Microbit Robotics Advanced Level 3

Lesson 1

Introduction To Superbit

Presented by Advanced Superlogic Team

Today's Topic

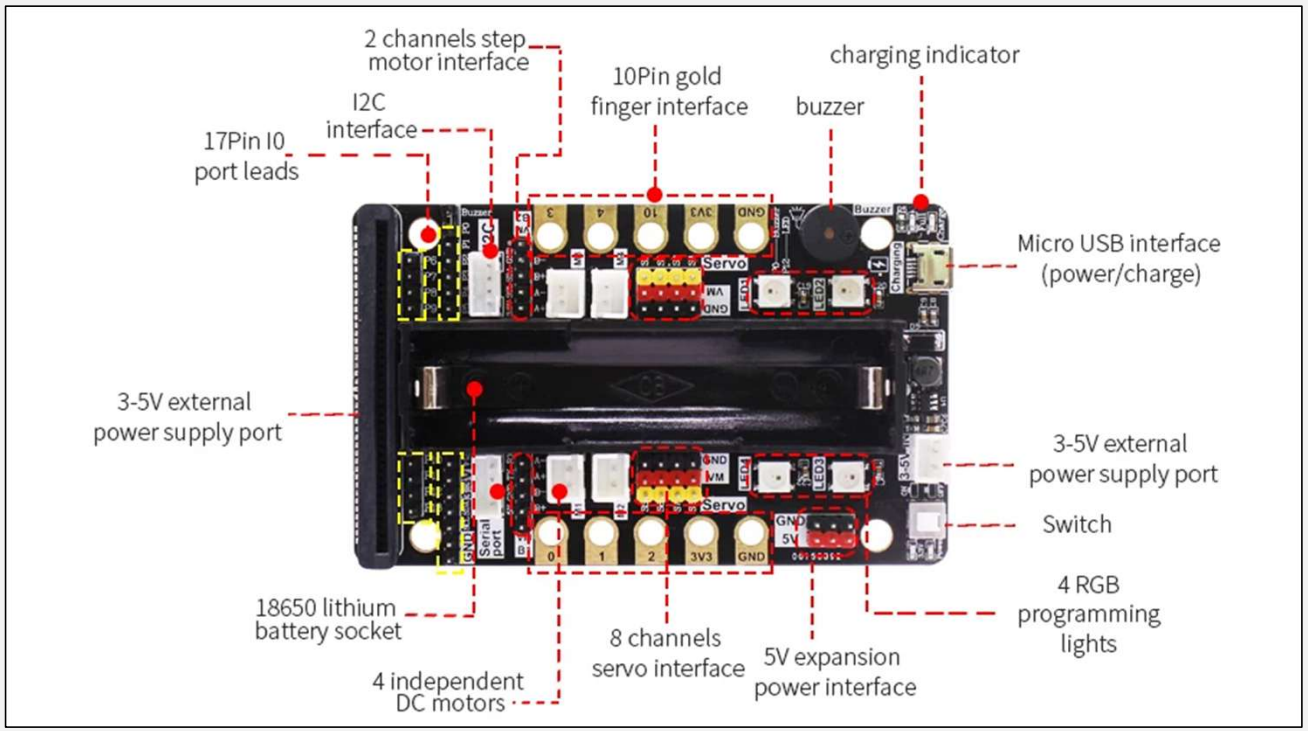
1. Introduction to SuperBit
2. What's in the box?
3. Build a mobile shooter
4. Program the mobile shooter
5. Challenge Time

Learning Outcome

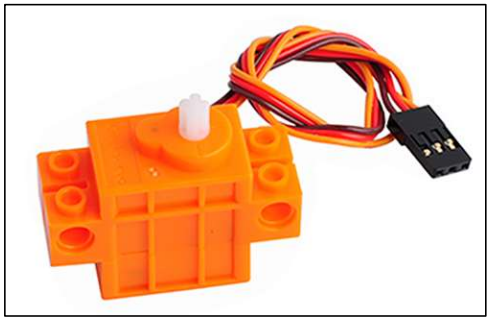
1. Able to identify and differentiate servos and motors
2. Understand the basic operation of SuperBit
3. Able to build a mobile shooter with the SuperBit
4. Able to do basic programming for mobile shooter

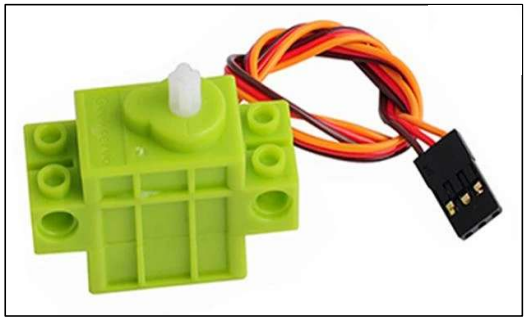
Introduction to SuperBit

SuperBit Expansion Board



Electronic Modules



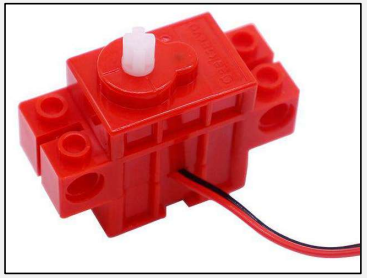


360° Continuous servo

3-wired servo:
Red wire: Positive
Brown wire: Negative
Yellow/Orange wire: data pin (control pin)

270° servo

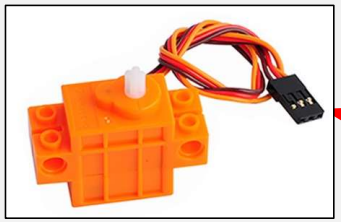
3-wired motor:
Red wire: Positive
Brown wire: Negative
Yellow/Orange wire: data pin (control pin)



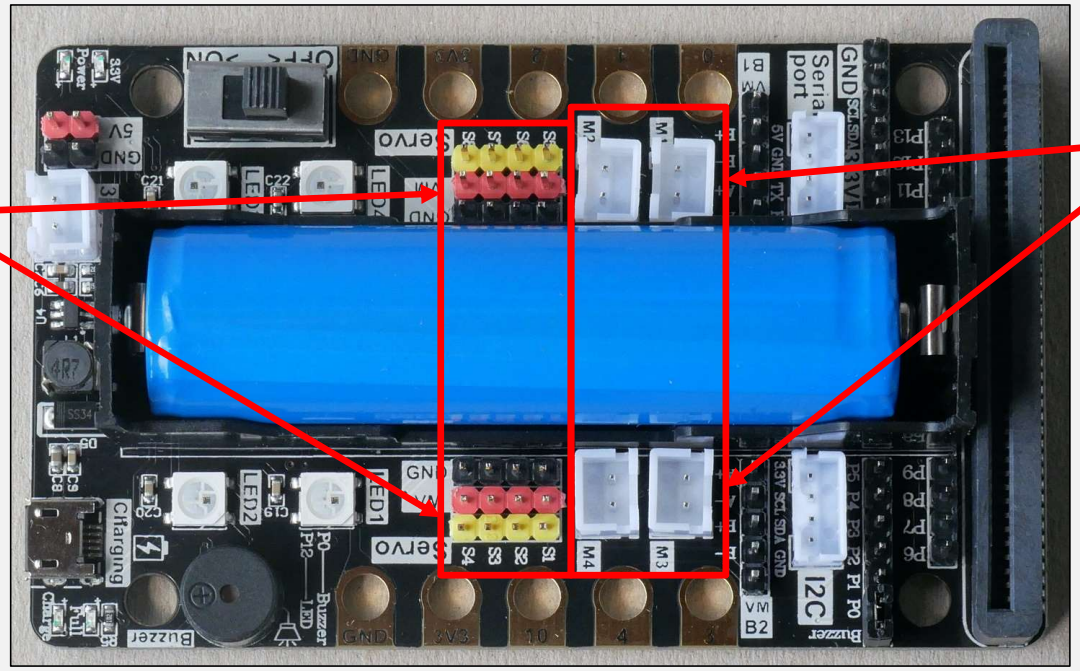
Continuous motor

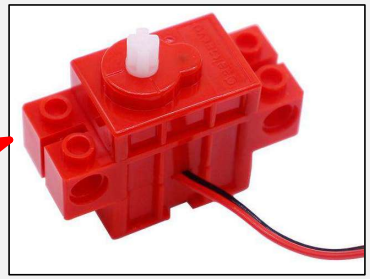
2-wired motor:
Red wire: Positive
Black wire: Negative

Servo & Motor



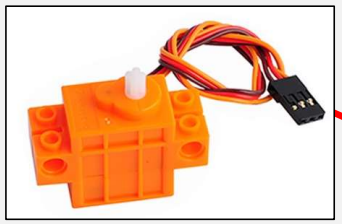
The 3-pin motor is to be connected to any of the S1 to S8 servo ports.



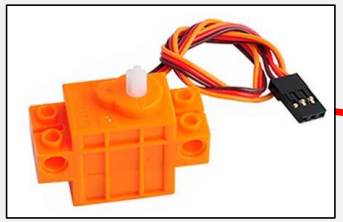


The 2-pin motor is to be connected to the port M1, M2, M3 or M4.

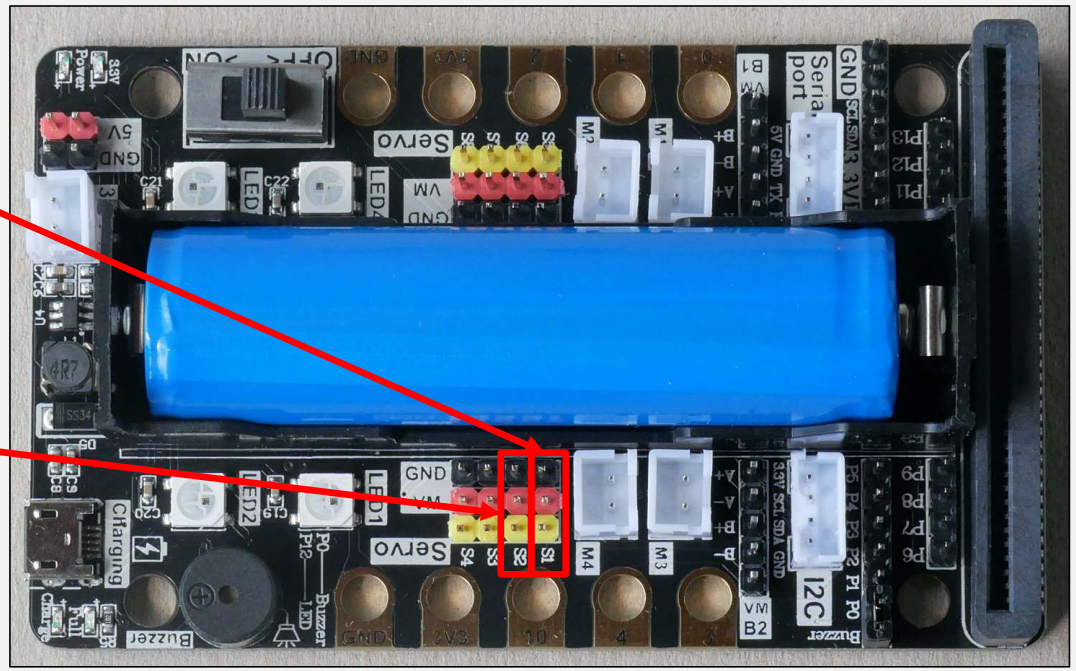
Servo Testing (S1 & S2)



S1



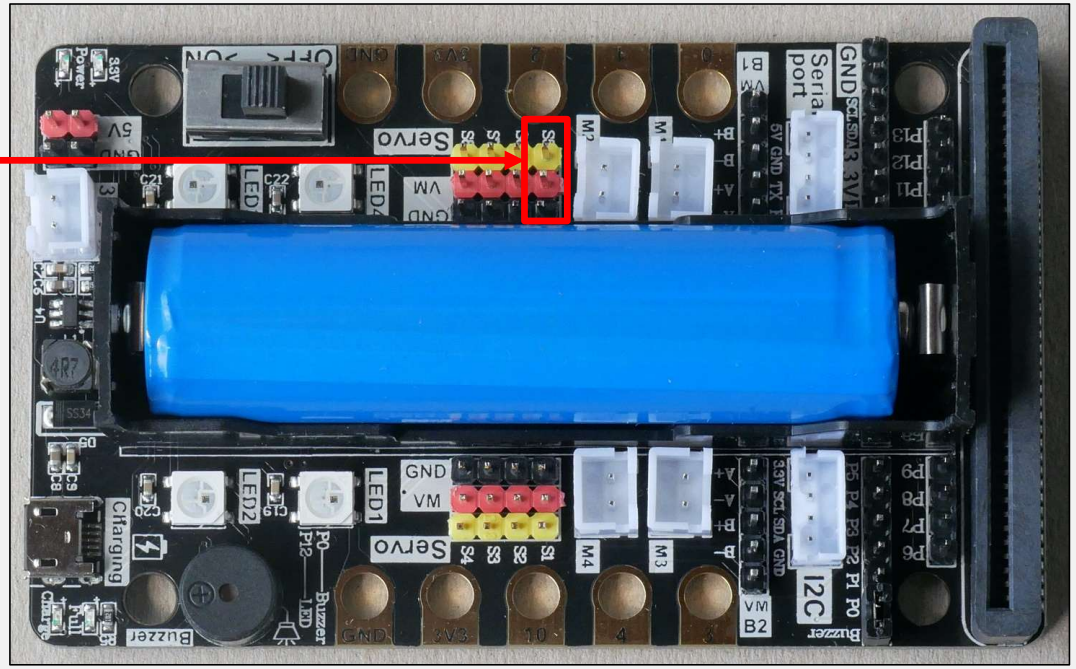
S2



Let's plug in a 360° servo in S1 and do testing with the make code program.

Servo Testing (S5)

S5



Then plug in a 270° servo (gray) in S5.

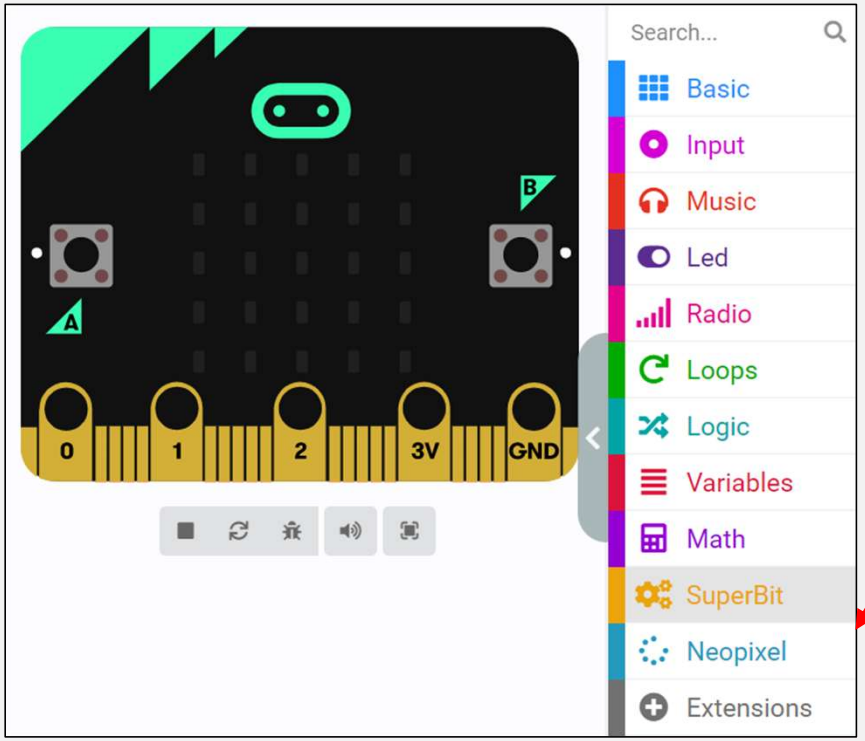
MakeCode Programming

First, we need to connect the micro:bit to the computer by USB cable. The computer will pop up a USB flash drive and click on the URL in the USB flash drive: http://microbit.org/ to enter the programming interface.

Add the Yahboom package https://github.com/lzty634158/SuperBit to program.

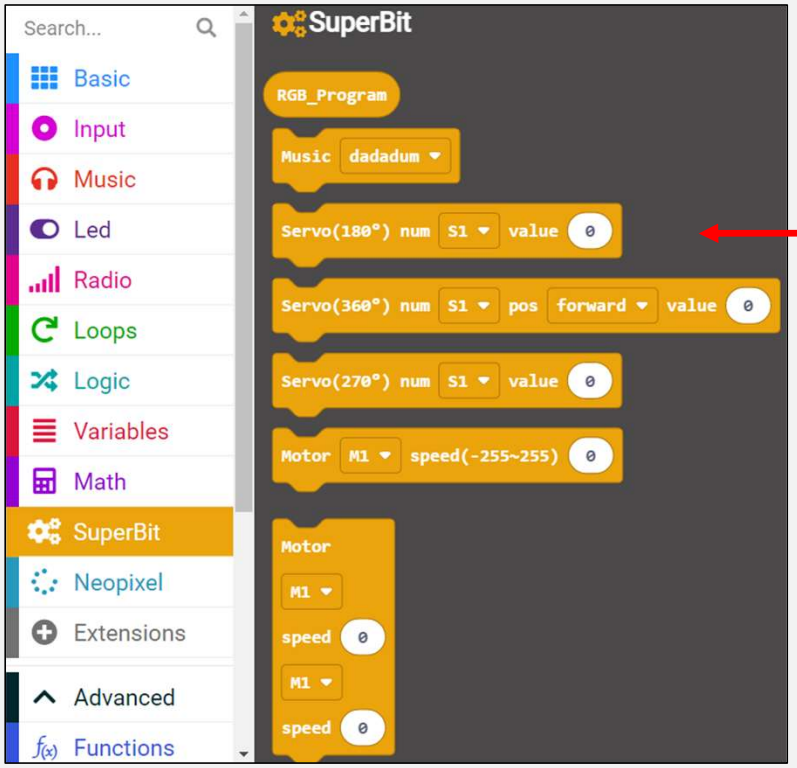
Or search lzty634158/SuperBit in the Microbit extension.

Superbit Extension



After importing the superbit extension, you will see **SuperBit** and **Neopixel** in your coding blocks.

Superbit Extension

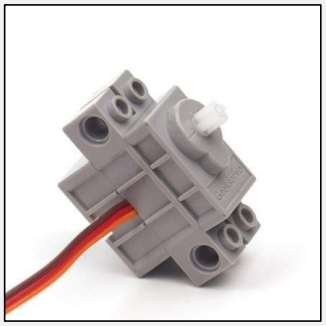


To program our Robot to move, we have some servo and motor blocks here.

Different type of servo requires corresponding coding block for it.

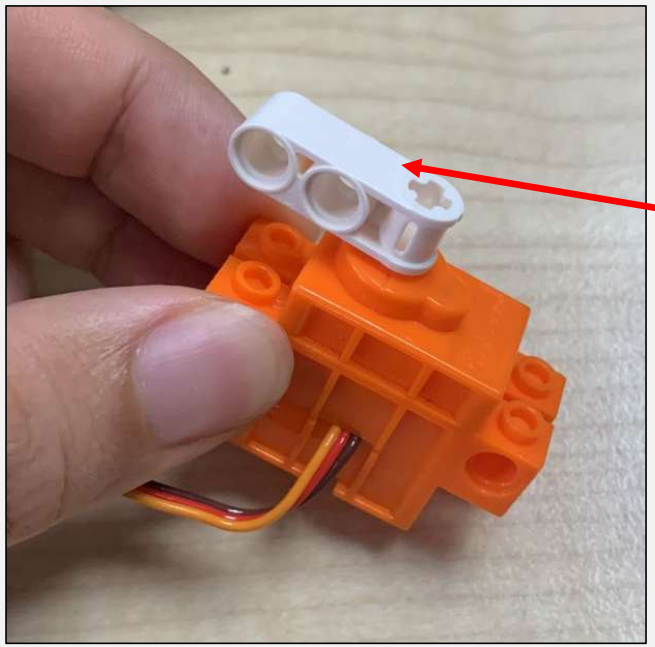
Servo and Motor

To make the **360° continuous servo** turn, we should program it with the **Servo(360°) block**, and select the port that we use in our model, and the action we want it to perform.



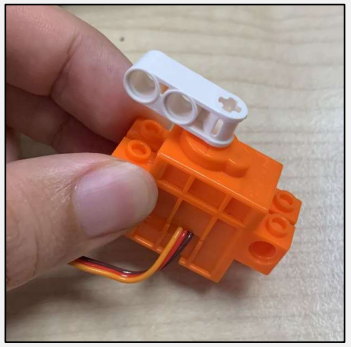
To make the **270° servo** turn, we should program it with the **Servo(270°) block**, and select the port that we use in our model, and the action we want it to perform.

How does servo turn with coding?



To observe better on how the servo is turned, attach a brick on 3 of the servos.

Servo 360° Continuous turn

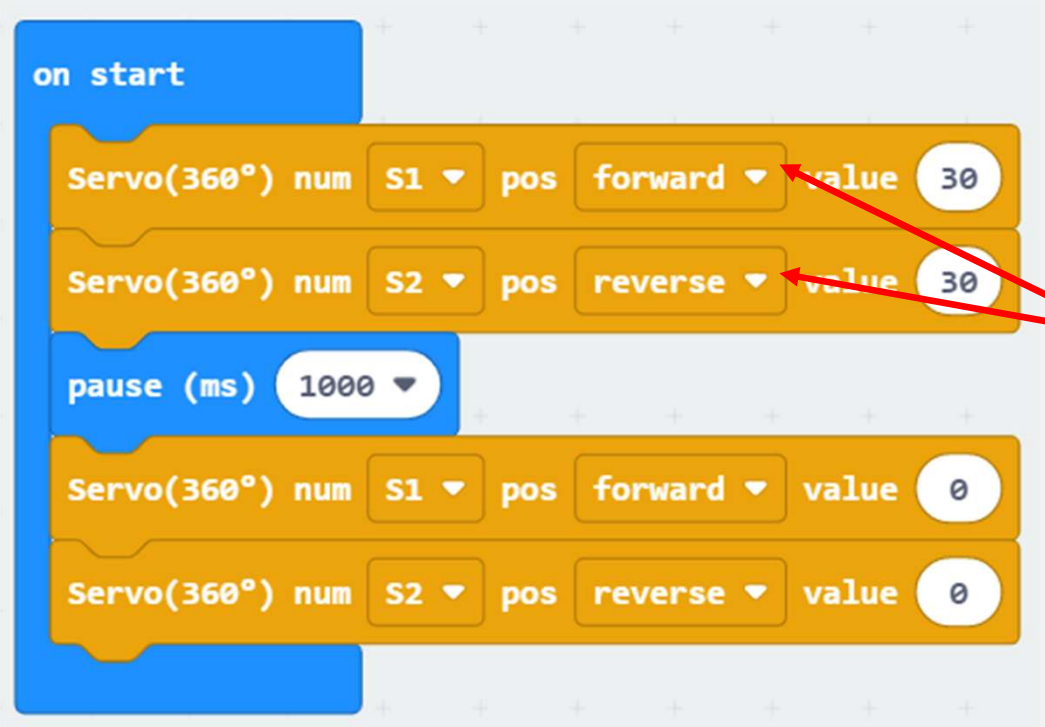


Try to drag these block, and download to your Microbit, it will keep turning forward (clockwise) with the speed of 30.

Servo 360° Continuous turn

To make it stop turning after 1 second, we can add a pause 1 second after the move forward, then followed by moving forward with 0 speed.

Move forward as wheel

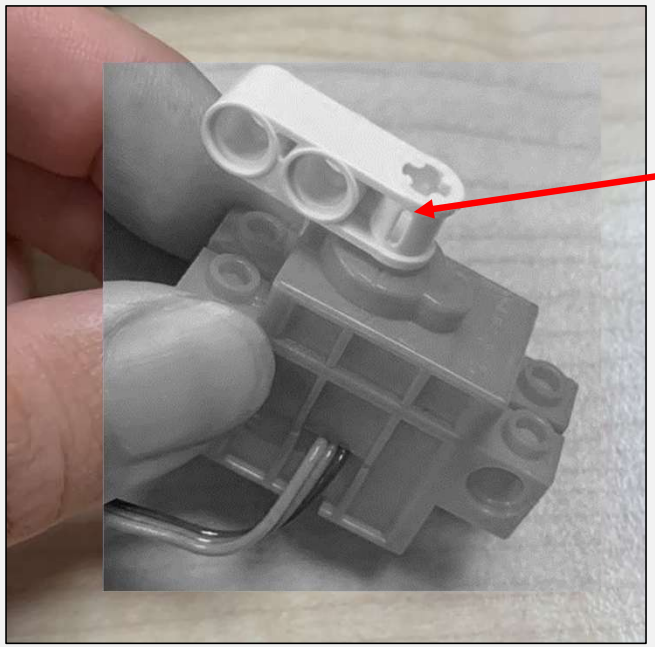


Since the placement of the servos are pointing outward and attached with wheels.

To make the robot move forward, one side of the servo must turn clockwise while another side should turn counter-clockwise, so we need to make S1 to move forward while S3 move in reverse direction.

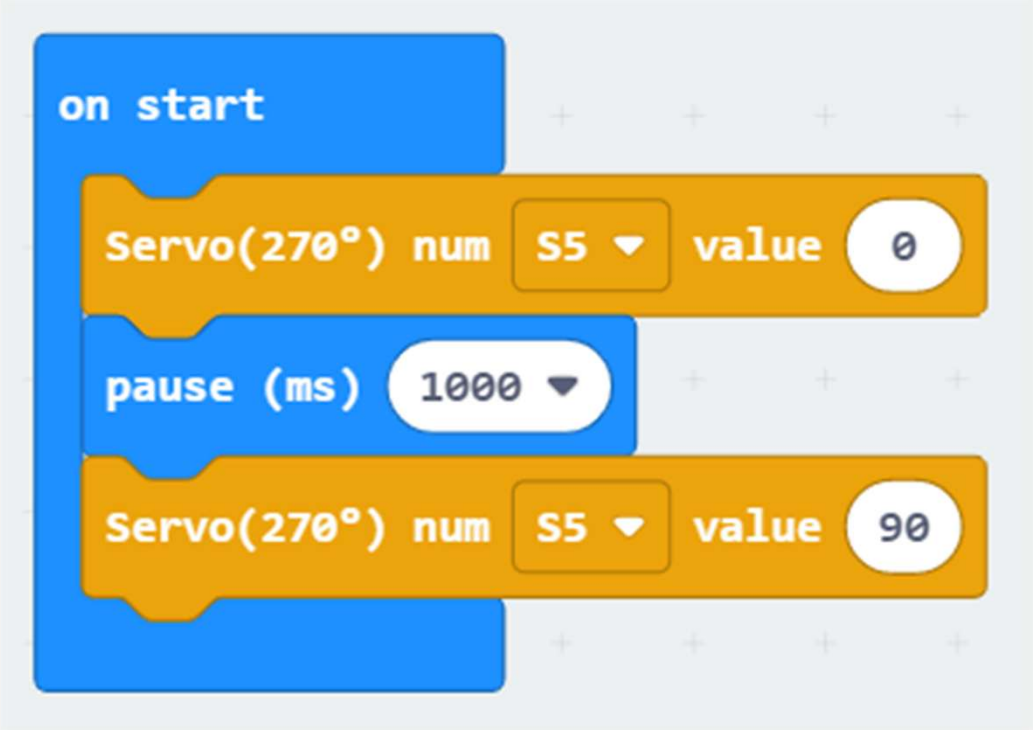
This program will make the robot move forward for 1 second, then stop.

Servo 270° Position Turn



Attach a brick on the 270° servo to observe the turning.

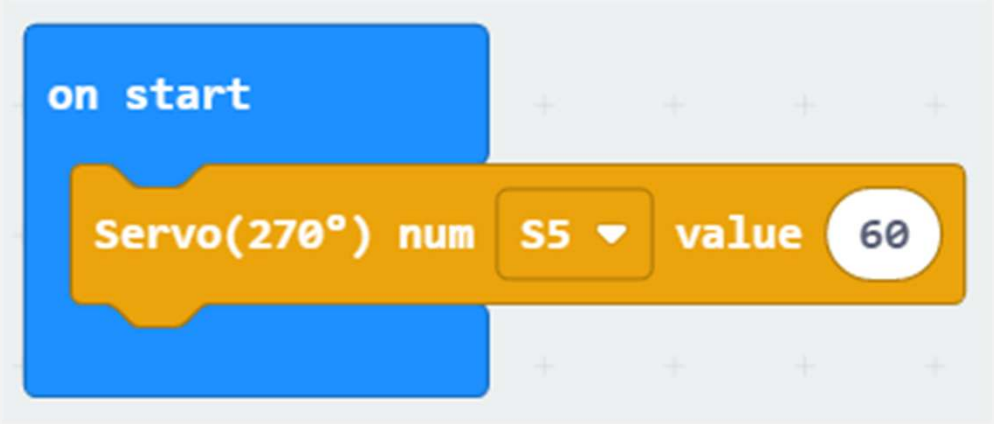
Servo 270° Position Turn Coding



This coding will make the 270° servo turn to zero position first at start, then it will turn to 90° position after 1 second.

As 270° servo has a range to turn, so we should not manually turn our servo, it may cause over-turned and break the servo.

Servo 270° Start position

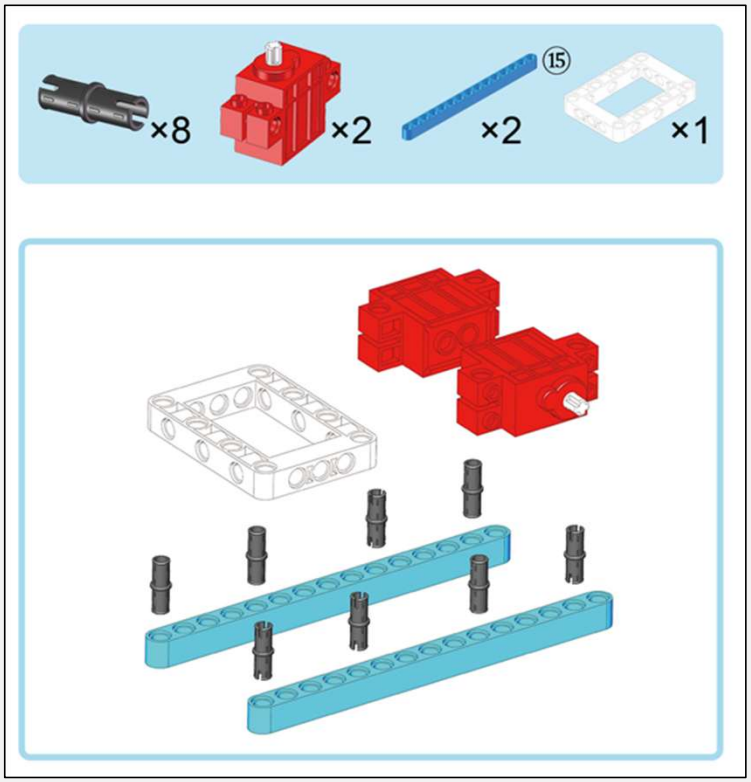


Let's download this before we start building.

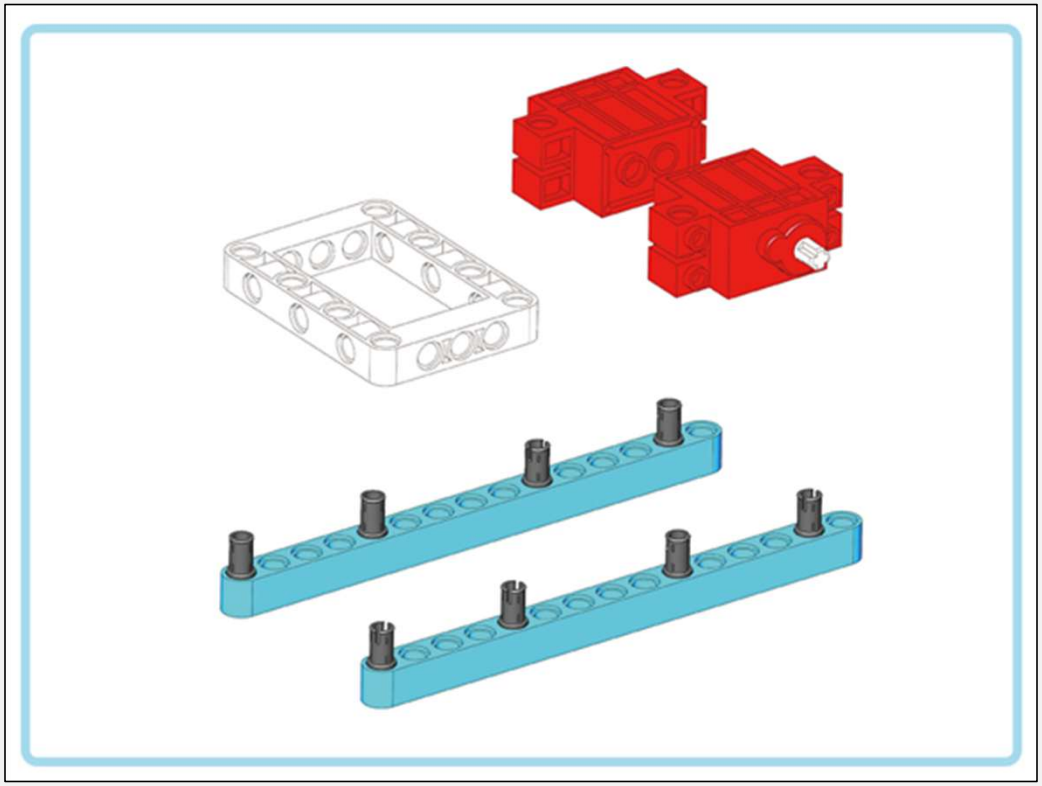
Today we are going to build a Mobile Shooter

Let’s build a Mobile Shooter

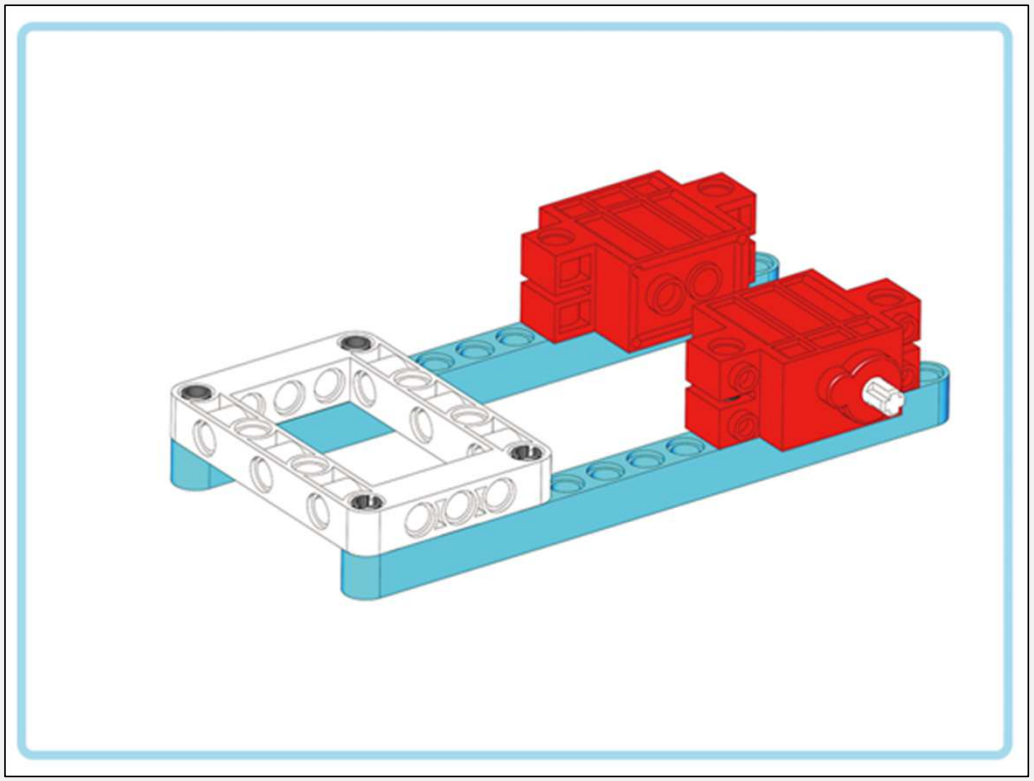
Step 1



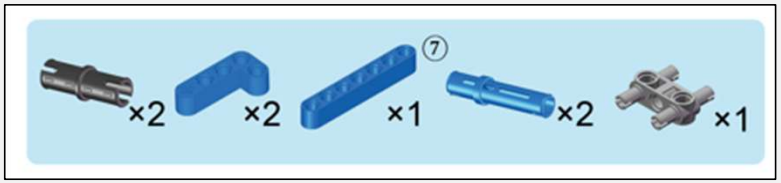
Step 2

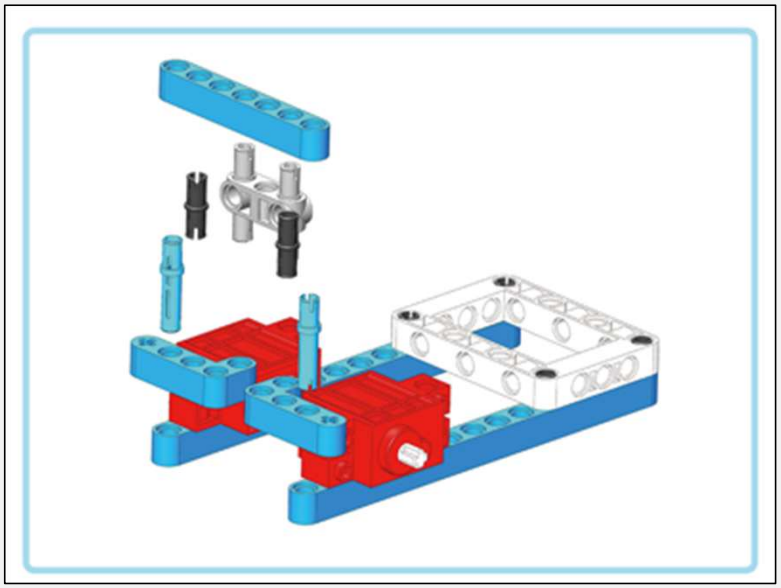


Step 3

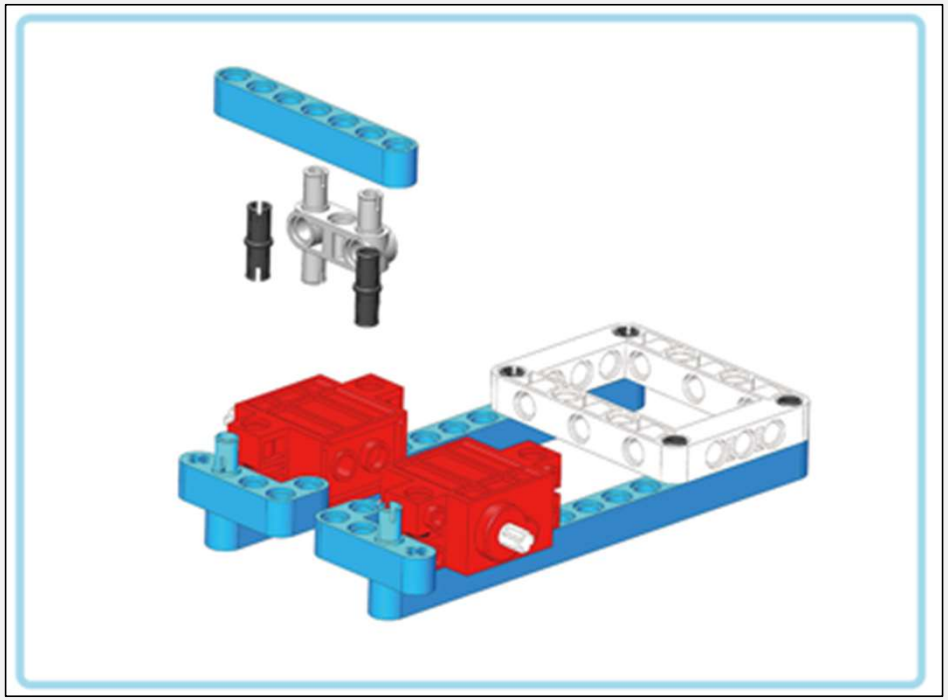


Step 4

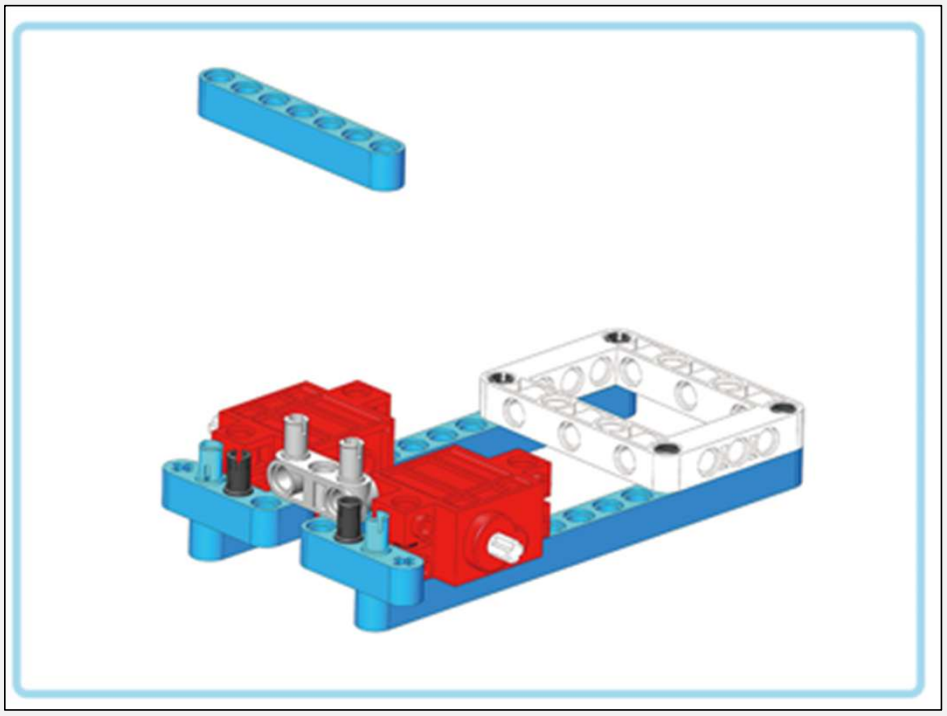




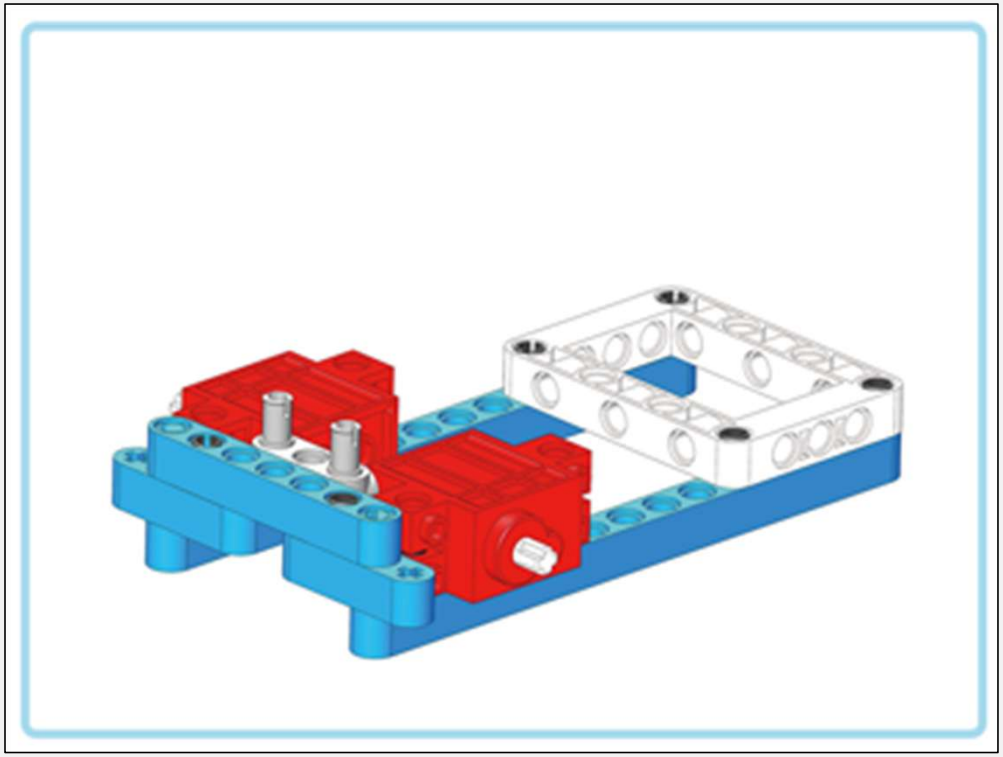
Step 5



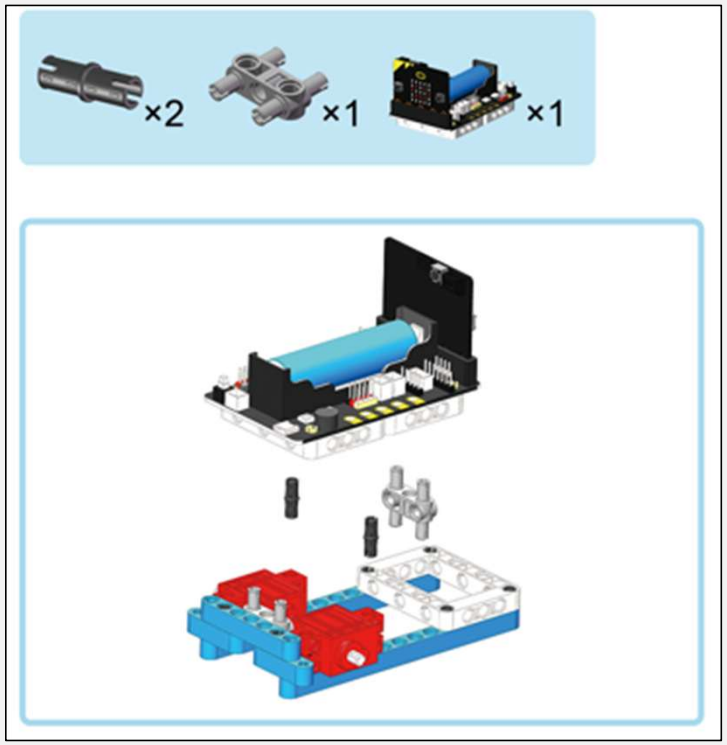
Step 6



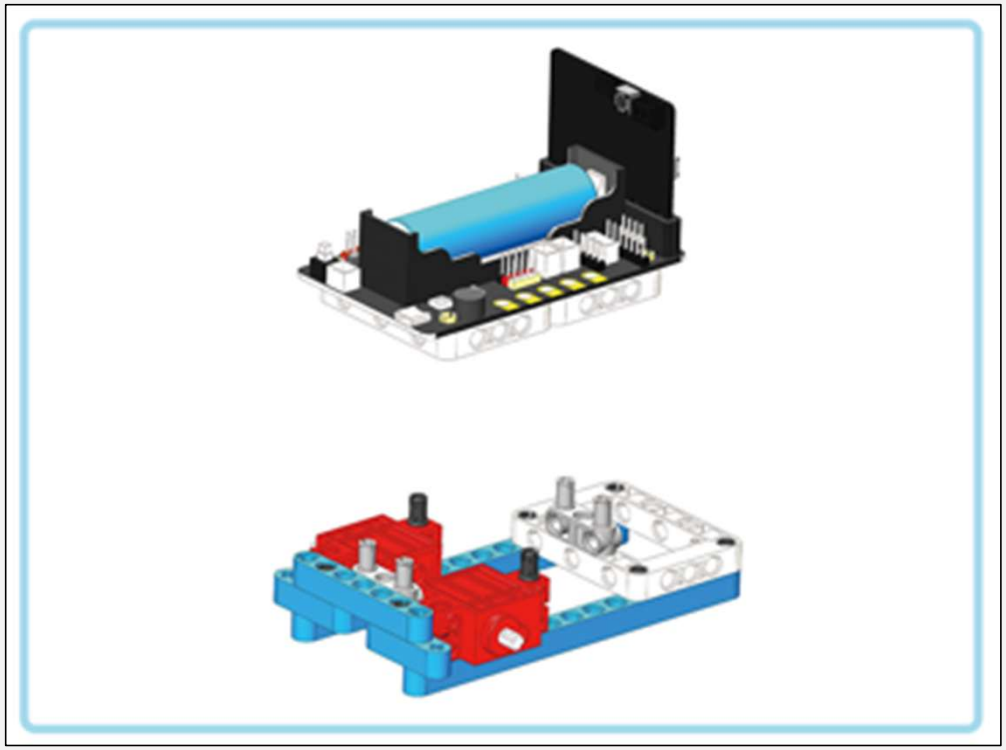
Step 7



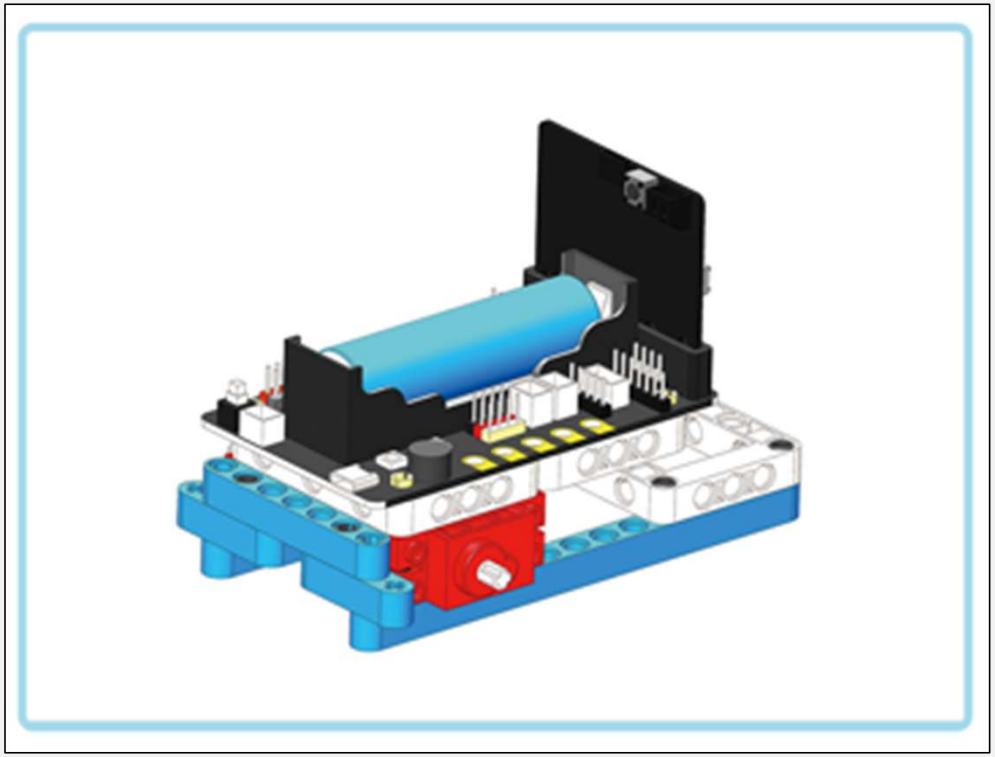
Step 8



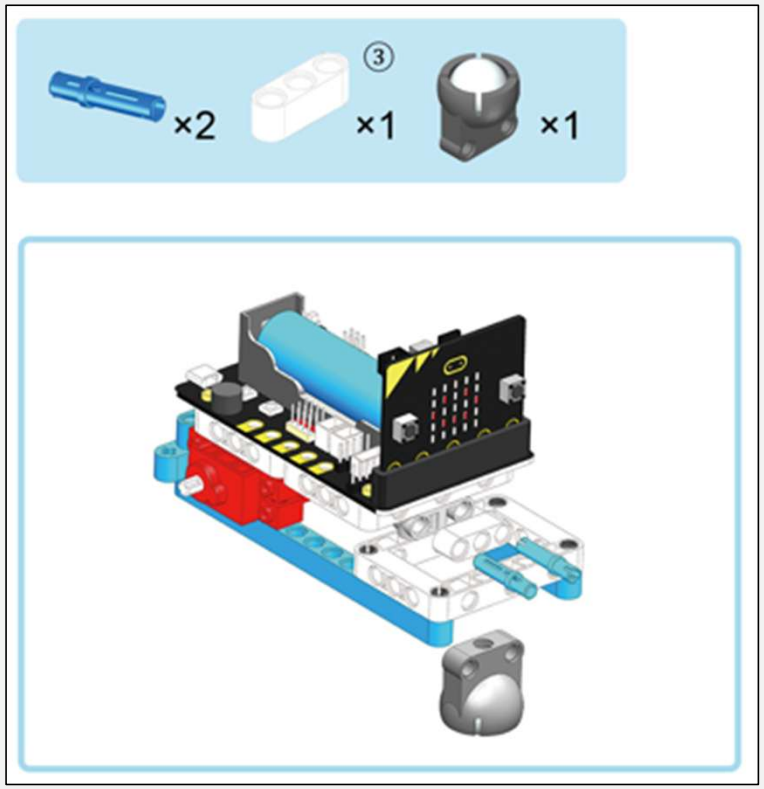
Step 9



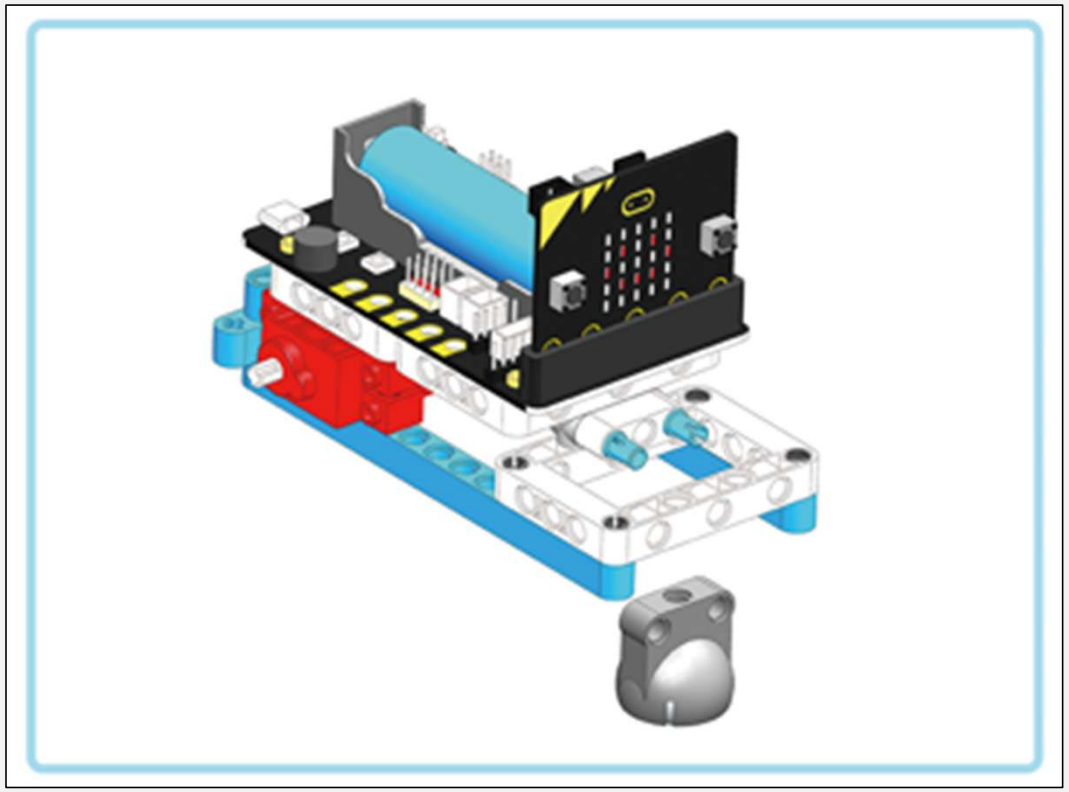
Step 10



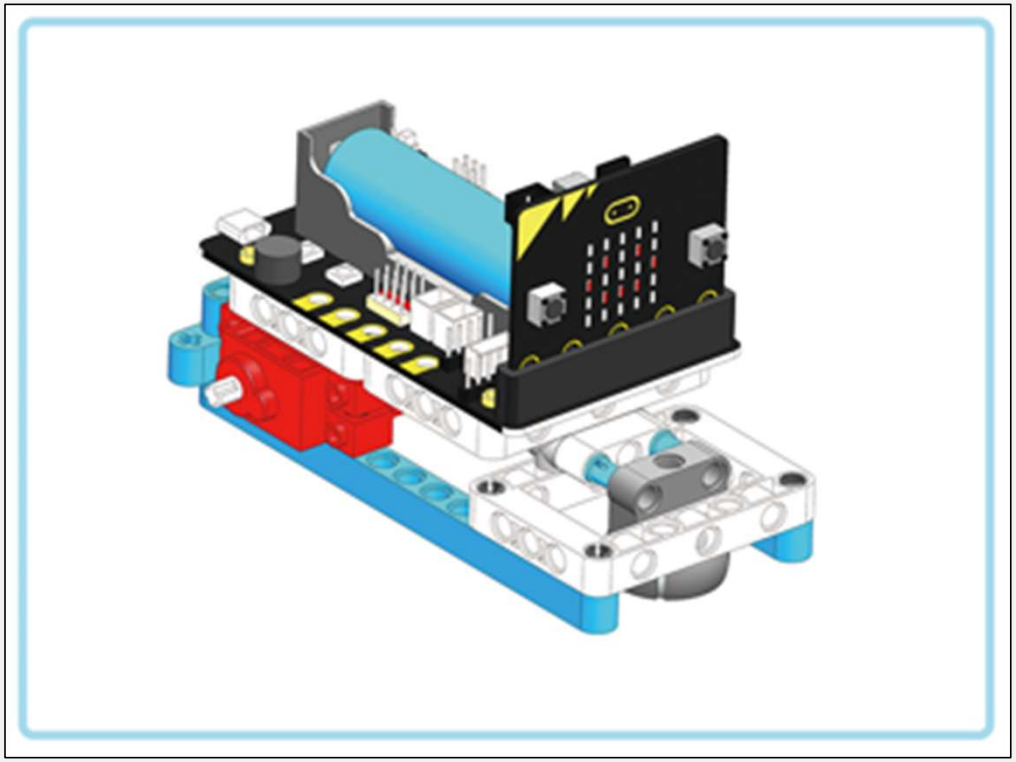
Step 11



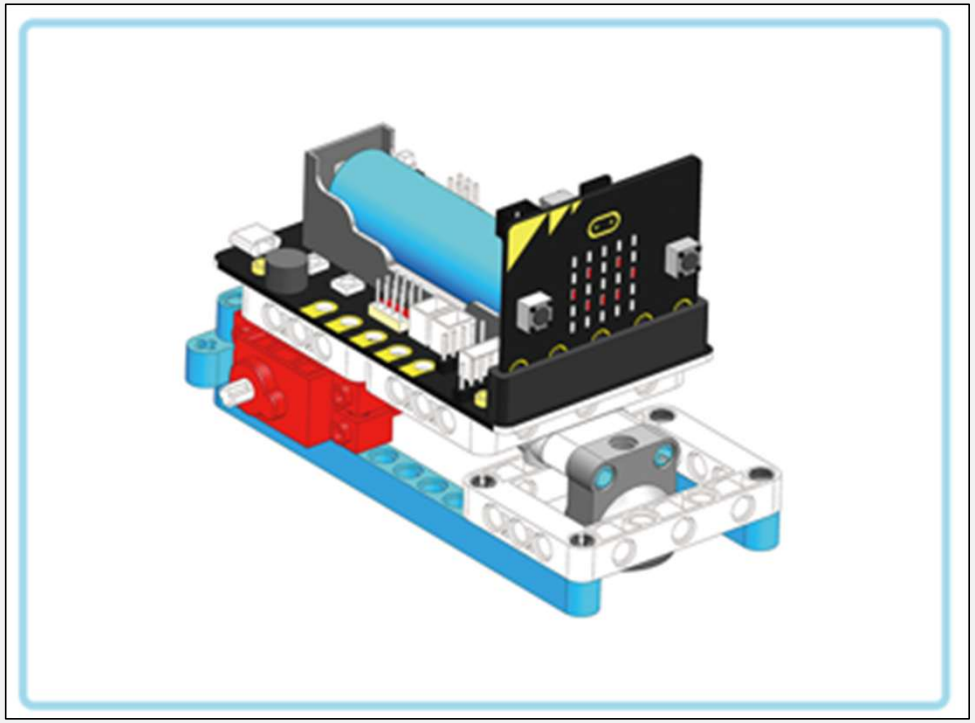
Step 12



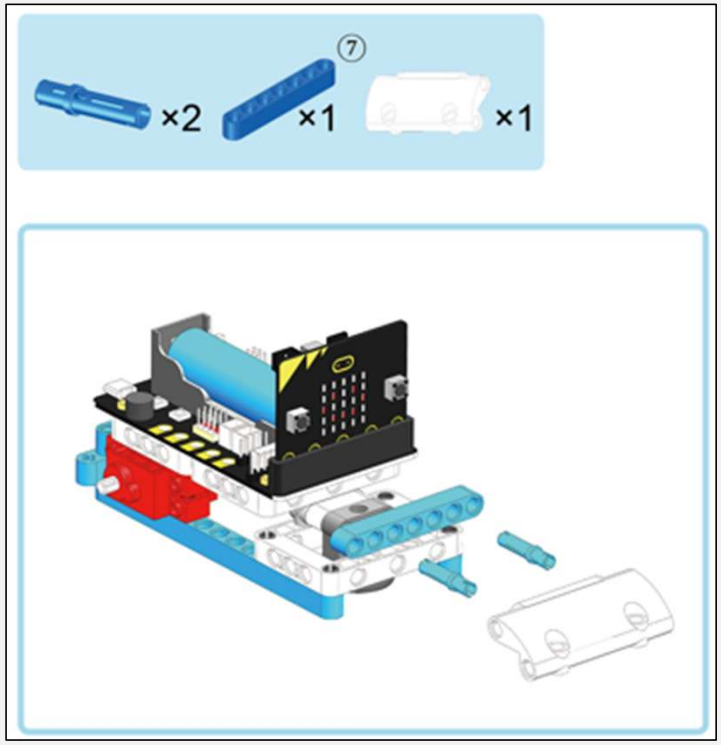
Step 13



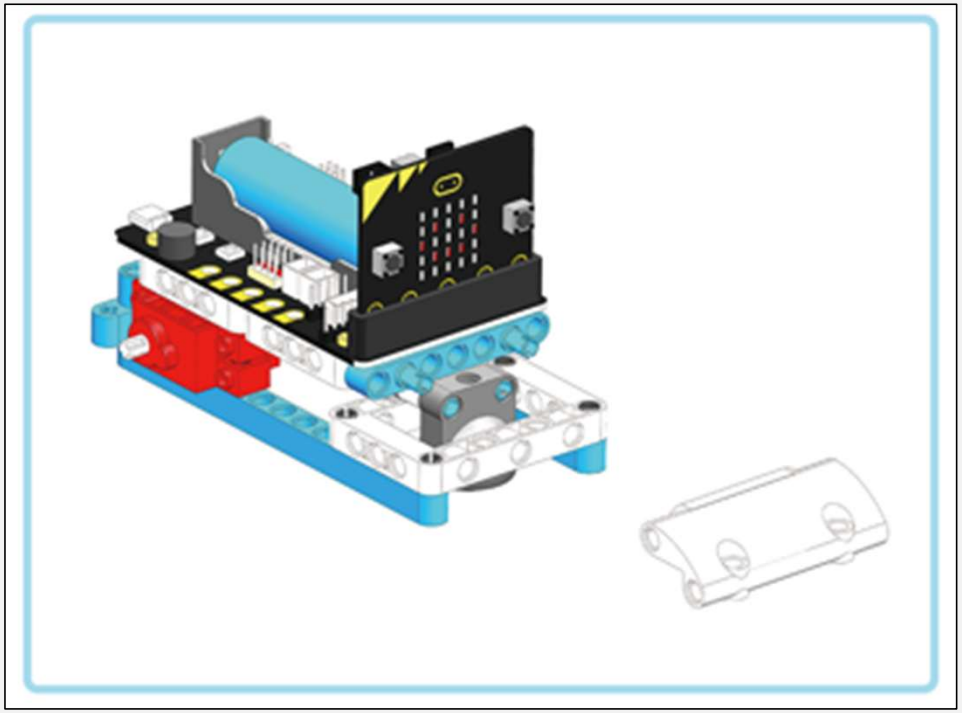
Step 14



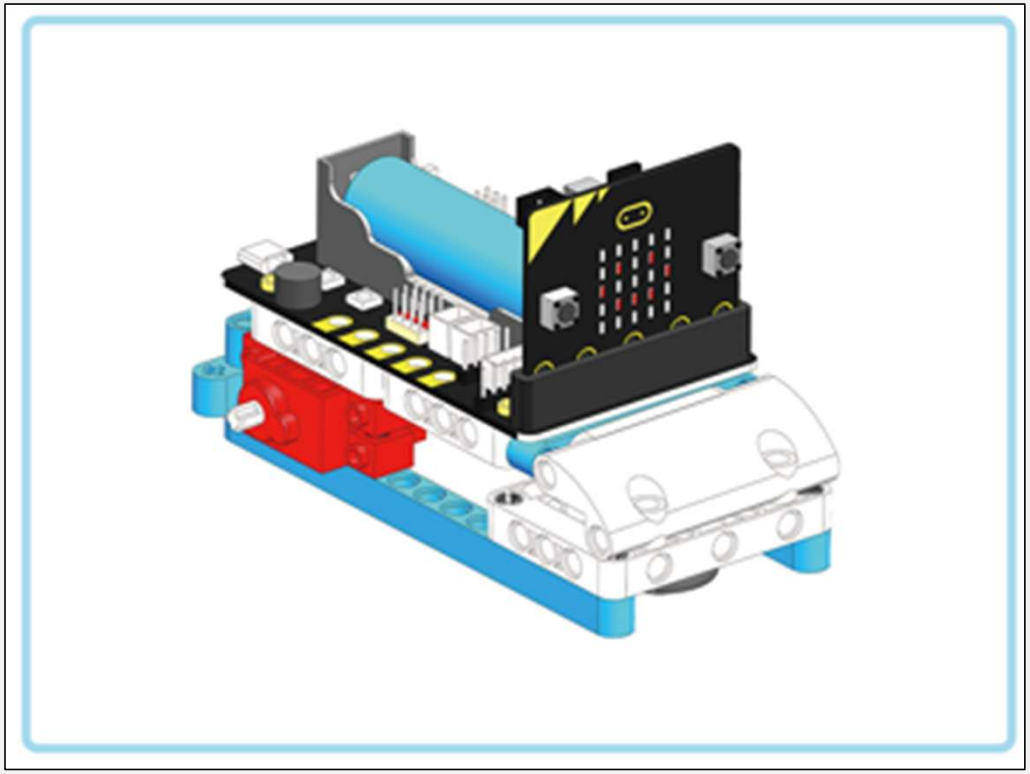
Step 15



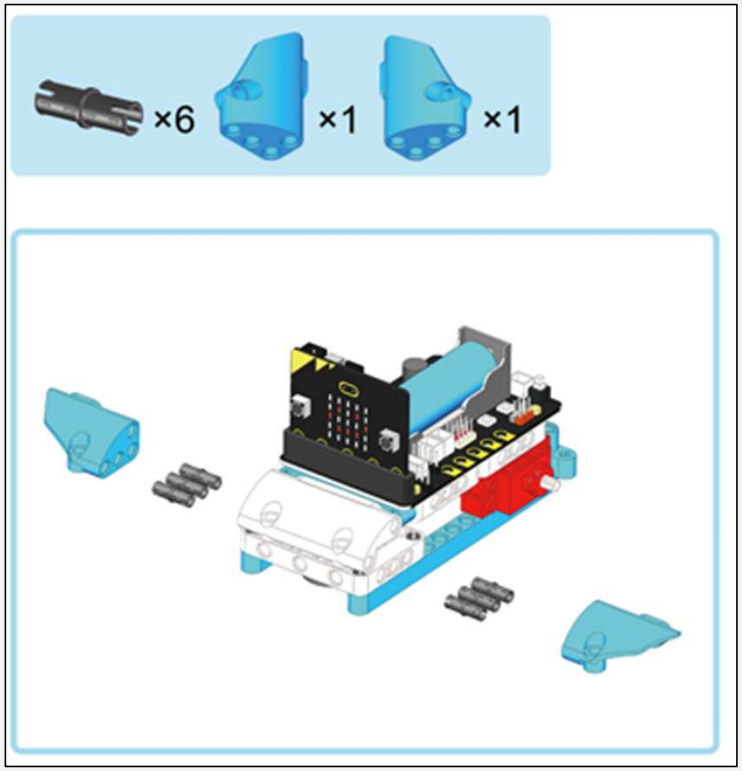
Step 16



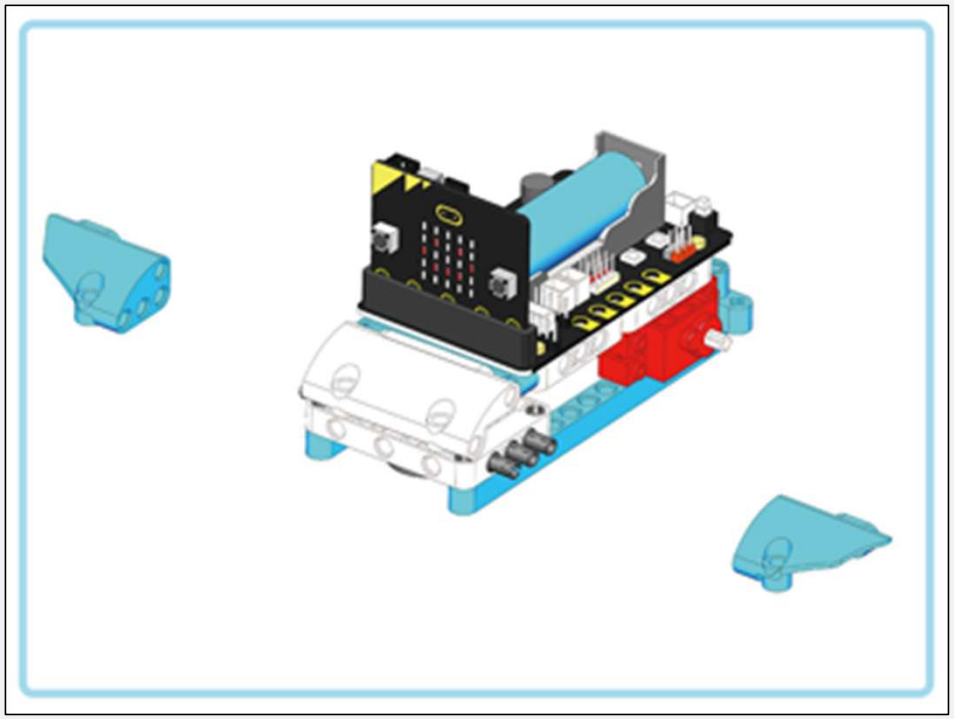
Step 17



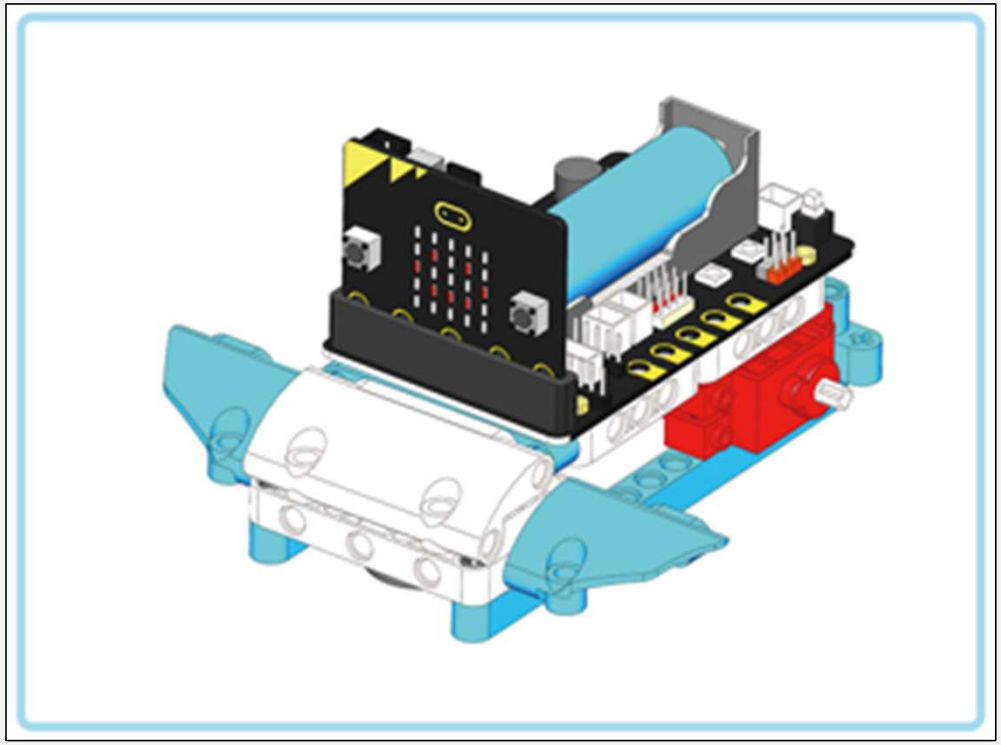
Step 18



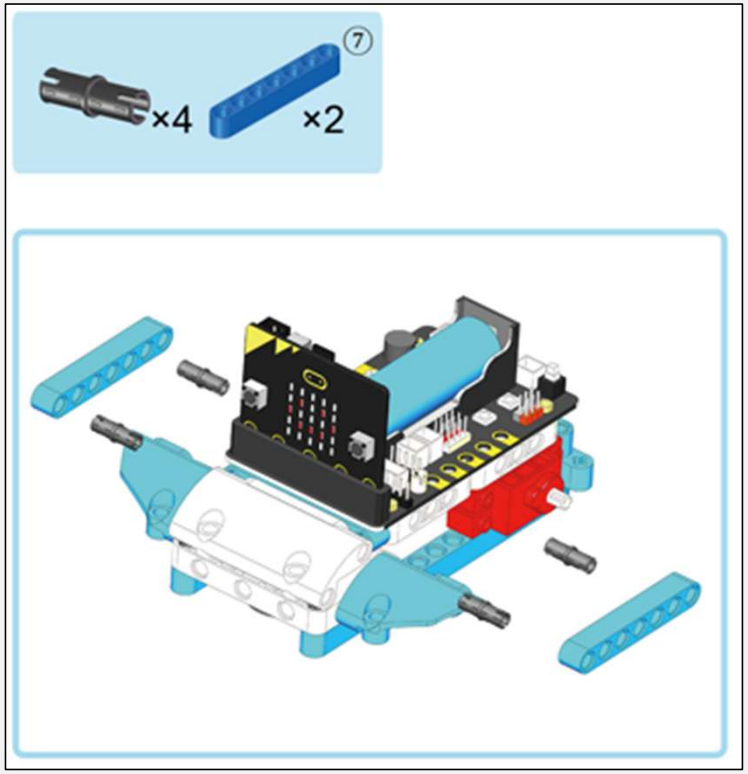
Step 19



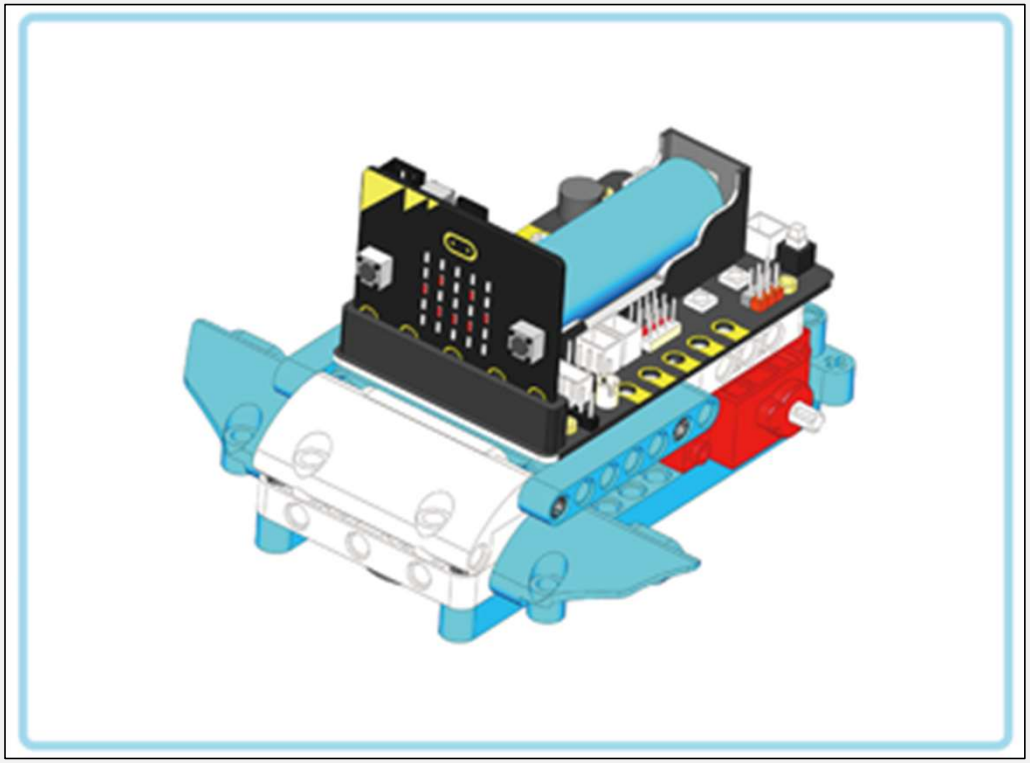
Step 20



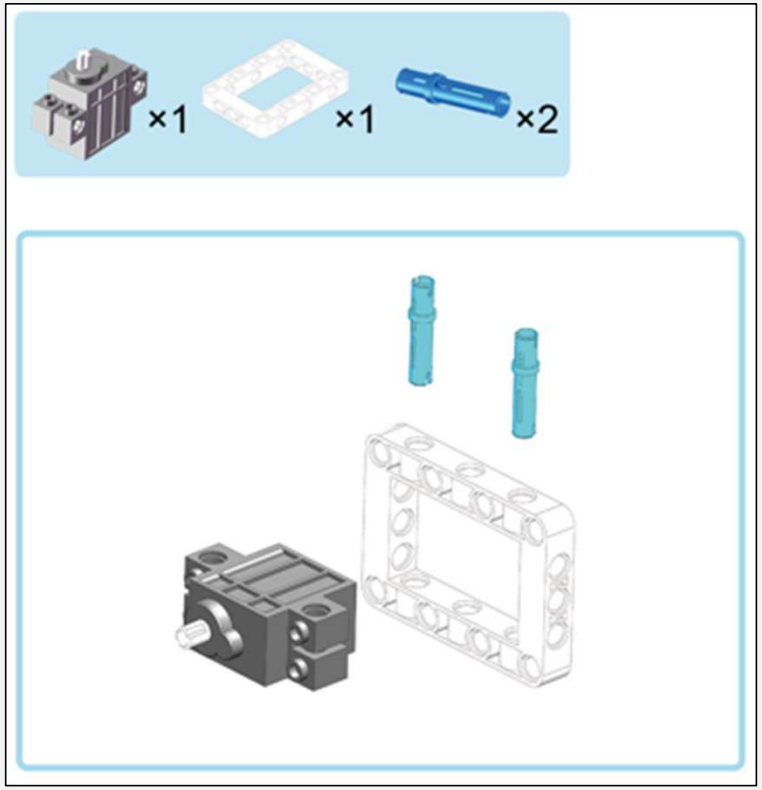
Step 21



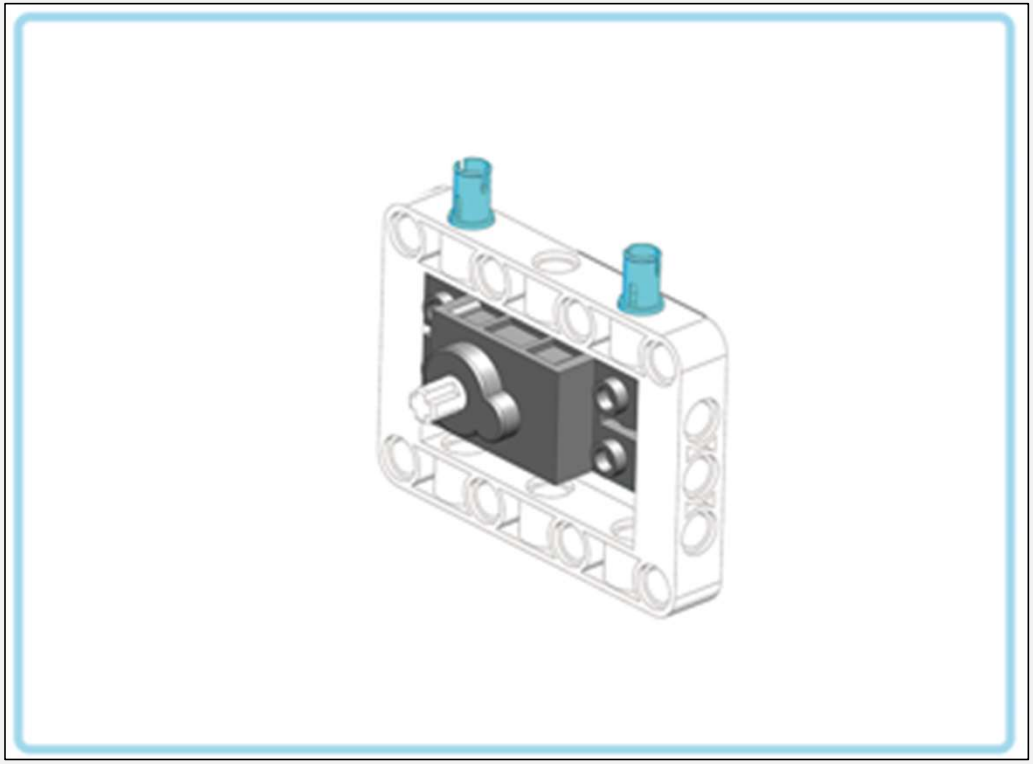
Step 22



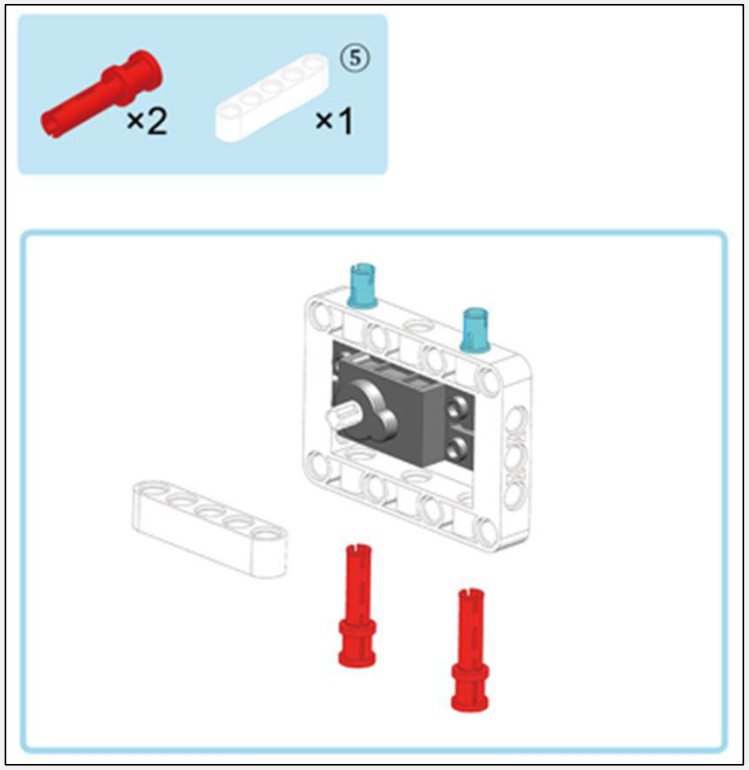
Step 23



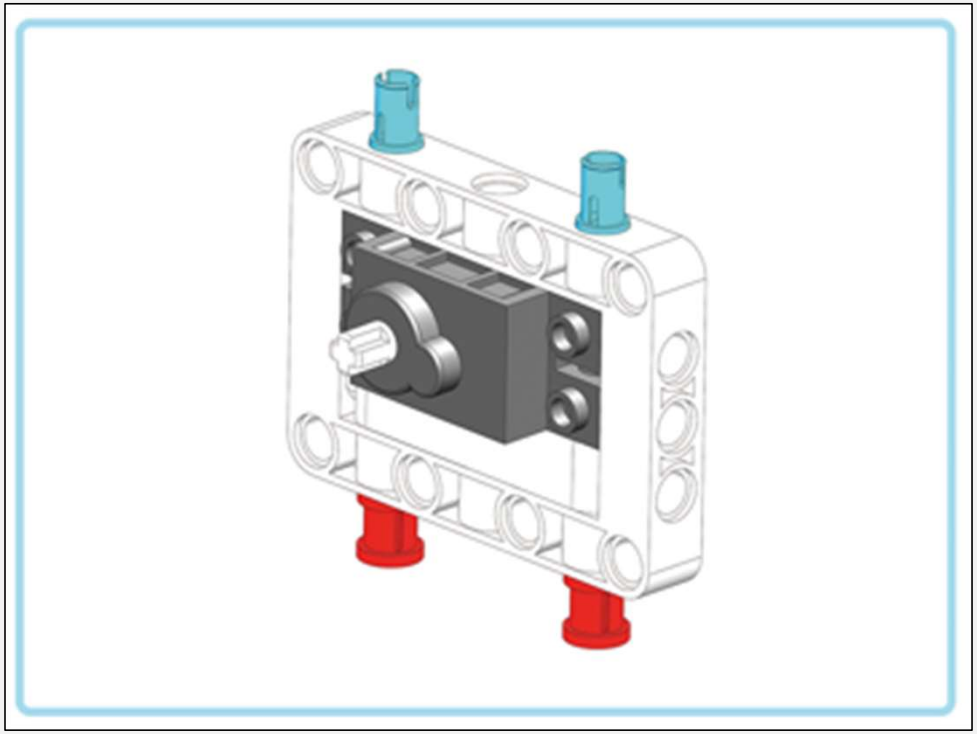
Step 24



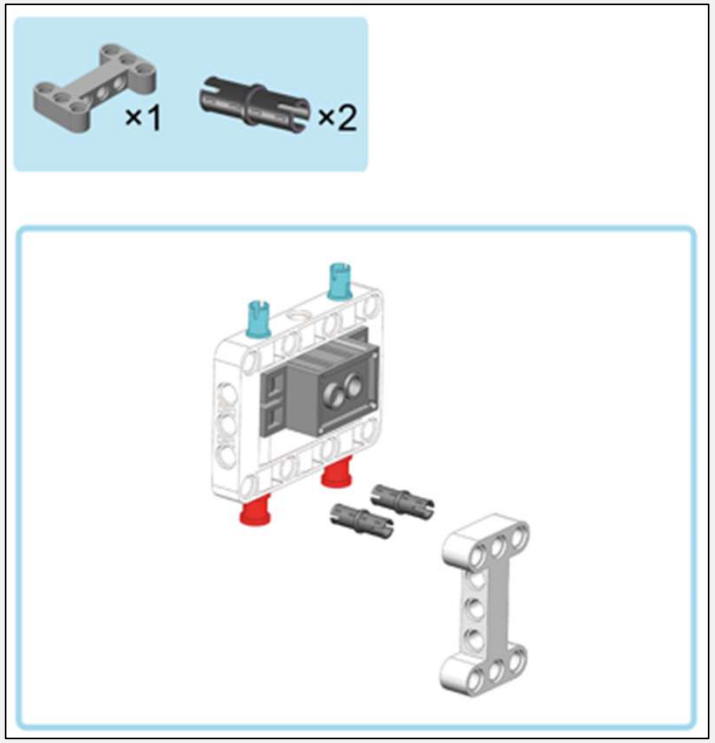
Step 25



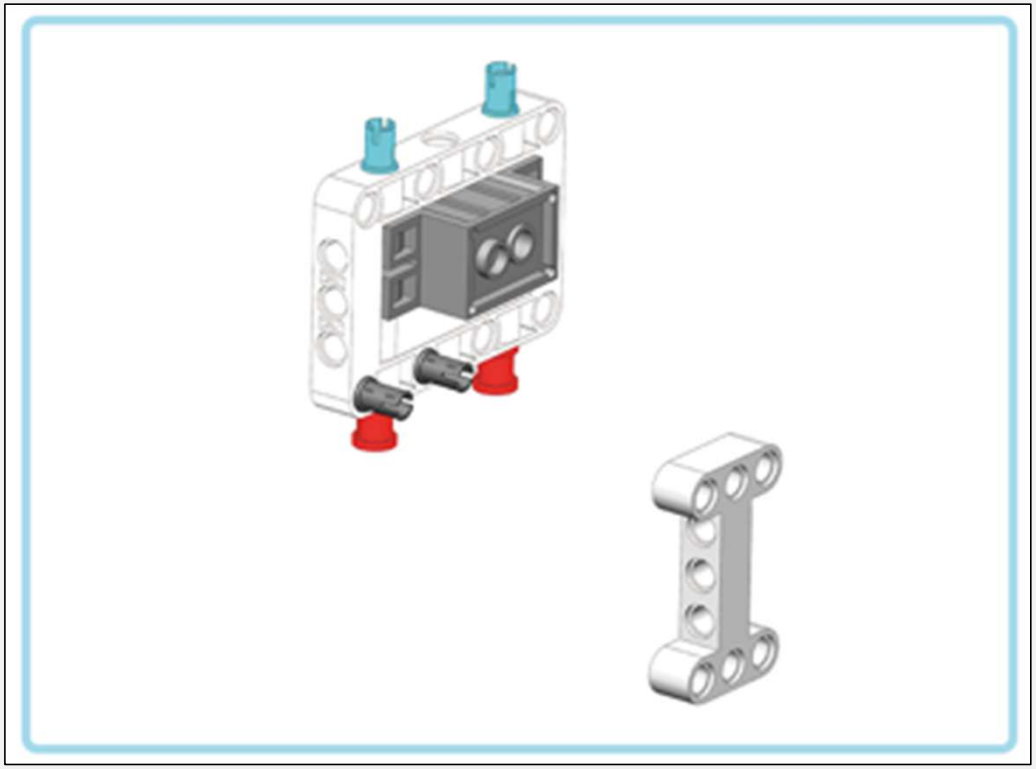
Step 26



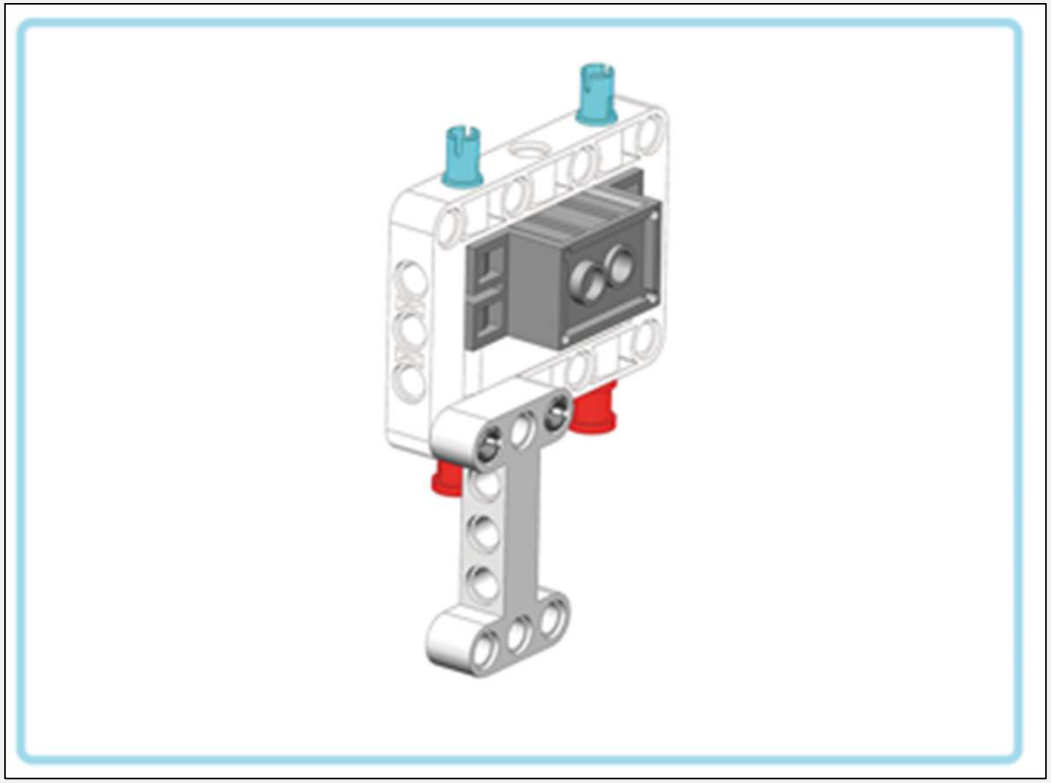
Step 27



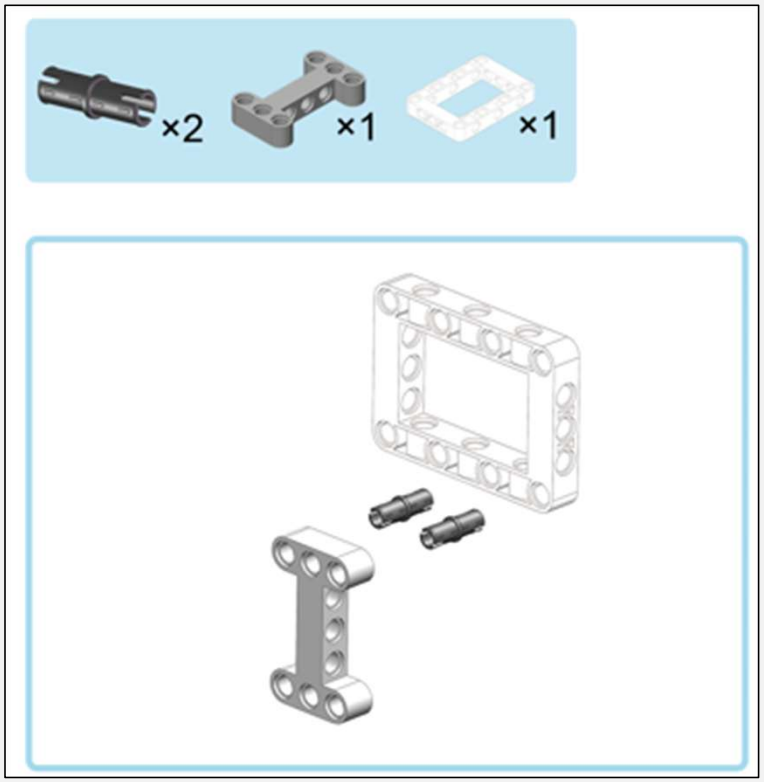
Step 28



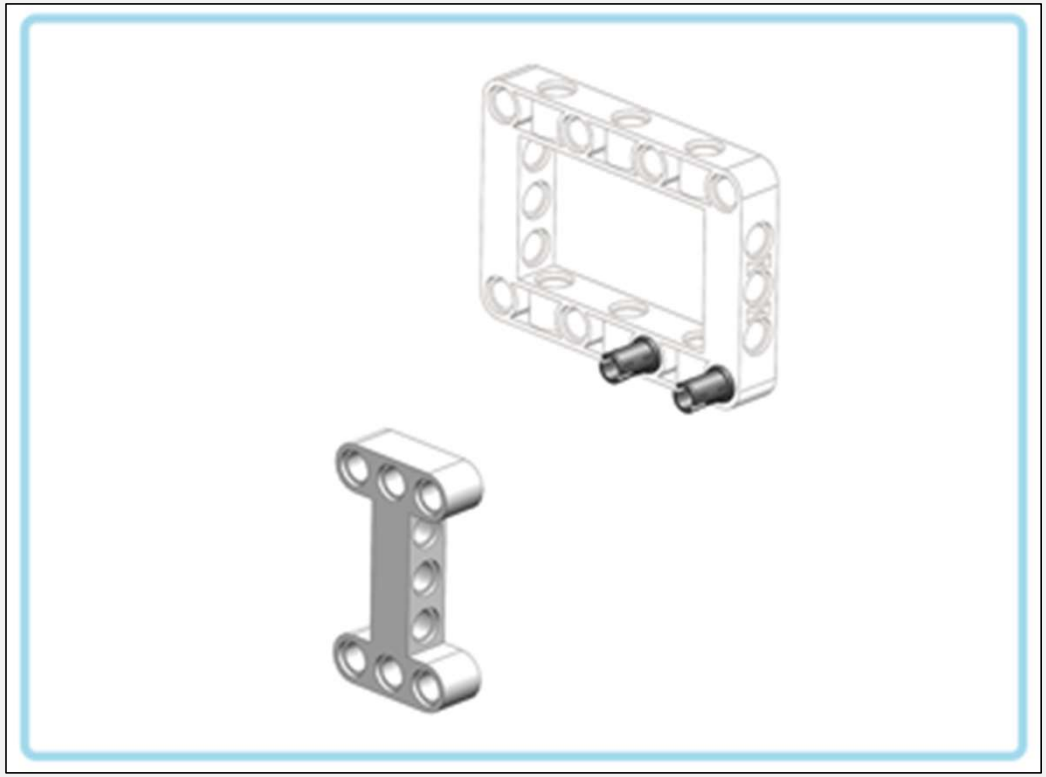
Step 29



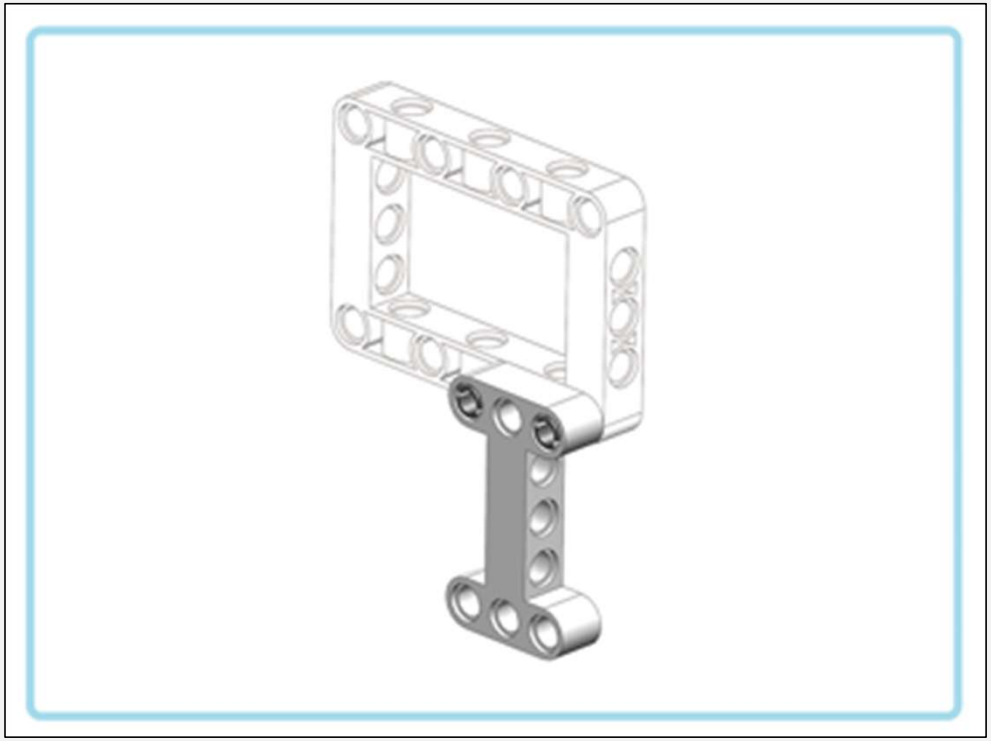
Step 30



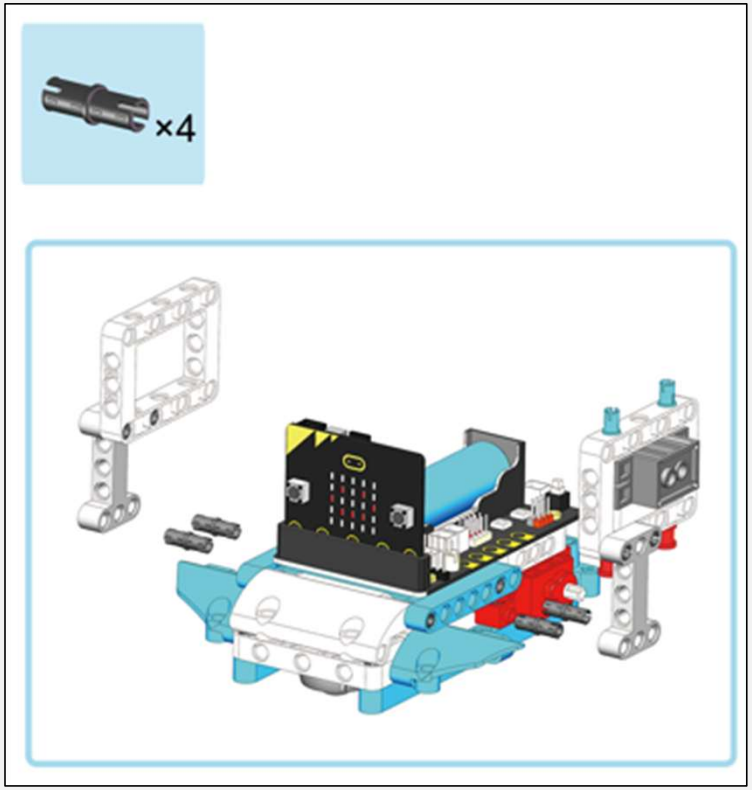
Step 31



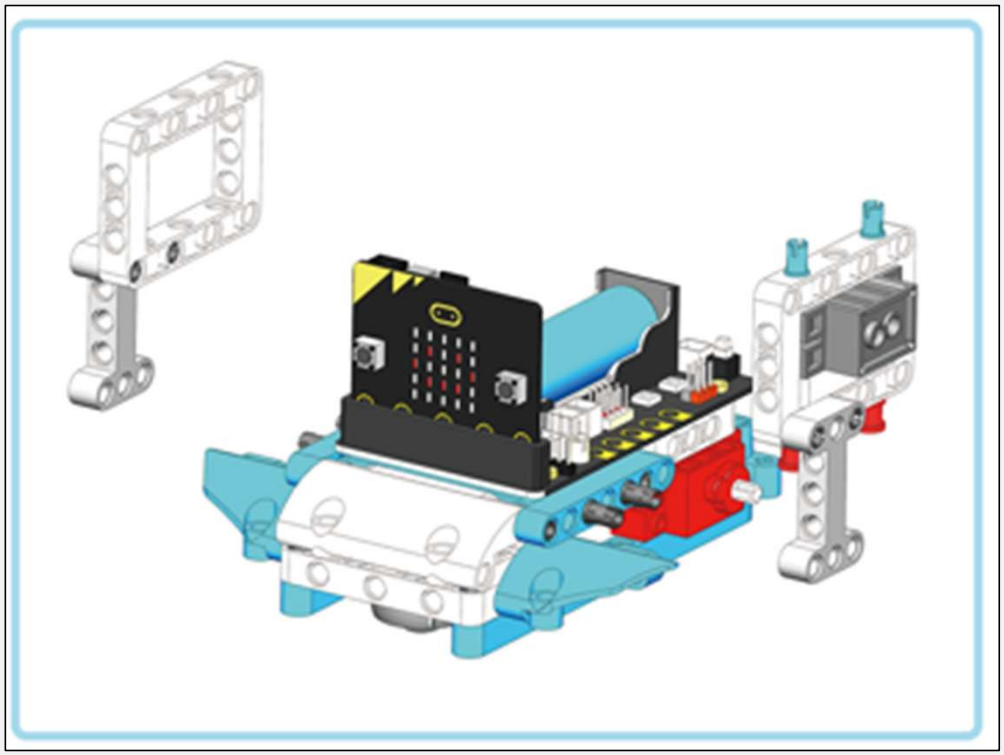
Step 32



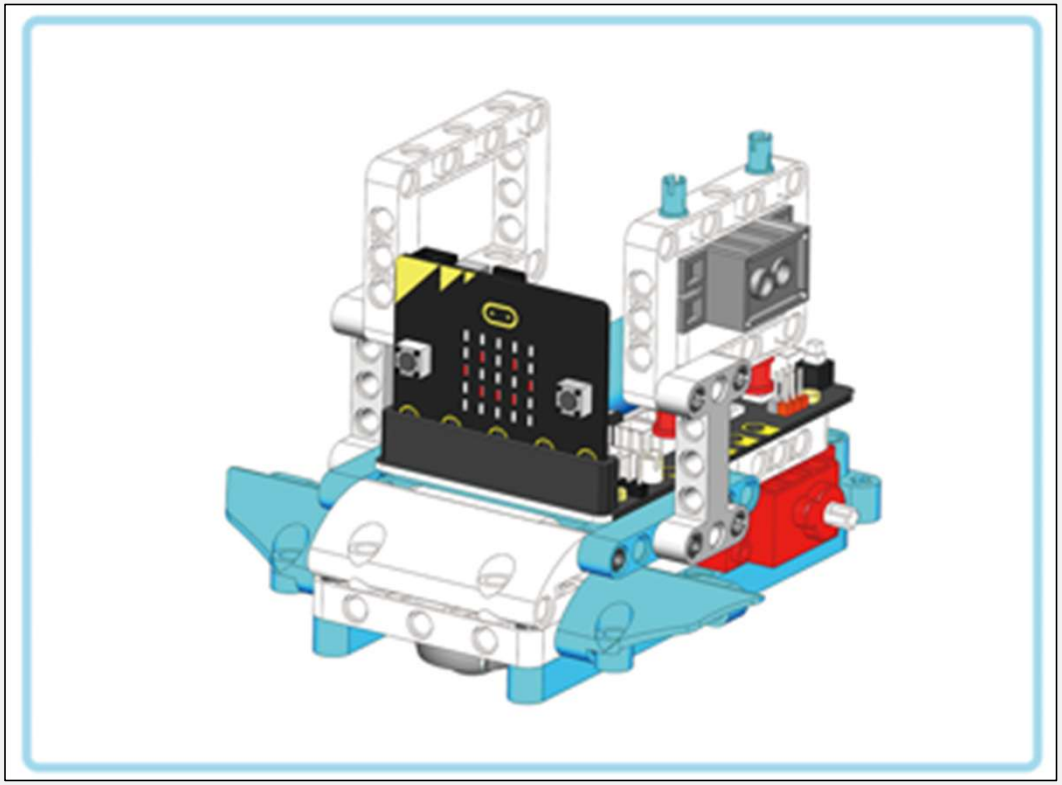
Step 33



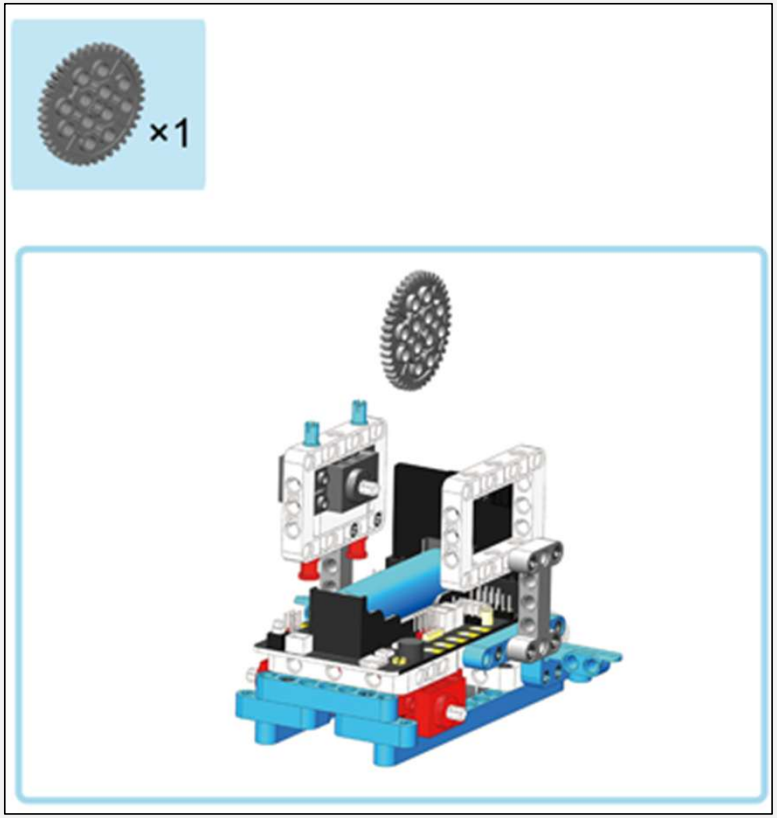
Step 34



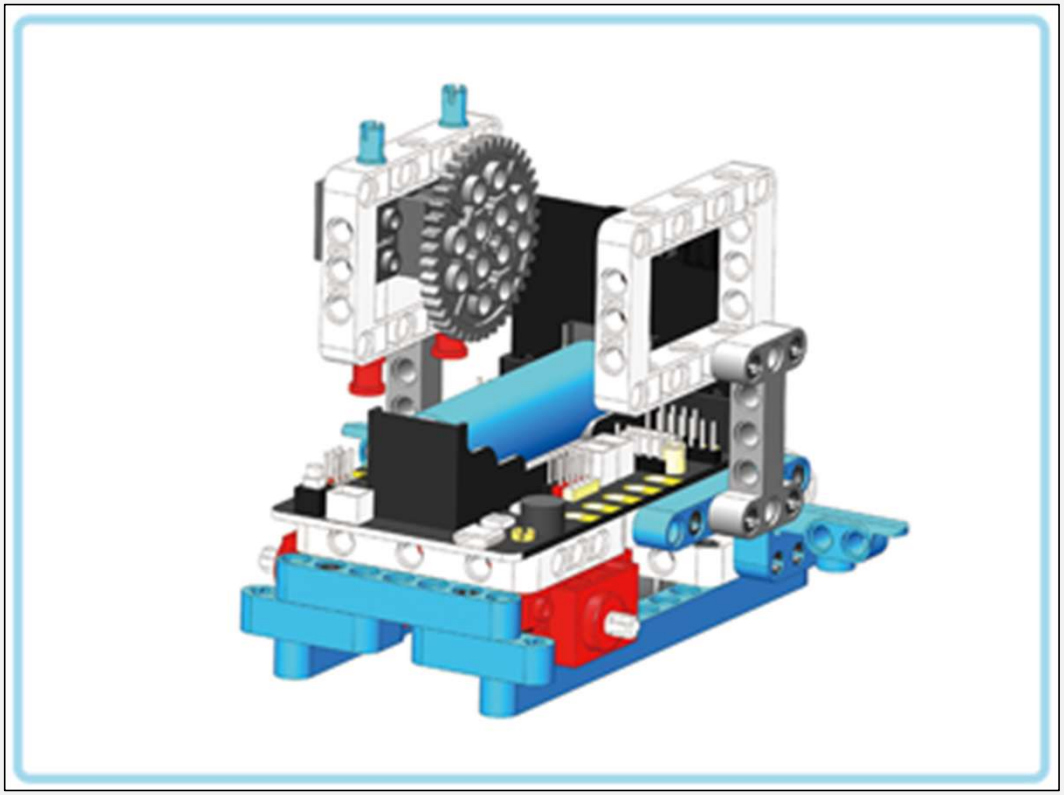
Step 35



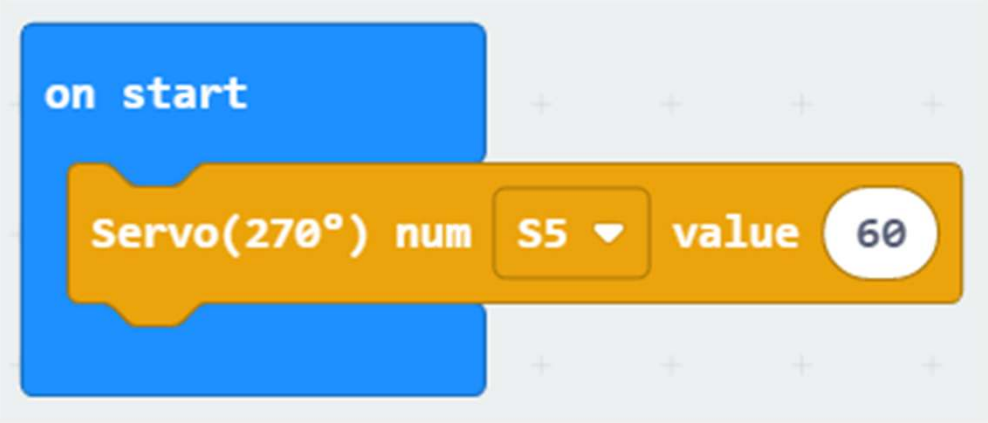
Step 36



Step 37

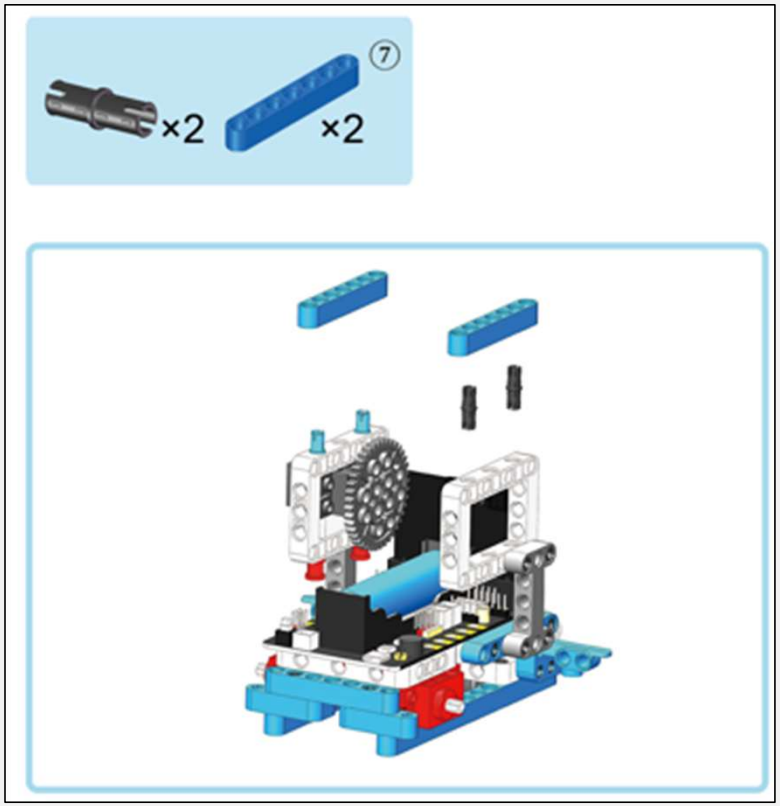


Servo 270° Start position

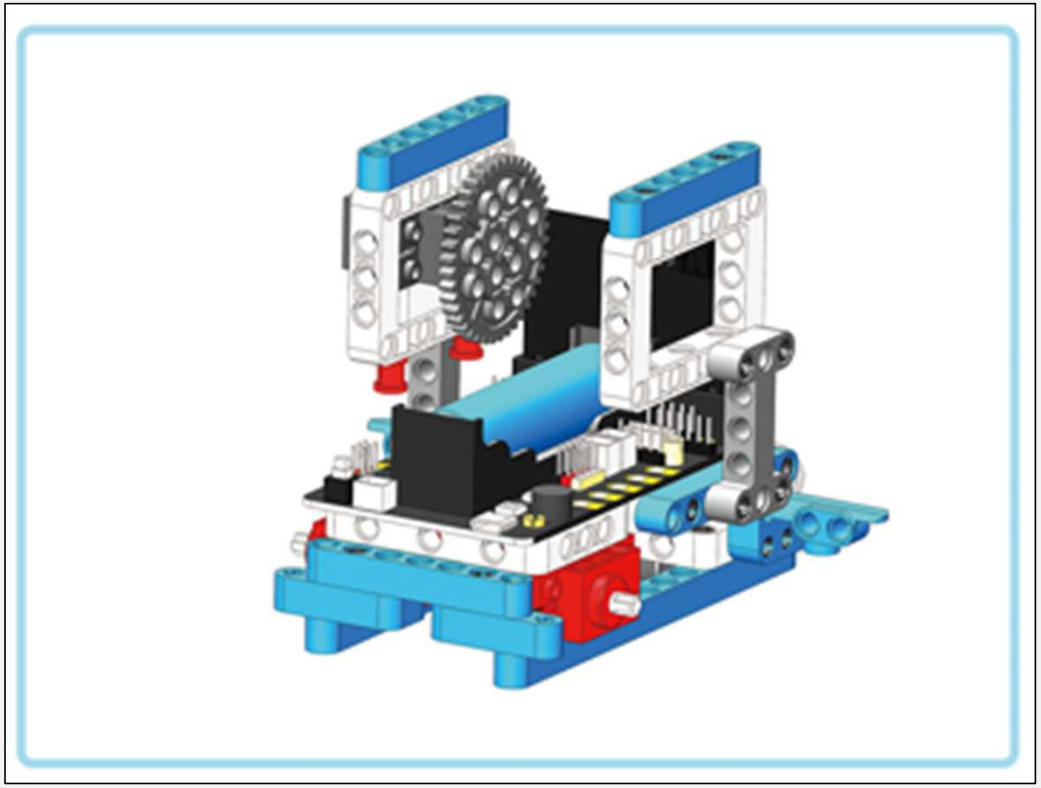


Before we continue, let's set our 270° Servo start at 60°.

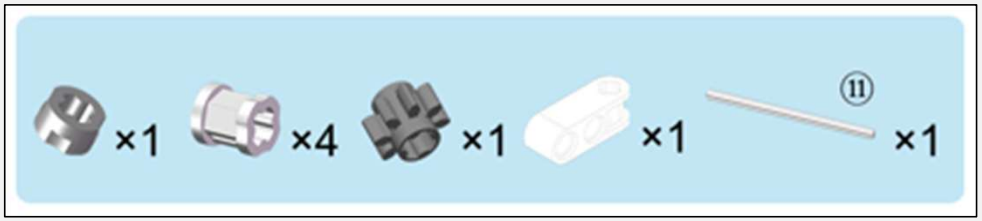
Step 38

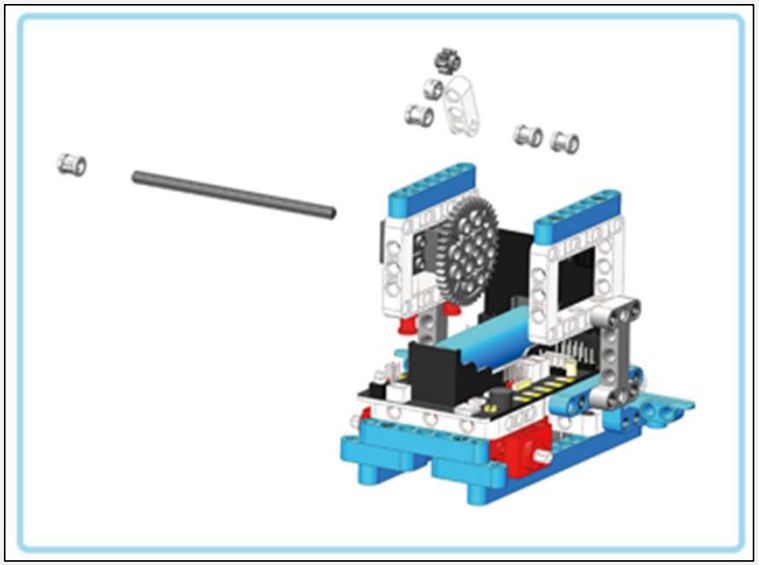


Step 39

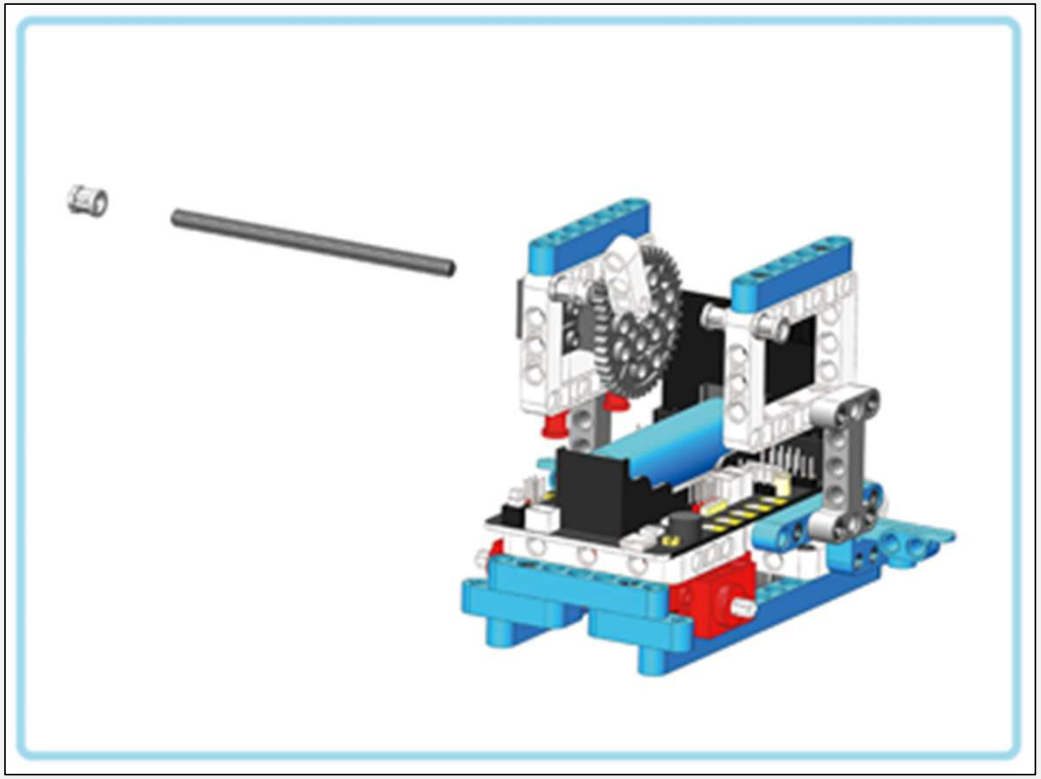


Step 40

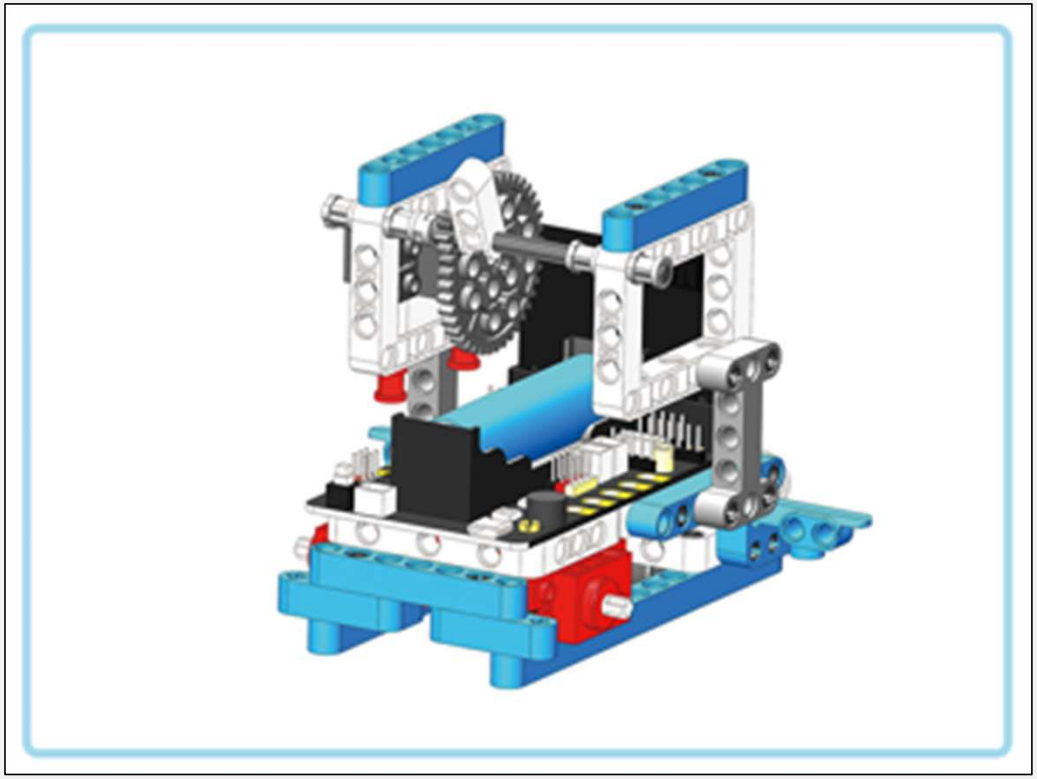




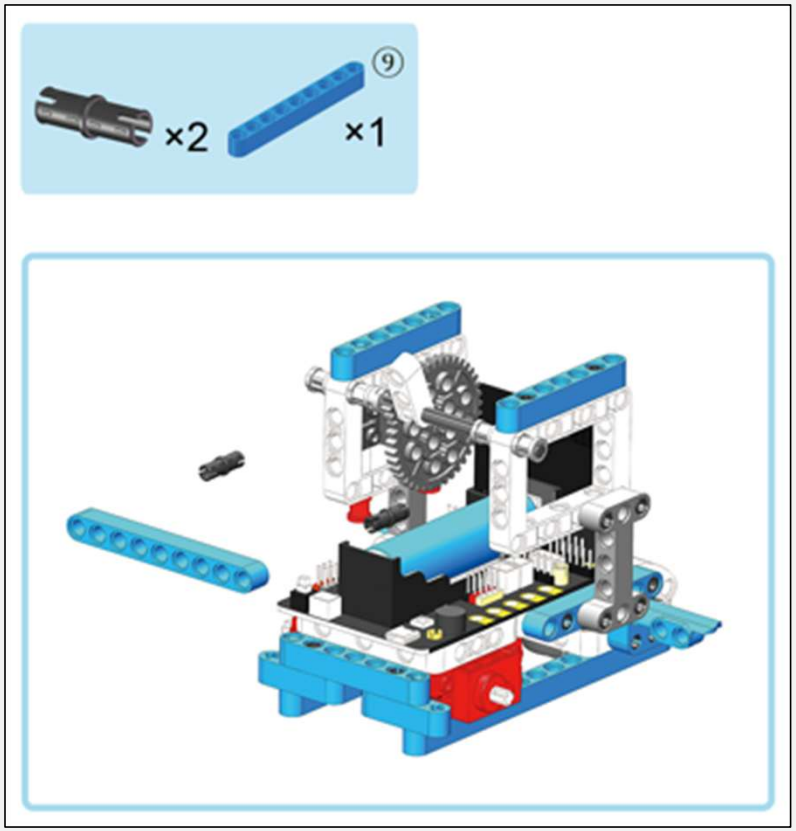
Step 41



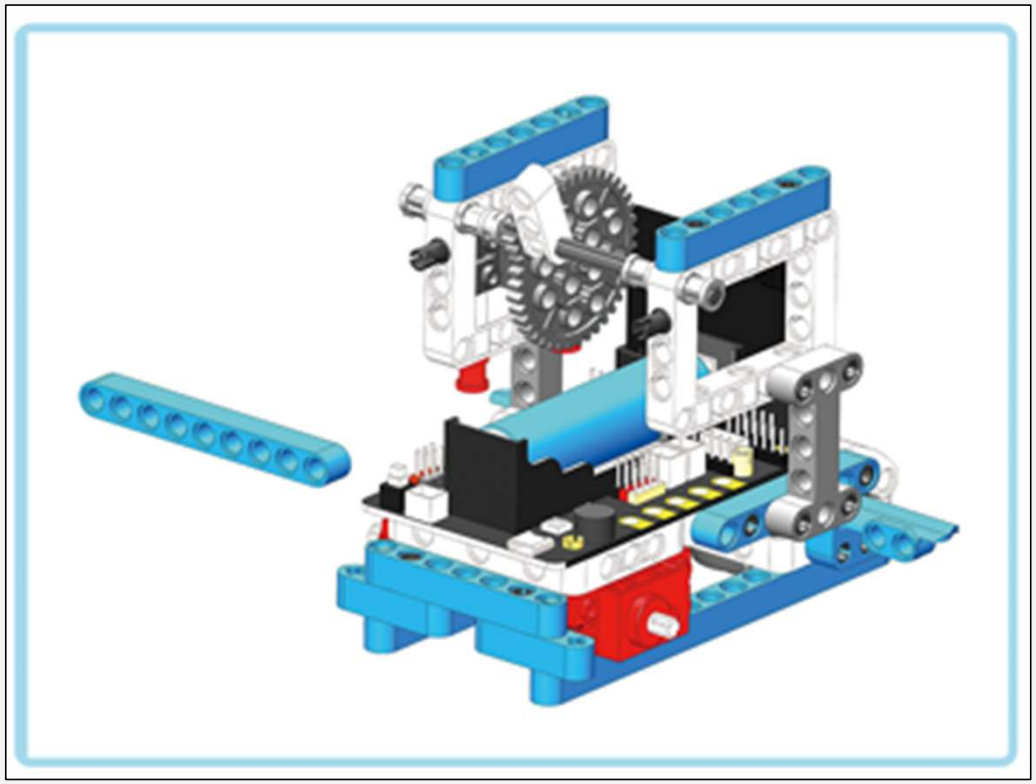
Step 42



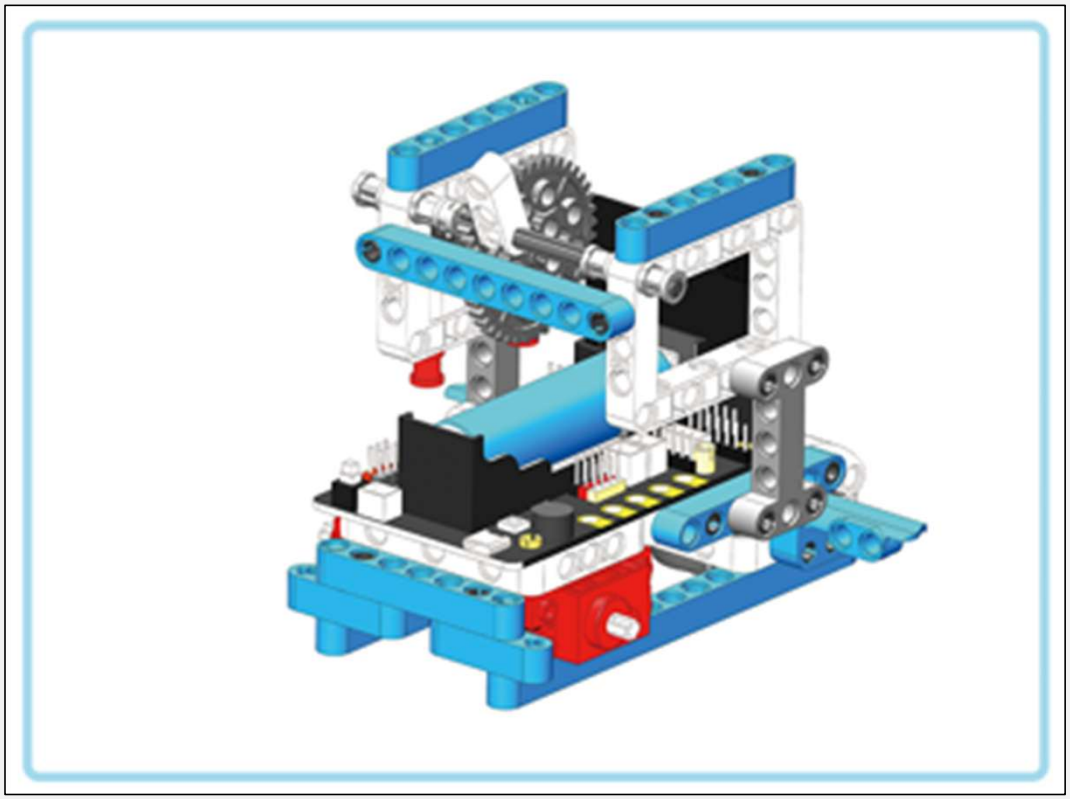
Step 43



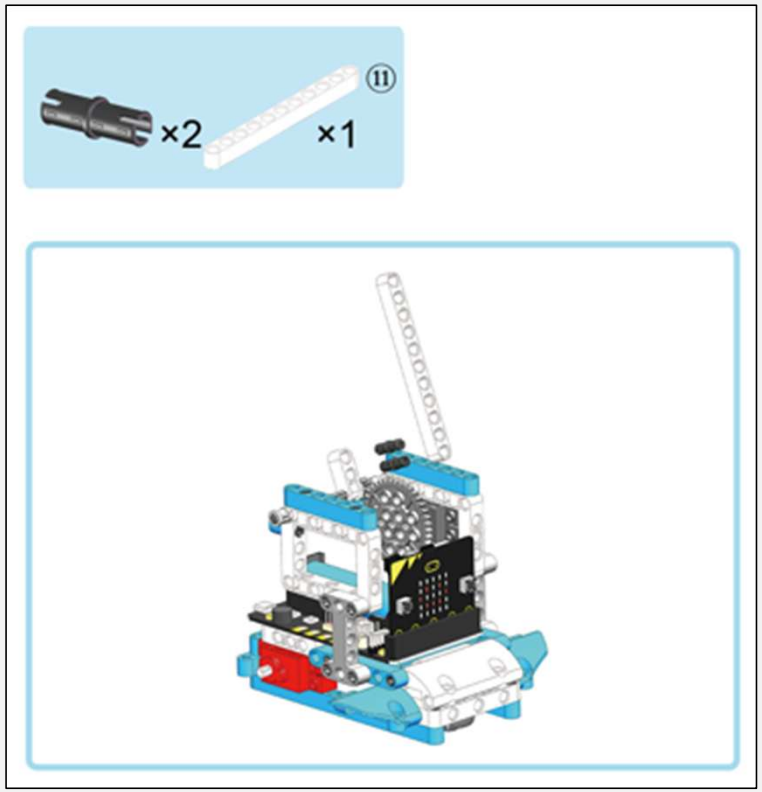
Step 44



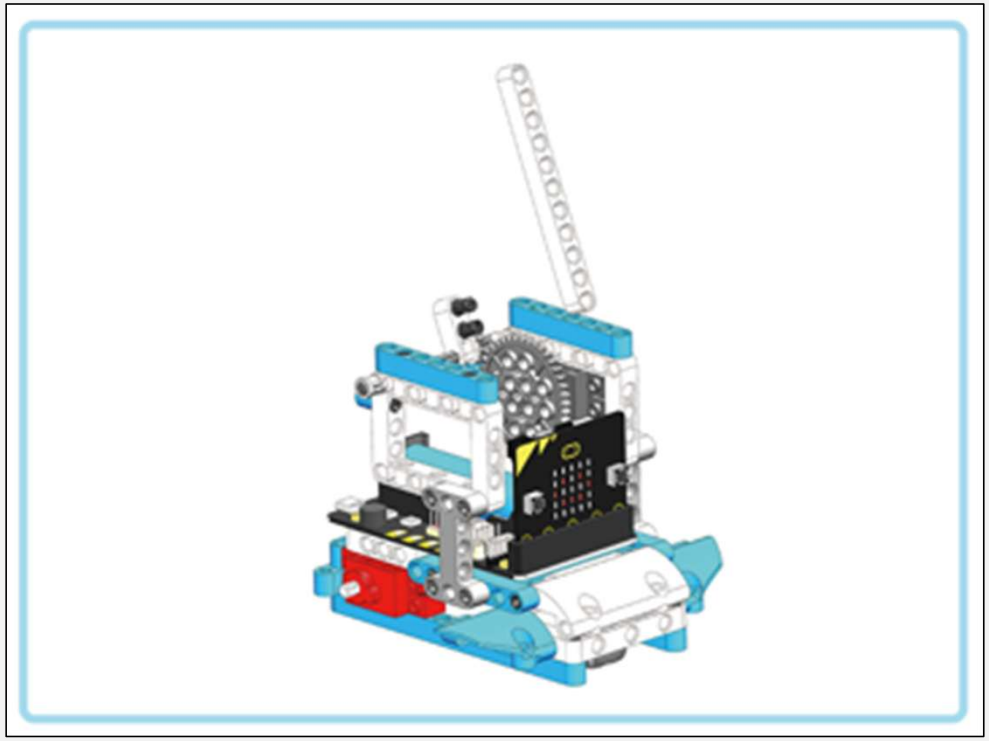
Step 45



Step 46

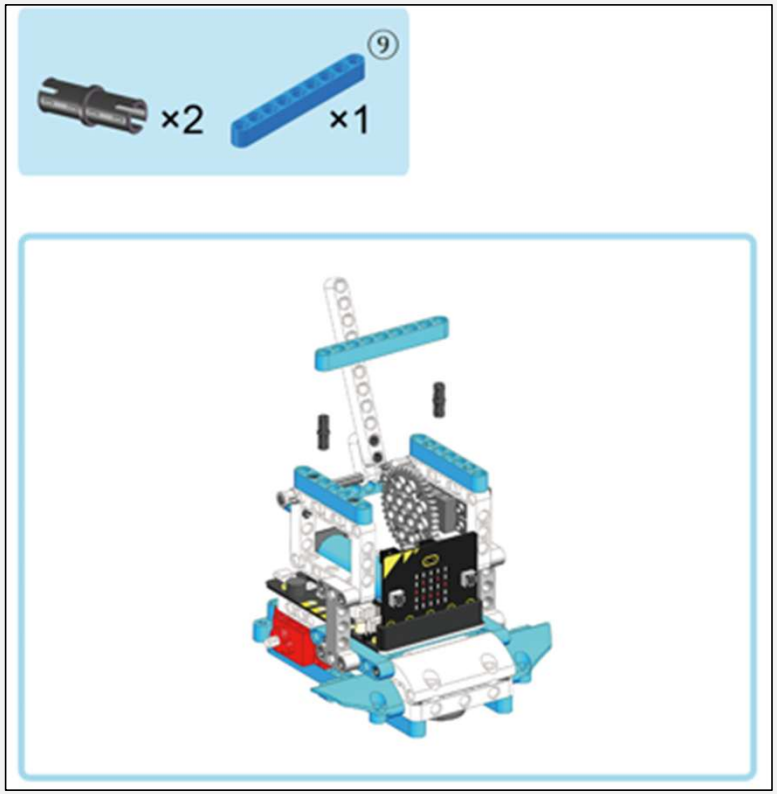


Step 47



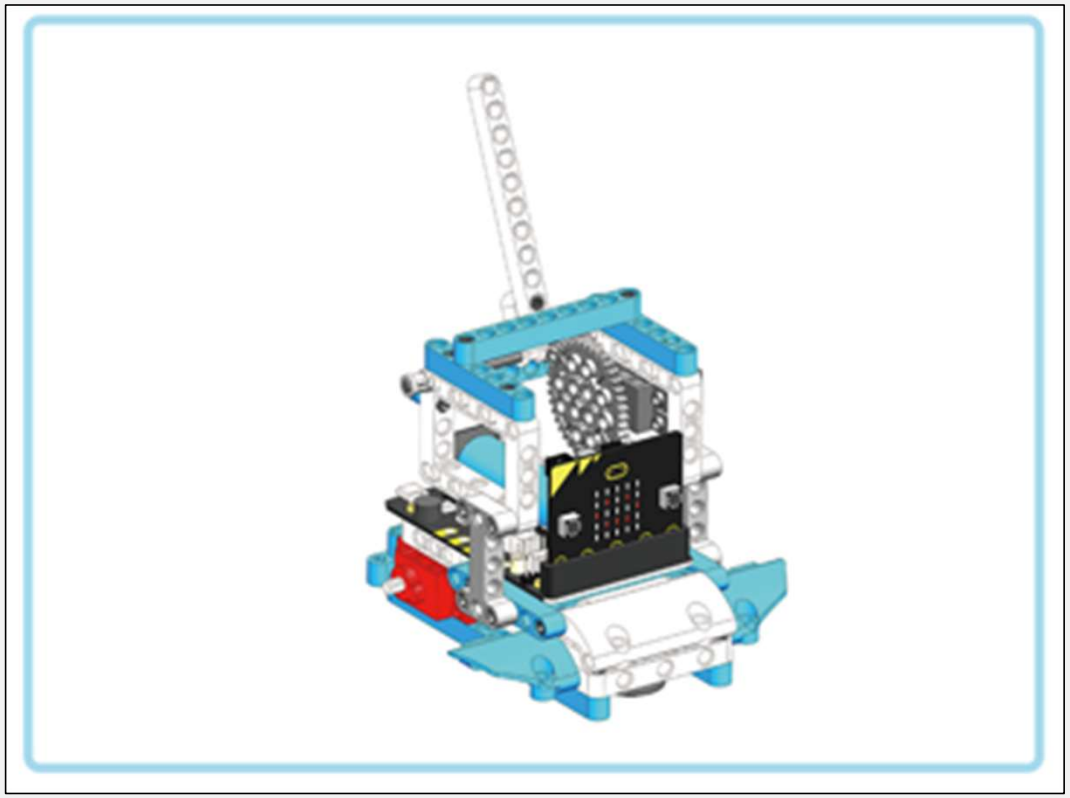
Step 48

Step 49



Step 50

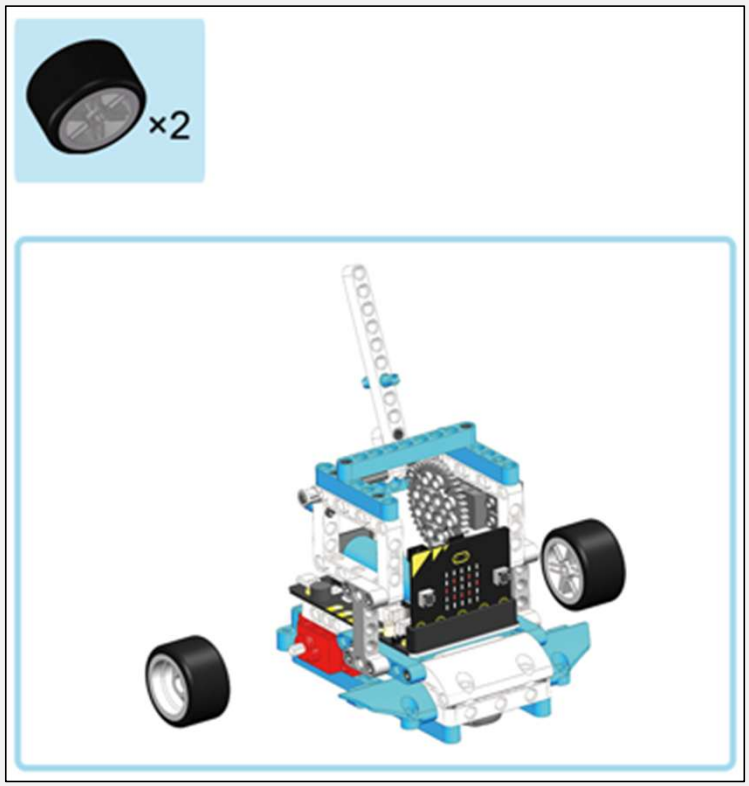
Step 51



Step 52

Step 53

Step 54



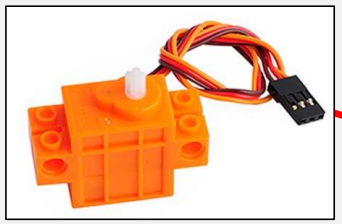
Step 55

Step 56

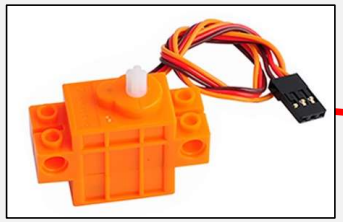
Step 57

Wire Connection

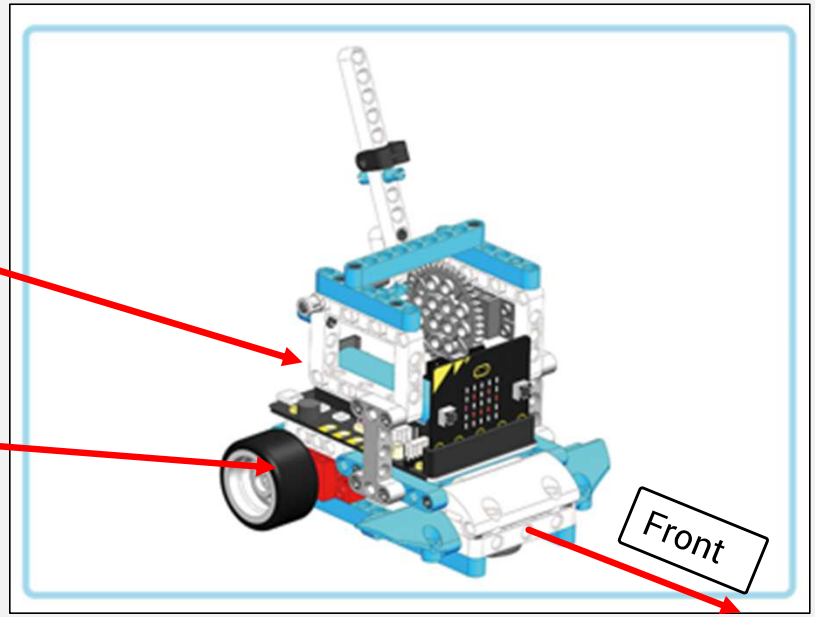
Connect the modules



S2



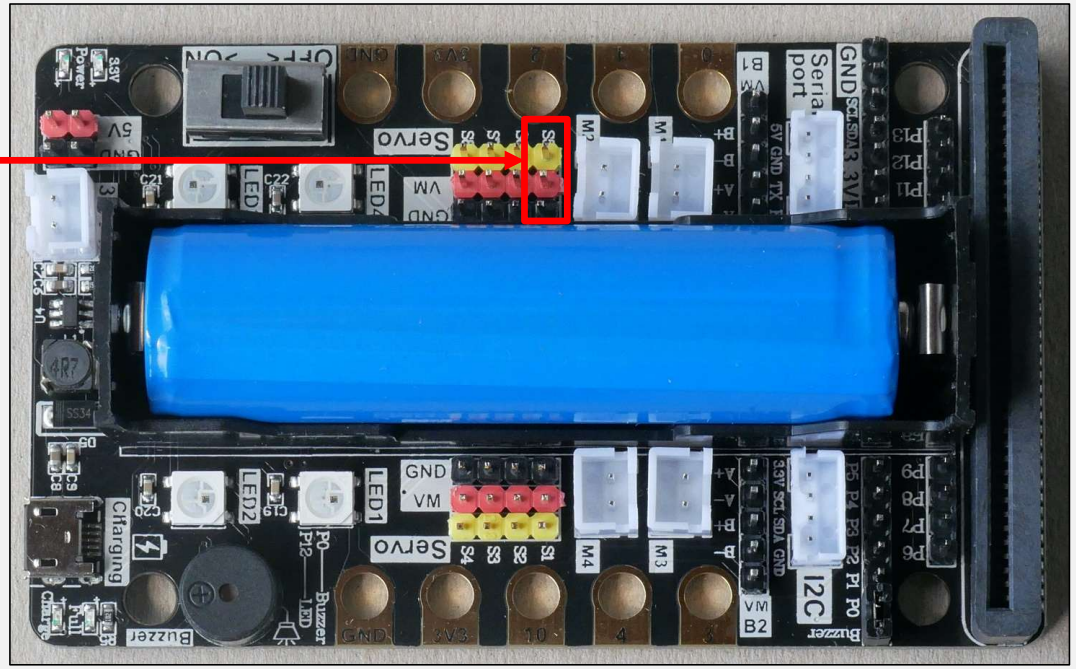
S1



Left motor plug in to **S2**, while **right** motor plug in to **S1**.

Connect the modules

S5



Then plug in a 270° servo (gray) in S5.

MakeCode Programming

First, we need to connect the micro:bit to the computer by USB cable. The computer will pop up a USB flash drive and click on the URL in the USB flash drive: http://microbit.org/ to enter the programming interface.

Add the Yahboom package https://github.com/lzty634158/SuperBit to program.

Or search lzty634158/SuperBit in the Microbit extension.

MakeCode Programming

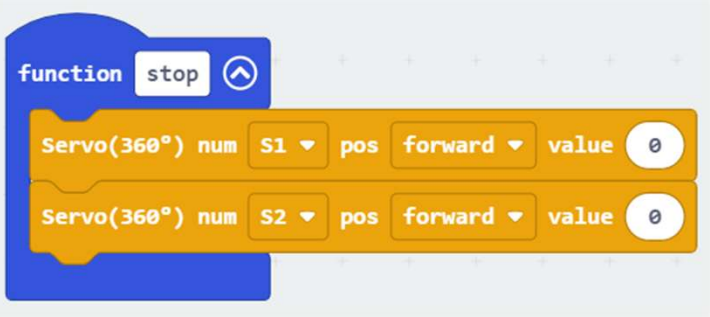
Program our starting state of our servos.

Make a Function - Move Forward

To make our program easy, we better make functions for our move forward, turn right, turn left and move backward as well as shooting.

Let's start with Move Forward.

Make a Function

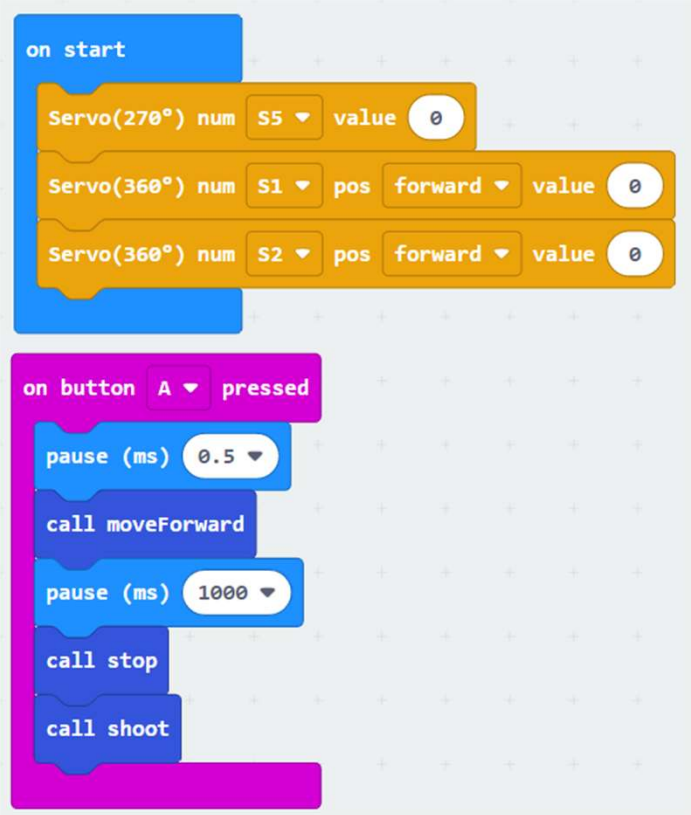


Let's program for the remaining functions.

Make a Function - Shoot

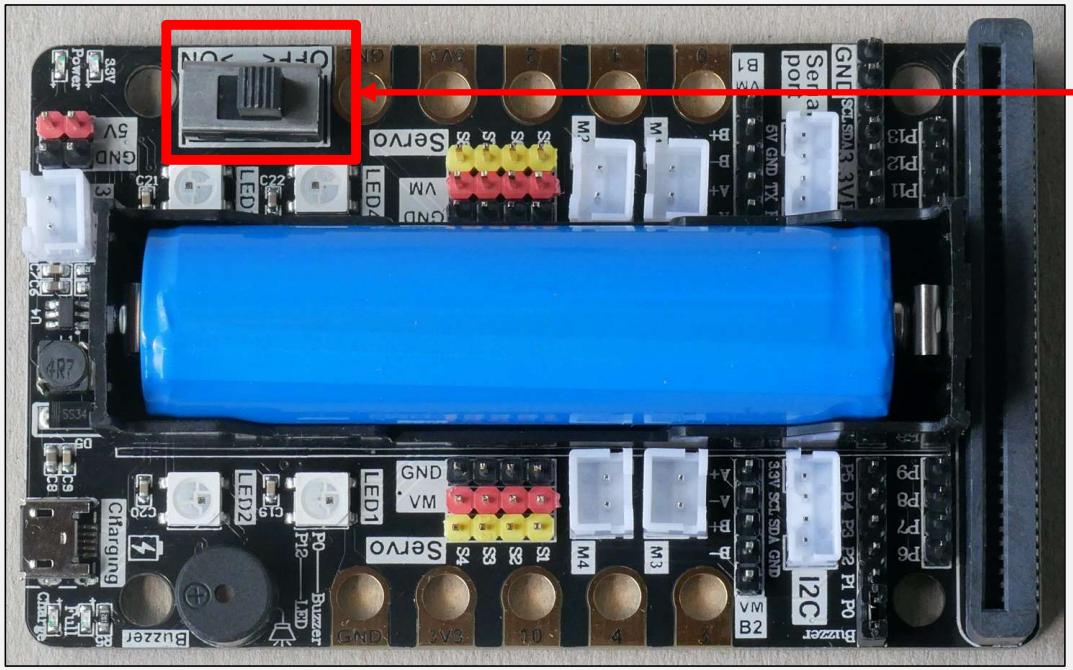
The shoot function is like shooting with a catapult, so we just need to turn our grey servo to 60°, then go back to 0° after 0.5 second.

Shooting time!



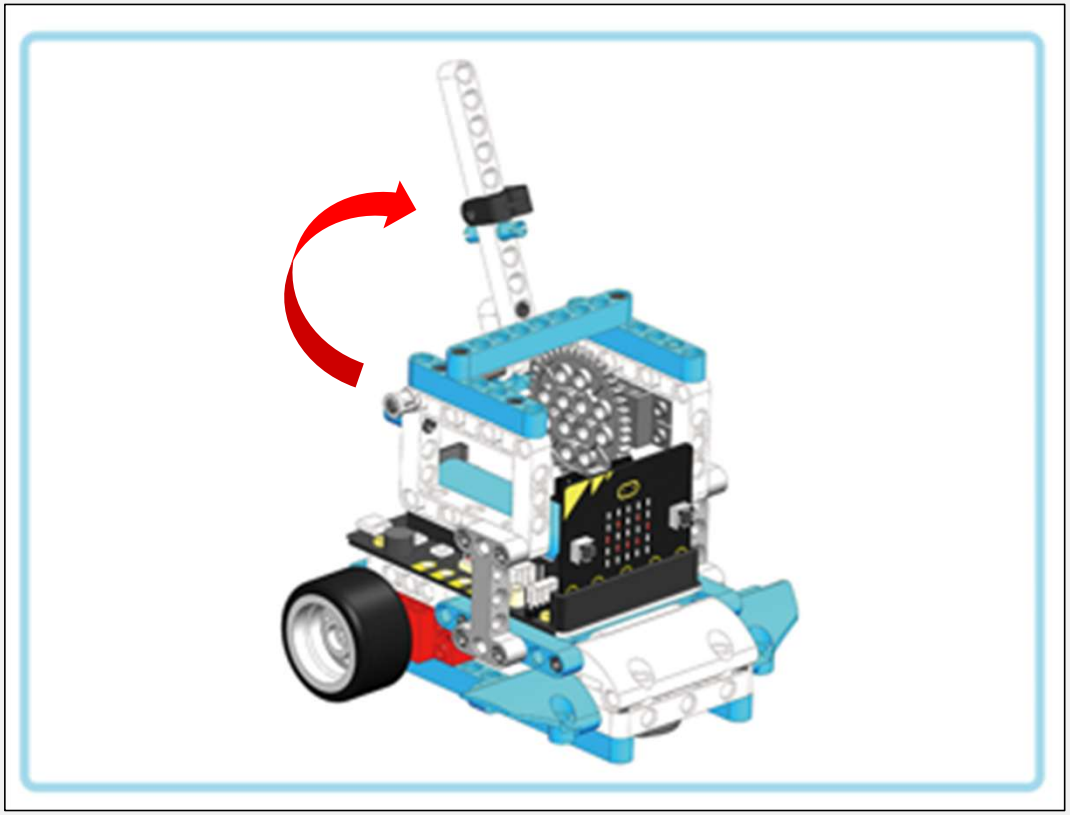
The shoot function is like shooting with a catapult, so we just need to turn our grey servo to 60°, then go back to 0° after 0.5 second.

MakeCode Programming



After the program is downloaded, turn on the SuperBit with the power switch to turn on your mobile shooter.

Phenomenon

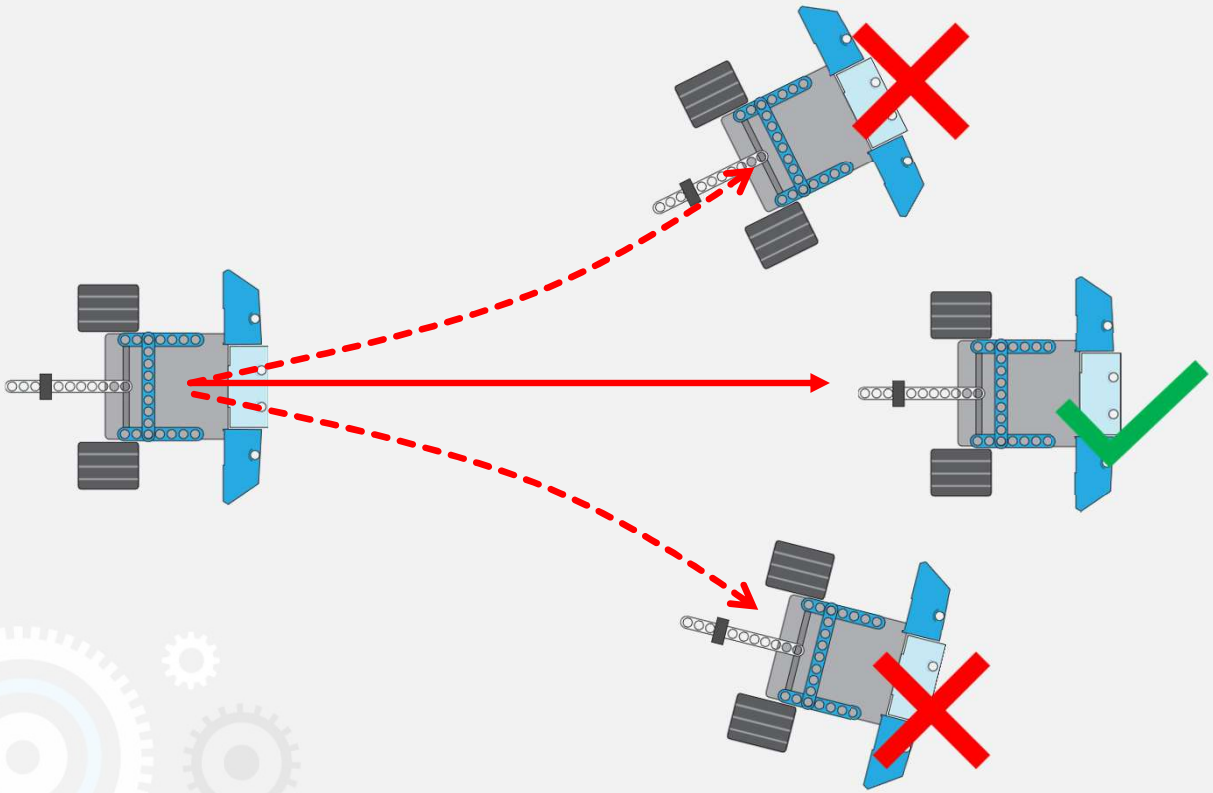


When the superBit is turned on, after button A is pressed, the mobile shooter will move forward for 1 second, then shoot.

30 Points

CHALLENGE
for : Lesson 1

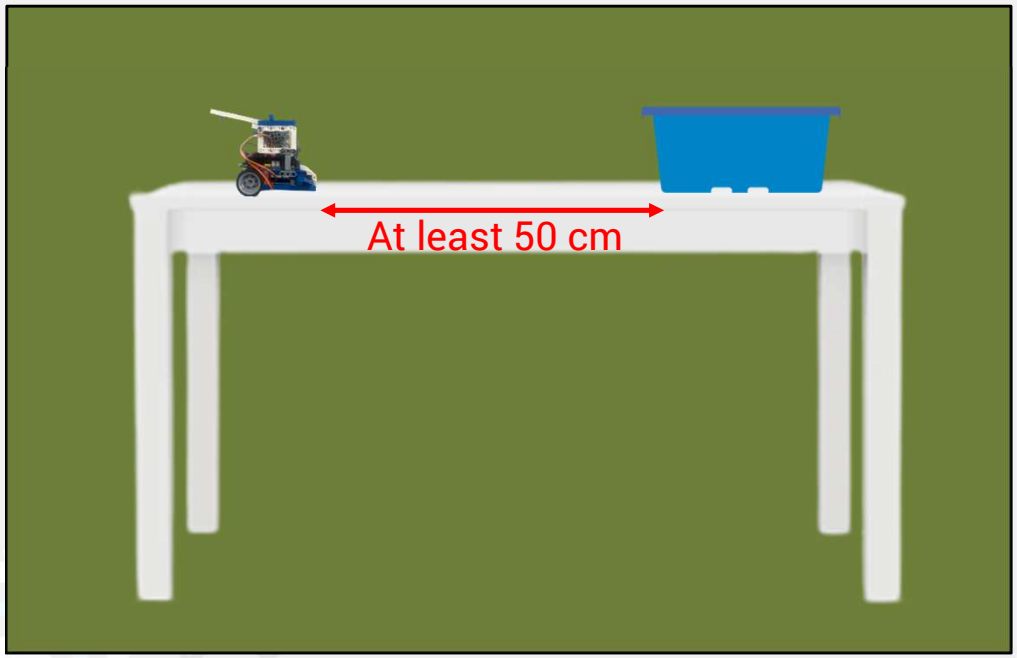
L1 – Challenge 1



When your mobile shooter is running, it might now run straight even the speed values for both servos are same, tune the speed of each servos to make it move forward in a straight line.

30 Points

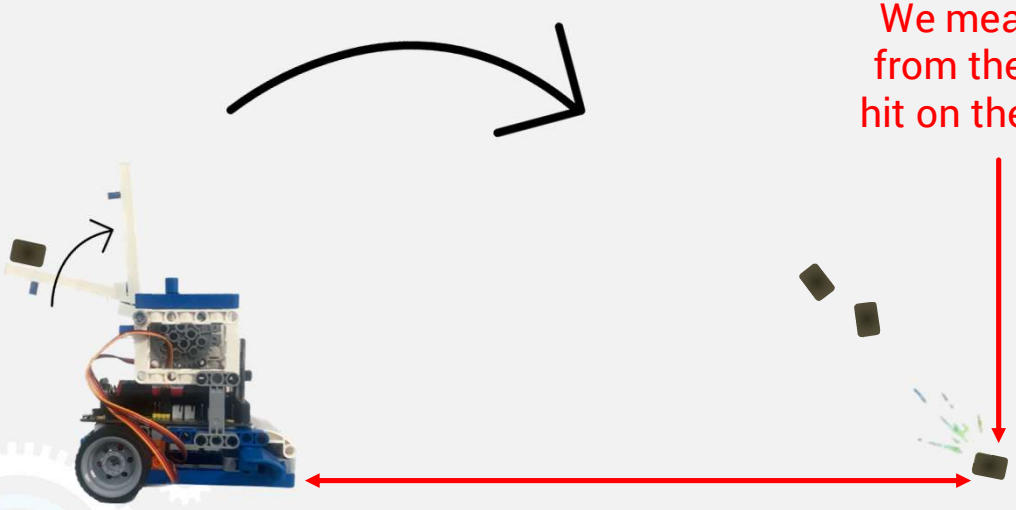
L1 – Challenge 2



Place your mobile shooter **at least 50cm** away from the box on the table and try to shoot the part into the box.

30 Points

L1 – Challenge Activity 3



30 Points

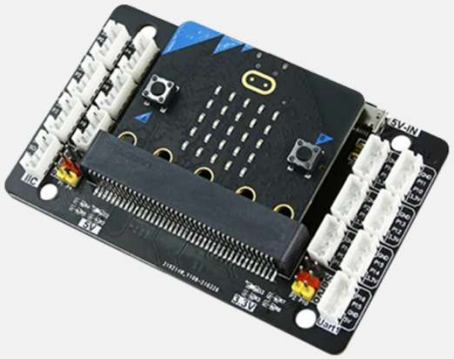
Now you can modify your shooter, but the part to be shot is the same.

Let's see which group can shoot the part with the furthest distance.

The distance is calculated based on the first hit on the floor.

Each group only has **3 shots**.

L1 – Mission



50 Points

Still remember we did the radio programming to send message from 1 microbit to another microbit?

We can use our WOM kits like rocker and button module to program a controller for our mobile shooter, then we can do a remote controller to control our mobile shooter.

Can you make a remote controller for your mobile shooter?