Assembling the robot

bill of materials

1x chassis
1x suspension
2x wheel
1x markerActuator
1x markerClamp
1x boardHolder
2x nema17 pancake stepper motors
2x servo42c stepper motor driver kits
8x m3x25 bolts
8x m3x5 bolts
4x m4x10 bolts
4x m4 nuts
1x 9g metal gear servo
1x pen spring
1x esp32 devboard
1x generic push switch
1x passive buzzer
1x ssd1306 oled
1x 5v regulator
1x 12v 1.2f capacitor bank
2x 6 pin header (m)
1x 3 pin header (m)
1x 2 pin header (m)
1x 4 pin header (f)
1x 2 pin header (f)
1x expo dry erase marker (chiseled tip)

step 1:

print all of the parts:

1x chassis
1x suspension
2x wheel
1x markerActuator
1x markerClamp
1x boardHolder

step 2:

install the driver board on to the motors
install the driver boards onto the motor as described in the manual, substituting the included bolts with m3x25 bolts

step 3: install the motors into the frame

use m3x5 bolts to attach the motors to the motor mounts on the chassis (install both motors)
install ONE of the wheels

step 4: install the suspension

position the suspension so that the holes on the chassis and the holes on the suspension align

thread a piece of 1.75mm filament (approximately 105mm long) through the holes to attach the suspension

at this point, the second wheel can be installed

step 5: install the back caster bearing

use 2 m4 nuts and bolts to attach the caster bearing to the suspension

step 6: install the front caster bearing and board holder

position the caster bearing so that it aligns with the mounting holes on the chassis.

Align the board holder with the holes on the chassis, then install 2 m4 nuts and bolts to secure the caster bearing and board holder, do not install the nuts

step 7: build the main board

build the main board in accordance to the following wiring diagram:

make sure to leave some free space at the bottom of the board (about 5mm)

step 8: install the motor cables

connect the included motor cables to the motors, ensure that the other end is long enough to reach the motor connectors on the main board

step 9: install the main board

use glue to attach the main board to the boardHolder (test the board to make sure that it works before doing this)

building the tool head

step 10: install the servo into the markerActuator

use 2 m2 bolts to attach the servo to the markerActuator

step 11: install the marker

slide the marker into the marker clamp with the tip of the marker facing the same direction as the rod on the marker clamp

install the m3 bolts into the marker clamp (do not tighten them all the way)

step 12: attach the marker clamp to the servo arm

use a piece of string to attach the servo arm to the marker clamp

finish the robot

step 13: install the tool head

use 2 bolts to attach the tool holder to the board holder of the robot

the hardware is now complete

software

step 14: installing thonny

follow the instructions on thonnys website to install it on your preferred operating system

step 15: flashing the firmware

download the latest micropython firmware from micropython.org
https://micropython.org/download/ESP32_GENERIC/
connect the esp32 to your computer and use thonny to flash the firmware onto the esp32
run > configure interpreter >
select micropython (ESP32) as the interpreter
click install or update micropython
select the port that the board is plugged into
select the firmware that you want to install
click installed
the firmware is now installing (if does not start installing, refer to the firmware installation instructions of the board you are using)

step 16: installing the software on the robot

you will need to install the following files onto the board
buzzer.py
config.py
hardwareTest.py
motorCalibration.py
navTest.py
rbtMain.py
ssd1306.py
stepperDriver.py
trig.py
the following files are optional
boot.py
you can find these files here: https://github.com/Matt2D3/floorPlanRobot under the robot folder

step 17: running the software on the computer

gcodeParser.py can be used to to convert gcode into a format that the robot can read
to use it, you can either import it and call convertGcode(file name (string), compress output (bool)
or you can replace the file name on line 98 with whatever file you want to parse and run the script

to generate the gcode, use the prusaSlicer profile (floorPlanRobot.ini) included on the github page

running the robot

step 18: marker calibration

have the robot draw something, adjust the marker up or down as needed
this can be done by loosening the screws, adjusting the marker position
and then tightening the screws
repeat this step until the marker is touching the floor just enough to draw

step 19: first drawing

slice an object with the included prusaslicer profile
run the gcode parser on it, copy the output
paste the output of the gcode parser into rbtMain.py on line 10 (cordList = )
click run in thonny, disconnect your computer, and place the robot where you want it to draw
press the start button on the robot and it should begin drawing

calibration

step 20: the calibration script

the calibration script has 3 modes, 1: calibrate steps per degree, 2: calibrate steps per mm, 3: calibrate servo
in mode 1, the robot will turn on the spot 3 times in 1 direction, pause, and turn 3 times in the other direction,
in mode 2, the robot will move forward 100mm, pause, and then move backwards 100 mm
if the robot undershoots or overshoots these targets, consult step 21
in mode 3, the servo will rotate to its highest position, wait for the start button to be pressed, and then the rotate the servo to its lowest position

step 21: calibrating the robot (mode 1)

in motorCalibration.py set mode = 1, then click run.
The robot will turn around 1 time, you will be prompted to enter a new steps per degree value
once you enter a value and press enter, the robot will repeat the calibration.

step 22: calibrating the robot (mode 2)

in motorCalibration.py set mode = 2, then click run.
The robot will move forwards 1 meter, you will be prompted to enter a new steps per mm value
once you enter a value and press enter, the robot will repeat the calibration.