

# PlenBot Dokumentation

## 1 Installation

Online Anleitung: <https://github.com/plenprojectcompany/PLEN2> Benötigte Software:

- Motion Installer und Repository für Bewegungen
- Control Server
- Arduino (hier sind die Treiber enthalten)

Zum Ausführen wird der Motion Editor benötigt:

<http://plen.jp/playground/motion-editor/>

## 1.1 Motion Installer und Repository

### Getting Started

---

1. Download this repository.
2. Get control boards and servo motors. [Buy all!](#)
3. Install firmware. (This is placed "/arduino/firmware/" directory.) [See also...](#)
4. Print and assemble all \*.stl files. (There are in the "/stl/" directory.)
  - [Print Details](#)
  - [Assembling Instructions](#)
5. Configure home positions of joints. (This step is done by using [PLEN Utils](#).) [See also...](#)
6. Write all motions. (Please check "/motions/\*/ " directory and use MotionInstaller.) [See also...](#)
  - In the future, this step will be included in the [PLEN Utils](#).
  - [MotionInstaller for Windows](#)
  - [MotionInstaller for OSX](#)
7. Hello, world where PLEN is!

Abbildung 1: Repository downloaden und entzippen

## 1.2 Download Motioninstaller

### Downloads

---

 [MotionInstaller\\_Win\\_v1.1.0.zip](#)

---

 [Source code](#) (zip)

---

 [Source code](#) (tar.gz)

### 1.3 Download ControlServer

## How to Use

---

1. Download latest version of the application from [here](#).
2. Unzip the downloaded file.
3. Run the `ControlServer.exe` or `ControlServer.app` .
  - If you are an OSX user, you need to run the application by following steps.
  - Click on the application's icon with `ctrl` key.
  - Choose `open` from menu items. (Using this method, you can turn off security alert tem
4. Connect your PLEN and a laptop using USB type micro B cable.
5. Start up any application to communicate with PLEN series robots. (s.a. [Motion Editor](#))

If you would like to use the application as a tuning up tool, Please [see also...](#)

### 1.4 Version auswählen

## Downloads

---

 [ControlServer\\_OSX\\_v2.3.1.zip](#)

---

 [ControlServer\\_Win\\_v2.3.1.zip](#)

---

 [Source code](#) (zip)

---

 [Source code](#) (tar.gz)

---

## 1.5 Download Arduino

### How to Install

1. Download (and install) Arduino IDE from [arduino.cc website](#). (Choose your environment, the button.)
2. Download this repository from [here](#).
3. Open the [firmware.ino](#) using the Arduino IDE.
4. Click the upload button (that is like a right arrow).

DOWNLOAD

## Download the Arduino Software



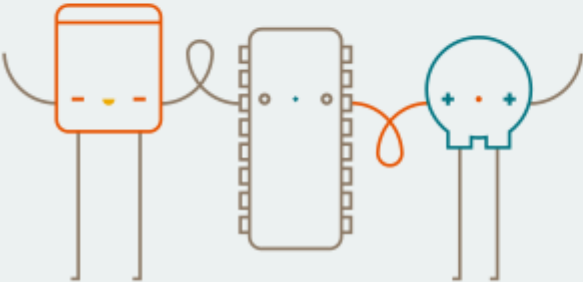
### ARDUINO 1.6.12

The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. It runs on Windows, Mac OS X, and Linux. The environment is written in Java and based on Processing and other open-source software.

This software can be used with any Arduino board. Refer to the [Getting Started](#) page for Installation instructions.

# Support the Arduino Software

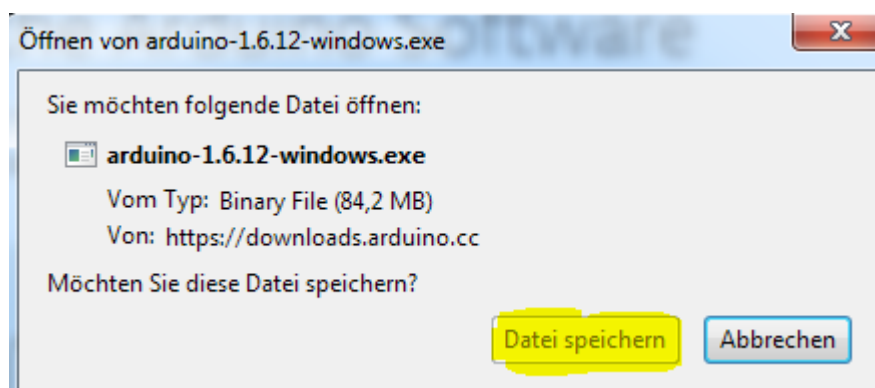
Consider supporting the Arduino Software by contributing to its development. (US tax contribution is not tax deductible). [Learn more on how your contribution will be used.](#)

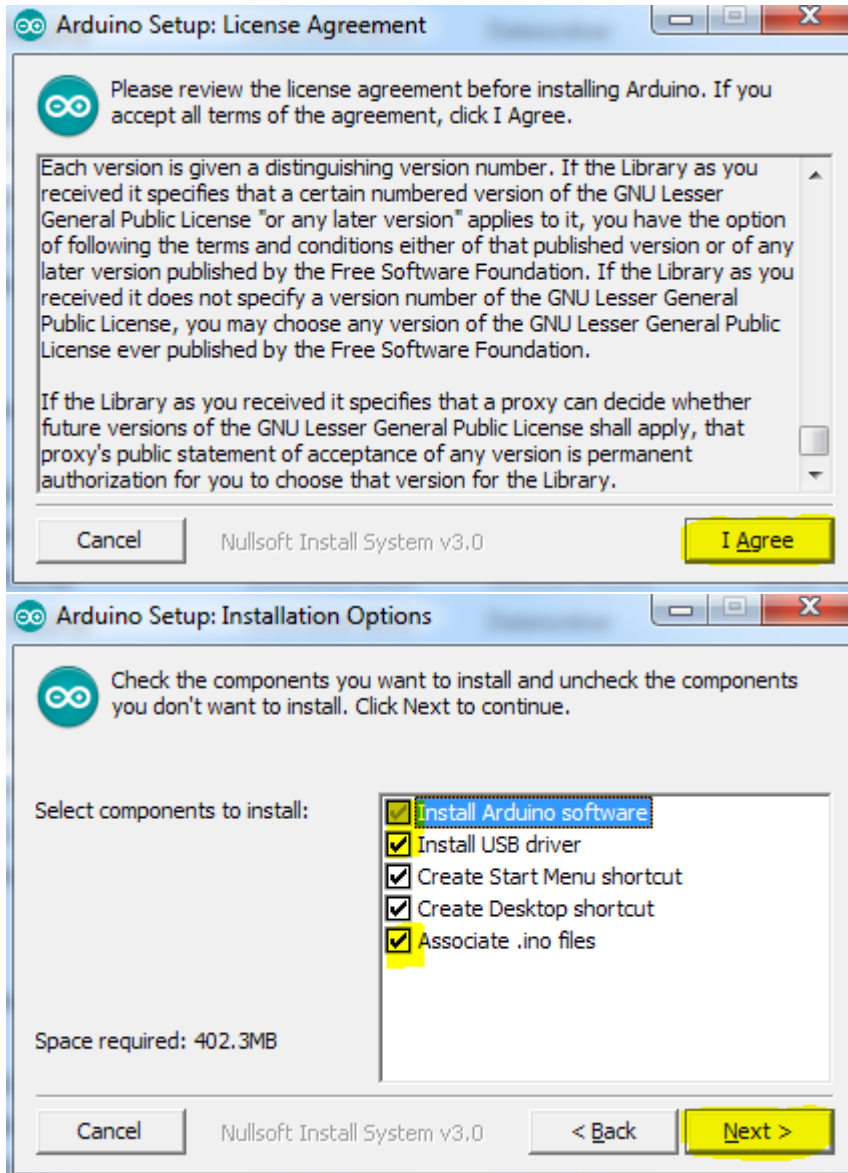


SINCE MARCH 2015, THE ARDUINO IDE HAS BEEN USED **100** TIMES. (IMPRESSIVE!) NO LONGER JUST FOR ARDUINO BOARDS, HUNDREDS OF COMPANIES AROUND THE WORLD USE IT TO PROGRAM THEIR DEVICES, INCLUDING COUNTERFEITS. HELP ACCELERATE ITS DEVELOPMENT WITH YOUR CONTRIBUTION! REMEMBER: OPEN SOURCE IS OPEN.

**\$3** **\$5** **\$10** **\$25** **\$50**

JUST DOWNLOAD





## 2 Zuruecksetzen des Plens

Aufgrund der Arbeit mit dem PlenBot kann es natuerlich auch passieren, dass man den Plen einmal softwaretechnisch außer Gefecht setzt. Dann ist ein Reset der Software des Plens notwendig. Hierfuer benoetigt man Arduino.

Schritt 1 Download des Repositories in dem sich die Firmware befindet unter:

<https://github.com/plenprojectcompany/PLEN2/tree/master/arduino>

Schritt 2 Veraendern der Build Config.h: Target Developer Edition muss auf true gesetzt werden

```
21  #define TARGET_PLEN14 false
22
23  /*!
24   @brief Configuration macro to build the firmware for PLEN2
25
26   @attention
27   This macro and TARGET_PLEN14 macro are incompatible.
28   You need to enable only one configuration.
29  */
30  #define TARGET_PLEN20 true
31
32  /*!
33   @brief Configuration macro to build the firmware for developer edition
34
35   If you enable this macro, it changes following feature(s):
36   - Serial communication speed
37  */
38  #define TARGET_DEVELOPER_EDITION false
```

Schritt 3 Firmware.ino mit Arduino starten

Schritt 4 Plen an PC anschliessen ueber USB und Upload per Arduino

Warnings und Mismatches sind hier normal, so lange in dem tuerkisen Bereich 'Done uploading' steht war der Upload erfolgreich.



```
/*
  Copyright (c) 2015,
  - Kazuyuki TAKASE - https://github.com/Guvalif
  - PLEN Project Company Inc. - https://plen.jp

  This software is released under the MIT License.
  (See also : http://opensource.org/licenses/mit-license.php)
*/

#define DEBUG false

/*!
  @note
  If you want to apply natural moving on PLEN, set the macro to "true".
  (The process is using a lot of calculation space, so the firmware becomes hard to input user action.)
*/
#define ENSOUL_PLEN2 false

#include <string.h>

#include <EEPROM.h>
#include <Wire.h>

#include "ExternaleEPROM.h"
#include "JointController.h"
#include "Motion.h"
#include "MotionController.h"
#include "Interpreter.h"
#include "Pin.h"
#include "Parser.h"
#include "Protocol.h"
#include "System.h"

#if ENSOUL_PLEN2
  #include "AccelerationGyroSensor.h"
  #include "Soul.h"
#endif

#if DEBUG
  #include "Profiler.h"
#endif

Done uploading.

sketch\Motion.cpp: In static member function 'static bool PLEN2::Motion::Header::set(uint8_t, const PLEN2::N'
```



Schritt 5 Starten des Control Servers und des lokalen Motion Editors (siehe auch Installation)

Schritt 6 Bei Load Motions alle Motions aus dem Verzeichnis hereinladen und anschliessend installieren. Der PlenBot ist dabei eingeschaltet und mit dem PC verbunden.