

Mecanum Drivetrain Building Guide: How to Build a Mecanum Chassis

The FTC Drive Base Kit – v2 from Studica Robotics includes all the parts needed to complete this Mecanum drivetrain build. This kit is available at <https://www.studica.com/studica-robotics/ftc-drive-base-kit-v2>

These build instructions will walk you through the steps of building your Mecanum Drivetrain. (Also referred to as a Mecanum Chassis or Mecanum Drive Base). This is an example of what your completed project will look like.

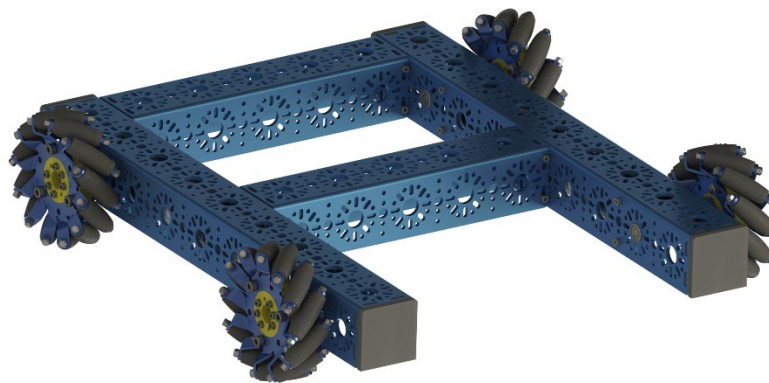


Table of Contents:




- **Parts List:**
 - Structure – page 2
 - Motion – pages 3 - 4
 - Screws, Nuts, Spacers – page 5

- **Mecanum Drive Base Building Instructions:**
 - Frame Assembly – page 6
 - Motion Components – pages 7-8
 - Final Assembly – page 9
 - Code for FTC (Onbot Java)– page 10






Parts List:

Structure:

Description	Part #	Quantity
432 mm U-Channel 	76010	2
240 mm U-Channel 	76014	2
End Piece Plate (2 pack) 	76143-2	2
U-Channel Bumper (4 Pack) 	76505-4	1

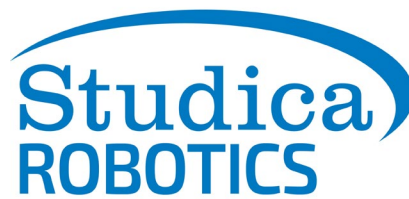
Motion:

Description	Part #	Quantity
NeveRest Orbital Gearmotor JST-VH-2 	am-3637b	4
Orbital Mount Plate 	76138	4
30 Tooth Bevel Gear (2 Pack) 	76219-2	4
6mm x 70mm D-Shaft (6 pack) 	76160-6	1

<p>Clamping Shaft Hub</p> 	<p>76280</p>	<p>4</p>
<p>6mm D-Shape Collar Clamp</p> 	<p>76320</p>	<p>4</p>
<p>100mm Slim Mecanum Wheel Set (Bearing Rollers, 2 left, 2 right)</p> 	<p>76246</p>	<p>1</p>

Screws, Nuts, Spacers:

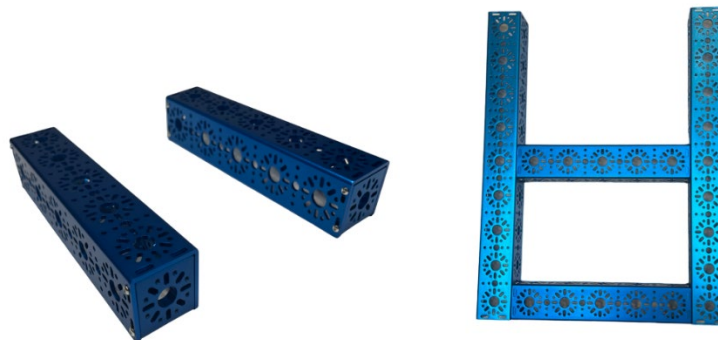
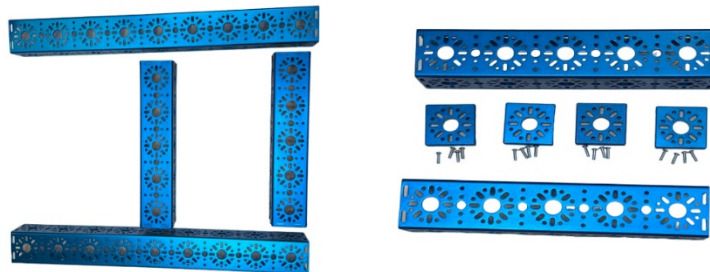
Description	Part #	Quantity
M3 x 6mm Socket Head Cap Screw (50 pack) 	76213-50	1
M3 x 10mm Socket Head Cap Screw (100 pack) 	76201-100	1
M3 x 12mm Socket Head Cap Screw (100 pack) 	76202-100	1
M3 x 25mm Socket Head Cap Screw (50 pack) 	76214-50	1
Shaft Spacer Plastic 6mm ID x 10mm OD x 1mm L (24 pack) 	76305-24	1

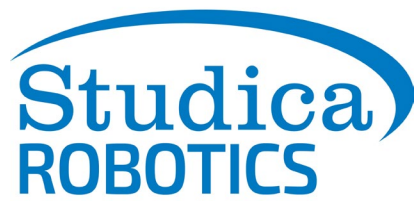


Mecanum Drive Base Building Instructions:

Frame Assembly:

1. Screw the 2 End Piece Plates to each of the 240mm U-Channels using the M3 x 10mm Socket Head Cap Screws
2. Screw the 240mm U-Channels to the 432mm U-Channels using the M3 x 10 mm Socket Head Cap Screws
 - a. The 240mm U-Channels should be placed on the end and center of the 432mm U-Channels (Center being the fifth hole pattern from the end)

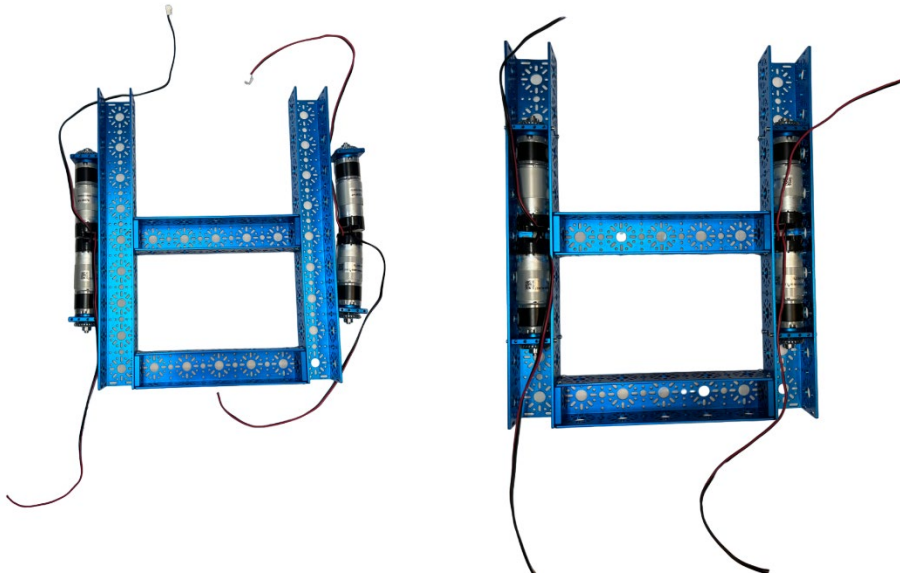




Motion Components:

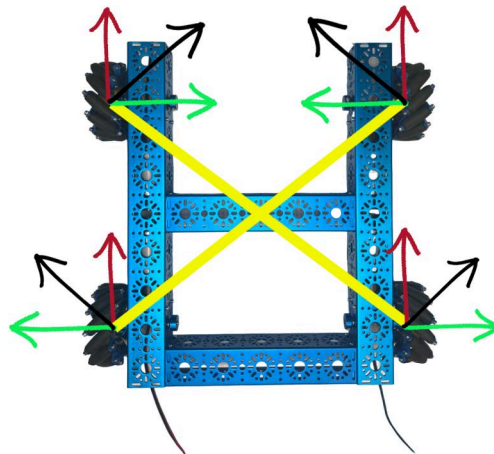
1. Screw the motors onto the Orbital Mount Plate using the M3 x 10mm Socket Head Cap Screws
2. Attach one of the 30 Tooth Bevel Gears to the shaft of the motor and tighten.
3. Secure the Orbital Mount Plate to the 432mm U-Channels using four M3 x 6mm Socket Head Cap Screws on each plate.
 - a. Motors should recess in the 432mm U-Channels as shown in the image.
4. Screw the Clamping Shaft Hub to the 100mm Slim Mecanum Wheels using six M3 x 25mm Socket Head Cap Screws for each wheel.



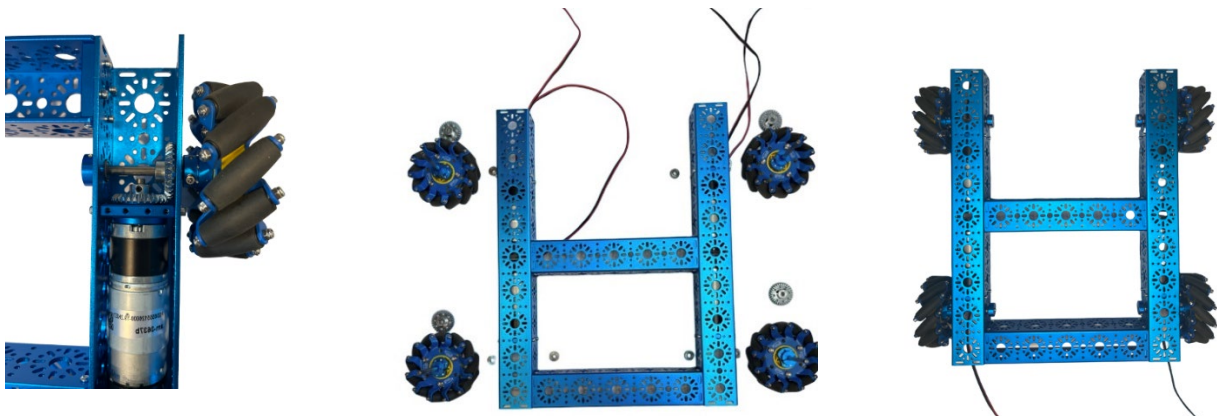


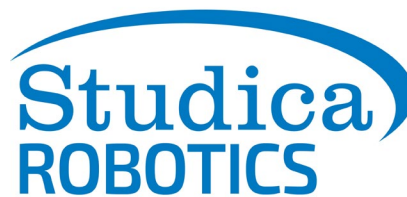
Final Assembly:

1. Place Flange Bearings onto the 432mm U-Channels where the shafts will be inserted, on the hole pattern adjacent to the Bevel Gear attached to the motor's shaft.
2. Insert the 6mm x 70mm D-Shaft into the Clamping Shaft Hub so it is about flush with the shaft.
3. Insert the 70mm D-Shaft through the Flange Bearings and 30 Tooth Bevel Gear on the 432mm U-Channel
 - a. Before tightening the set screws for the Bevel Gears, make sure the driving gear on the motor shaft and the driven gear on the wheel shaft mesh with one another.
4. The wheels of the drivebase should be oriented like the image below. The rollers should be angled towards the center of the robot in an "X" pattern. This is so the drivebase can drive in a as well as perform rotations about its center of mass.

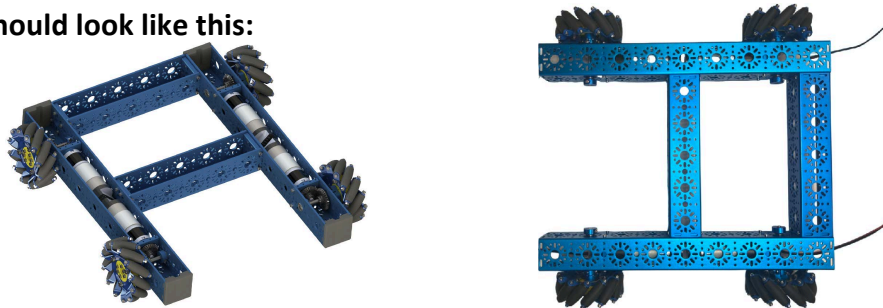


5. Place the 6mm D-Shape Collar Clamp on the end of the 6mm x 70mm D-Shaft and secure it by tightening the clamp with the given set screw.





Your Build should look like this:



Code for FTC (Onbot Java):

Below is the code for the FTC Drive Base Kit – v2 in Onbot Java for FTC. The variables “x”, “y”, and “rx” represent the 3 axes of movement the mecanum drivebase. The left stick controls x and y axis movement (forward, backward, and strafing left to right). The denominator variable is the absolute value of the largest motor power or 1. This is to maintain the power ratio between -1 and 1.

Java Code:

```
package org.firstinspires.ftc.teamcode;

import com.qualcomm.robotcore.eventloop.opmode.LinearOpMode;
import com.qualcomm.robotcore.eventloop.opmode.TeleOp;
import com.qualcomm.robotcore.hardware.DcMotor;
import com.qualcomm.robotcore.hardware.DcMotorSimple;
import org.firstinspires.ftc.robotcore.external.JavaUtil;

@TeleOp(name = "MecanumDriveCode2 (Blocks to Java)")
public class MecanumDriveCode2 extends LinearOpMode {

    private DcMotor frontRight;
    private DcMotor backRight;
    private DcMotor frontLeft;
    private DcMotor backLeft;

    /**
     * This function is executed when this OpMode is selected from the Driver Station.
     */
    @Override
    public void runOpMode() {
        float y;
        double x;
        float rx;
        double denominator;

        frontRight = hardwareMap.get(DcMotor.class, "frontRight");
        backRight = hardwareMap.get(DcMotor.class, "backRight");
        frontLeft = hardwareMap.get(DcMotor.class, "frontLeft");
        backLeft = hardwareMap.get(DcMotor.class, "backLeft");

        // Put initialization blocks here.
        frontRight.setDirection(DcMotorSimple.Direction.REVERSE);
        backRight.setDirection(DcMotorSimple.Direction.REVERSE);
        waitForStart();
        if (opModeIsActive()) {
            // Put run blocks here.
            while (opModeIsActive()) {
                // Put loop blocks here.
                y = -gamepad1.left_stick_y;
                x = gamepad1.left_stick_x * 1.1;
                rx = gamepad1.right_stick_x;
                denominator = JavaUtil.maxOfList(JavaUtil.createListWith(JavaUtil.sumOfList(JavaUtil.createListWith(Math.abs(y), Math.abs(x), Math.abs(rx))), 1));
                frontLeft.setPower(((y + x) + rx) / denominator);
                backLeft.setPower(((y - x) + rx) / denominator);
                frontRight.setPower(((y - x) - rx) / denominator);
                backRight.setPower(((y + x) - rx) / denominator);
                telemetry.update();
            }
        }
    }
}
```