Content

Preface

Mindstorms Robot Inventor Set 51515 continues the successful history of the Mindstorms series. The new generation is compatible with the electronic components of Boost 17101, Spike Prime and Powered-Up.

As the decisive difference to Boost and Powered-Up, Mindstorms is not generally “remote controlled” by a PC or Handeheld but the programs can run independently on the Hub. Unfortunately, the official set only comes with models that are remote controlled. Whereas the Ball-Inventor can be used independently from a PC, as soon as the program is uploaded to the hub.

The Document has three segments:

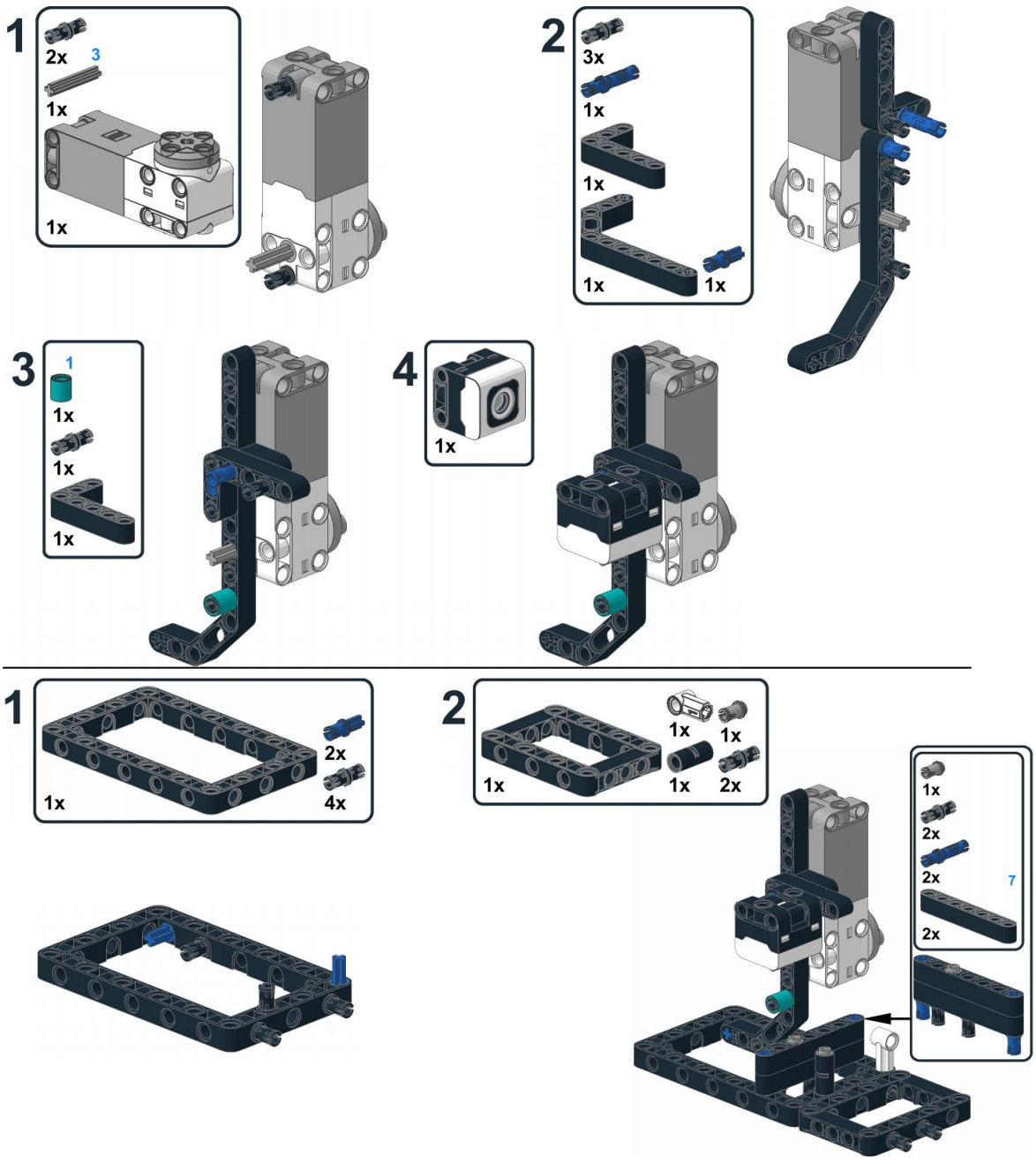
- BUILD – Building instructions
- CODE – Programming the Hub
- Play - Operate and customize

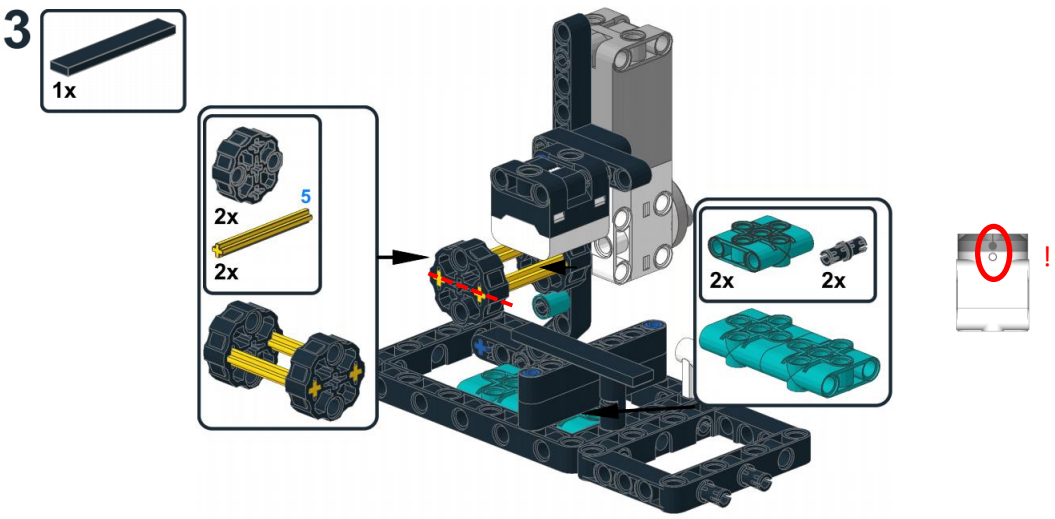
Ball-Inventor – the marble-run

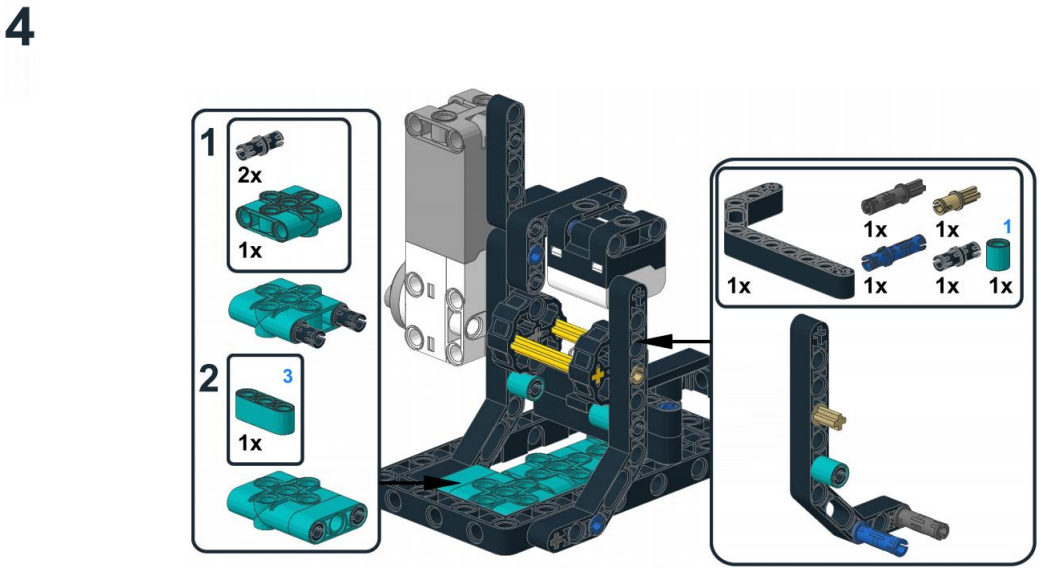
The Ball-Inventor marble run can teach us some basic automation techniques. I addition, you can paly with it, of course. The ramp which the marble rolls down can be customized to your personal liking. If other LEGO technic parts are available, you may even want to extend the ramp by additional elements. You may want to build some courses, so a marble can take different routes on its way downward.

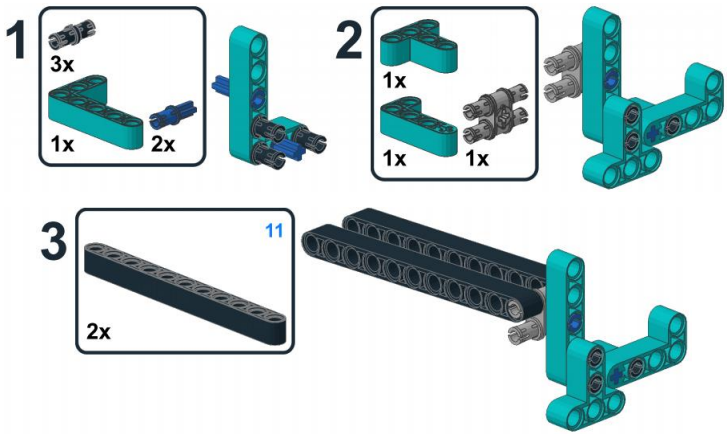
Building instructions

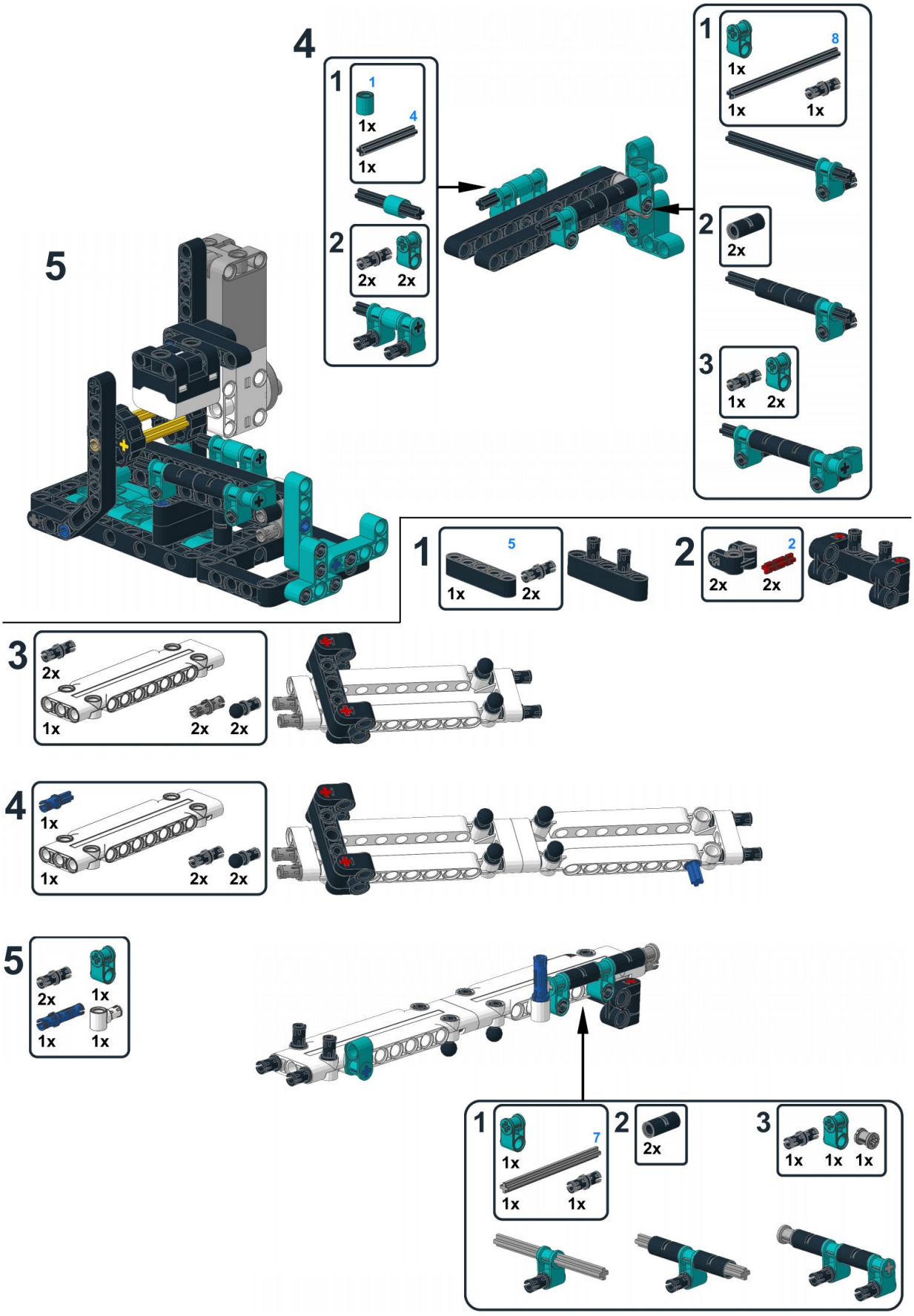
Ball-Inventor can be built with just the parts in the 51515 Mindstorms Inventor set. You do not need any additional parts besides a PC, tablet, or phone to upload the program.

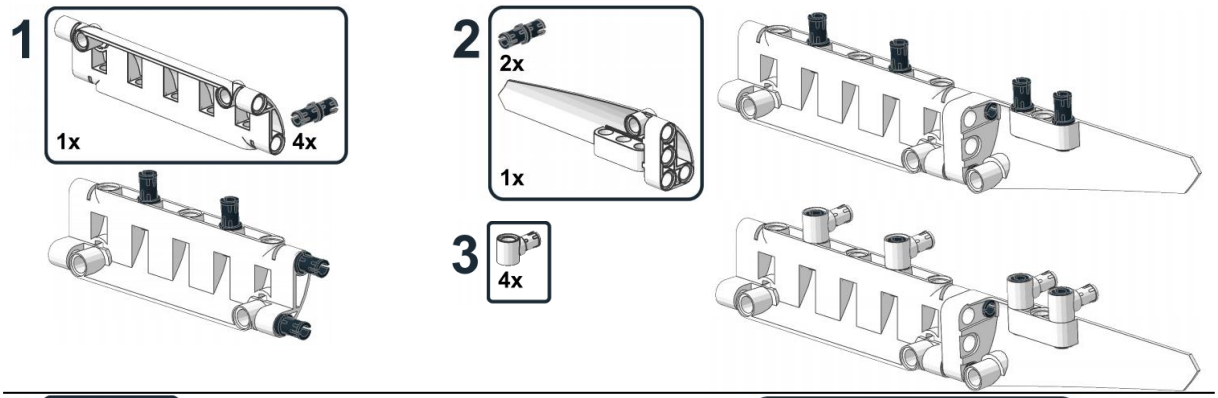


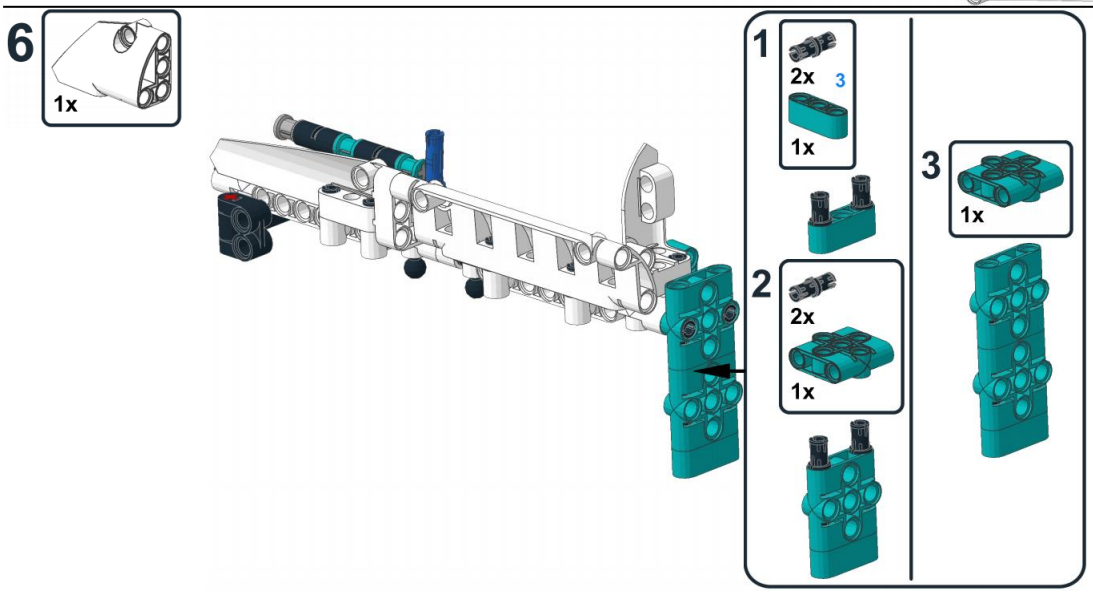

3
1x
2x
5
2x
2x
2x
!

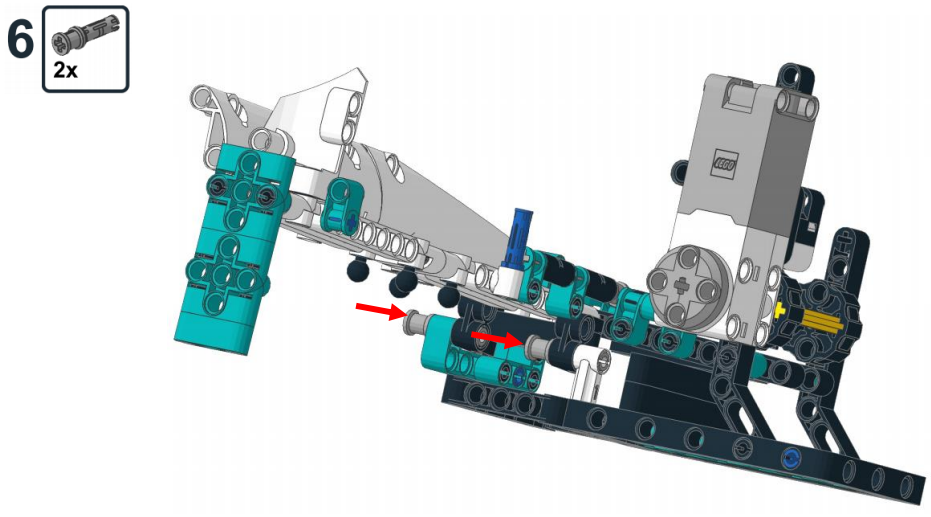

4
1
2x
1x
2
3
1x
1x
1x
1x
1x
1x
1x
1

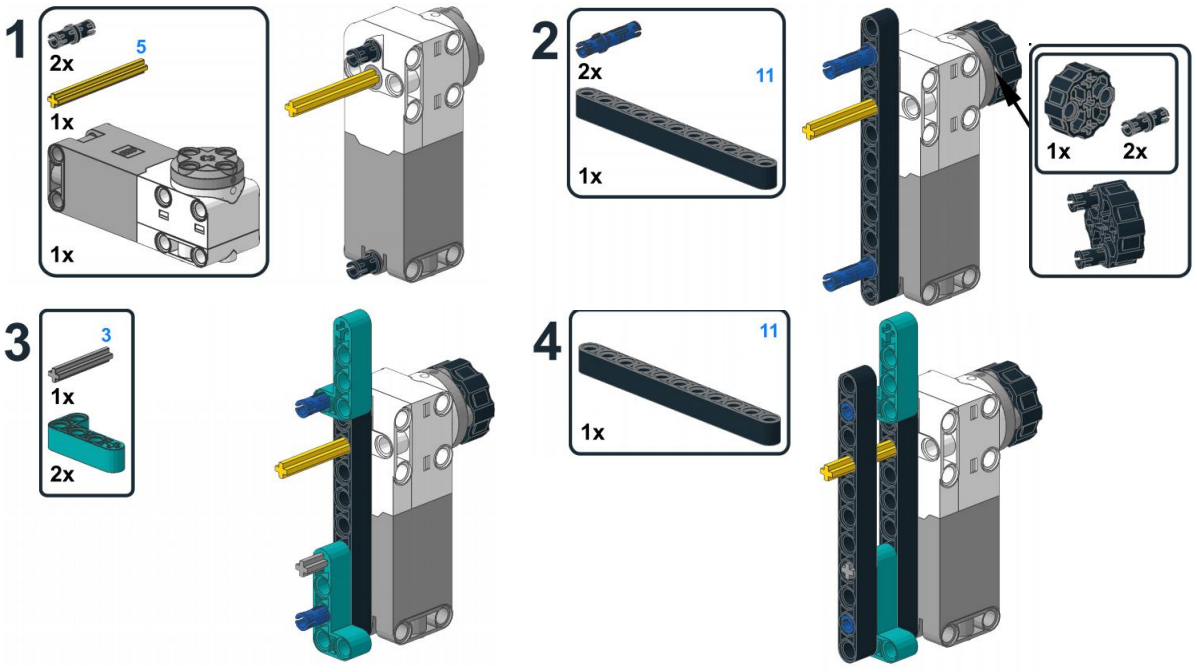

1
3x
1x
2x
2
1x
1x
1x
3
11
2x

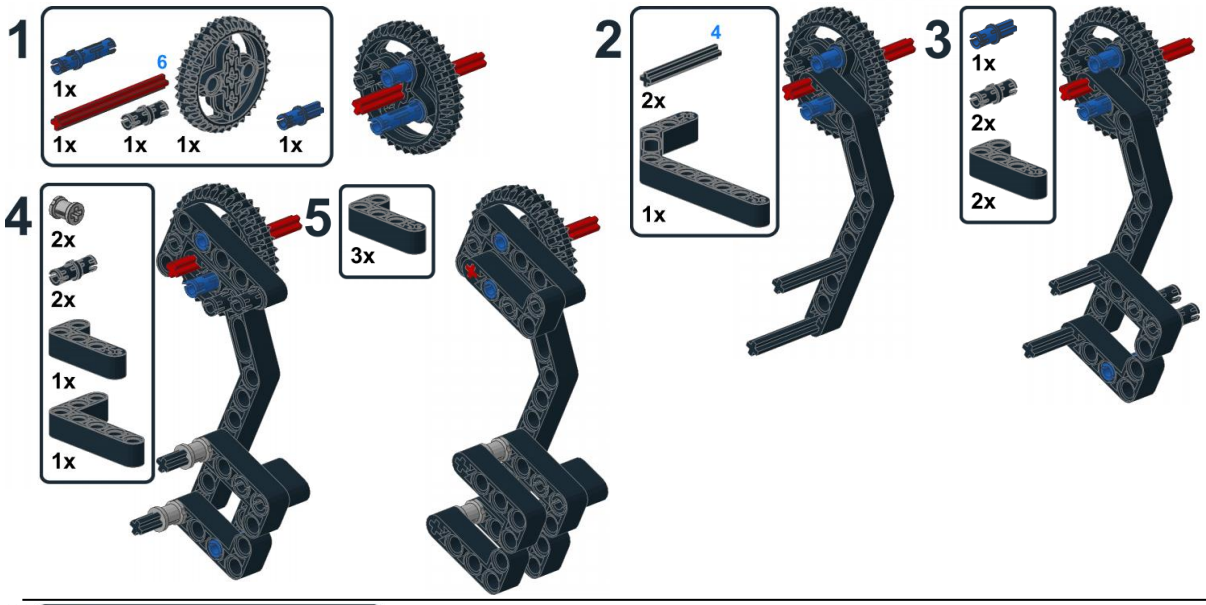

4
1
1
1x
4
1x
2
2x
2x
1
8
1x
1x
1x
2
2x
3
1x
2x
5
1
5
1x
2x
2
2
2x
2x
3
2x
1x
2x
2x
4
1x
1x
2x
2x
5
2x
1x
1x
1x
1
7
1x
1x
1x
2
2x
3
1x
1x
1x

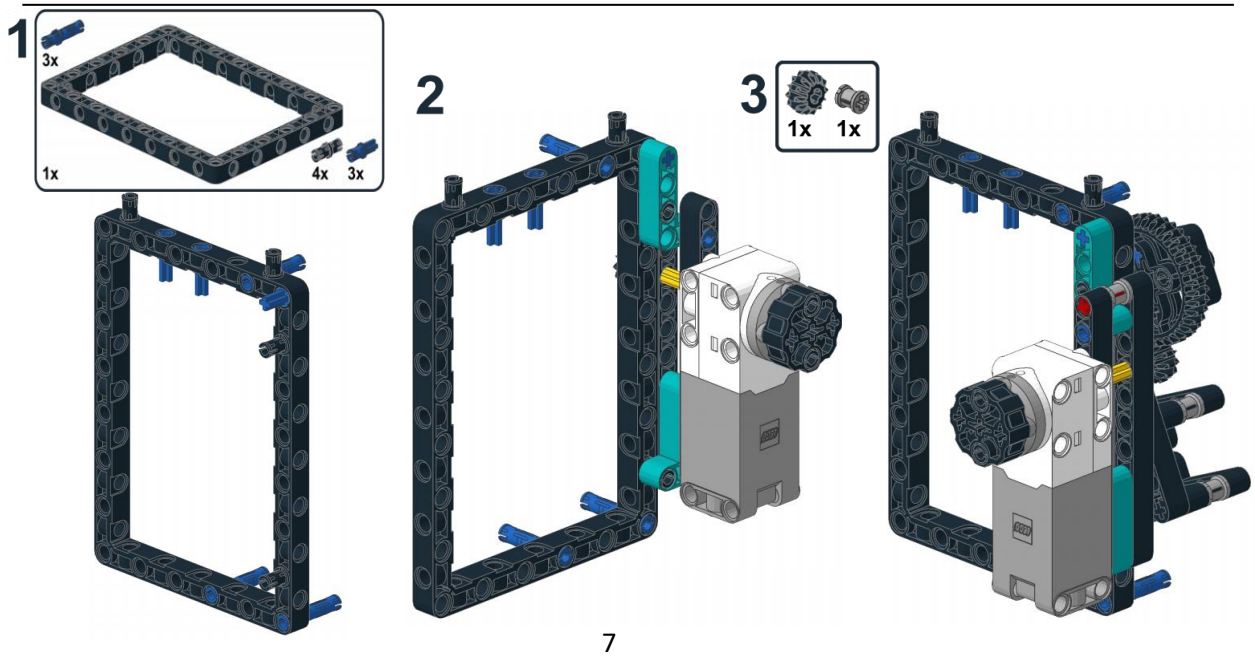

1
1x
4x
2
2x
1x
3
4x

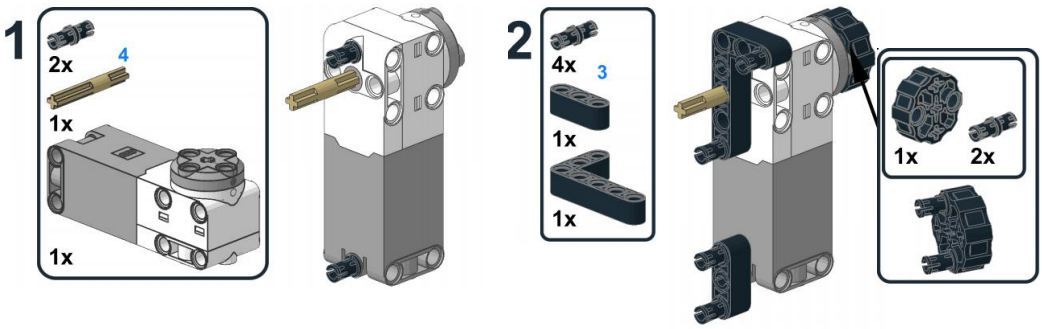

6
1x
1
2x
3
1x
2
2x
1x
3
1x

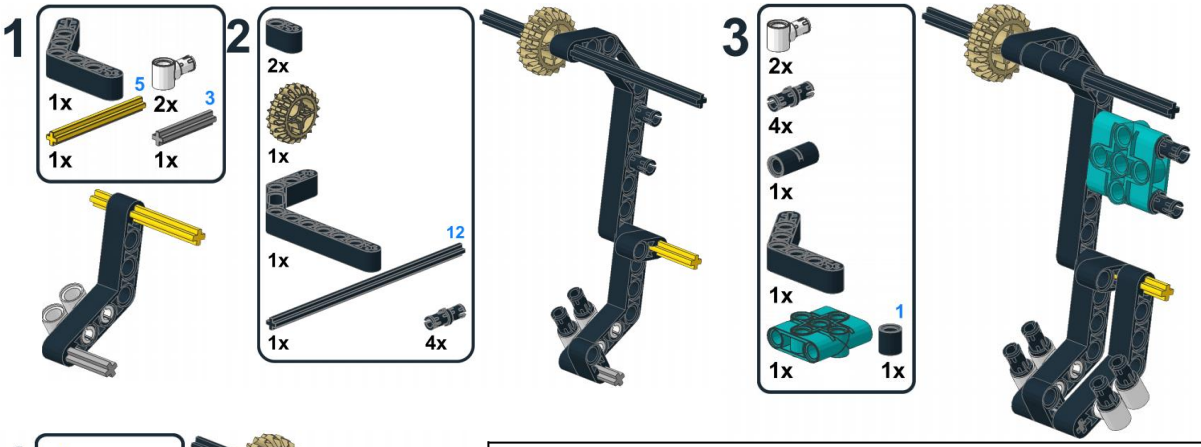

6
2x

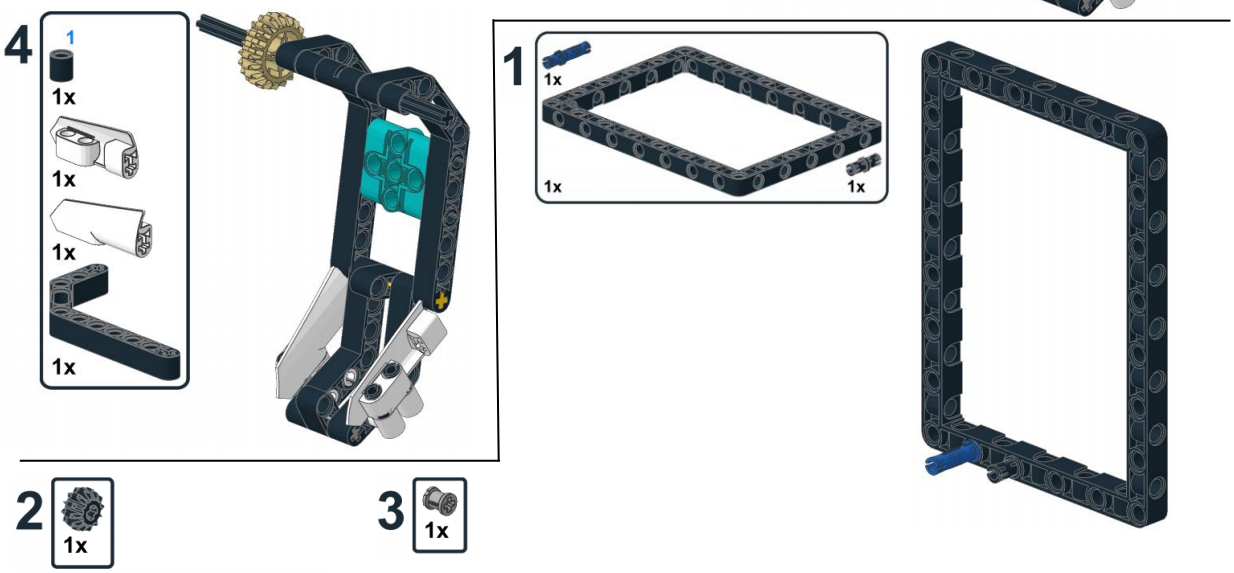

1
2x
5
1x
1x
2
2x
11
1x
1x
2x
3
3
1x
2x
4
11
1x

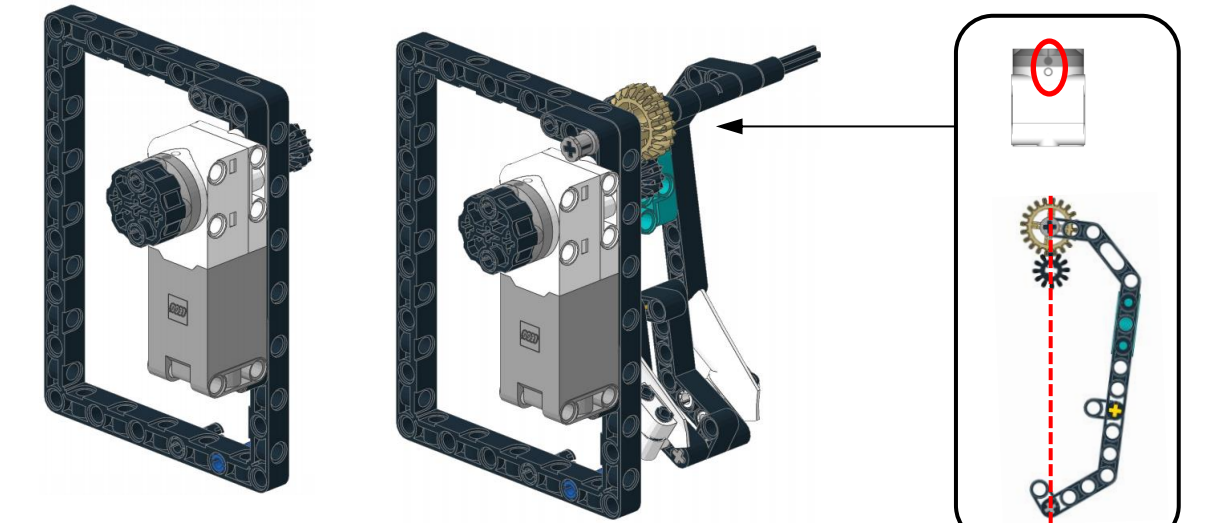

1
6
1x
1x
1x
1x
1x
2
4
2x
1x
3
1x
2x
2x
4
2x
2x
1x
1x
5
3x

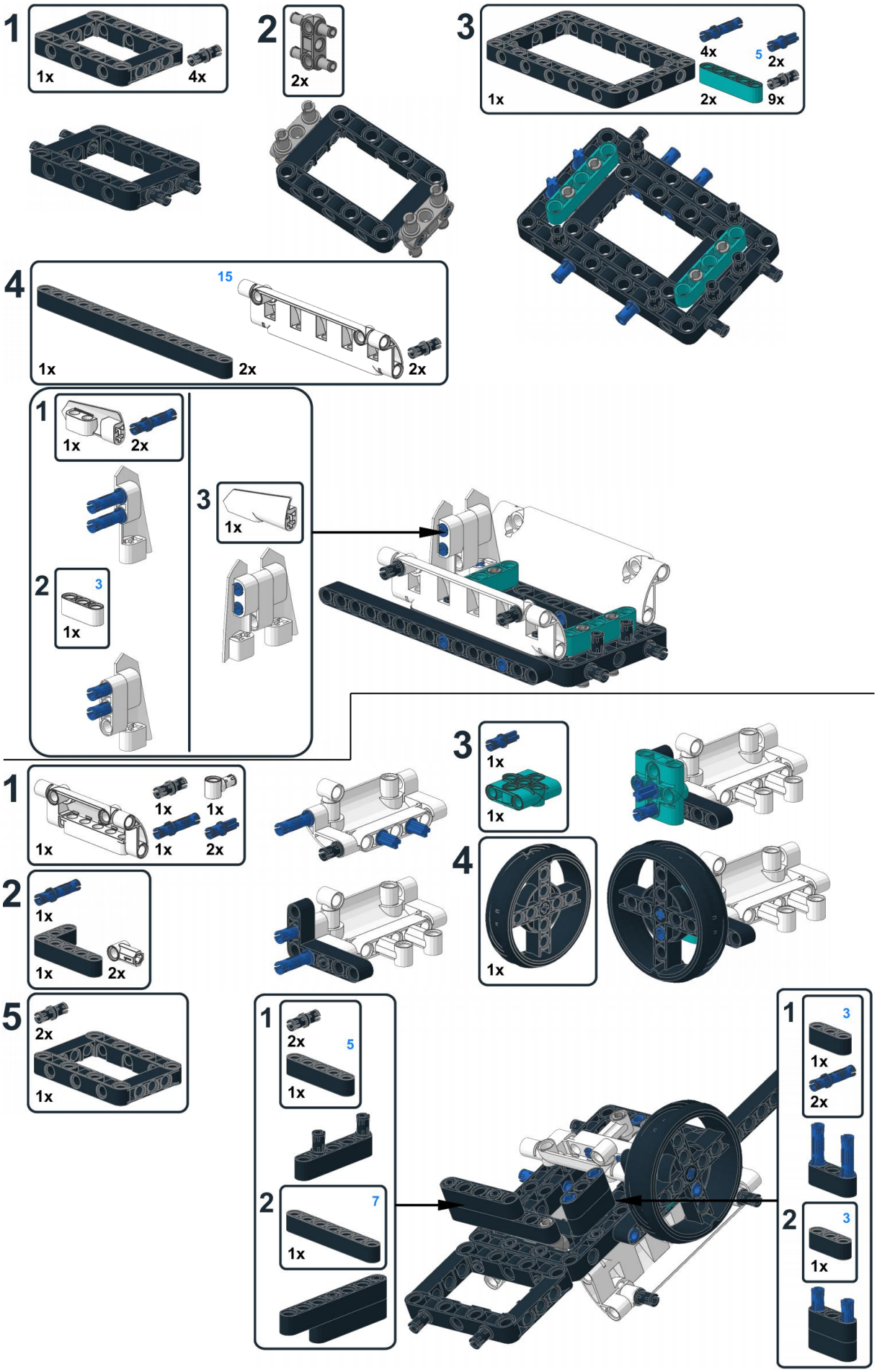

1
3x
1x
4x
3x
2
3
1x
1x

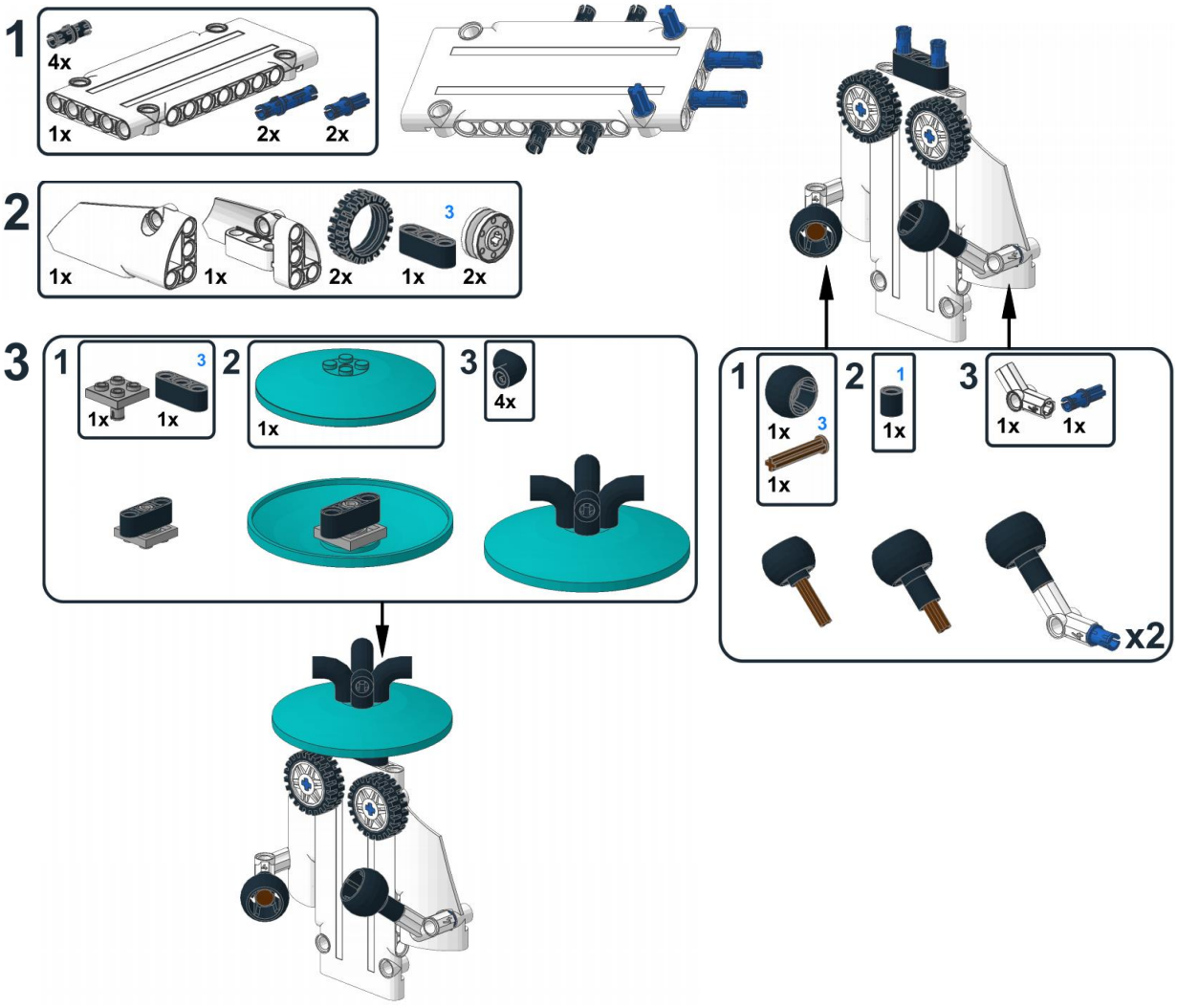

1
2x
4
1x
1x
2
4x
3
1x
1x
1x
2x

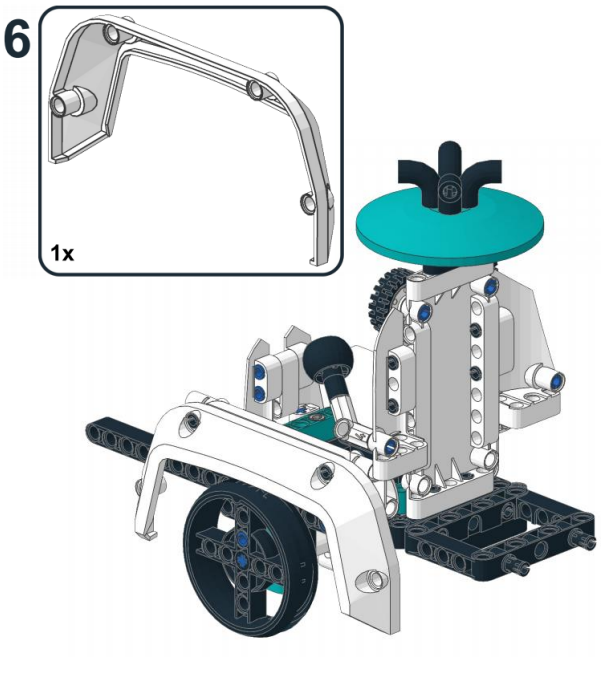

1
1x
5
2x
3
1x
1x
2
2x
1x
12
1x
4x
3
2x
4x
1x
1x
1
1x
1x

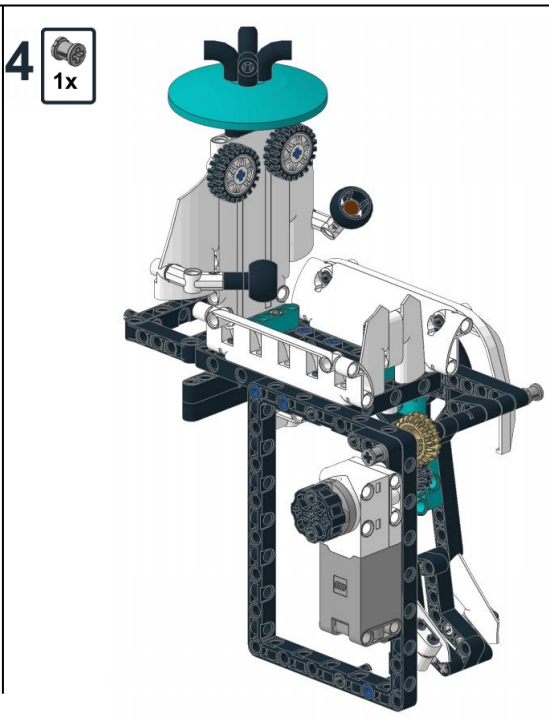

4
1
1x
1x
1x
1x
1
1x
1x
1x


2
1x
3
1x

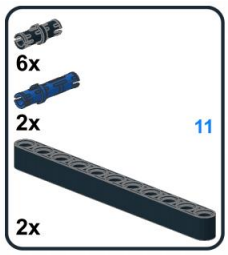

1
1x
4x
2
2x
3
1x
4x
5
2x
2x
9x
4
15
1x
2x
2x
1
1x
2x
2
3
1x
3
1x
1
1x
1x
1x
1x
2x
2
1x
1x
2x
3
1x
1x
4
1x
5
2x
1x
1
2x
5
1x
2
7
1x
1
3
1x
2x
2
3
1x


1
4x
1x
2x
2x
2
1x
1x
2x
3
1x
2x
3
1
3
1x
1x
2
1x
3
4x
1
1x
3
1x
2
1
1x
3
1x
1x
1x
x2

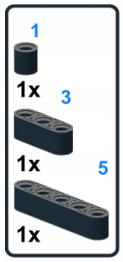

6
1x

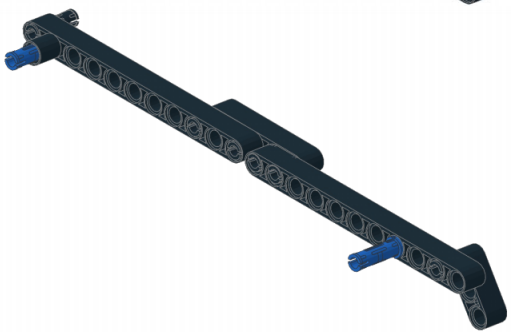

4
1x

1

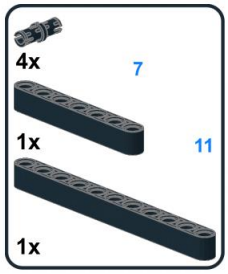


2

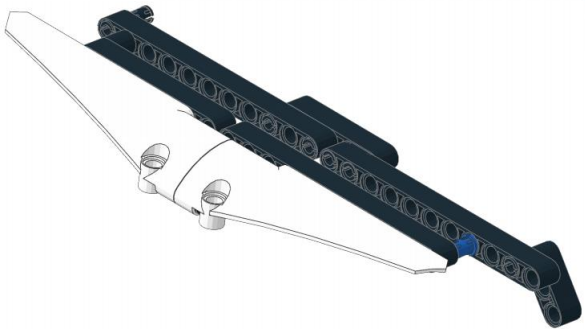




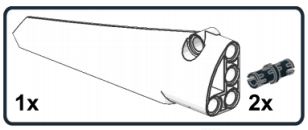
3



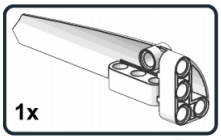
4



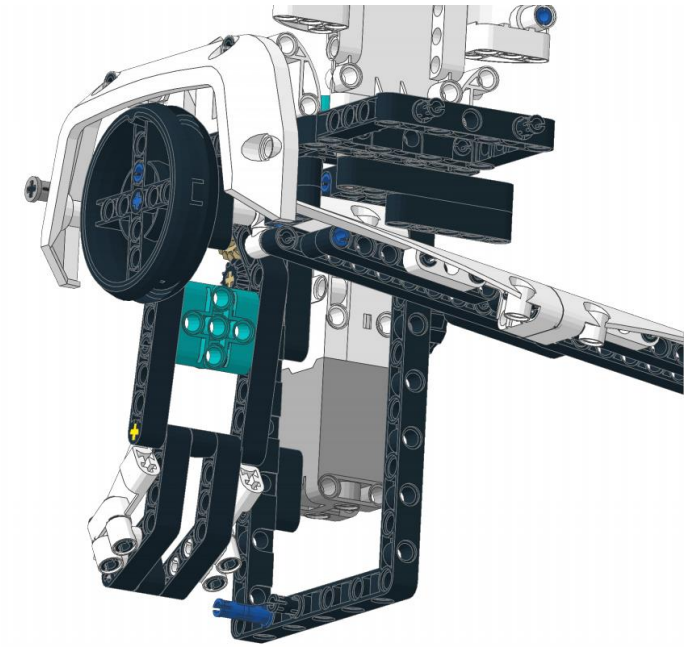
1



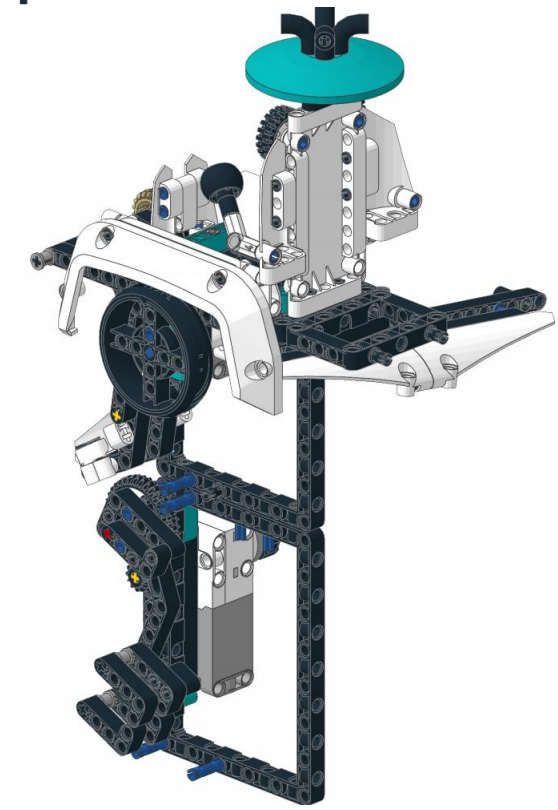
2



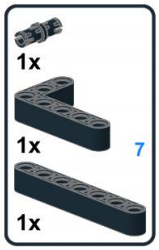
5

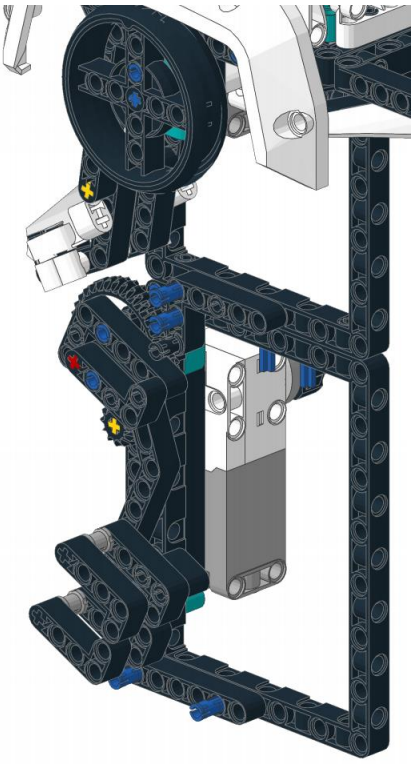


4

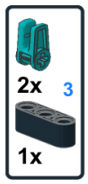


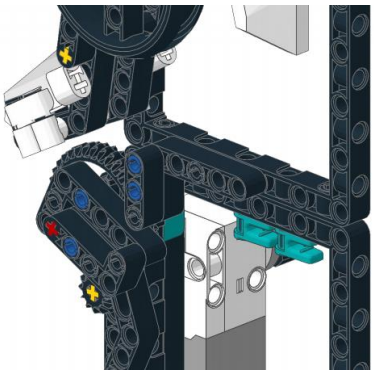
5



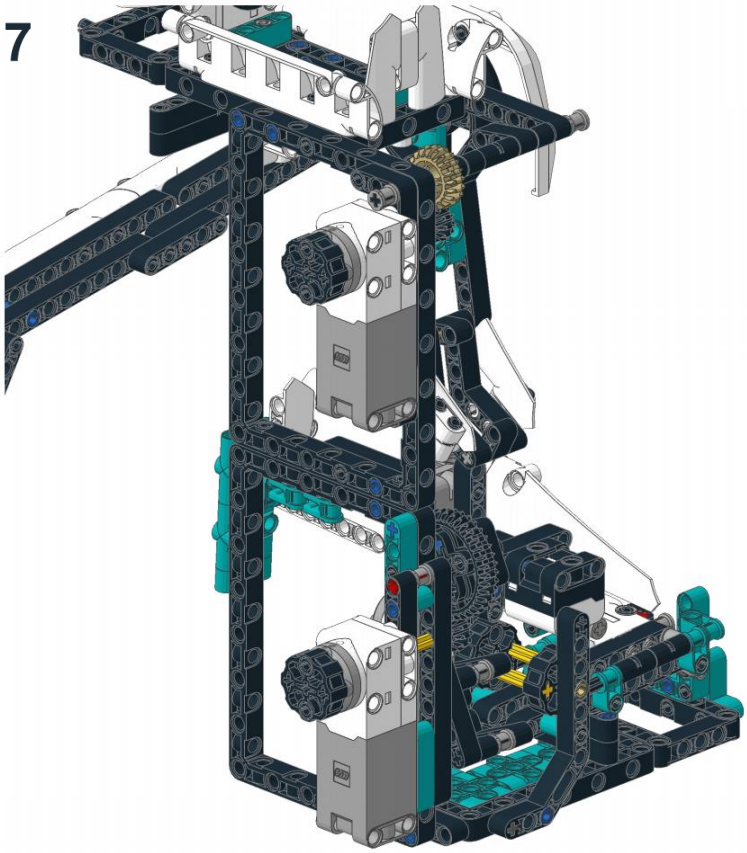


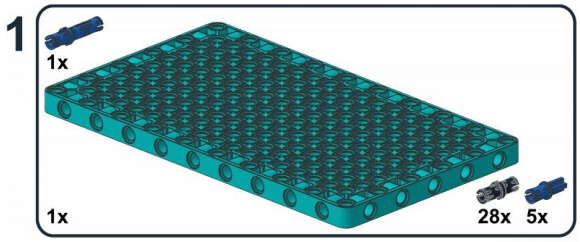
6

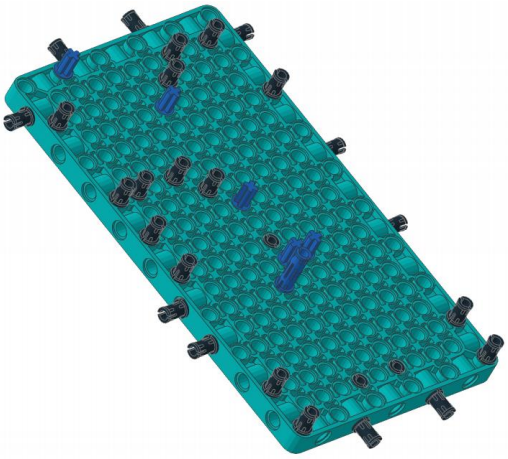


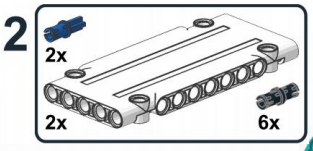


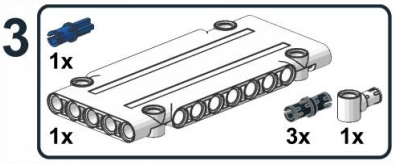
7

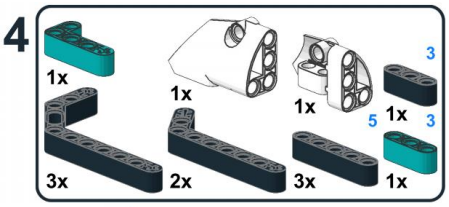


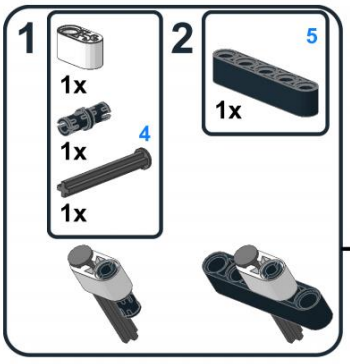

1
1x
1x
28x
5x

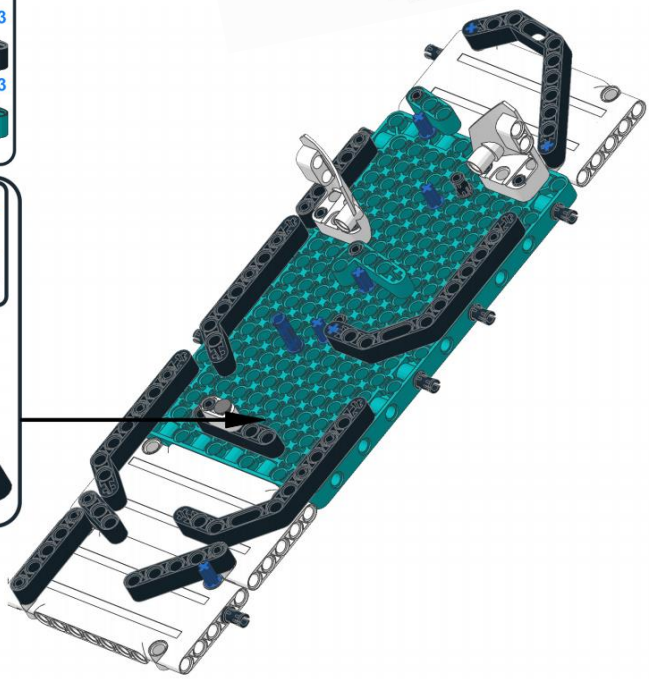


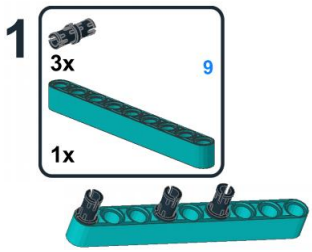

2
2x
2x
6x

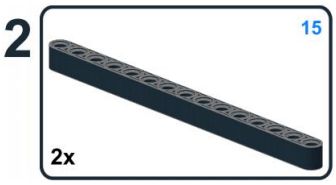

3
1x
1x
3x
1x

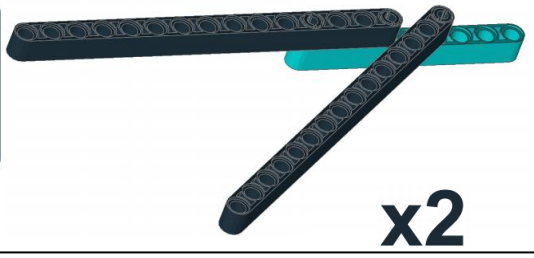

4
1x
1x
1x
5
1x
3
3
3x
2x
3x
1x

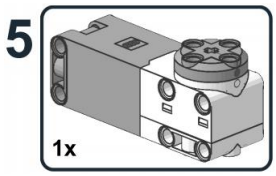

1
1x
1x
4
1x
2
5
1x

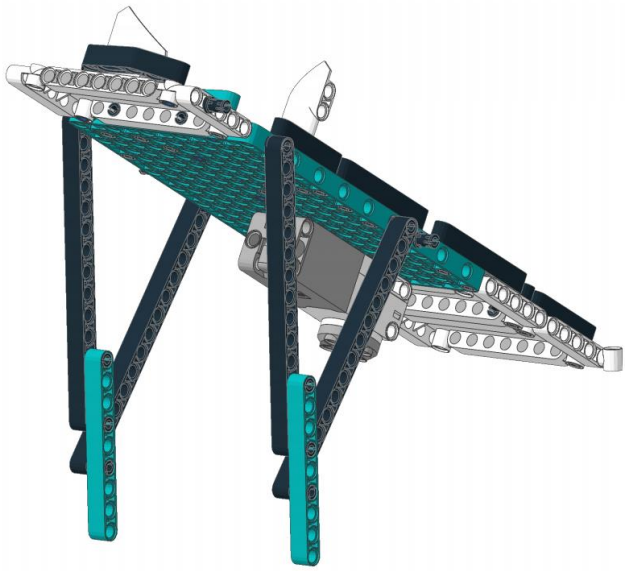


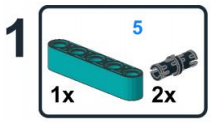

1
3x
9
1x

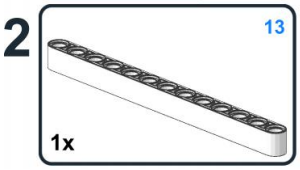

2
15
2x

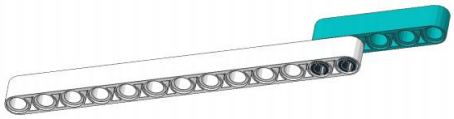

x2

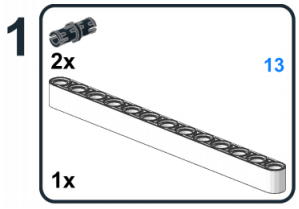

5
1x

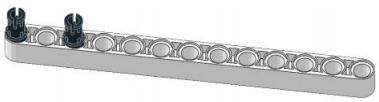


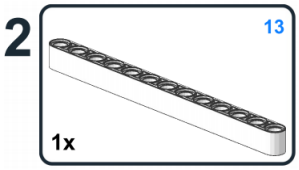

1
5
1x
2x

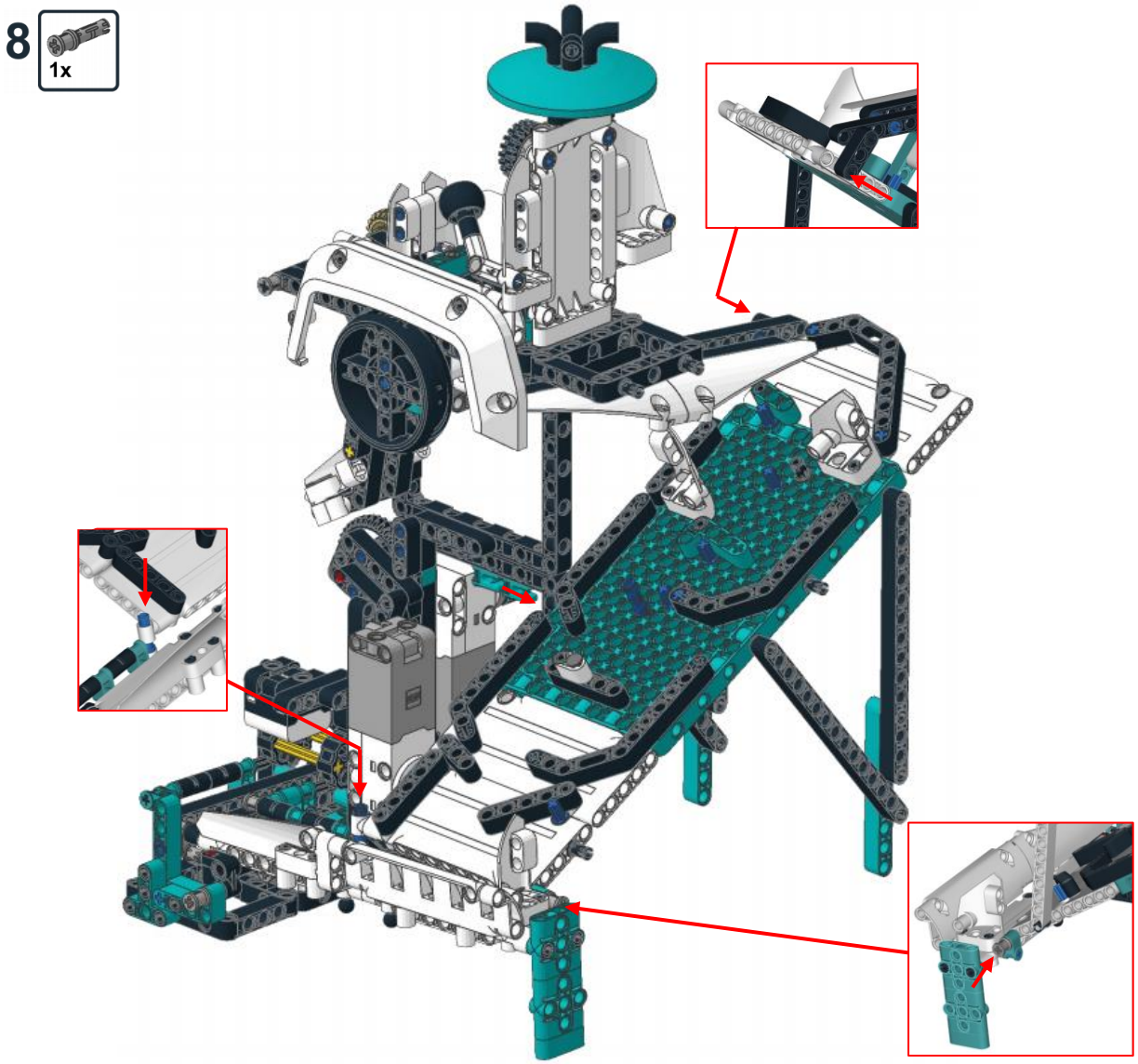

2
13
1x

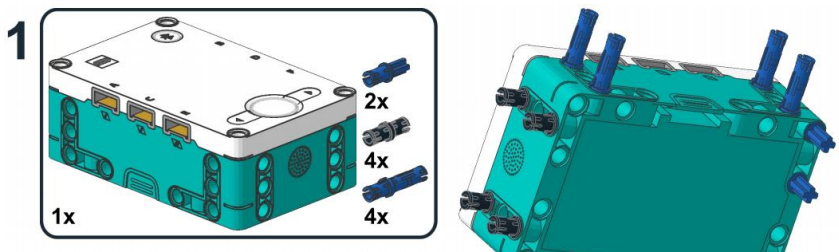



1
2x
13
1x

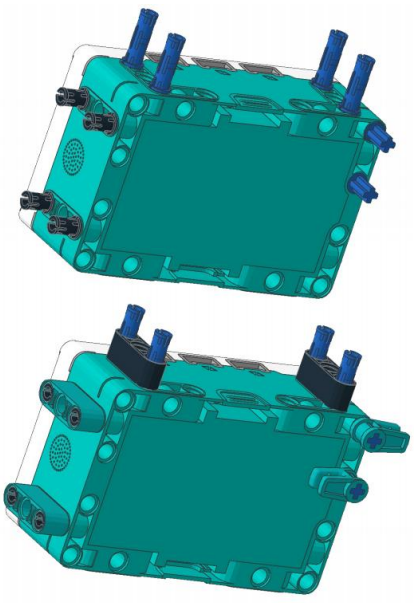



2
13
1x

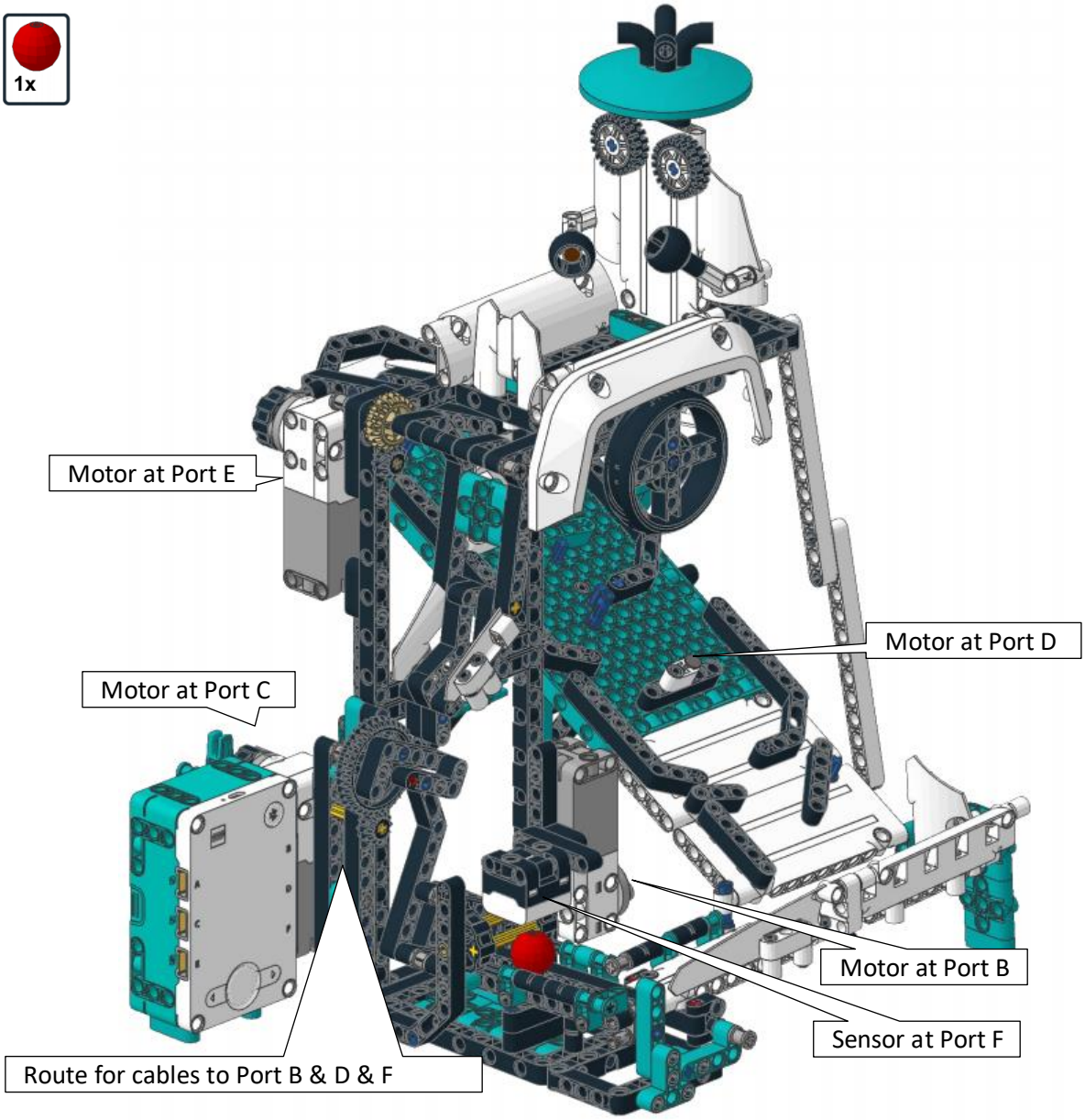

8
1x


1
2x
4x
1x
4x


2
2x 3
2x 3
2x



10 1x


Motor at Port E
Motor at Port D
Motor at Port C
Motor at Port B
Sensor at Port F
Route for cables to Port B & D & F

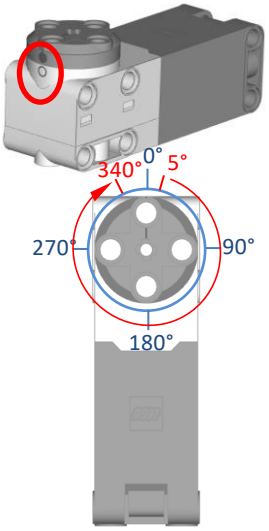
Alternative Motor-Positioning

There are several ways to position a motor. The easiest option is to use the WordBlock "`go shortest path to position`". This moves the motor to one specific position.

To be able to move a motor to an "absolute Position" is a new feature. Previous Mindstorms sets and the Boost set did not offer this option. These sets only offered the "relative position".

So, what is the difference between the two options?

Absolute Positioning:



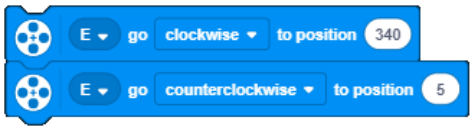
The robot inventor motors have a mark on their casing and the turntable to indicate their zero-position. The command "`go shortest path to position 0`" will bring the motor to this position. The same way, you can move the motor to any position between 0 and 359 degrees.

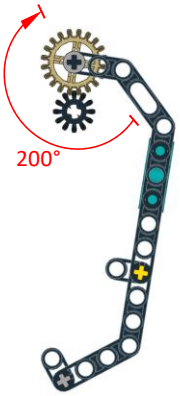
Values below or above 359 are not accepted. So, this command does not allow to turn multiple times before stopping at a specific position.

The direction of rotation can be influenced by selecting "`clockwise`" or "`counterclockwise`". When selecting "`shortest path`" the algorithm will decide themselves in which direction to turn.

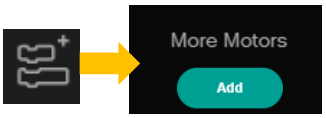
Ball-Inventor uses absolute positioning for the marble separator drum (Motor at Port B) and the movement of the upper arm (Motor at Port E).

As the gear ratio of the upper arm is 12:20, the motor needs to turn 335 degrees for a 200 degrees movement of the arm. The motor will move between its 5- and 340-degree positions.



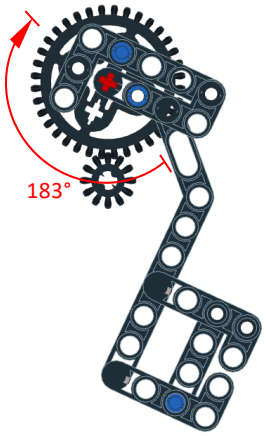


Relative Positioning:



Commands for relative Positioning are located within the “More Motors” extension. You must add them to your command block menu.

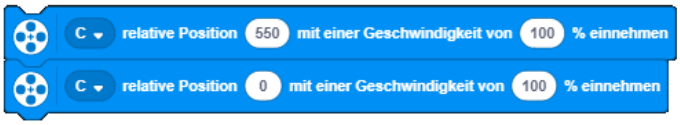
Relative Positioning does not have a pre-defined zero-point. We can demonstrate this functionality with the movement of the lower arm. With 12:36 the gear ratio equates to 1:3. For the required arm movement by 183 degrees, the motor must turn more than one full turn. From its self-defined zero-point, the motor will turn 550 degrees to move the arm by 183 degrees.



When starting the program, the current position of the arm is unknown to us. The absolute position does not help, as the arm could have different positions for one and the same motor position. This is particularly true if the gear ratio is even higher (e. g. as sown in the egg-Inventor or the Inventor-Writer) or for other applications (e. g. driving). Very often, there is no indication about where parts of the model are initially located. Thus, the program will have to first evaluate the position of the motor. This is usually called “initialization” of the model and you can observe this also in the “official sets” – although this functionality is often hidden in pre-defined, model specific blocks.

To initialize the Ball-Inventor, we start to turn the motor with a low level of power (30%) counterclockwise towards the mechanical stop. As soon as the arm touches the stop, the speed lowers and we stop the motor and set its position to -25 degrees. This represents the angle to bring the arm from the stop to the hand-over-position.

The arm can now be moved via the blocks “`go to relative position`”. Thanks to the high gear ratio, the arm can be positioned more accurately compared to an arm directly mounted to the motor.



However, the biggest advantage of relative over absolute positioning is the possibility to directly approach positions across multiple turns. Relative positioning also allows negative positions.

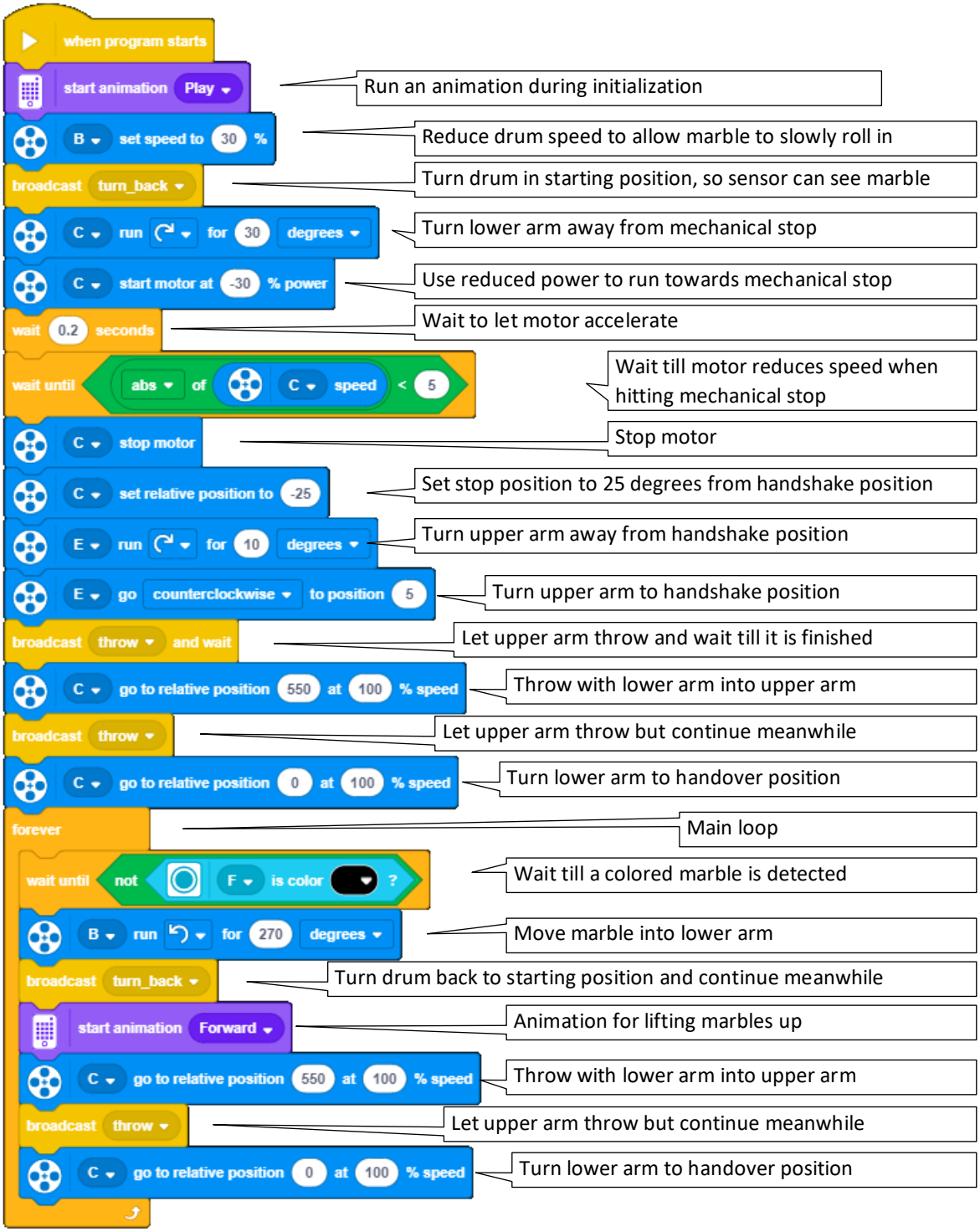
Reduction of motor power

Speed and power of the motor can be controlled separately. When setting a speed, the motor will try to keep this speed even when the model hits resistance and higher torque is needed.

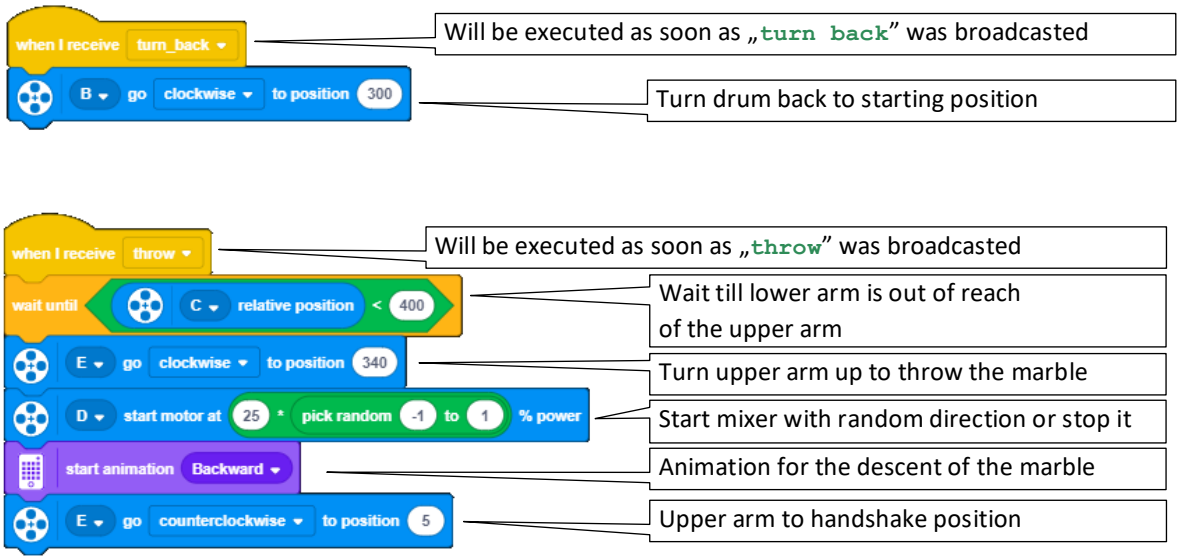
When setting a specific power, the motor will run with a certain torque. When there is more resistance, speed will reduce until the motor stops. Setting a low power can be used to initialize relative positioning for a motor without the need for a separate sensor. Furthermore, the power setting can be used to establish a customized control loop for motors. Such control loops are used within the Inventor-Balancer, Inventor-Writer and Egg-Inventor.

Program the Ball-Inventor

A relatively simple WordBlock program can run the Ball-Inventor.



There are no self-defined blocks for sub-routines required but the program relies on broadcasting blocks to run program sequences in parallel.



Install downloadable program on the Hub

As an alternative to creating the program yourself, you can download a ready to use version and upload it via the LEGO Mindstorms APP.

`Ball.lms`

Open your Mindstorms-APP on a PC. Via the menu „File" and „Open..." you can load any `.lms` file into the APP. The program can then be uploaded to the hub the same way you upload any other program to the hub.

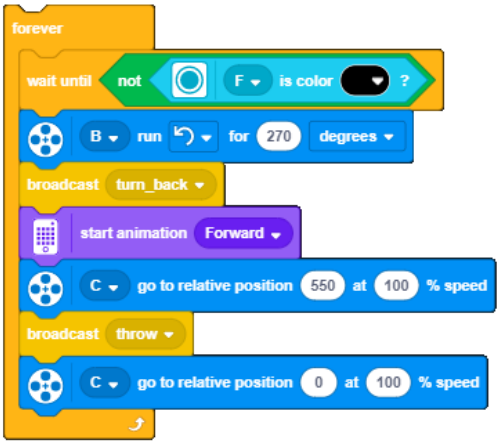
Play

As soon as the program is started, all motors are brought to their starting position and the arms are emptied in case an marbles remain on them from a previous run. After this initialization and emptying of the arms, the main loop will start.

Unfortunately, the red plastic marble included in the set is not perfectly round and very light. This causes the marble to occasionally stop in front of the marble separation drum. Alternatively, you can use LEGO-Basketballs or LEGO-Soccer balls with 14mm diameter. Heavier metal or glass marbles up to a diameter of 16.5mm do run even better.

Black or transparent glass marbles cannot be detected by the color sensor and need a slight change in the program. The easiest option is to just delete the “Wait”-Block including the color detection. The ball inventor will then continuously transport marbles up, even if no marbles are in front of the sensor. As long as you have more then 3 marbles on the marble run, there might be no difference to the version including the color sensor. With less marbles, the arms will move the arms empty part of the time. This is what the color sensor is preventing in the original program.

Program with color sensor



Program without color sensor