
    "id": "0",
    "value": "12000"
  }, {
    "id": "1",
    "value": "34200"
  }, {
    "id": "22",
    "value": "-17100"
  }]
}
```

This example motor3 has two memorized positions 1 and 22. It was given an order to go to position 22 and it is running with 15 sec. And 200000 steps remaining. Numbers are arbitrary don't try making sense of them.

Please note the field "target" that will be present only for motors that execute "goto": "position xx" command.

Please note that "stopped" is used for both "stop" and "run" do not expect to see "running" in the report. If stopped = 1 than the motor is running. If stopped = 0 than the motor has stopped.

Normally the GET reports are long as they cover all motors and they are primarily meant to be processed by computer that parses JSON files.

10.7 Instant responses to POST command

Instant responses to the POST commands are available. They are much shorter than the GET response and indicate current situations such as errors and problems.

The response merely confirms that communication channel is good and the syntax of the command is OK, the command was properly recognized and put for execution.

Example of an instant response.

```
{
"motor1": "0",
"motor3": "2",
"motor4": "0"
}
```

This is a response to JSON command that affected motors 1, 3 and 4. The commands for motors 1 and 4 were received and they were in proper syntax and were executed. The commands for motor 3 were not executed because there was a bad line in it and all lines were ignored. Motor 2 is not mentioned in the report because the JSON commands did not include any commands for Motor 2.

Here is the list of error codes used in the instant response

1. Improper JSON format. Command was ignored.
2. Improper argument. For example number 3 in the "stopped" field where only 0 and 1 are expected.
3. Non existing motor name.
4. Non existing target location in the "goto" command.
5. Improper format. For example letters in numerical fields
6. Improper command. Command that the software doesn't recognize for example "durection" instead of "direction"
7. Previous power loss. Needs update of "position 0". Will NOT execute any "goto" position commands.
8. Reserved
9. Reserved
10. Motor is already in motion and can't execute another command until finishes the previous one.
11. RPM not set.

10.8 "Sleep" signal.

Sleep signal is active all the time when motor is not running to save energy. You can choose the polarity of the sleep signal in order to accommodate wide variety of stepper motors and drivers. Use this JSON command for any of the motors (example is for motor 1).

```
{motor1:{"sleep": x}}
```

where x=1 for positive and 0 for negative polarity.

10.9 UDP "CallAll"

Zumungo listens on UDP port 60560 (programmable) for command "CallAll". It responds by the following line

Name, IP address, Ports, Location, software versions.

Example:

```
StepperTest, IP=192.168.1.100, TCP=8000, HTTP=80, Office, FW=2.71, LW=1.90\r\n
```

where the IP address is the address of the board and the motor names are the ones assigned in the software. Port 60560 is default and can be changed in the board's settings.

10.10 Power loss and recovery

Power loss has an effect on the motors only if it happen during active motor movement. If it happens than "current position" may not be accurate. That's why Zumungo will not execute any "goto" position commands (will produce error=7). If it happens you must position the motor manually in it's starting position 0 and issue this command to reset the starting position

Example: {motor1:{"position 0":0}}

All memorized positions remain the same and are not affected.