

# Getting started morobot

Digital Factory – FH Technikum Wien



# Contents

- 1      What is required? ..... 3
- 2      Mechanical Setup ..... 3
- 3      Electrical Setup ..... 4
- 4      Programming ..... 5
  - 4.1 Installation of the Libraries ..... 5
  - 4.2 First Example Program ..... 6
  - 4.3 Upload to Arduino ..... 8

# 1 What is required?

For the successful and smooth operation of the online laboratory the following is required:

- 1x morobot
- 2x Servo-connection cable (included in the morobot)

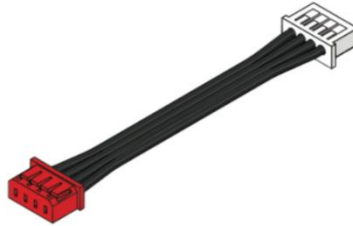


Figure 1: Servo-connection cable

- 1x Arduino <-> Servo connection cable (included in the morobot)

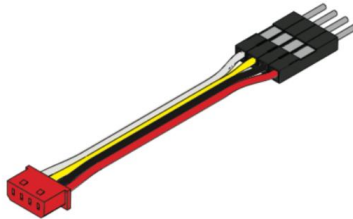


Figure 2: Arduino<->Servo connection cable

- 1x Arduino Mega (included in the morobot)
- USB <-> Arduino cable (not included)
- Mounting device for morobot (not included) (chapter 2)
  - M4 screws with sufficient length.
- Power supply 9-12V at least 3A (included in the morobot)
  - Recommended 12V 6A with a hollow plug (5.5mm/2.1mm)

## 2 Mechanical Setup

To ensure a safe use of the morobot it is necessary to mount it on a heavy object or fixed point in space. For mounting, four M4 threads are provided on the back of the morobot. The dimensions of the mounting holes are shown in the following figure.

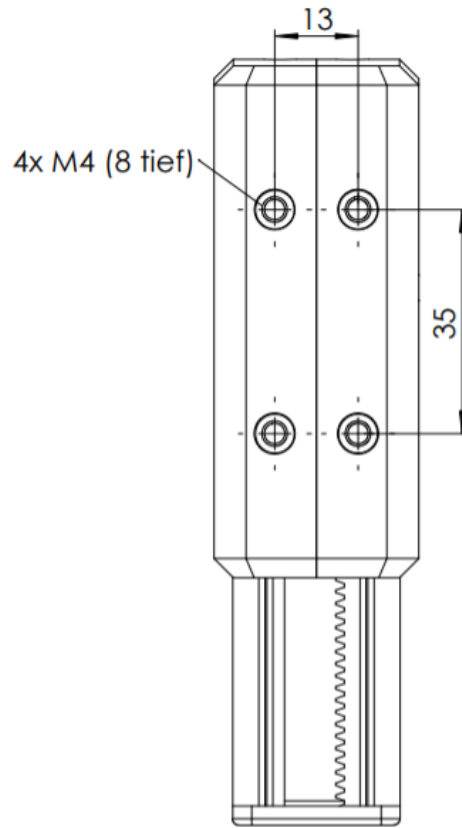


Figure 3: Mounting thread morobot

### 3 Electrical Setup

Now that the robot is fixed, the cables can be connected. The following figure shows the robot and an end effector. The robot roughly consists of the base and the following axes - in this case 3 axes. The labels of the axes are also shown in the next figure.

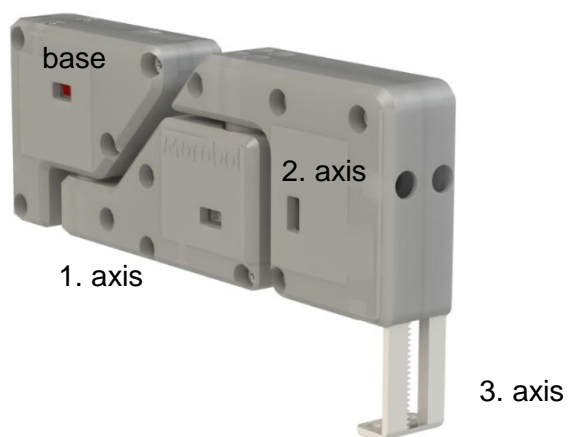


Figure 4: Labelling of the axes

A board with at least two serial interfaces (Arduino Mega, ESP32) is recommended for the control, since one is required for communication with the servo motors. The second serial interface can therefore be used for interaction with the user. A board with only one serial port (Arduino Uno) can also be used, but attention must be paid to the lack of user interaction during operation. Or this can be circumvented with a digital serial interface. Further information can be found here:

<https://www.arduino.cc/en/Reference/softwareSerial>

The following description is explained using an Arduino Mega Board as an example.

First the Arduino<->Servo cable is connected to the base. Then the base is connected to the 1st axis using the servo cable, then the 1st axis is connected to the 2nd axis and so on. (if there are more than 3 axes). Make sure that the red plugs of the cables are connected to the red sockets of the axes and base and the white plugs to the white sockets.

The base and end effector are connected to the Arduino with the individual header connectors.

1. Base
  - a. Black -> GND
  - b. Red -> VIN
  - c. White -> 17 (RX)
  - d. Yellow ->16 (TX)

Afterwards the PC is connected to the Arduino with the USB cable. Finally, the power supply is connected to the hollow connector. The additional power supply is for the servo motors, which requires 9 to 12V DC.

## 4 Programming

For the first programming of the Arduino Mega the Arduino IDE is required. This can be found at <https://www.arduino.cc/en/main/software>. This tutorial was done with the Windows Installer version on a Windows 10 PC, but all other versions should also work. After downloading and installing the Arduino software, various libraries have to be installed.

### 4.1 Installation of the Libraries

The Makeblock library is found on this link

<https://github.com/Makeblock-official/MSmartServo-Driver>

This can be downloaded using the code button and then the Download ZIP button (see Figure 5).

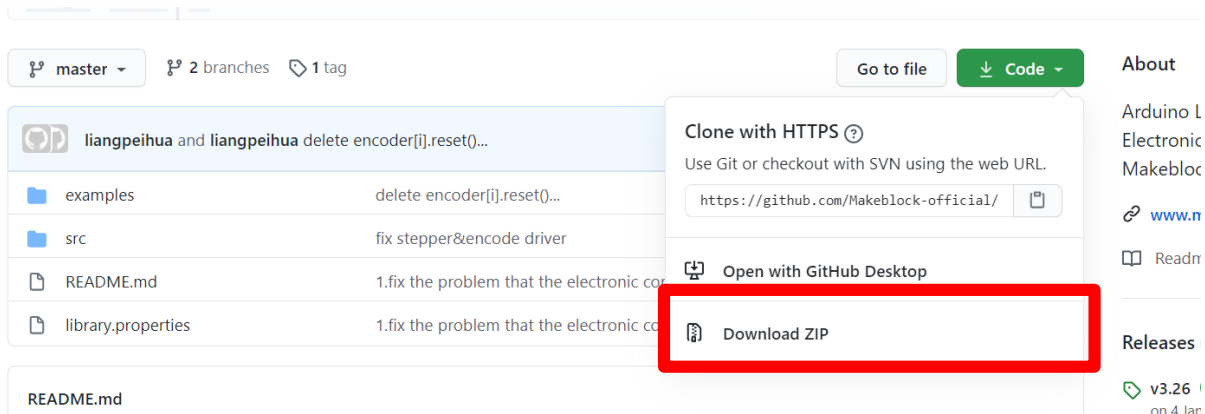


Figure 5: download Makeblock library

The downloaded ZIP file contains the folder "MSmartServo", which must be copied into the libraries folder of the Arduino installation directory. Under Windows the folder is "[Arduino-path]\libraries\", while under MacOS it is "[Arduino-path]\contents\Java\libraries\". In addition, "Manage Libraries" can be used to install "Listed Libraries" directly if any are missing.

## 4.2 First Example Program

The following code example allows the first two axes of the robot to move 20° to the left and then to the right, opening and closing the extension axis. At the end of the sequence, the built-in LED on the Arduino lights up briefly.

```

#include <SoftwareSerial.h>
#include "MeAuriga.h"

MeSmartServo MyRobot(PORT5);
Servo MyEndeffector;
uint8_t inc=0;

void setup() {
  MyEndeffector.attach(5);
  MyRobot.begin(115200);
  delay(5);
  MyRobot.assignDevIdRequest();
  delay(50);
  pinMode(LED_BUILTIN, OUTPUT);
}

void loop() {
  switch (inc)
  {
    case 1:
      MyRobot.move(1, 20, 10);
      break;
    case 2:
      MyRobot.move(2, 20, 10);
      break;
    case 3:
      MyRobot.move(1, -20, 60);
      break;
    case 4:
      MyRobot.move(2, -20, 60);
      break;
    case 5:
      MyEndeffector.write(0);
      break;
    case 6:
      MyEndeffector.write(180);
      break;
    default:
      inc = 0;
      digitalWrite(LED_BUILTIN, !digitalRead(LED_BUILTIN));
      delay(500);
      digitalWrite(LED_BUILTIN, !digitalRead(LED_BUILTIN));
      break;
  }
  delay(2000);
  inc++;
}

```

To run the example, a new sketch is created in Arduino with "File->New", insert the code and set these parameters in the drop down menu:

- Board: Arduino Mega
- Port: COM (the Arduino is connected to)

- In Windows this can be seen in the Device-Manager  
Afterwards, the code is checked with the check mark (see Figure 6).



Figure 6: Verification of the program

If no error messages appear, all libraries are present and the code is correct.

### 4.3 Upload to Arduino

To finally move the robot, the program must be transferred to the Arduino. To do this, start the upload with the arrow (see Figure 6). After the upload the implemented sketch is started.