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. The Draw Inventor can also be used independently from a PC, as soon as the program is uploaded to the hub.

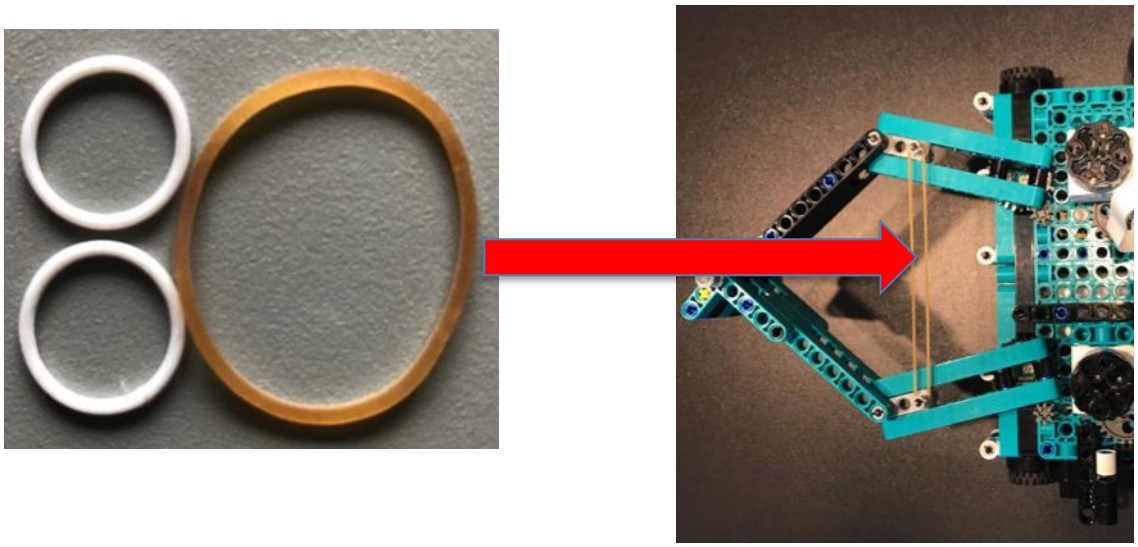
The Document has three segments:

- BUILD – Build the Cube-Inventor
- CODE – Program the Hub
- Play - Operate and optimize

Draw Inventor – Draw and Write

Draw Inventor can use any pen to draw or write. You can attach the pen to the arms using the rubber bands within the Inventor set or you can use any household rubber band you have available. The pen should be firmly attached tot he drawing arm to enable exact positioning oft he pen tip.

In addition to the rubber bands for the pen, you should connect the two main levers by another household rubber band. This rubber band eliminates the play in the Lego-gears. The rubber band should be under tension in any position oft he lever arms. However, the force oft he rubber band should not be too strong, it could prevent the levers from reaching the desired positions.

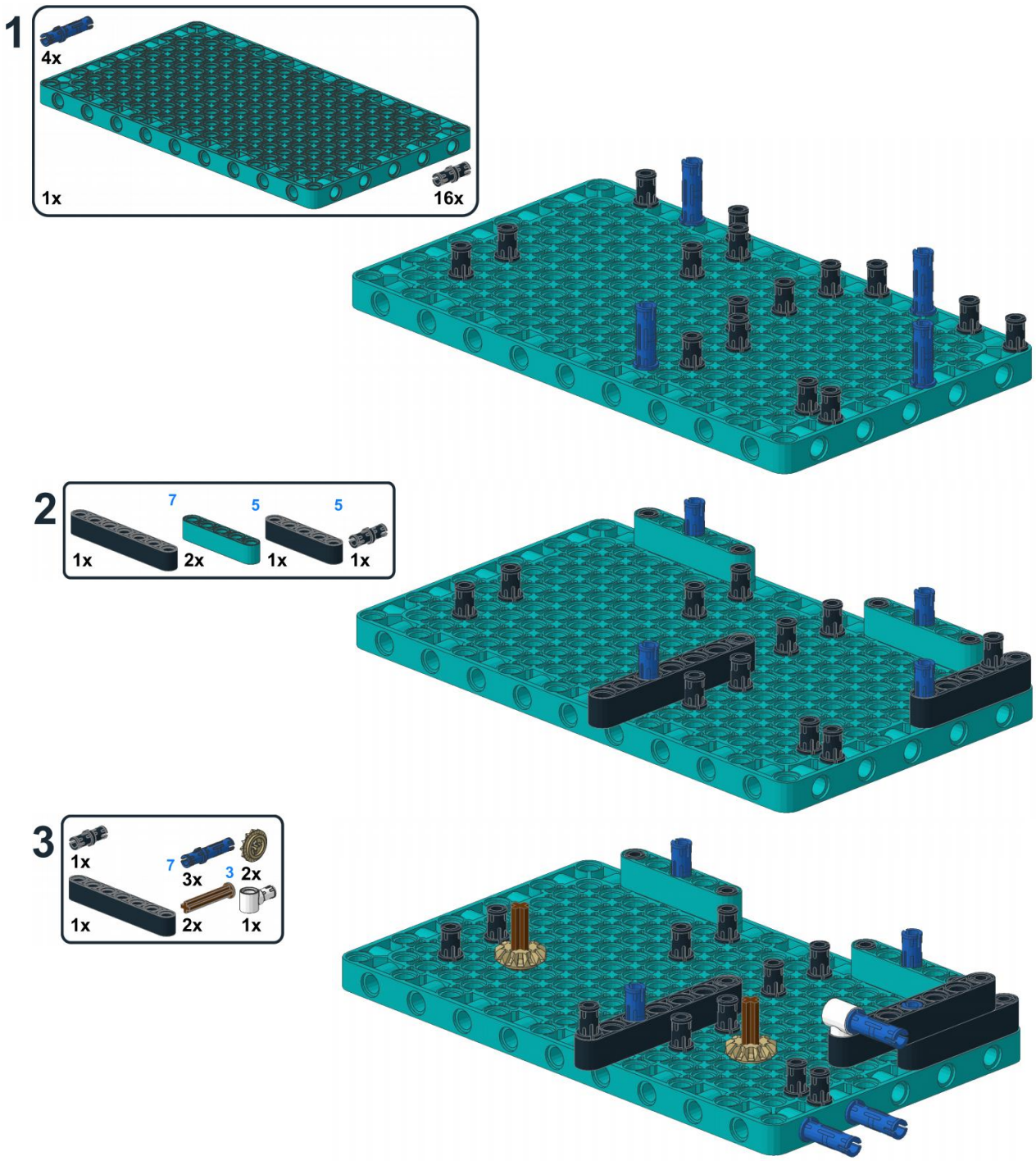


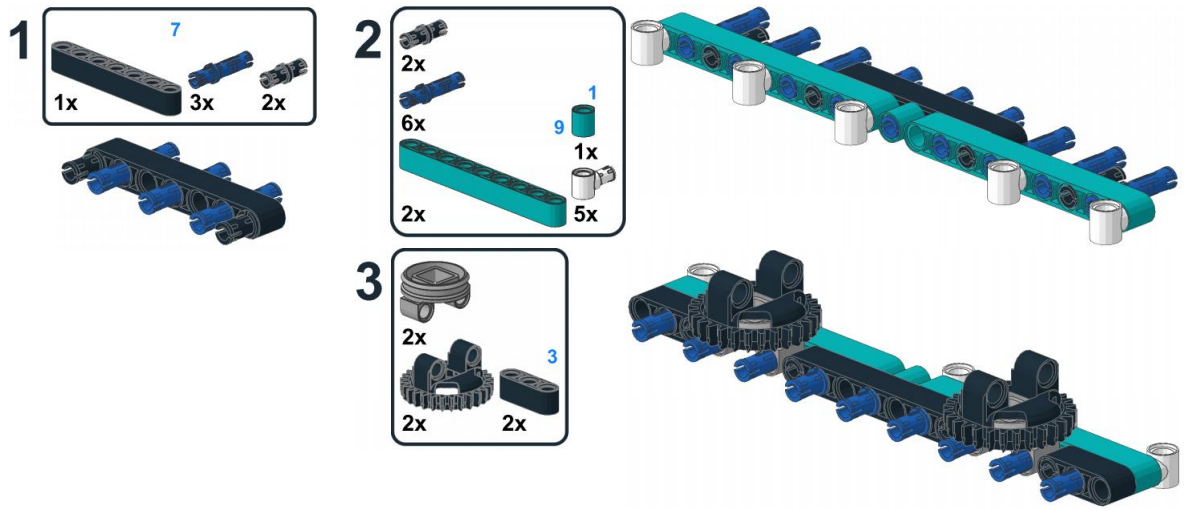
Furthermore, the pen should move smoothly over the paper. The height oft he pen tip should be adjusted in a way to just barely touch the paper when the pen is in the lowered position. When the pen is pressed too hard tot he paper, the friction may prevent the pen from reaching the programmed positions.

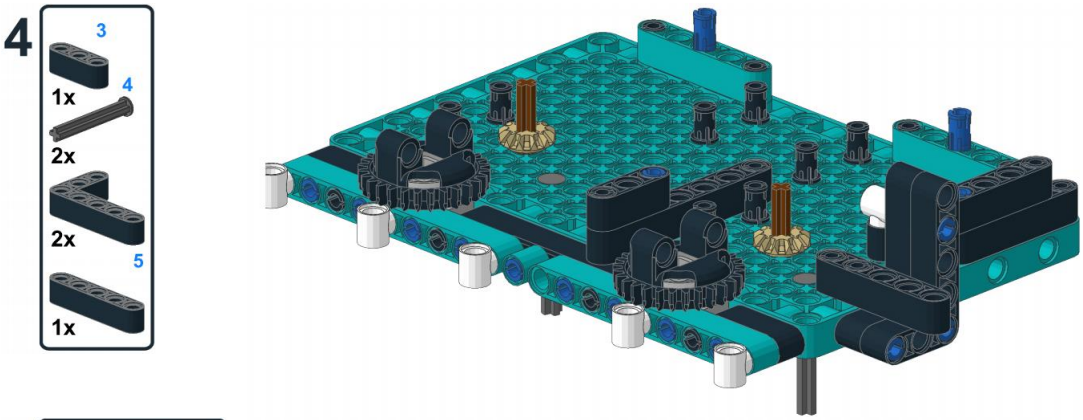
When in the upper position, the pen should usually be about 9mm above the paper, but you might have to experiment a bit to get the best results.

Building Instruction

Draw Inventor can be built with the standard content of the Mindstorms Robot Inventor set 51515 alone. Apart from a pen, 1 to 3 rubber bands and a PC, no additional components are needed.

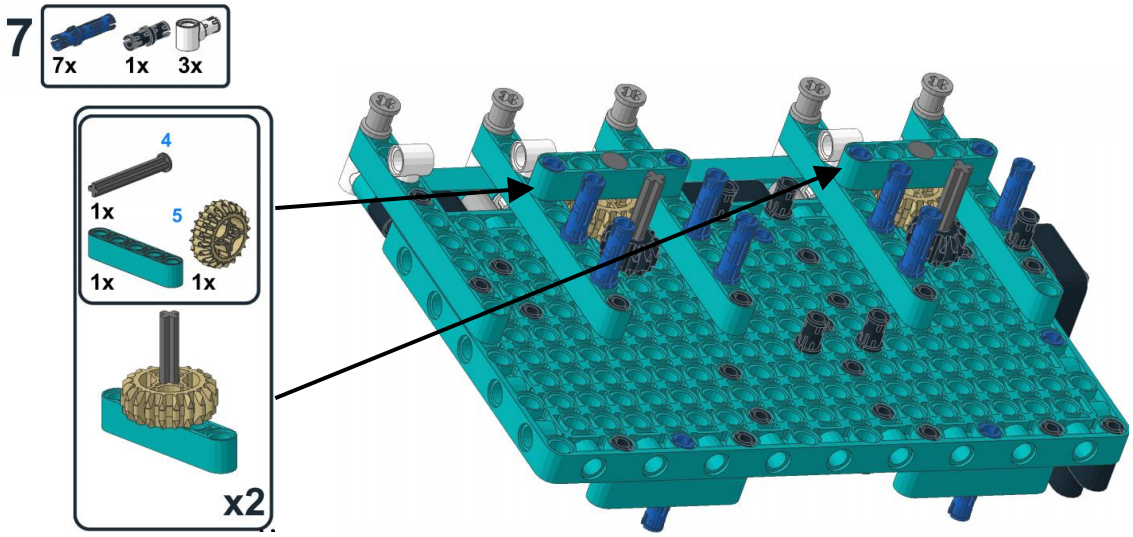


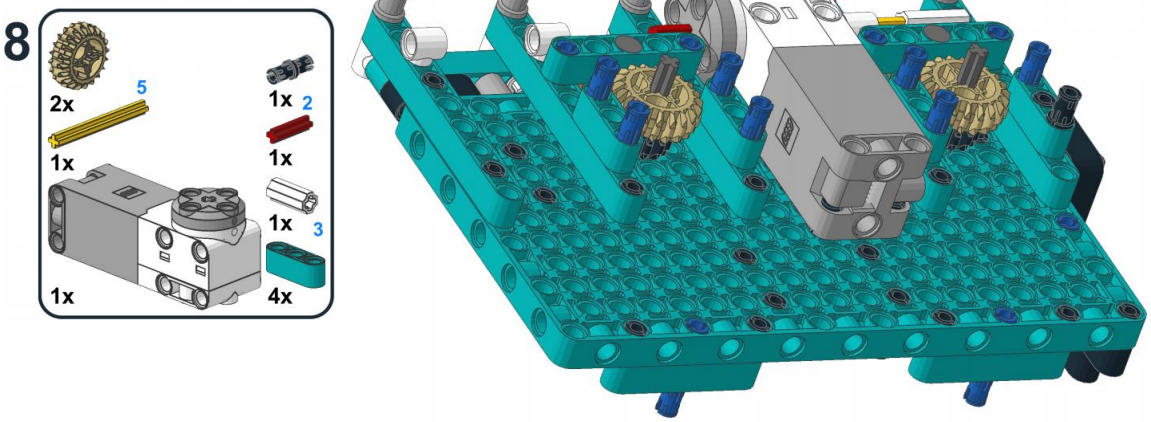

1
7
1x
3x
2x
2
2x
6x
9
1
1x
2x
5x
3
2x
3
2x
2x

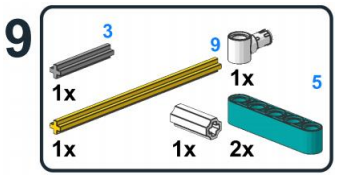

4
3
1x
4
2x
2x
5
1x

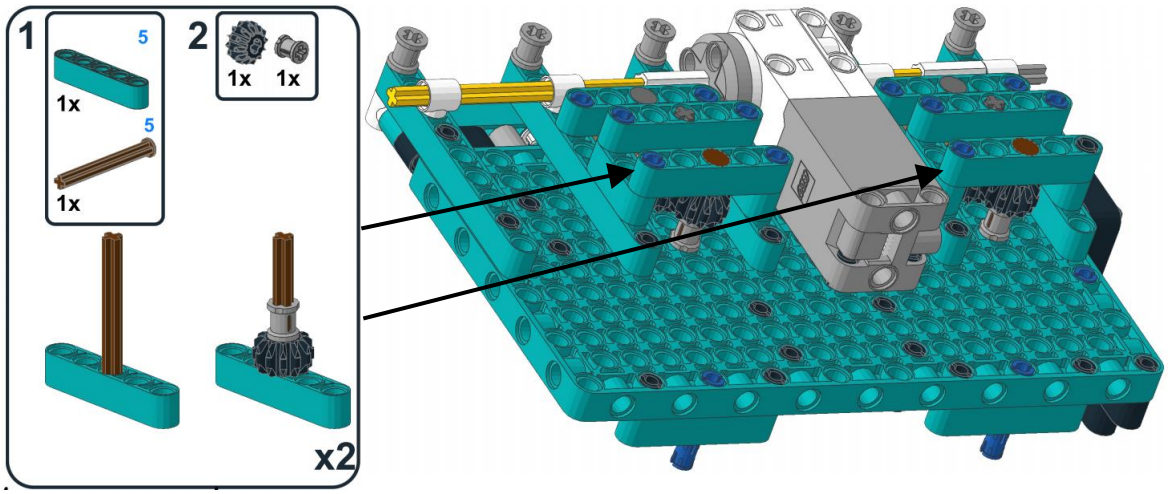

5
4x
10x
2x

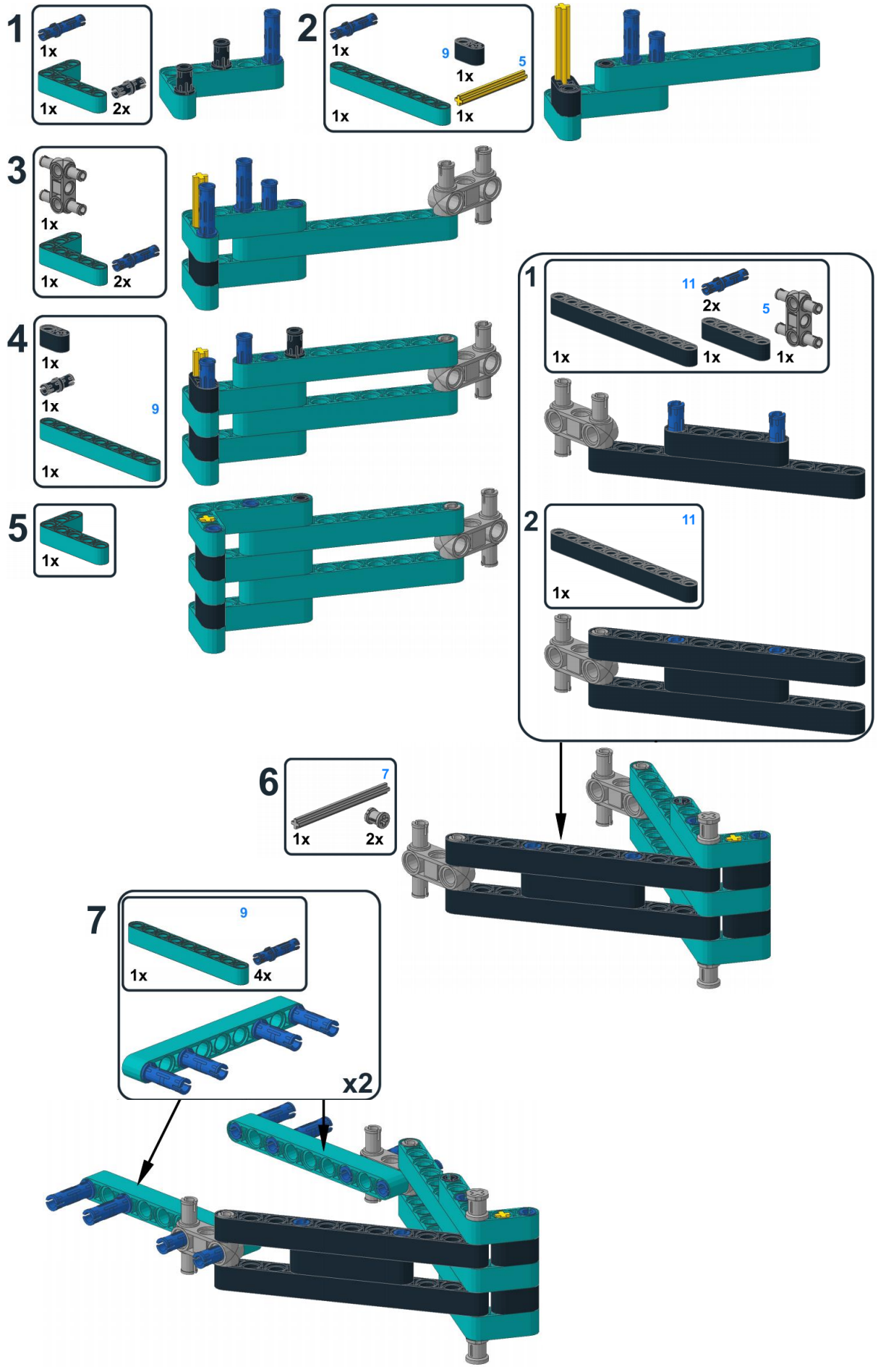

6
5x
9
5x


7
7x
1x
3x
4
1x
5
1x
1x
x2

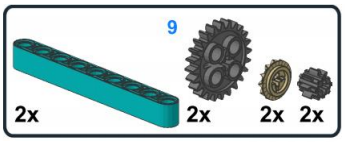

8
2x
5
1x
1x
2
1x
1x
3
1x
4x

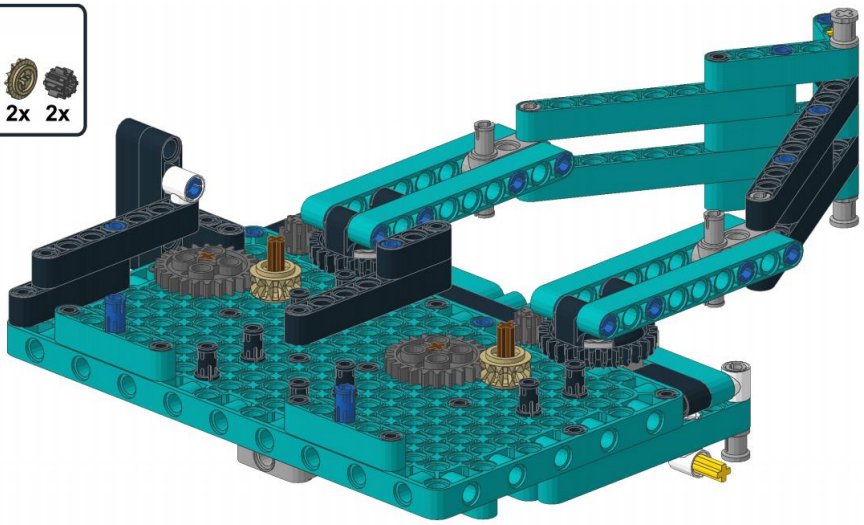

9
3
1x
9
1x
1x
1x
5
1x
2x


1
5
1x
5
1x
2
1x
1x
x2

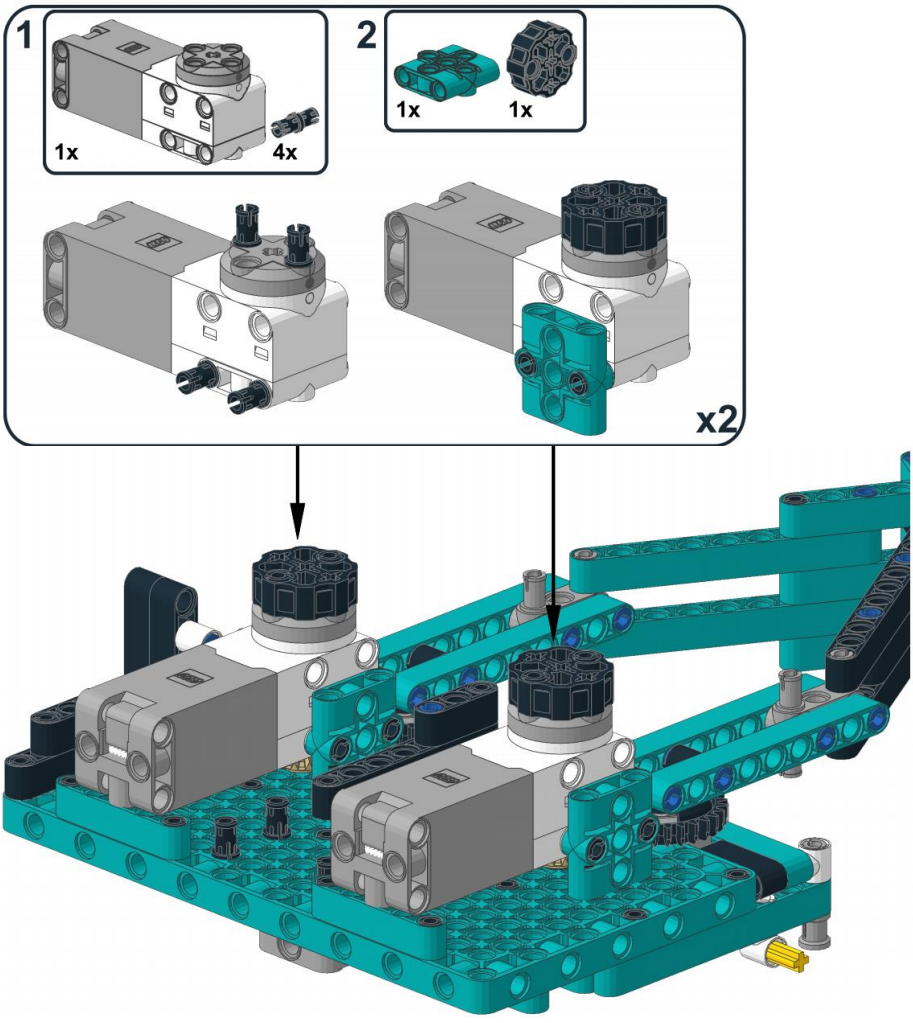

1
1x
1x
2x
2
1x
9
1x
5
1x
1x
3
1x
1x
2x
4
1x
1x
9
1x
5
1x
1
11
2x
5
1x
1x
1x
2
11
1x
6
7
1x
2x
7
9
1x
4x
x2

10



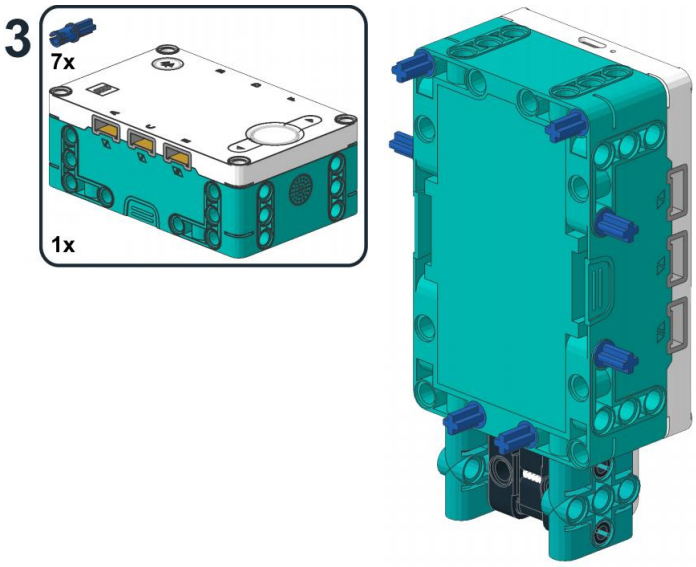


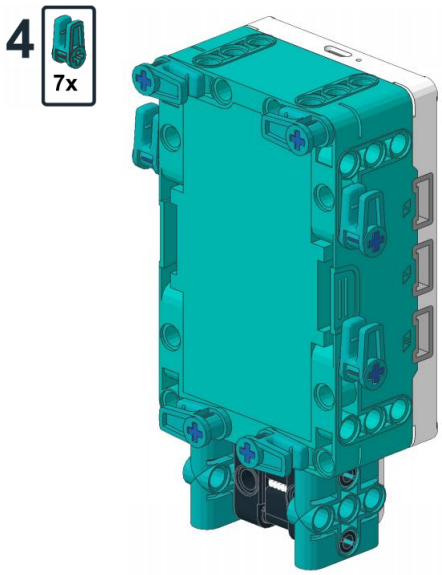
11

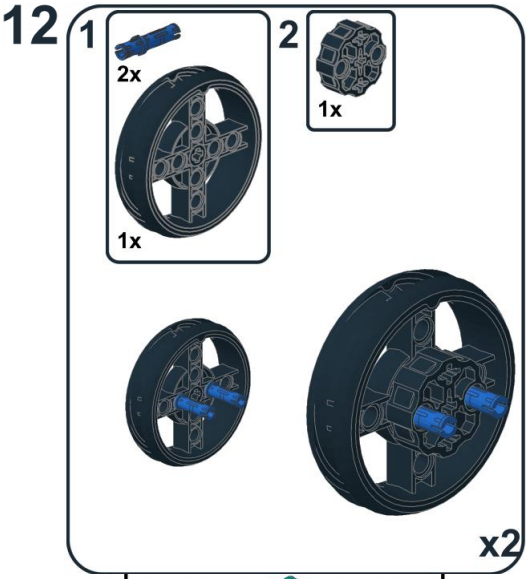


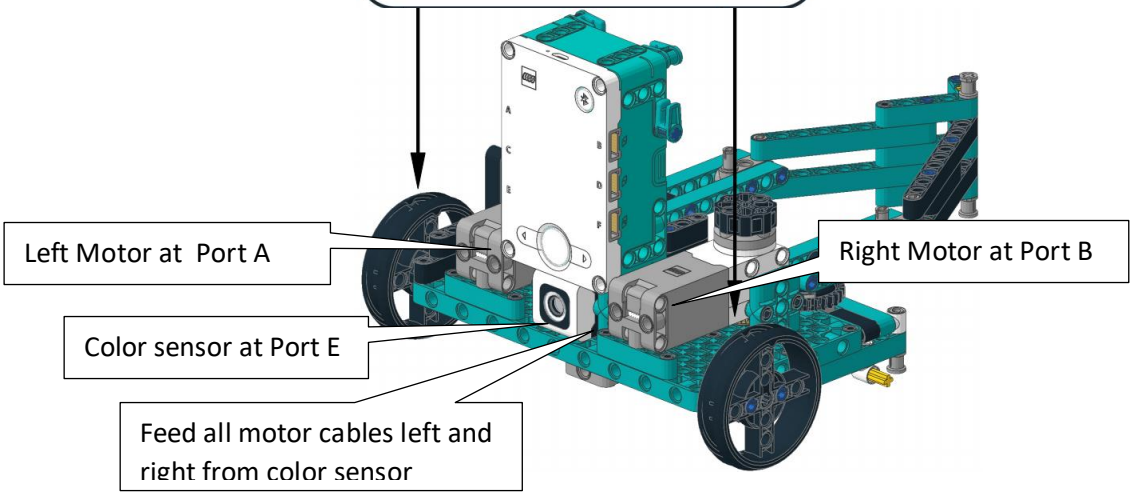

1
1x
4x

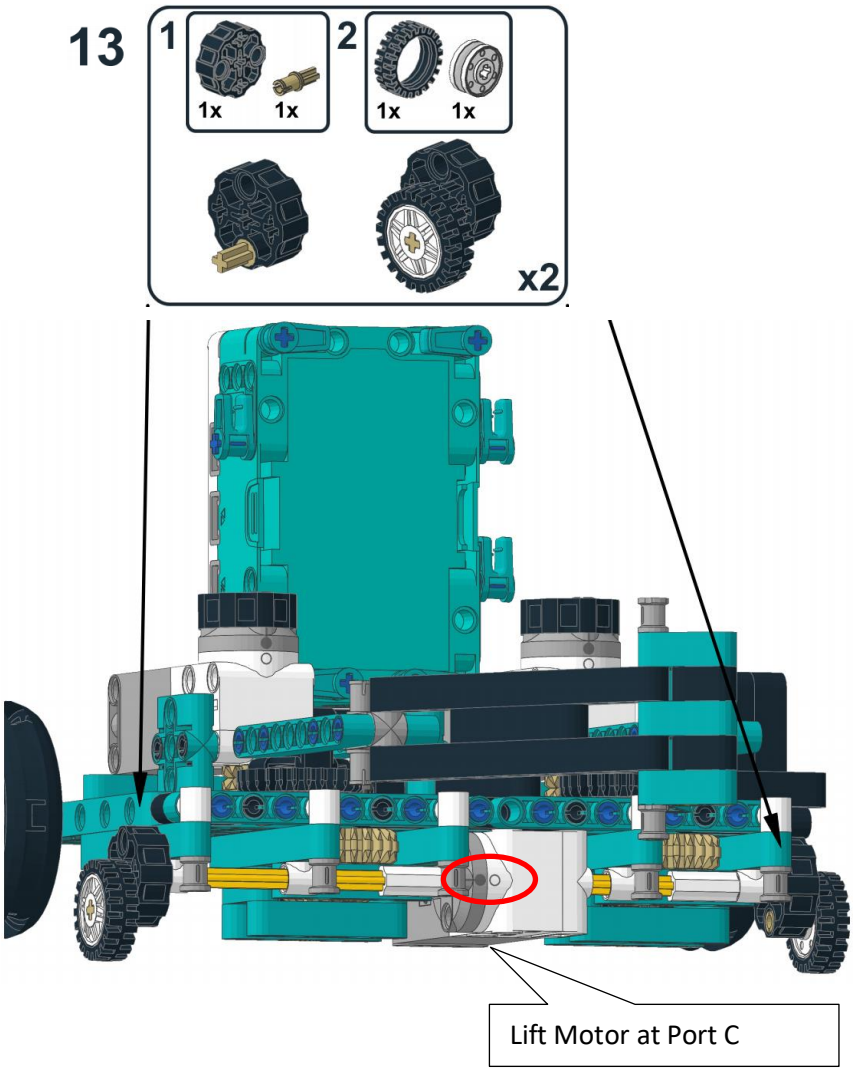

2
4x
2x


3
7x
1x


4
7x


12
1
2x
1x
2
1x
x2


Left Motor at Port A
Right Motor at Port B
Color sensor at Port E
Feed all motor cables left and right from color sensor


13
1
1x
1x
2
1x
1x
x2
Lift Motor at Port C

Installation on the Hub

Two files are supplied fort he Draw Inventor:

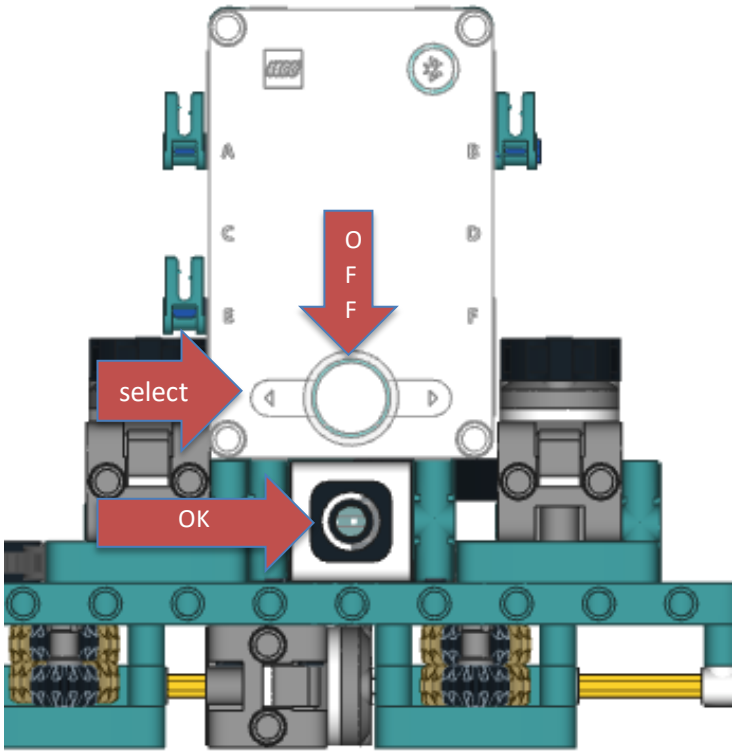
`draw.lms` - Hub-program for drawing

`write.lms` - Hub-program to write

The programs `draw.lms` and `write.lms` can be uploaded tot he hub using the standard LEGO Mindstorms APP.

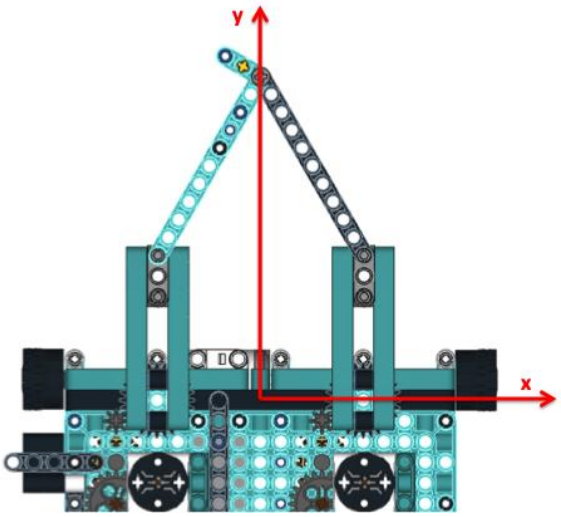
Open your Mindstorms-APP on a PC. Via the menu „File“ and „Open...“ you can load any `.lms` file into the APP. This program can then be uploaded to the hub the same way you upload any other program to the hub.

After starting one of the programs on the hub, you can select one of the 5 drawings (Heart, House, Smiley, Teddy, Minion) or a text (0 = MINDSTORMS INVENTOR, 1= HAPPY PLAY) by using the arrow buttons on the hub. To start the drawing or writing, you cover the color sensor with a finger. Clicking the center button of the hub will end the program.



Coordinate system

The position of the pen is defined in a normal x, y coordinate system. Millimeters are used as units. X and y axis intersect in the center between the two turntables of the main levers.



Own texts

In the first row of the program `write.lms` the variablen `writelist` defines the standard texts „MINDSTORMS INVENTOR“ and „HAPPY PLAY“.

```
writelist = ['&-35,100 $2 MINDSTORMS&-35,80 INVENTOR ', '&-30,105 $2.5 HAPPY &-30,80 PLAY']
```

In addition to the plain text, there are some control commands for the starting position (&) and scale ($) of the text.

Text and all commands have to be enclosed within quotation marks “ or ´ . In the supplied program, only capital letters can be written. More letters or symbols can be added.

The special character & followed by comma separated x and y coordinates set the starting position for the next letter. Example: `&-35,100`

The special character $ followed by a number sets the scale factor for the following text. A value of 1 prints the text in the originally defined size. A value of 2 writes double size and 0.5 in half the original size. Example: `$2`

After the numbers of a command, you have to add a space before the next text element. This space will not be printed.

If the coordinate of a letter exceeds the reach of the pen, this letter will not be printed. Printing will continue with the next letter within the printable area. E.g. after a new starting point was set.

Own drawings

The first few rows of the code in `draw.lms` describes the basic command syntax for drawing. This syntax is also used for the definition of the letters in `write.lms`.

Every drawing is defined as a list of basic commands in a variable. The list syntax is defined by basic python list syntax. Commands are defined by a letter followed by one or several numeric parameters.

„S“ = Scale: following coordinates and radius are multiplied by this scale factor.

„O“ = Offset: these coordinates are used as the zero point for x and y

„M“ = Move: move to x, y coordinate without drawing (pen up)

„L“ = Line: Draw a line from current position to x y position

„C“ = Curve: sub-circle defined by starting angle, circle angle, radius and rotating direction

The exact syntax and some examples is described in the first few lines of the program code. den.

The variable names of the drawings are listed in the variable `drawings` as a Python list. (the supplied program has this list in line 141)

For the selection menu, a symbol is defined for each drawing in the variable `draw_symbols`. (the supplied program has this list in line 142)

If a coordinate exceeds the reach of the pen, the program will end with error message „math domain error“.