

AUTODESK
Instructables

ESP32 Small Robot Dog (closed)

By [Gleb Devyatkin](#) in [CircuitsRobots](#)



Introduction: ESP32 Small Robot Dog (closed)



Disclaimer

This project has been done mostly for fun about a year ago. Some libraries have been updated, so it basically does not work any more. Not every planned feature was released. No PCB is made and it creates a lot of trouble for most of you. I'm sorry to say, but I'm not going to provide any support for it any more. It's exhausting and sometimes just destroys any intentions to continue. This is not commercial project and I'm not going to do something like Donation. Maybe I'm disappointing someone. Sorry. Project closed.

ESP32 small robot dog is my attempt to make quadruped dog that become very popular after Boston Dynamics Spot. Work on the project still in progress and hardware and software can be changed.

Features

- Only ESP32 required (you don't need additional PWM i2c, Bluetooth modules, etc)
- Arduino IDE
- Web based interface with telemetry, to control robot you will need just you smartphone or tablet
- CLI interface for calibration and debug
- True Inverse kinematics code
- Configurable Gait sequence and settings (just if you need it)
- Fun

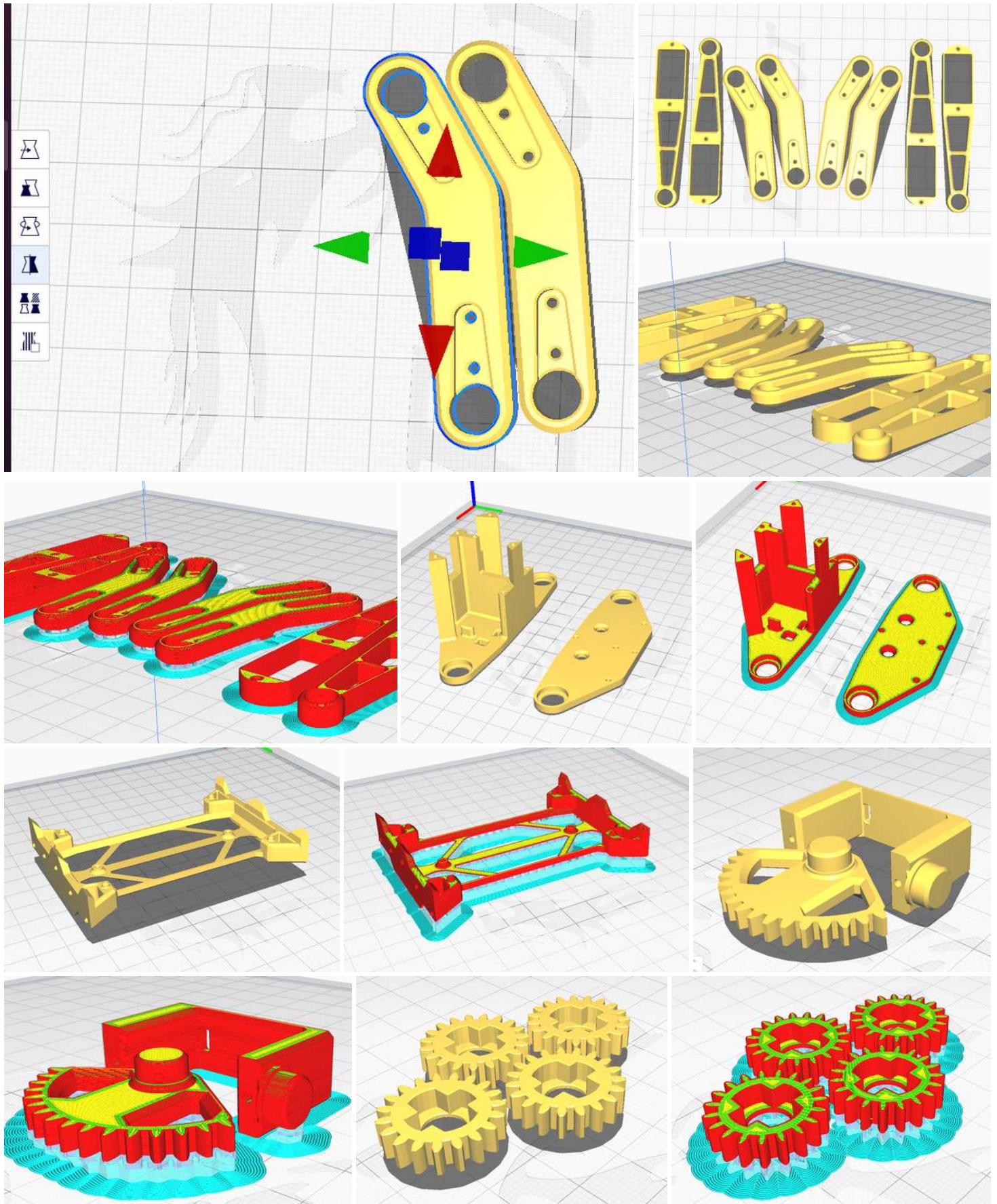
Electronics

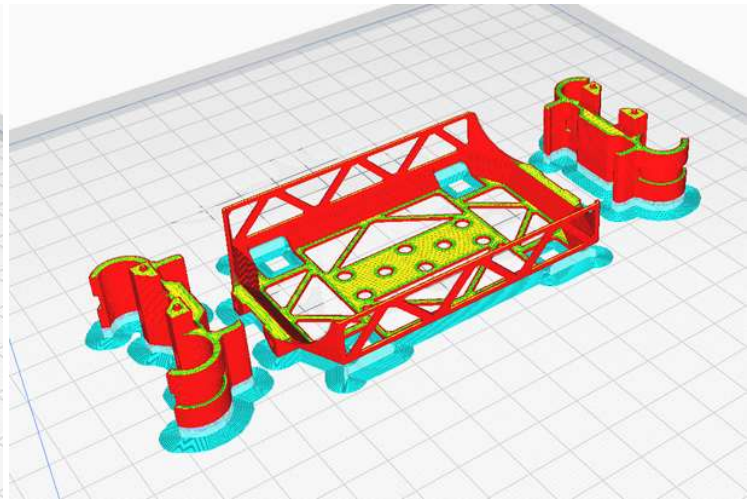
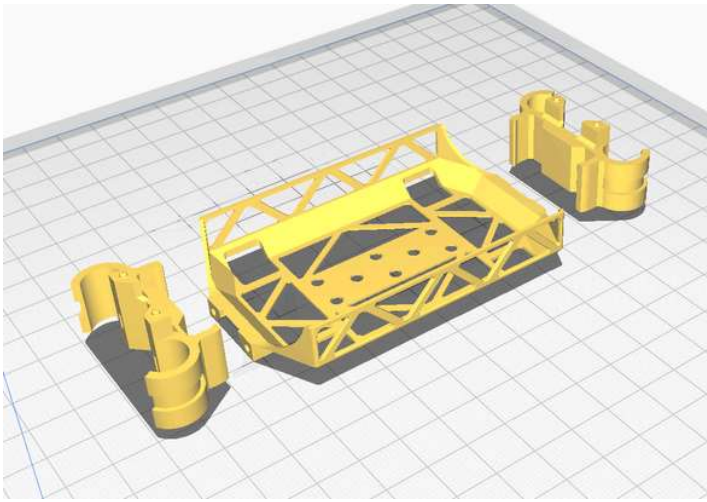
- 1 x ESP32 with 38pin
- 1 x 50x70 mm green prototyping board
- 12 x TowerPro MG90D or MD90S (cable should be out at the bottom of servo, thanks to [triawan](#)) servos (**it can be tricky to use other servos, as size may vary**, please have a look at this images of three different mg90-like servos: <https://www.instagram.com/p/COolyvzrLZt/>)
- 1 x INA219 (optional)
- 1 x MPU9250 (optional, still has not been implemented, WIP)
- 3 x Mini360 (DC-DC Buck Converter Step Down Module) or similar, 2 for front/hind legs (or more), 1 for ESP32
- 1 x 18650 Battery Holder for 2 elements (try to find "18650 battery holder smt")
- 2 x 18650 Battery
- some capacitors

Other parts

- 8 x 8x12x2.5mm bearings
- 4 x small cable ties
- Super glue (cyanoacrylate) to glue all parts together

Step 1: Print Parts





All STLs to print can be downloaded from Thingiverse page: [Robot dog, quadruped robot, 12DOF, 3DOF per leg](#)

Some part should be mirrored to be able to print correctly. Please see attached images.

My printer have issue with first layer thickness, aka Elephant Foot. Gears should be printed as perfect as possible, so I add very tiny spacer to avoid issue, you will see that some part "flying" over surface, so please enable support for it.

I'm strongly recommend print parts with best possible quality, especially gears. As also use brim and ironing.

Full list of parts to print:

Printable parts:

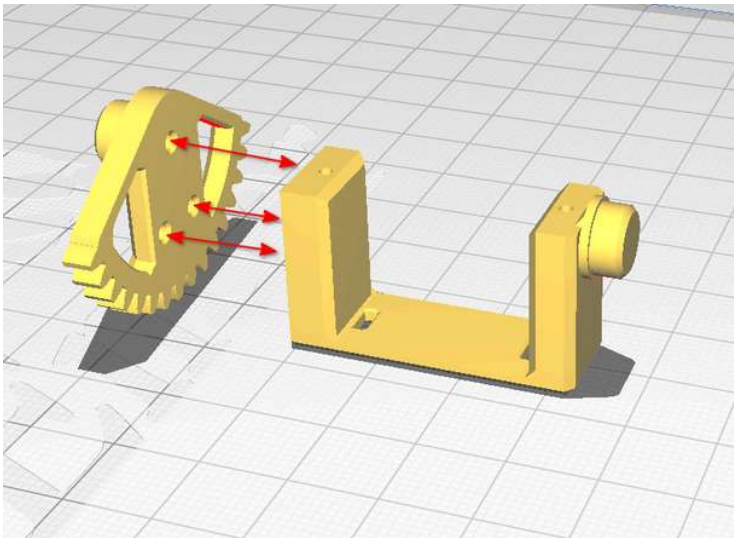
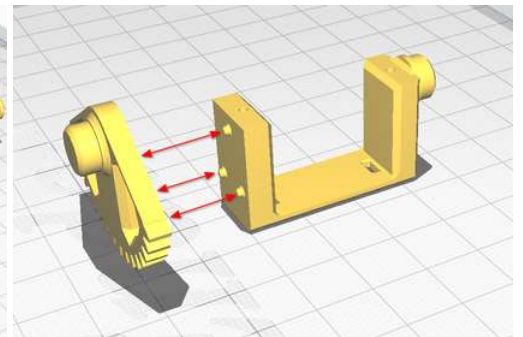
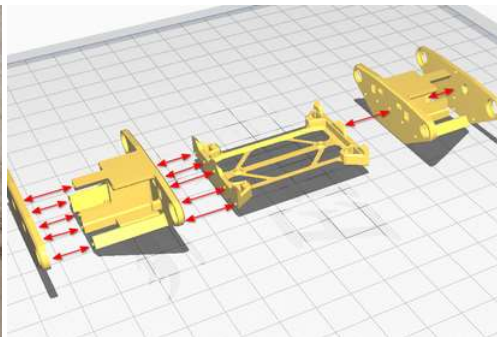
- 1 x Body
- 2 x Leg top
- 2 x Leg top (mirrored)
- 2 x Leg bottom
- 2 x Leg bottom (mirrored)
- 4 x Shoulder part1
- 4 x Shoulder part2
- 2 x Legs holder part1
- 2 x Legs holder part2
- 4 x Servo gear
- 4 x Leg shoes, print with flexible filament (not required)
- 1 x Cover
- 2 x Cover clamps

Printable helping tools (see description on [GitHub](#)):

1. 1 x CALIBRATION_LEG-Beta_and_Gamma
2. 1 x CALIBRATION_LEG-Beta_and_Gamma (mirrored)
3. 1 x CALIBRATION_LEG-Alpha
4. 1 x CALIBRATION_SERVO-10deg

I'm using Ender 3 Pro, Cura, use Support and choose Dynamic quality. Plastic is PLA from OPY (AliExpress). For more details please visit Thingiverse page: <https://www.thingiverse.com/thing:4822059>

Step 2: Let's Make It Better and Glue Together



Preparation

Before we start, please check that bearings can be easily fit to body parts and over shoulder parts.

Also check that holes in Shoulder and body parts printed fine and servo **cables** (temporary disassemble servo connector) can be fit inside.

Do it

I tried to make body as light as possible, as small servos not very powerful. That is why you will need to glue some parts together. Promise it is not so hard =)

Make sure that gluing surfaces are flat, use some flat mini file and make sure parts can be connected to each other without space between. Then glue it together as shown on images.

Servo gears

Cut servo horn (see image) to fit it inside servo gear part and glue it together.

Step 3: Legs

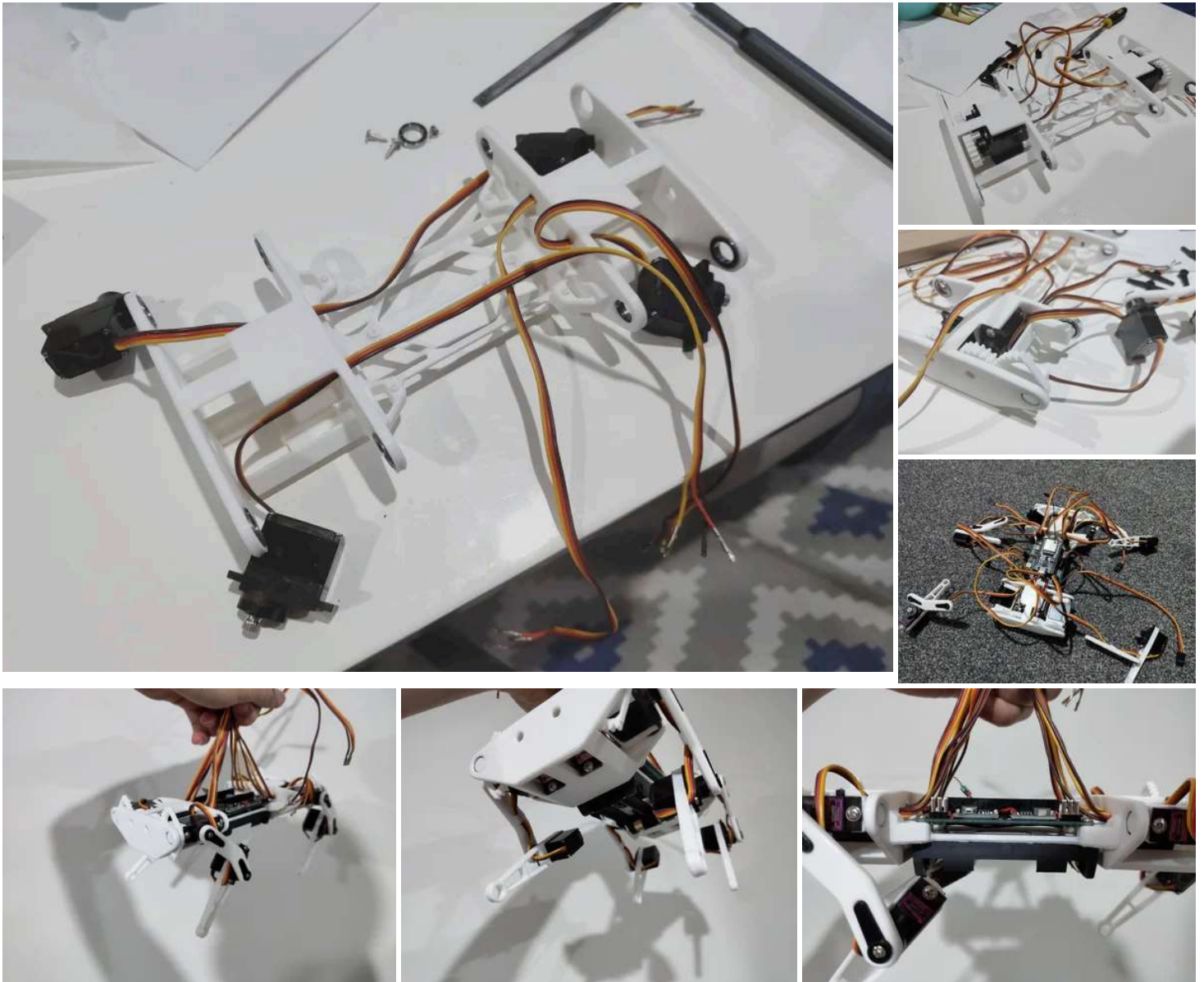
Glue short servo horns into "leg top" parts.

Insert servo into "leg bottom part" and insert cable as shown on image.

Pull another servo cable through one of the hole on the "shoulder" part.

Repeat with other legs.

Step 4: Assemble Body and Legs Together



As was shown on previous pictures, pull all cables through legs and body holes.

Set all servos to middle using Arduino/ESP32 or servo tester.

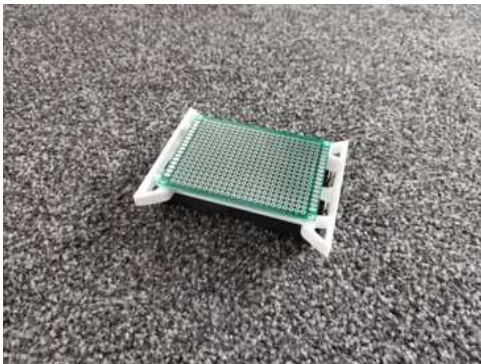
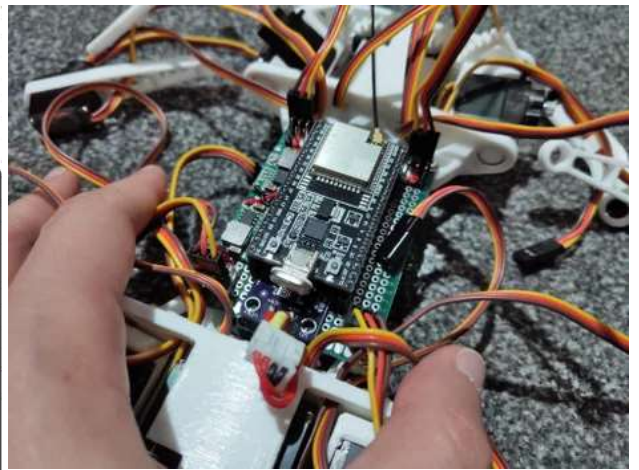
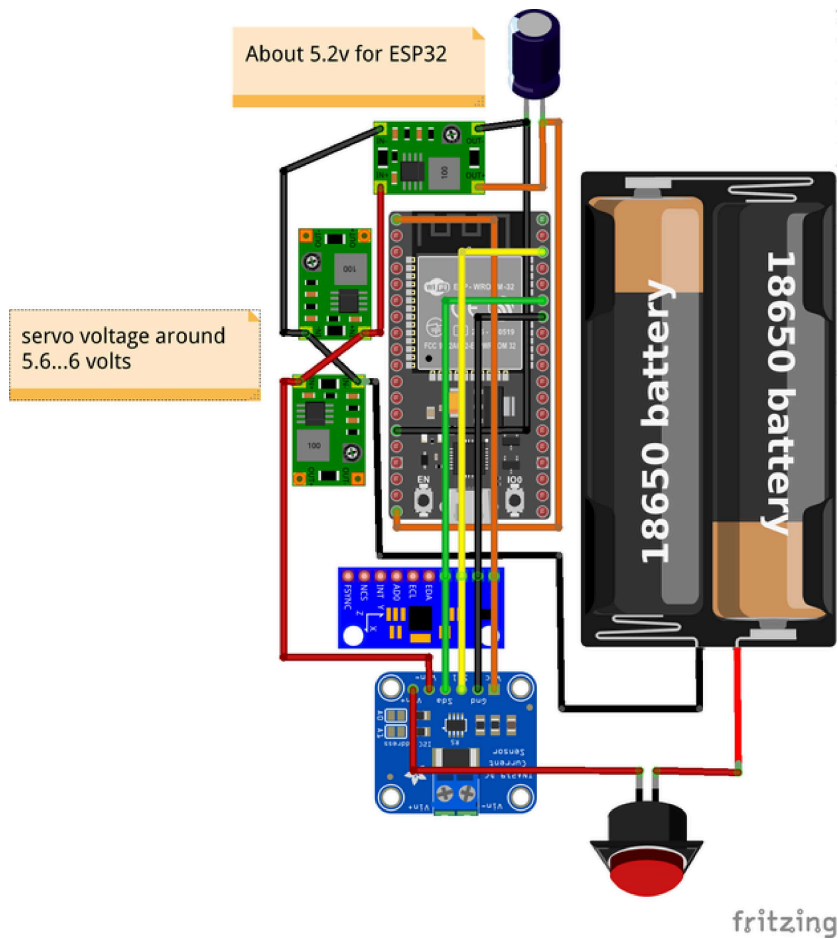
Cut servo horn to make it fit inside "servo gear", glue it and install on the servo as usual. Insert servos into body. Screwing it can be challenging.

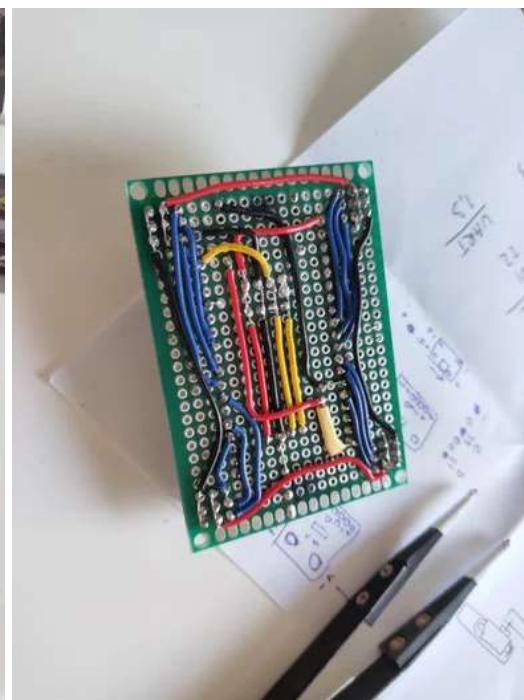
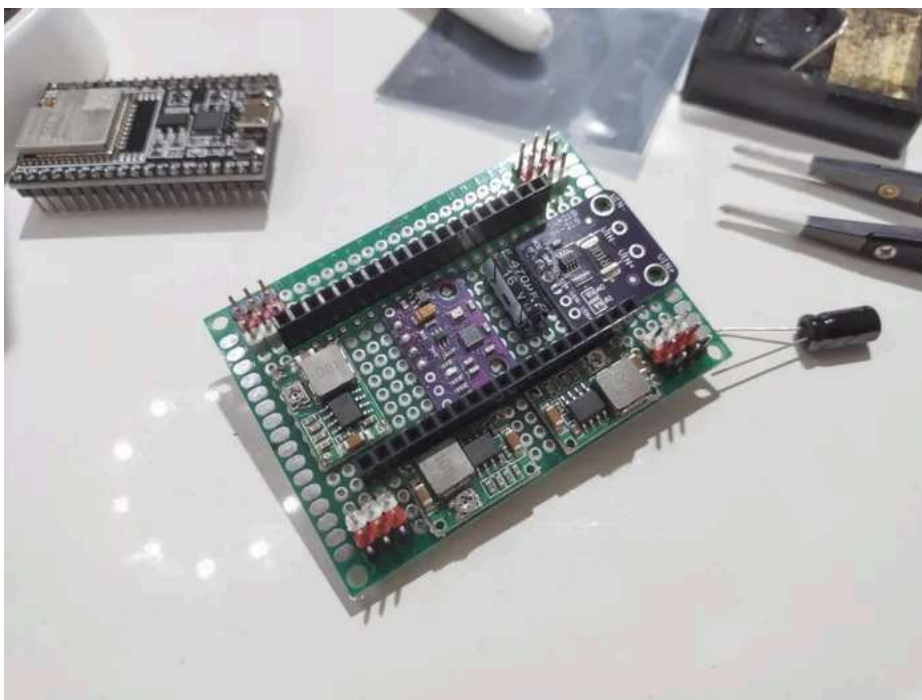
Squeeze "shoulder" part without servo (but with cable in place) and insert it between bearings.

Repeat with other servos and parts.

You don't need to fix legs with bolts at that moment, later you will need to calibrate legs following instruction in my GitHub repository: <https://github.com/SovGVD/esp32-robot-dog-code>

Step 5: PCB





PCB configuration is up to you. I can only show what I did.

Sensors connected to default I2C bus of ESP32 (GPIO 21 and 22).

Servos connected to:

- Left front: 25 (body servo, servo gear, alpha), 26 (body to leg servo, shoulder, beta), 27 (leg servo, gamma)
- Right front: 16, 18, 17
- Left hind (back): 13, 12, 14
- Right hind (back): 4, 2, 15

You can reconfigure it in the code, but make sure that [ESP32 ISR Servo library](#) support it.

Step 6: Code

Setup [Arduino IDE to work with ESP32](#) but don't use esp32 version 2 and higher. There were some strange changes related to WiFi, so one of the core just freezing in v2.x.x

Install addition libraries (most of them could be installed in Arduino IDE "Sketch" menu -> "Include Library" -> "Manage Libraries...", **please use only specified version**):

- [ESP Async Web Server](#) with [Async TCP](#)
- [ESP32 ISR Servo](#) version 1.1.0
- [MPU9250_WE](#) version 1.1.3
- [INA219_WE](#)

Setup WiFi Access point credentials.

1. Rename config_wifi.example.h to config_wifi.h
2. Change APssid to any name of your robot, e.g. SmallRobotDog
3. Change APpass to password (8 or more chars), e.g. MySup3rPassw0rd
4. After code upload to ESP32 you should be able to find SmallRobotDog WiFi with your phone or tablet.

Open browser and input URL <http://192.168.4.1/> and you will see two virtual joystick to control robot dog.

Step 7: Good Luck!

Hope you like that project. More code and instruction updates on the go and hopefully will be release soon.