n

# *DON'T PANIC*

**The Complete User's Manual to Troubleshooting the Multitude of Problems that Inevitably Come Up With MarkerBot 2000**

Table of Contents

## Introduction

Robots never, ever work like they're supposed to.

That's not even an exaggeration. In my (at the time of this writing) three-years-and-one-summer of experience designing, building and sometimes coding full-size competition robots for my after-school robotics team, not a single robot has yet failed to malfunction. Sometimes it's a small malfunction and a correspondingly easy fix. Sometimes it's a massive malfunction and you have to take apart half your robot to fix it. No matter how small it is, a robot always breaks down somehow. It's best to accept that fact as soon as possible.

But that's not an excuse to give up! When your robot inevitably breaks down, you don't toss it in the trash, proclaiming that it's worthless and that you'll never touch another robot again. You definitely do *not* kick the problem down the road for a future user of the robot to deal with.

You fix the darn robot.

Luckily, you have your trusty copy of *DON'T PANIC: The Complete User's Manual to Troubleshooting the Multitude of Problems that Inevitably Come Up With MarkerBot 2000* with you. Now you have absolutely nothing to fear! Not even fear itself, because with this manual, you'll be able to fix any problem that MarkerBot 2000 happens to stumble into.

Are you ready?

You'd better be ready, because no one is going to fix this robot except you. Think of yourself as the fabled hero of the prophecy, destined to descend from the heavens and smite whatever inconvenience has foolishly decided to wreak havoc upon the innocent patrons of the UCSF Makers Lab.

Now that I put it like that, are you more inclined to fix the robot? I hope so. Have fun trying to fix whatever problem you encountered!

# How to Use MarkerBot 2000

Before fixing a robot, you should probably know how to use it in the first place.

*In remote-control mode:*

1. Use the joystick to control whether MarkerBot goes forward or turns right or left. At the time of this writing, MarkerBot does not go backwards, since we couldn't buy any motor controllers.
    a. If anyone would like to modify MarkerBot to be able to go backwards, be my guest. However, make sure to update this user's manual with any required troubleshooting tips, and you'll probably need to 3D print a new joystick case since the current one physically prevents the joystick from going backwards.
2. Use the button to lower or raise the rack-and-pinion assembly that moves the pen up and down. The button acts as a toggle, so you only need to press it once to move the pen up or down. No need to hold it

*In autonomous mode:*

1. Using the Arduino IDE, open the program "autonomous.ino" (can be downloaded from my website).
2. Write your code inside the function called main. That is, in the program you will see a line beginning with `int main().` Write your program within the brackets directly after that. Your code would look something like this:

```
int main(){
        example_code; //Your code here!
}
```

3. I've written some functions that you can use to program your robot. If you're more well-versed in coding you can probably deduce what they do and skip ahead. If not, read the lines that are greyed-out at the top of the program "autonomous.ino". Those are called comments, and they do not affect the rest of the program; nonetheless, they contain valuable information about how to program MarkerBot.
4. After you're done coding, upload it to MarkerBot's bottom Arduino board. You can do this by plugging a cable from the laptop to the board, then clicking the "upload" button at the top left corner of the Arduino IDE (ask Jenny for a cable).

## Troubleshooting Problems with Motors

*General things to check first, before moving on:*

1. Ensure the AA battery pack is plugged into the breadboard.
2. Ensure all AA batteries are fully charged.
3. Ensure the motors are plugged into the breadboard. Consult the wire labels for proper position of the motor jumper wires.
4. Ensure that both top and bottom Arduino boards are plugged into their 9v power supplies.
5. Ensure that both 9v batteries are fully charged.
6. Check the wire labels to make sure everything is plugged in correctly.

*During remote-control mode:*

If both motors are not running:

1. Using a computer in the Makers Lab, or your own computer:
    a. Ensure that the bottom Arduino board has the program "remote_control.ino" loaded onto it.
    b. Ensure that the top Arduino board has the program "servo_control.ino" loaded onto it.
    c. These programs can be downloaded from my website: nicolaskyoung.github.io
2. Ensure that all wires leading from the joystick to the bottom Arduino board are connected to the correct places.

If only one motor is not running:

1. Perform same steps as if both motors were not running first.
2. Ensure jumper wire connecting motor to breadboard has not broken off from the motor lead. If it is, solder it back on.

If motors are running backwards:

1. Reverse the connection of the backwards-turning motor's jumper wires. Essentially, plug in the red wire where the black wire goes, and vice versa.

*During autonomous mode:*

1. Ensure that all wires are connected to the right places, as indicated by the wire labels.
2. Go through the same steps as you would if it were in remote-control mode.
3. Ensure your autonomous code is uploaded to the bottom Arduino board.

## Troubleshooting Problems with the Servo

*General things to check first, before moving on:*

1. Ensure that all three wires of the servo is plugged in correctly. Consult the wire labels to check which wires go where.
2. Ensure that the I2C connection between the bottom and top Arduino boards is set up correctly. (Consult the wire labels.)
3. Ensure that both Arduino boards are plugged into their 9v power supplies.
4. Ensure that both 9v batteries are fully charged.

*During remote-control OR autonomous mode:*

If the servo rotates (i.e. the pen moves up and down) without any apparent input:

1. Ensure the bottom Arduino board is plugged in as well.
2. Check the 9v batteries and make sure they are both charged.
3. Unplug the top Arduino board and plug the bottom Arduino board in. Then, plug in the top Arduino board. Essentially, turn it off and turn it back on.
4. Ensure that the program “servo_control.ino” is uploaded into the top Arduino board, and that the program “joystick_control.ino” is uploaded into the bottom Arduino board. Both these programs can be downloaded from my website.
5. Remove the rack (the white piece of plastic that holds the pen) from the pinion (the red gear on the servo). If the servo as a result of doing this, then that was probably the problem (the servo was held in a position that it shouldn’t have been in). Replace the rack.
6. Contact an exorcist, because the servo is obviously possessed.

*During remote-control mode:*

If the servo does not respond to button presses:

1. Check that the controller’s button is still connected to the long wires making up its cable, as the solder may have broken. In this case, consult the wire labels, and re-solder the joint.
2. Check that the wires leading from the controller are still plugged in to the bottom Arduino board.

*During autonomous mode:*

1. Follow the same steps that the motor troubleshooting page has.
2. Consult the wire labels to ensure everything is correctly plugged in.

## Final Resort

*If you have tried everything in this manual, and nothing has helped you fix whatever problems you're having with MarkerBot 2000:*

1. Step away from MarkerBot 2000 for an hour or so. Take a walk, go eat lunch, or go back to class or work. After that, come back and try attacking the problem again. You might find a solution to a new problem, in which case, add it to this manual!
2. Email me at nicolaskyoung@gmail.com. I'm the one who built the robot in the first place, so it stands to reason that I should be able to fix it. Include a thorough description of the symptoms of the problem and which steps can be taken to replicate the issue. *Make sure to try everything that could be a possible solution in the manual!* If you try everything that could possibly fix the problem, and the problem still exists, *that* is when you email me to fix it. If it turns out to be something like a dead battery, all further emails asking for help will go straight to my junk folder.
3. File a bug report on my website. Same conditions apply as writing me an email.