

**Installation**

Semesterprojekt SS 2018

Brain2Machine Interface

# Inhaltsverzeichnis

Inhaltsverzeichnis 2

1 Einleitung 3

2 Olimex Open EEG 4

2.1 Überblick der verwendeten Komponenten 4

2.2 Konfiguration 5

2.2.1 Verkabelung 5

2.2.2 USB-Treiber installieren 5

2.2.3 Gerät konfigurieren 5

2.3 Anbringen der Elektroden am Kopf 6

2.4 Testen des EEGs 8

3 EEG-Aufzeichnung mit Matlab 9

3.1 Überblick der verwendeten Komponenten 9

3.2 Installation von Psychtoolbox 9

3.3 Matlab Skripts zum Aufzeichnen von EEG-Daten 10

3.3.1 readserialio 10

3.3.2 writebrain 10

3.3.3 readbrain 11

4 Lego Mindstorms Greifarm Steuerung in Java 12

4.1 Überblick der verwendeten Komponenten 12

4.2 Einrichten der Entwicklungsumgebung 13

4.2.1 Java installieren 13

4.2.2 Installation der leJOS Tools 13

4.2.3 Eclipse aufsetzen 13

4.3 Vorbereitung des EV3 Bricks 15

4.3.1 Formatieren der MicroSD Karte 15

4.3.2 Erstellen der leJOS-fähigen MicroSD Karte 15

4.3.3 MicroSD Karte an EV3 Brick anschließen 16

4.4 Herstellen einer Bluetooth-Verbindung von Rechner zu EV3 Brick 16

4.5 Lego Mindstorms Greifarm Steuerung 17

4.5.1 Klassendiagramm 17

4.5.2 Einrichten der RXTX Library in Eclipse 17

4.5.3 Funktionsweise 18

4.5.4 Erstellen einer ausführbaren Datei 18

# Einleitung

Dieses Dokument dient dazu, projektfremden Personen das Aufsetzen der Entwicklungsumgebung für das Projekt Brain2Machine Interface zu ermöglichen.

Das Projekt besteht aus drei Hauptbereichen:

* **Olimex Open EEG**: als Grundvoraussetzung für die nachfolgenden Bereiche ist das Olimex Open EEG vorzubereiten und zu konfigurieren
* **EEG-Aufzeichnung mit Matlab**: mithilfe von erstellten Matlab-Skripts werden EEG-Daten eingelesen, gespeichert und ausgewertet
* **Lego Mindstorms Greifarm-Steuerung in Java**: eine eigens entwickelte Software, das EEG-Daten liest, Gesten daraus herausfiltert und einen Lego Mindstorms Greifarm mit leJOS steuert

Genaueres zur Systemarchitektur finden Sie im Dokument /Documentation/TechnicalDocumentation/**SystemArchitecture.docx**.

Zur Entwicklung des sind einige Werkzeuge notwendig, um Source-Code zu verfassen, testen sowie eine Release-Version zu erstellen.

Systemvoraussetzung für dieses Projekt ist **Windows 10**.

# Olimex Open EEG

## Überblick der verwendeten Komponenten

|  |  |
| --- | --- |
| **Komponente** | **Details** |
| **Olimex EEG-SMT** | <https://www.olimex.com/Products/EEG/OpenEEG/EEG-SMT/open-source-hardware> |
| **4x EEG-AE (Aktive Elektrode)** | <https://www.olimex.com/Products/EEG/Electrodes/EEG-AE/open-source-hardware> |
| **1x EEG-PE (Passive Elektrode)** | <https://www.olimex.com/Products/EEG/Electrodes/EEG-PE/open-source-hardware> |
| **USB-A-B-Kabel** | <https://www.olimex.com/Products/Components/Cables/USB-A-B-CABLE/>  zur Verbindung des EEGs mit dem PC |
| **Virtual COM Port Driver** | <http://www.ftdichip.com/Drivers/VCP.htm> |
| **Electric Guru** | <https://realization.org/downloads/ElecGuru40.zip>  optional  empfehlenswert, um nach dem initialen Konfigurieren des EEGs das Gerät auf seine korrekte Funktionsweise zu testen |

## Konfiguration

### Verkabelung

Zu Beginn werden alle Elektroden sowie das USB-Kabel am EEG (und PC) angeschlossen.

EEG AE



USB-A-B

PC USB Port

EEG PE

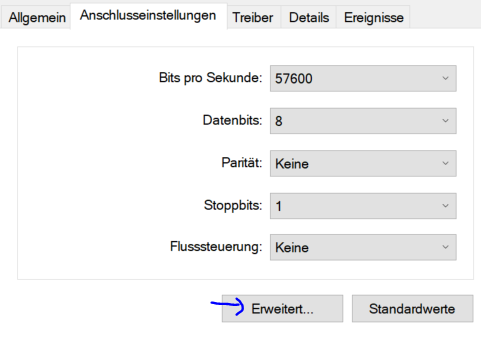
### USB-Treiber installieren

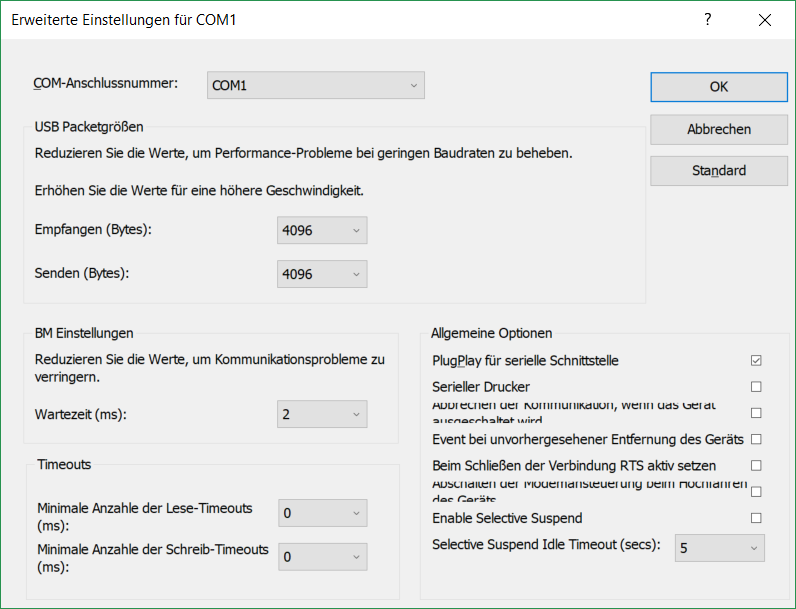
Damit das EEG verwendet werden kann, muss der Virtual COM Port Driver installiert werden. Folgende Vorgehensweise ist dazu notwendig:

1. Öffnen des Gerätemanagers; „andere Geräte“. Beim Gerät OLIMEX „Treiber aktualisieren“ auswählen und „Meinen PC nach Treibern durchsuchen“.
2. „Aus einer Liste verfügbarer Treiber auf meinem Computer auswählen“
3. Mittels „Datenträger“ und „Durchsuchen die Datei „ftdibus.inf“ vom entpackten Virtual COM Port ausgewählt.

### Gerät konfigurieren

Weiters wird im Gerätemanager beim OLIMEX-Gerät folgende Eigenschaften getroffen:

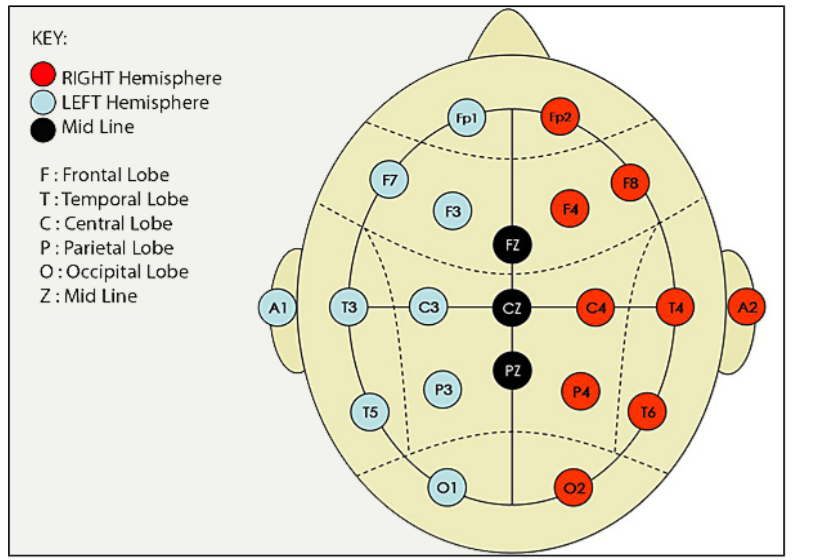




## Anbringen der Elektroden am Kopf

Wichtig für die Messung ist, dass die Elektroden so kompakt wie möglich am Kopf angebracht werden.

Die Reihenfolge und Positionierung der Elektroden ist wie folgt:



**CH2+**

**CH1+**

**CH1-**

**CH2-**

**DRL**



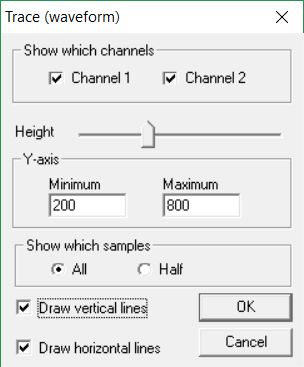
Als Hilfsmittel zur Anbringung der aktiven Elektroden (CH1, CH2) am Kopf wurde im Projektverlauf eine **Badehaube** verwendet. Die passive Elektrode (DRL) wurde mit einer Wäschekluppe am linken Ohr angeklemmt.



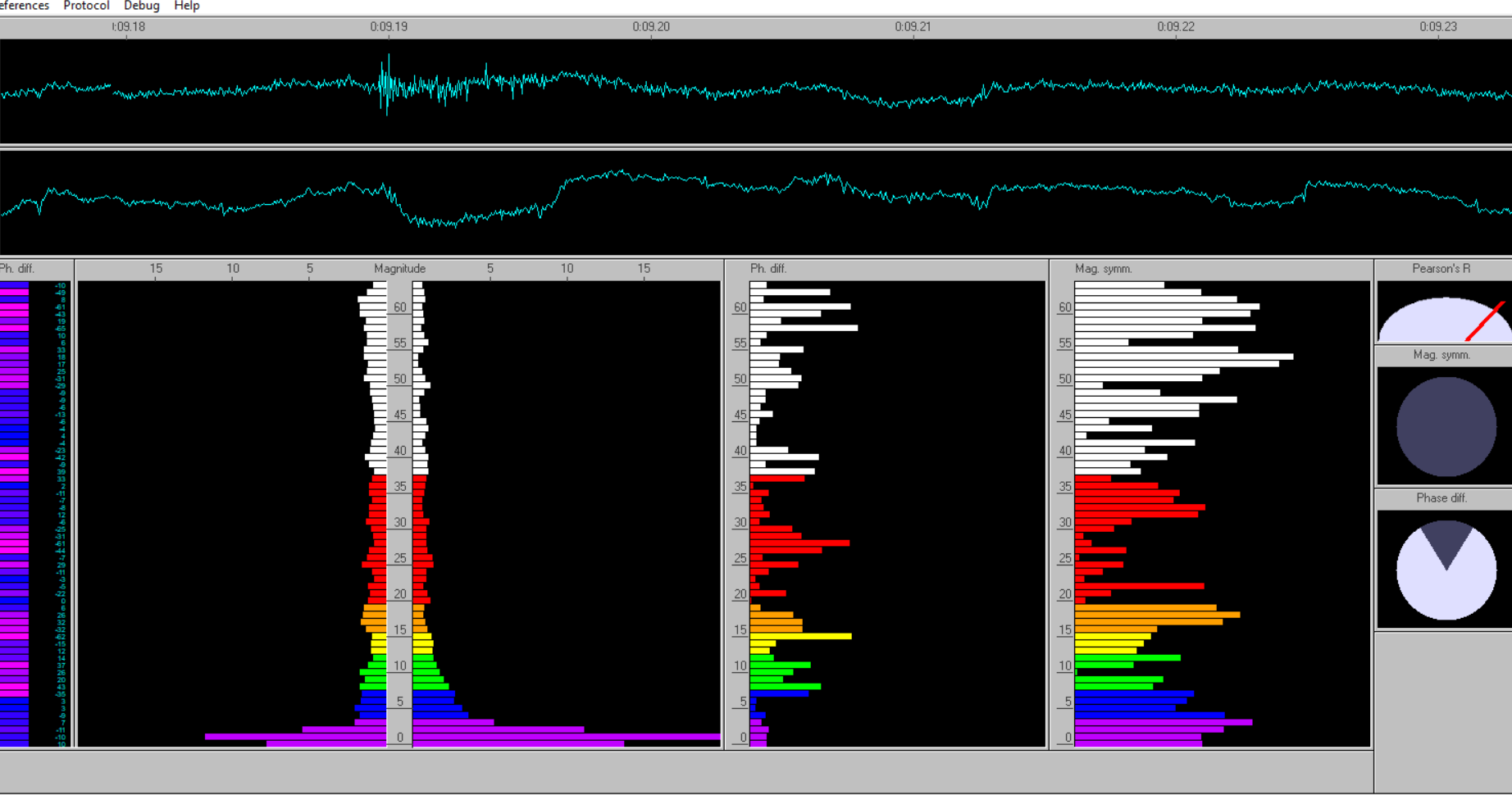
## Testen des EEGs

Um die Funktionsfähigkeit und Einsatzbereitschaft des EEGs zu testen, wird das Gerät vorerst mit Electric Guru getestet.

1. Dazu muss Electric Guru von obenstehendem Link heruntergeladen und entpackt werden.
2. Die Datei „ElectGuru.exe“ ausführen.
3. Unter Preferences > Trace folgende Konfiguration vornehmen:



1. Weiters ist auch sicherzustellen, dass unter Preferences > Serial port der richtige COM-Port konfiguriert ist (siehe Gerätemanager 🡪 OLIMEX).
2. Auf der linken Seite kann mit dem obersten Icon („Start EEG“) die Aufzeichnung gestartet werden. Das Resultat sollte etwa untenstehender Abbildung entsprechen:



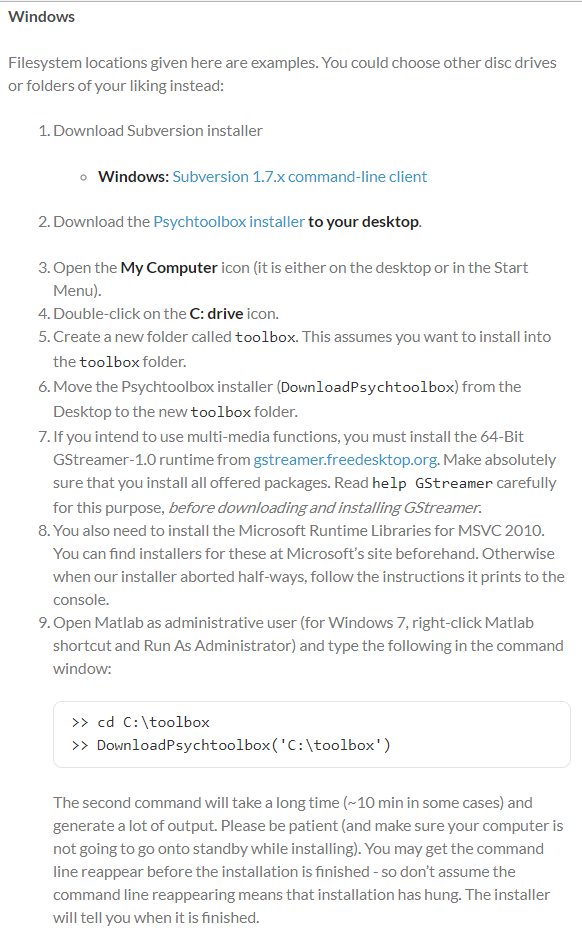
# EEG-Aufzeichnung mit Matlab

## Überblick der verwendeten Komponenten

|  |  |
| --- | --- |
| **Software und Version** | **Details** |
| **Matlab (am besten aktuellste Version)** | <https://de.mathworks.com/downloads/> |
| **Psychtoolbox 3** | <http://psychtoolbox.org/download/#Windows>  liefert wichtige Funktionen für Matlab, um Daten über serielle Ports auszulesen |

## Installation von Psychtoolbox

Nachdem Matlab installiert wurde, folgen Sie am besten den Instruktionen auf der Psychtoolbox Website <http://psychtoolbox.org/download/#Windows>.



## Matlab Skripts zum Aufzeichnen von EEG-Daten

Vorweg basieren die erstellten Matlab-Files auf der Anleitung dieses Blog-Eintrags: <https://bakerdh.wordpress.com/2013/01/31/a-first-look-at-the-olimex-eeg-smt/>

Die erstellten Matlab-Files sind eine wichtige Grundlage für die Analyse der EEG-Daten. Mithilfe der erfassten Aufzeichnungen wurden anschließend die Parameter für das Java-Programm im Punkt 4 Lego Mindstorms Greifarm Steuerung in Java getroffen.

Folgende Matlab Prozeduren wurden vom Projektteam zur Aufzeichnung und Auswertung von EEG-Daten mit Matlab geschrieben:

### readserialio

Funktion

Implementation\SourceCode\matlab\readserialio.m

Diese Funktion kümmert sich um das Einlesen der EEG-Daten anhand der Eingabeparameter „portSpec“ (COM-Port, zB „COM1“) und „secondsToRead“ (gewünschte Dauer der Aufzeichnung in Sekunden).

Dabei werden die empfangenen Rohdaten entsprechend aufbereitet in folgender Form:

Number[6][n]:

0.50342,0.5044,0.49756,0.50049,0.50147, ...

0.5044,0.5044,0.48974,0.49267,0.49267,0.51222, ...

...

Die 6 Zeilen stehen dabei für die von der Firmware unterstützten maximalen Anzahl von 6 Kanälen (in diesem Projekt sind nur die ersten beiden Kanäle relevant).

„n“ ergibt sich aus dem Lesevorgang und spiegelt die Länge des EEG-Datenstroms mit der gewünschten Dauer („secondsToRead“) wider.

Jeder Wert liegt in einem Wertebereich von 0 bis 1.

### writebrain

Skript

Implementation\SourceCode\matlab\writebrain.m

Dieses Skript schreibt EEG-Daten in der Form, wie oben bei „readserialio“ beschrieben, auf eine Datei. Dabei werden Daten von 10 Sekunden erfasst.

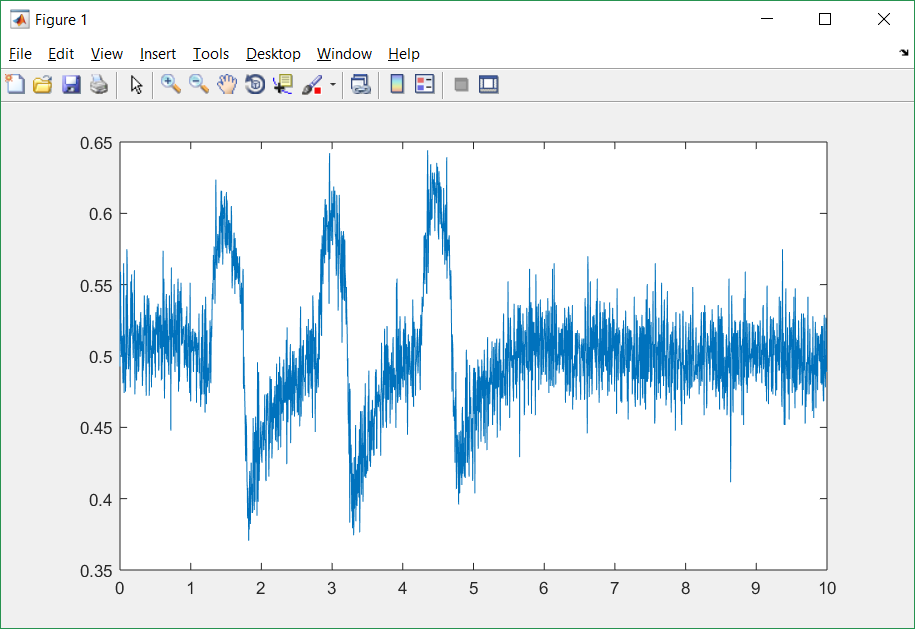
### readbrain

Skript

Implementation\SourceCode\matlab\readbrain.m

Hierbei werden vorhin von „writebrain“ geöffnete Daten eingelesen und grafisch dargestellt.

Beispiel für einen Signalverlauf auf Channel 1 bei dreimaligem Nach-links-Sehen:



# Lego Mindstorms Greifarm Steuerung in Java

## Überblick der verwendeten Komponenten

|  |  |
| --- | --- |
| **Komponente** | **Details** |
| **LEGO Mindstorms Bausatz** | <https://www.lego.com/en-us/mindstorms/build-a-robot> |
| **LEGO Mindstorms EV3 Brick**  https://sh-s7-live-s.legocdn.com/is/image/LEGO/45500?$PDPDefault$ | <https://shop.lego.com/de-AT/Intelligenter-EV3-Stein-45500>  Lego Mindstorms Brick, der verwendet wird, um die Motoren anzusteuern; Brick ist in jedem Lego Mindstorms Bausatz enthalten. |
| **MicroSD Card** | 2GB bis 32GB im FAT32-Format; SDXC nicht unterstützt! |
| **MicroSD Card Reader** | Notwendig ist jedenfalls ein Adapter, mit dem die MircoSD Karte schließlich an den Entwicklungsrechner angeschlossen werden kann. |
| **SD Memory Card Formatter** | <https://www.sdcard.org/downloads/formatter_4/> |
| **Java JDK 1.8 32 bit** | [http://www.oracle.com/technetwork/java/ javase/downloads/jdk8-downloads-2133151.html](http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html)  Zur Ausführung neuester Eclipse-Versionen ist Java 8 empfehlenswert. |
| **Eclipse IDE für Java Developers** | <https://www.eclipse.org/downloads/> |
| **leJOS Plugin for Eclipse** | siehe Eclipse Market Place |
| **leJOS EV3 0.9.1-beta (EXE) Tools** | <https://sourceforge.net/projects/ev3.lejos.p/files/> |
| **JRE für Lego Mindstorms** | <http://www.oracle.com/technetwork/java/embedded/downloads/javase/javaseemeddedev3-1982511.html> |
| **RXTX Library** | <http://rxtx.qbang.org/wiki/index.php/Download>  Java Library für die Kommunikation über eine serielle Schnittstelle |
| **Commons Math Library** | <http://commons.apache.org/proper/commons-math/download_math.cgi>  Java Library für erweiterte mathematische Funktionen wie zB Fourier, Regression, usw. |

## Einrichten der Entwicklungsumgebung

### Java installieren

Folgende Java-Tools müssen installiert werden:

* **JDK 1.8 32 bit** zur Ausführung von Eclipse und Ausführen der  
   Greifarm-Applikation

**Hinweis:** Die Verwendung von 64-Bit Java hat bei der  
 Verwendung der Library von RXTX für Fehler gesorgt.  
 Stattdessen wird also die 32-Bit Version gewählt.

### Installation der leJOS Tools

Damit ausführbare Programme für den EV3 Brick entwickelt werden können, sind die leJOS Tools notwendig. Durch Installation der Tools (C:\Program Files\leJOS EV3) werden

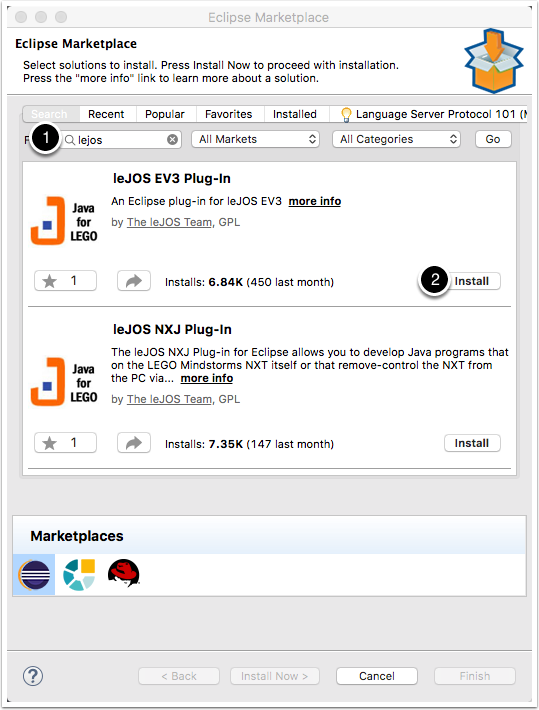
* Java Libraries
* sowie das leJOS Betriebssystem für den Brick

bereitgestellt.

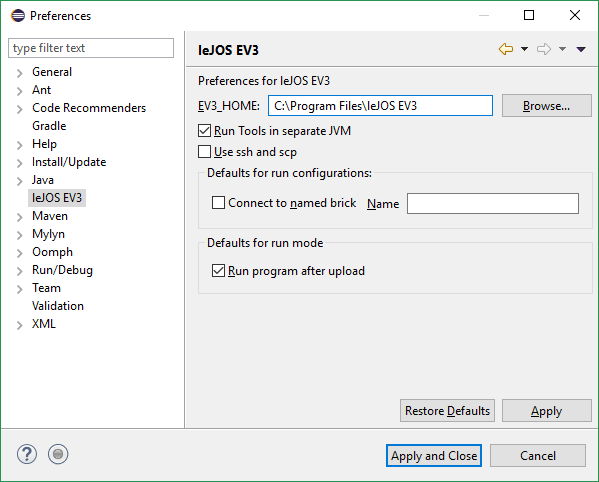
### Eclipse aufsetzen

Entsprechend der obigen Auflistung wird Eclipse für Java Developers auf dem Entwicklungsrechner installiert.

Weiters dient das leJOS Eclipse Plugin später zur Installation von leJOS Projekten auf dem Brick. Dazu wird folgendes Paket für Eclipse installiert:



Unter Window>Preferences können die Settings für leJOS nochmals überprüft werden.



## Vorbereitung des EV3 Bricks

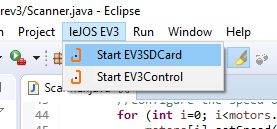
### Formatieren der MicroSD Karte

Zu Beginn wird die MicroSD Karte mit passendem Adapter mit dem Entwicklungsrechner verbunden.

Mithilfe des Programms SD Memory Card Formatter (siehe oben) wird die MicroSD Karte mit FAT32 formatiert.

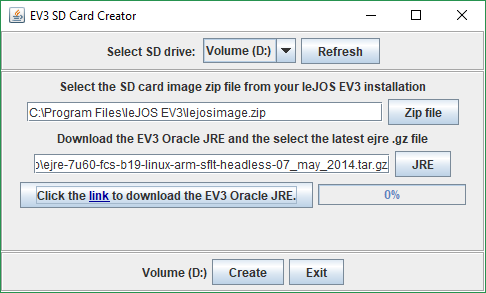
### Erstellen der leJOS-fähigen MicroSD Karte

Von Eclipse aus wird der leJOS SD-Card Creator gestartet.



Drei Dinge werden dabei festgelegt:

1. Auswahl des Laufwerkbuchstabens der MicroSD Karte
2. Pfad zum leJOS Image (standardmäßig: C:\Program Files\leJOS EV3\lejosimage.zip)
3. Pfad zum JRE für Mindstorms (zB C:\...\Downloads\ ejre-7u60-fcs-b19-linux-arm-sflt-headless-07\_may\_2014.tar.gz)



Mit „Create“ wird die Vorbereitung der MicroSD Karte abgeschlossen.

### MicroSD Karte an EV3 Brick anschließen

Die mit leJOS ausgestattete MicroSD Karte wird an den Lego Mindstorms Brick angeschlossen.

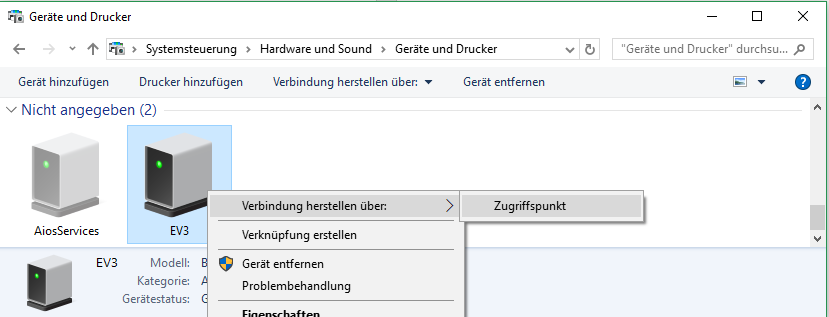


Wird der Brick anschließend gestartet (mittlerer Knopf), wird von MicroSD Karte gebootet und leJOS gestartet.

## Herstellen einer Bluetooth-Verbindung von Rechner zu EV3 Brick

Als ersten Schritt muss vom PC ein Pairing mit dem Brick als Bluetooth-Gerät hergestellt werden.

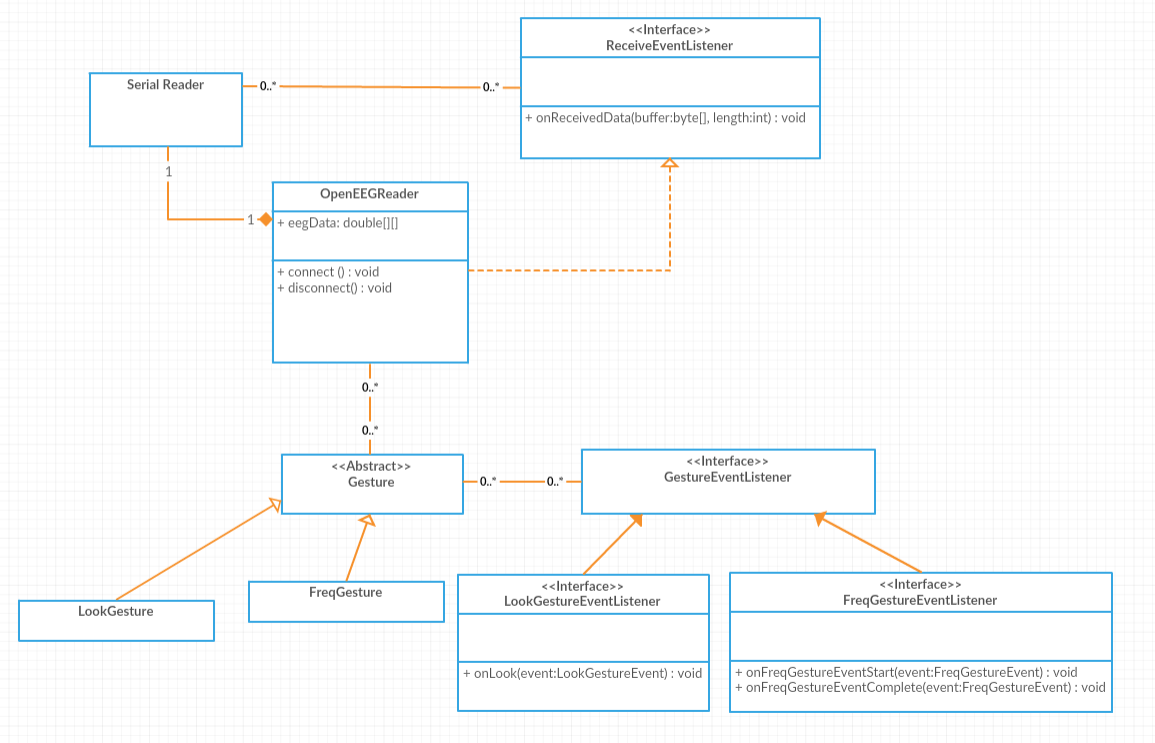
1. Am EV3 Brick wird im Bluetooth Menü die Sichtbarkeit eingestellt.
2. In Windows 10 kann nun der Brick unter „Bluetooth- und andere Geräte“ gefunden werden (Bluetooth- oder ein anderes Gerät hinzufügen).
3. Sobald der Brick gefunden ist (Name: „EV3“) wird die Kopplung mit dem Pin des Bricks bestätigt. Dieser lautet standardmäßig „1234“.
4. Als letzten Schritt muss ein PAN mit dem Brick hergestellt werden (ähnlich wie ein Wireless Netzwerk), damit eine TCP/IP Kommunikation zwischen Rechner und Brick erfolgen kann. Die Vorgehensweise ist wie folgt im Screenshot:



## Lego Mindstorms Greifarm Steuerung

### Klassendiagramm

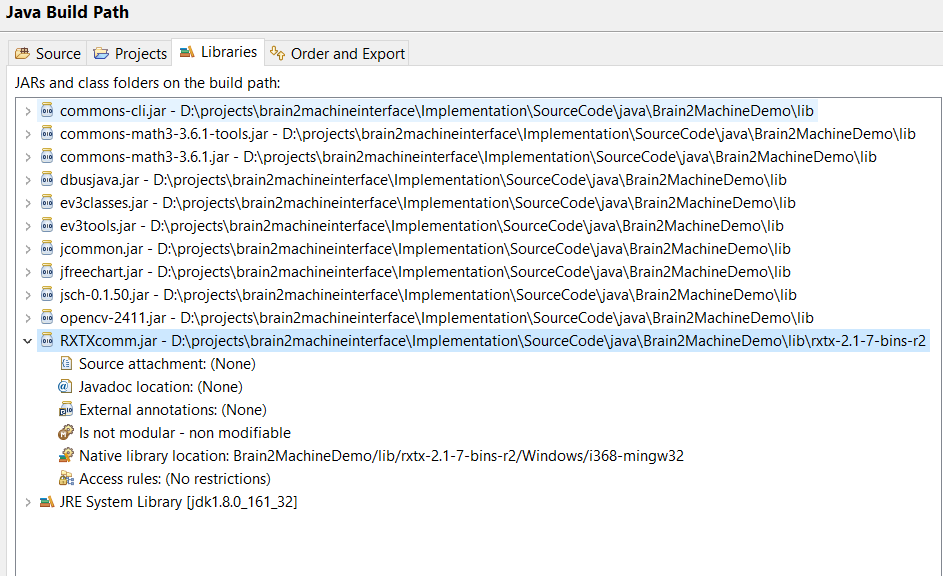
Zur Abwicklung der Verarbeitung und Auswertung der EEG-Daten für die Gestensteuerung wurde folgendes Konzept erstellt:



### Einrichten der RXTX Library in Eclipse

Da die RTXT auf eine native Library zugreift, muss dies bei der Einbindung der RXTXComm.jar berücksichtigt werden.

Dazu muss unter Projekt > Properties > Java Build Path folgende Vorkehrung getroffen werden:



Dies setzt auch eine sinnvolle Strukturierung (wie den Pfaden oben zu entnehmen) des lib-Ordners voraus.

### Funktionsweise

Näheres zur Funktionsweise und der Verwendung der restlichen Libraries finden Sie in der JavaDoc direkt im Programmcode im Ordner Implementation\SourceCode\java\Brain2MachineDemo\src.

### Erstellen einer ausführbaren Datei

Folgende Schritte sind zum Exportieren einer Release-Version notwendig:

1. Projekt > Export > Runnable JAR > Extract required libraries into generated JAR; Name: „Brain2MachineInterface.jar“
2. Nachdem das Projekt auf eine native Library von RXTX verweist, muss in späterer Folge die JAR Datei mit einem zusätzlichen Parameter in der Kommandozeile ausgeführt werden. Ein Beispiel befindet sich in der Datei Implementation\SourceCode\java\Brain2MachineDemo\Brain2MachineInterface.cmd

java -Djava.library.path="<Path to rxtx-2.1-7-bins-r2/Windows/i368-mingw32>" -jar Brain2MachineInterface.jar

1. Bei der Ausführung ist sicherzustellen, dass ein 32-Bit Java verwendet wird, der Pfad zur Native Library korrekt ist und der Lego Mindstorms Brick per Bluetooth Zugriffspunkt verbunden ist.