

Manuelle Installation und Konfiguration von leJOS

Anleitung für die Manuelle Installation und Konfiguration von leJOS für die Programmierung von Lego-Mindstorm Roboter mit Java und Linux.



www.informatics4kids.de

Inhaltsverzeichnis

1 Allgemeines.....	2
1.1 Nach dem Auspacken... ..	2
1.2 Voraussetzungen für die korrekte Installation von leJOS unter OpenSuse.....	3
1.3 Wie gehen wir vor.....	3
2 Voraussetzungen Überprüfen.....	4
2.1 Überprüfe alle notwendigen Tools installiert sind.	4
3 leJOS installieren.....	9
3.1 Übersetzen der Software.....	10
3.2 Den Bibliothekpfad (Library Path) setzen:.....	11
3.3 Setzen der usb rechte.....	12
4 leJOS NXJ Eclipse Konfiguration.....	12
4.1 leJOS Compiler.....	13
4.2 leJOS Uploader.....	14
4.3 External Tools Menu.....	15
5 Eclipse-Plugin.....	15
5.1 Installation des Plugins.....	15
5.2 Konfiguration des Eclipse leJOS Plugin.....	18
6 Die Firmware updaten.....	19
6.1 Mindstorm für das Flaschen vorbereiten.....	19
6.2 Flaschen.....	20
7 Anhang	21
7.1 A: Quellen.....	21
7.2 B:Über dieses Dokument.....	21

1 Allgemeines

Bevor man mit der Installation von **leJOS** und damit mit dem Java programmieren des *Lego-Mindstorm* Roboter beginnen kann, muss überprüft werden ob die Voraussetzungen dafür erfüllt sind. Ist dies nicht der Fall, sollte man erst dafür sorgen das die Voraussetzungen erfüllt sind bevor man weiter liest.

1.1 Nach dem Auspacken...

Nachdem das Verzeichnis leJOS-1.0.zip ausgepackt wurde ist folgende Verzeichnisstruktur entstanden:



Abbildung 1: Verzeichnisstruktur nachdem die Datei leJOS-1.0.zip entpackt wurde

Bedeutung:

- 97-lego.rules: **Konfigurationsdatei**
- lejos_NXJ_0_8_5beta.tar.gz: **Lego Software die es zu installieren gilt :-)**
- ManuelleInstallationKonfigurationleJOS.pdf: **Anleitung**

1.2 Voraussetzungen für die korrekte Installation von leJOS unter OpenSuse

1. **Eclipse** muss installiert und lauffähig sein. Das **SunJava SDK 1.6** sollte installiert sind. (siehe Artikel: *Installation von Eclipse*)
2. Überprüfe ob das Tool **ant** und die Bibliotheken **libusb-1_0-devel**, **libusb-compat-devel** installiert sind.
3. Überprüfe ob **gcc**, **gcc-c++** und **bluez-devel** installiert sind.
4. Zum installieren der leJOS Software werden Rootrechte benötigt

1.3 Wie gehen wir vor

1. Die leJOS Software wird installiert und konfiguriert.
2. Die Eclipse Entwicklungsumgebung wird für die Verwendung von leJOS eingerichtet.
3. Die leJOS Firmware auf dem Lego-Mindstorm Roboter übertragen (geflasht).

Alle Ausgaben von der Konsole oder Quelltexte sind eingerahmt, es wird nicht empfohlen diese Texte abzuschreiben. Man kann Sie z.B. mit KPWF in die Zwischenablage kopieren und als Datei abspeichern.

TIPP

2 Voraussetzungen Überprüfen

2.1 Überprüfe alle notwendigen Tools installiert sind.

1. Suche und öffne das Menü **YaST (Systemeinstellungen)**.
2. Gehe dann zum Eintrag **Software** und dort zu **Software installieren oder löschen**.
3. **Falls die Kontrollkästchen ant, libusb-1_0_devel, libusb-compat-devel, gcc, gcc-c++ und bluez-devel nicht aktiviert ist, aktiviere sie und drücke Akzeptieren**, damit sie installiert werden. Wenn sie bereits installiert sind drücke auf Abbrechen und verlasse YaST. Alle Voraussetzungen für die leJOS Installation sind jetzt erfüllt.

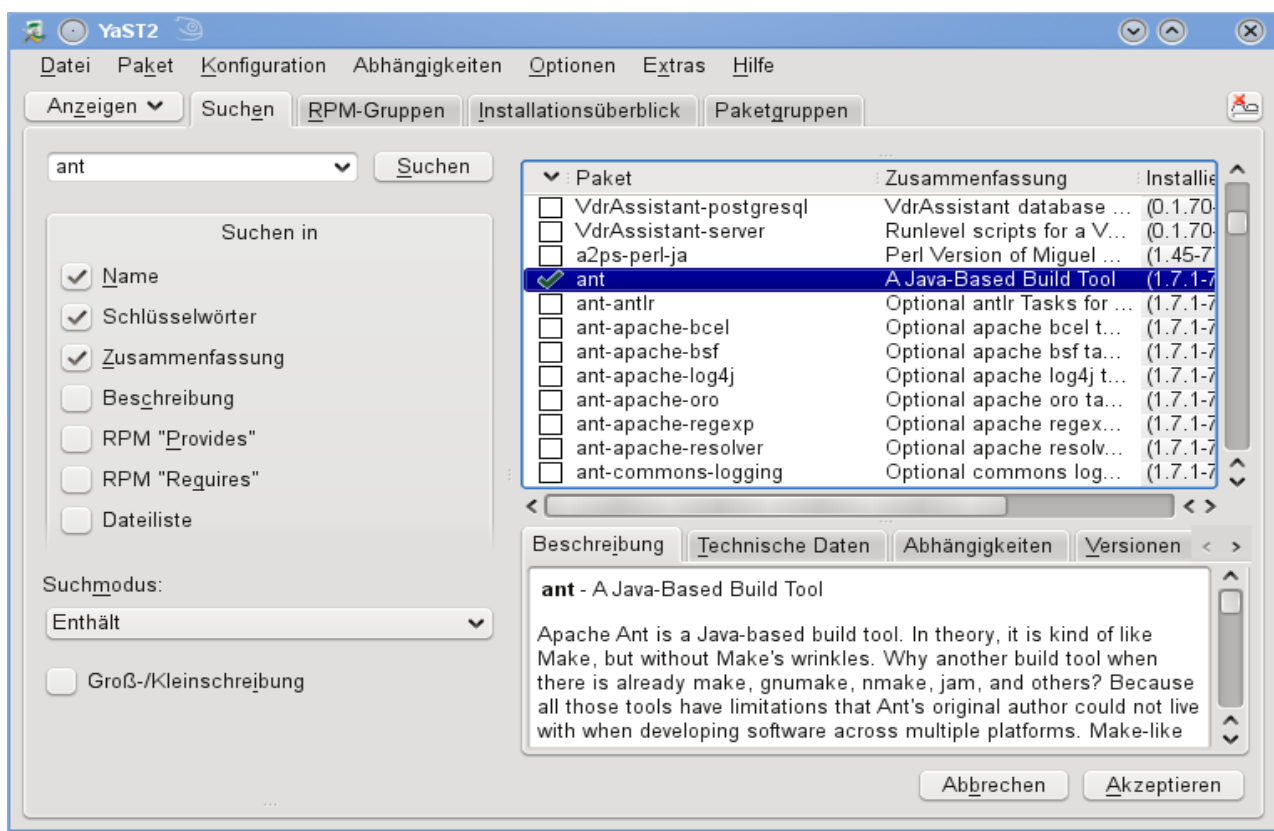


Abbildung 2: Das Tool **ant** wird für das übersetzen von **leJOS** benötigt.

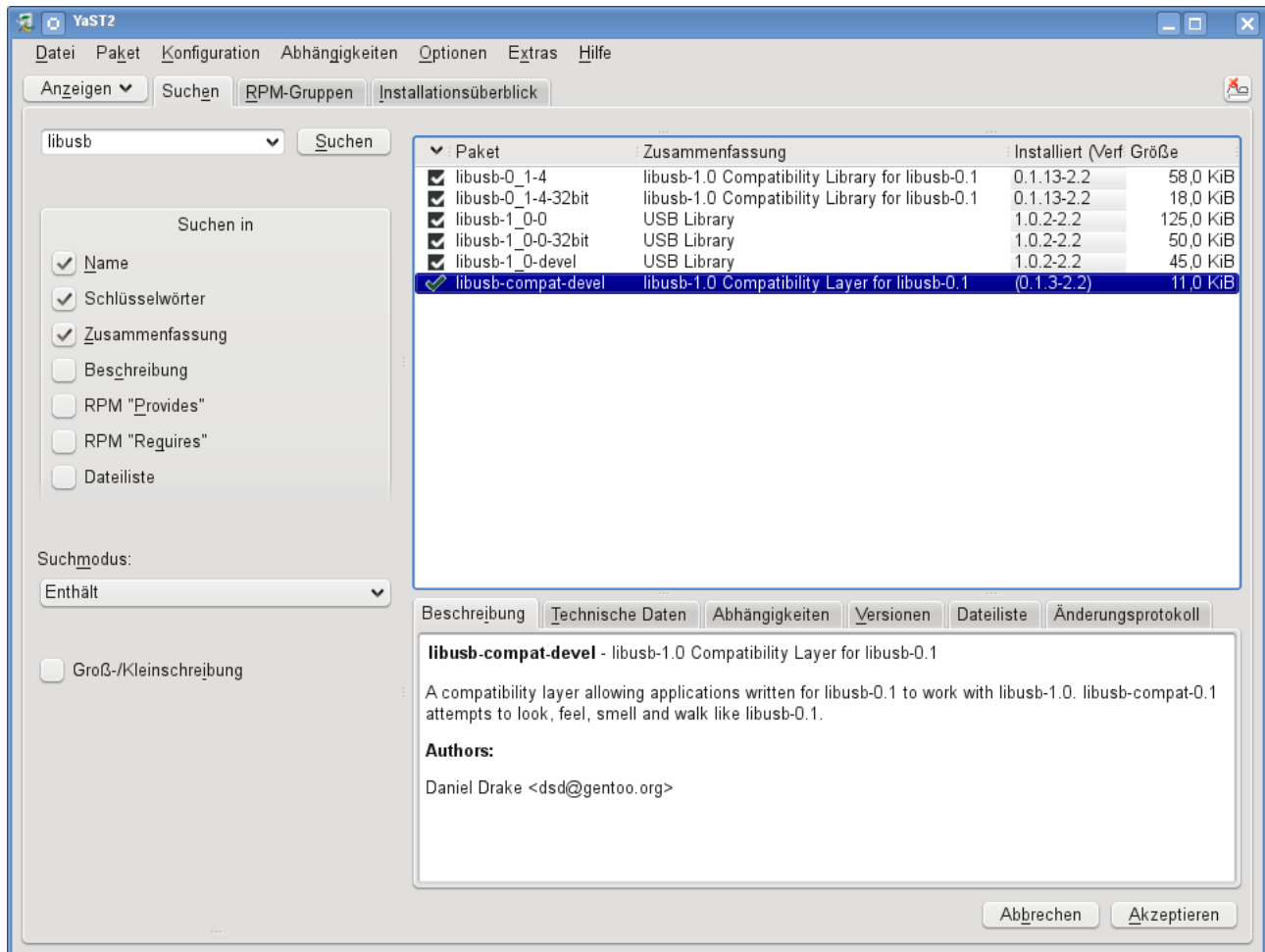


Abbildung 3: Überprüfe ob **libusb-1_0-devel** und **libusb-compat-devel** installiert sind.

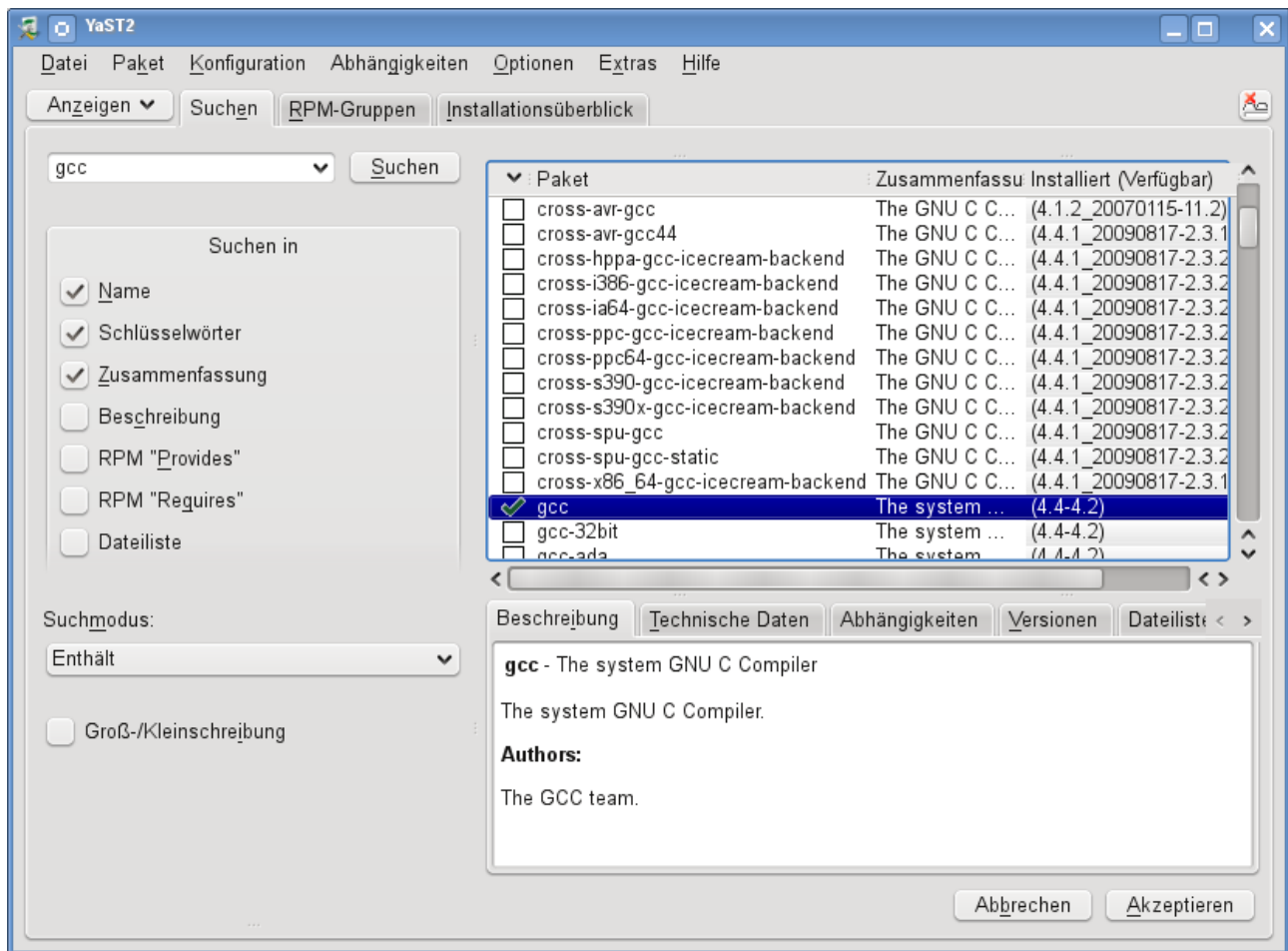


Abbildung 4: Überprüfe ob der gcc Compiler installiert ist.

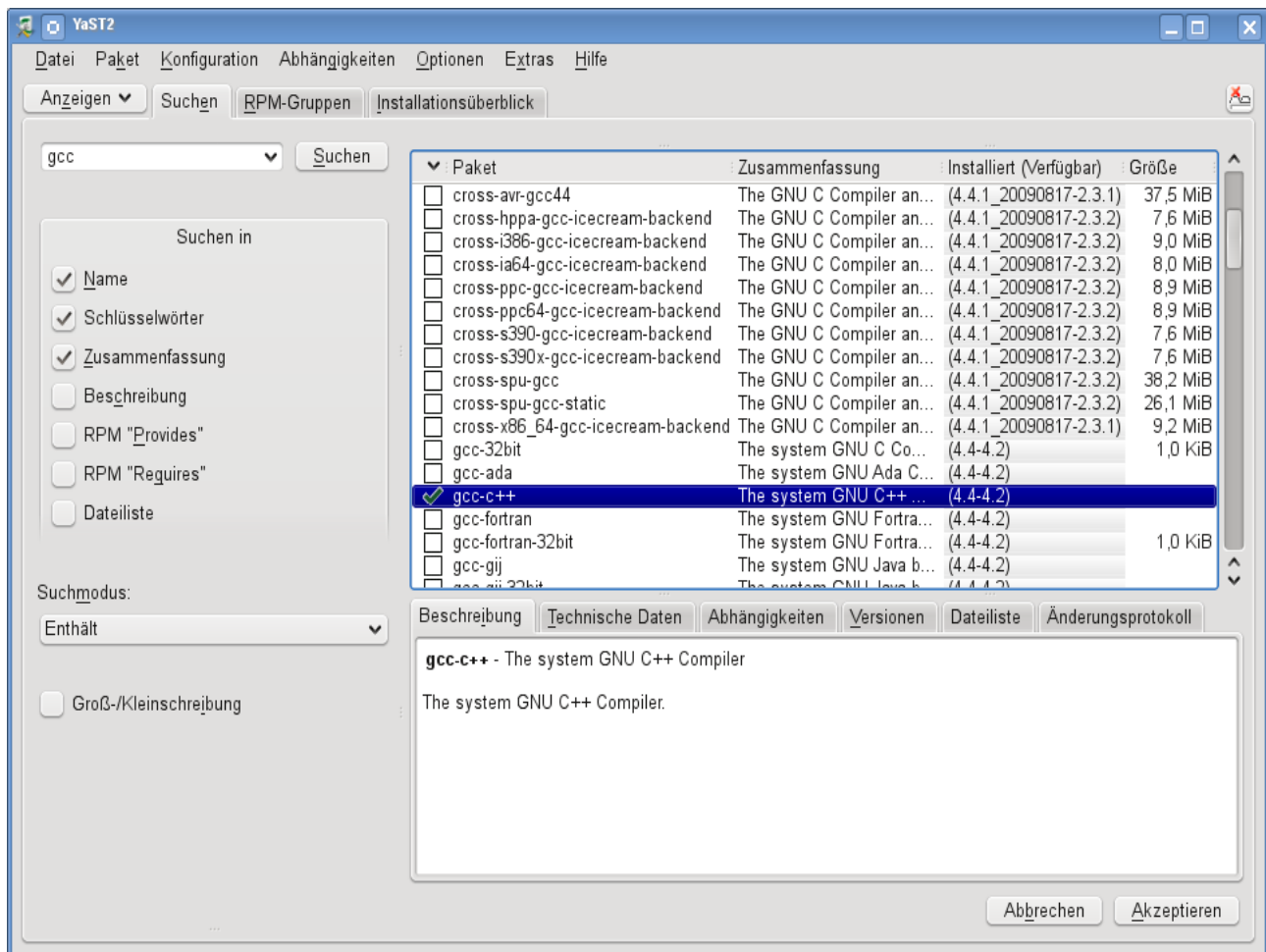
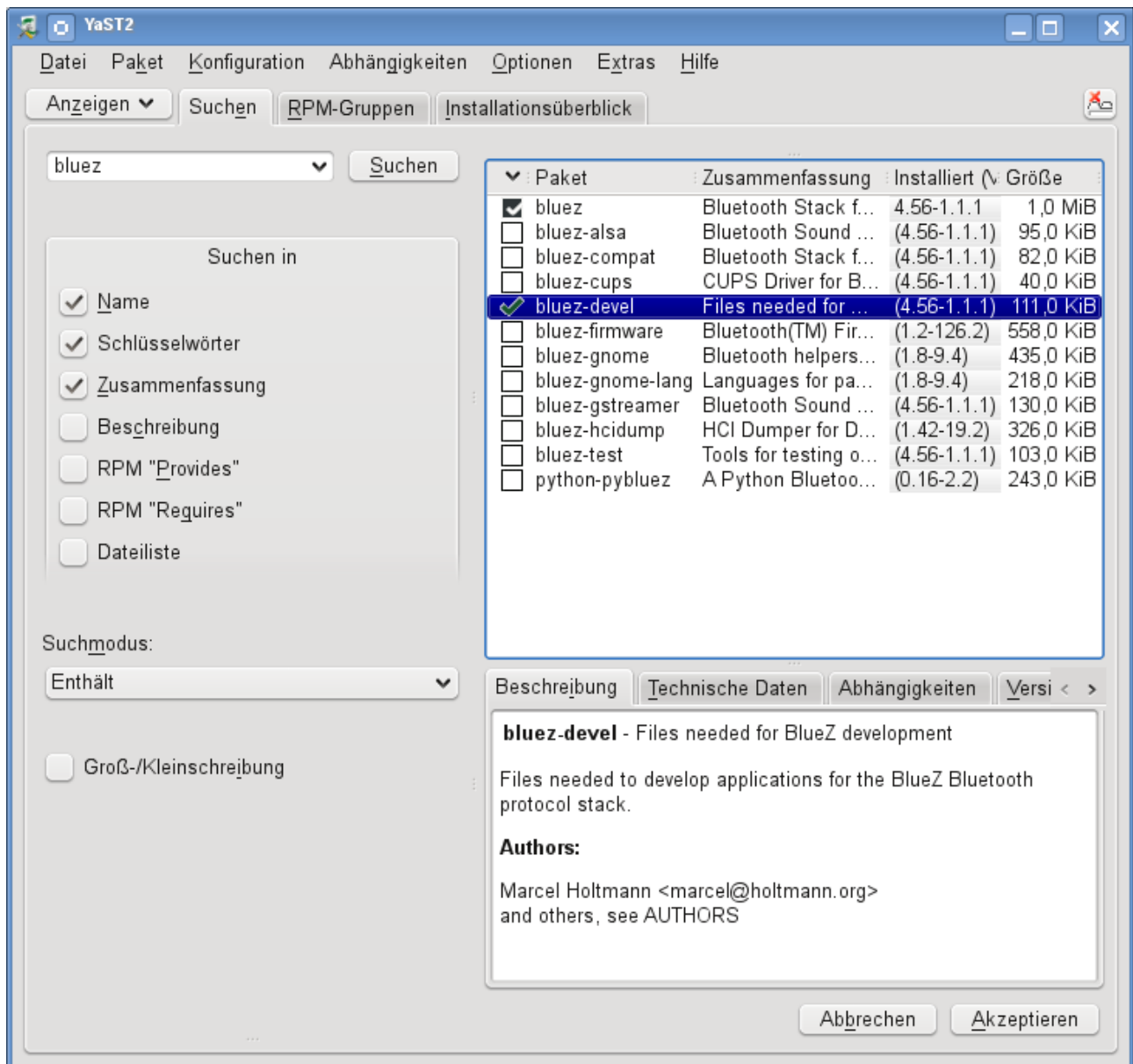


Abbildung 5: Überprüfe ob der gcc c++ Compiler installiert ist.

Abbildung 6: Überprüfe ob **bluez-devel** installiert ist.

3 leJOS installieren

Drücke **Alt+F2** und gebe *kdesu konqueror* ein um einen Dateimanager mit Rootrechten (Administratorrechten) zu öffnen.

Wenn der Konqueror geöffnet ist drücke **F9** um in das Dateisystem zu gelangen.

TIPP

Kopiere die Datei **lejos_NXJ_0_8_5beta.tar.gz** in das Verzeichnis `/opt` und wähle über das Kontextmeü (rechte Maustaste) *Archiv hierher auspacken*.

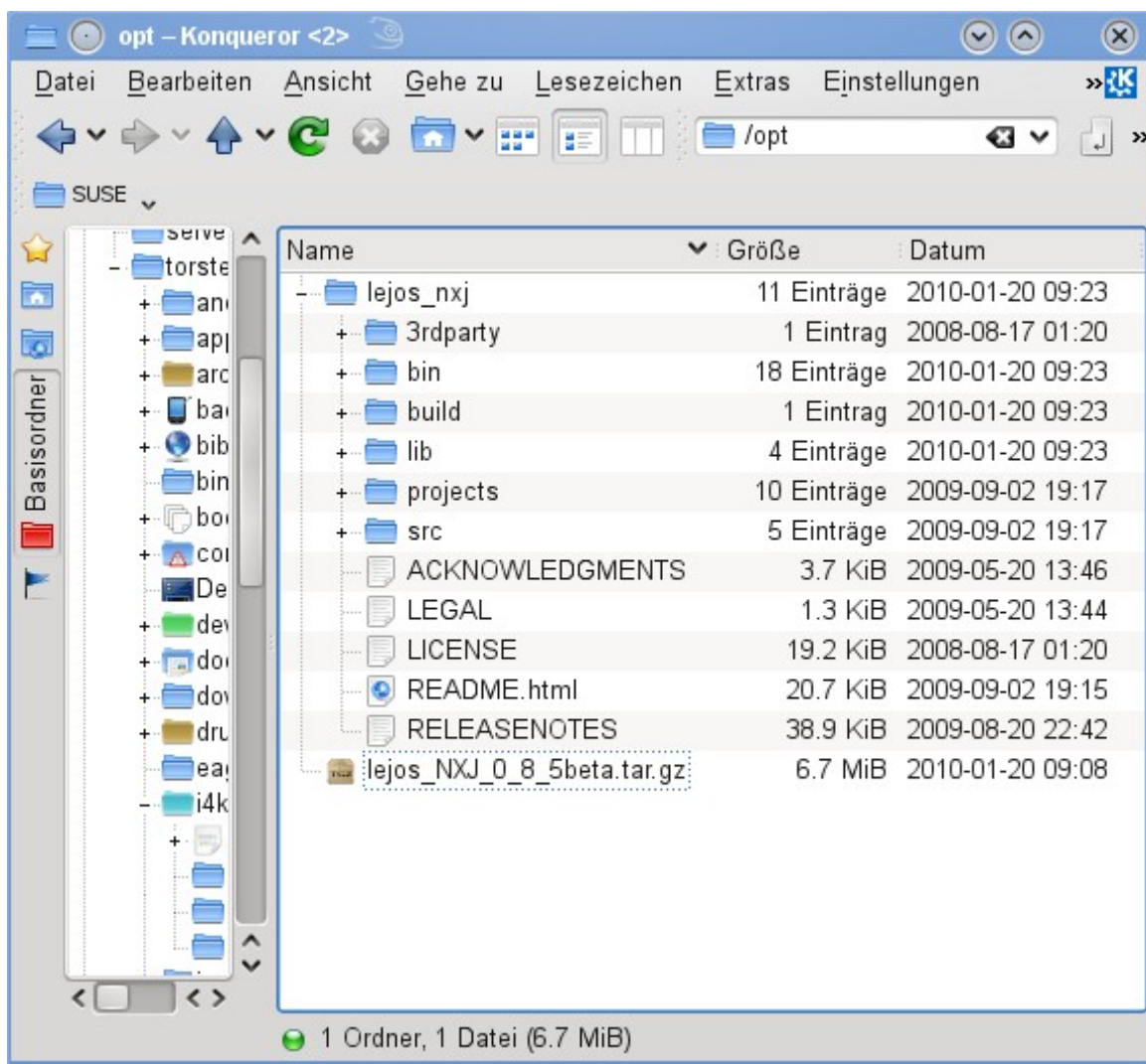


Abbildung 7: Entstandene Verzeichnisstruktur nach dem Auspacken im `/opt` Verzeichnis

3.1 Übersetzen der Software

Um die Software zu übersetzen wechsle in einer Rootkonsole ins Verzeichnis `/opt/lejos_nxj/build` und gebe dort den Befehl `ant` ein. Überprüfe ob im Verzeichnis `/opt/lejos_nxj/bin` jetzt `*.so` Dateien vorhanden sind, ansonsten hat die Übersetzung nicht geklappt.

Mit dem Befehl `su` (SuperUser) gelangst Du in den Rootmodus, dabei erfolgt eine „blinde Passwort abfrage“, d.h. Du siehst die eingegebenen Buchstaben nicht.

HINWEIS

```
demokrit:/opt/lejos_nxj/build # ant -v -buildfile build.xml
Apache Ant version 1.7.0 compiled on September 22 2007
Buildfile: build.xml
Detected Java version: 1.6 in: /usr/lib64/jvm/java-1.6.0-sun-1.6.0/jre
Detected OS: Linux
parsing buildfile /opt/lejos_nxj/build/build.xml with URI =
file:/opt/lejos_nxj/build/build.xml
Project base dir set to: /opt/lejos_nxj/build
[antlib:org.apache.tools.ant] Could not load definitions
from resource org/apache/tools/ant/antlib.xml. It could not
be found.
parsing buildfile jar:file:/usr/share/java/ant-
1.7.0.jar!/org/apache/tools/ant/types/conditions/antlib.xml
with URI = jar:file:/usr/share/java/ant-
1.7.0.jar!/org/apache/tools/ant/types/conditions/antlib.xml
Build sequence for target(s) `build' is [clean, libnxt,
jbluez, copy.binaries, clear, build]
Complete build sequence is [clean, libnxt, jbluez,
copy.binaries, clear, build, ]
...
...
clear:

build:
    [echo] Done.

BUILD SUCCESSFUL
Total time: 1 second
demokrit:/opt/lejos_nxj/build #
```

Wechsle in das Verzeichnis `/opt` und gebe dem Befehl `chmod -R 755 /opt/lejos_nxj` ein, um für die Richtigen Ausführungsrechte zu sorgen, insbesondere müssen alle Dateien im `bin`-Verzeichnis ausführbar sein.

3.2 Den Bibliothekspfad (Library Path) setzen:

In der Datei `/etc/ld.so.conf` wird festgelegt wo das System nach Bibliotheken sucht. Um die Lego Bibliothek dem System verfügbar zu machen editieren wir diese Datei und fügen eine Zeile (hier grün) hinzu.

```
demokrit:/etc # cat ld.so.conf
/usr/X11R6/lib64/Xaw3d
/usr/X11R6/lib64
/usr/lib64/Xaw3d
/usr/X11R6/lib/Xaw3d
/usr/X11R6/lib
/usr/lib/Xaw3d
/usr/x86_64-suse-linux/lib
/usr/local/lib
/opt/kde3/lib
/lib64
/lib
/usr/lib64
/usr/lib
/usr/local/lib64
/opt/kde3/lib64
/opt/lejos_nxj/lib
include /etc/ld.so.conf.d/*.conf
```

Aus Performance gründen wertet Linux diese Datei allerdings nicht direkt aus, sondern speichert sie als Binärdatei `ld.so.cache`. Um diesen Cache zu aktualisieren genügt ein Aufruf des Programms `ldconfig`.

```
demokrit:/etc # ldconfig
```

Damit haben wir dem System die Lego-Bibliothek bekannt gemacht.

3.3 Setzen der usb rechte

Um auch nicht Root Benutzern den Zugriff auf die USB-Schnittstelle zu erlauben wird eine Datei **97-lego.rules** im Verzeichnis **/etc/udev/rules.d** mit folgenden Inhalt angelegt.

Falls bereits eine Datei mit der Startnummer 97-Dateiname existiert, einfach eine andere noch freie Nummer wählen. Dateien mit kleineren Nummern werden nach dem Systemstart zuerst abgearbeitet.

Den Inhalt der Datei (diese Datei steht auch als Download zur Verfügung) zeigen wir mit dem cat Befehl an.

demokrit:/etc/udev/rules.d # cat 97-lego.rules

```
# Lego NXT
BUS=="usb",SYSFS{idVendor}=="03eb", GROUP="users",MODE="0660"
BUS=="usb", SYSFS{idVendor}=="0694", GROUP="users", MODE="0660"
```

Die Datei *97.lego.rules* im Verzeichnis */etc/udev/rules.d*

Um die neu angelegte „Rules“ dem System bekannt zu machen, es ist notwendig die Regeln neu zu laden.

demokrit:/etc/udev/rules.d # udevadm control --reload-rules

Damit sind die USB Rechte gesetzt.

4 leJOS NXJ Eclipse Konfiguration

Um bequem mit *Eclipse* programmieren zu können, richten wir Eclipse so ein, das das Compilieren des Java Quellcodes und das Uploaden auf dem Mindstorm direkt von Eclipse aus dem Menü heraus erfolgen kann.

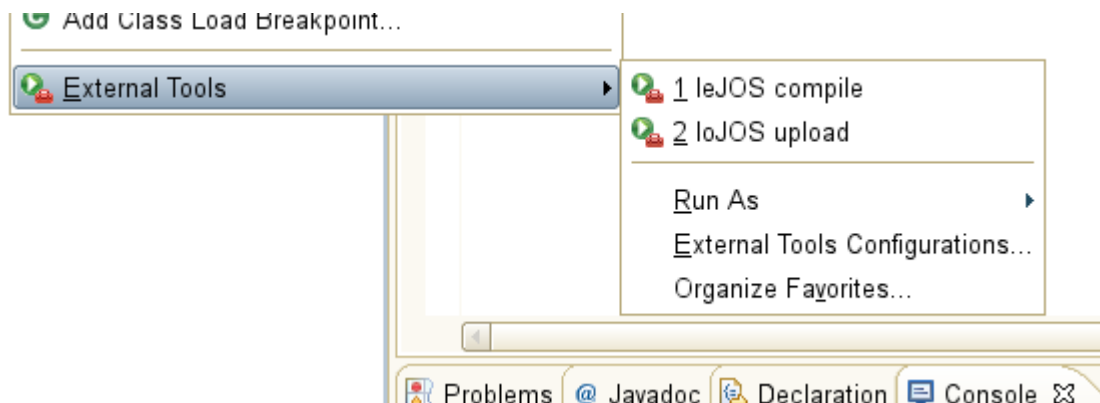


Abbildung 8: Mit 1 leJOS compile und 2 leJOS upload läßt sich nach der Konfiguration bequem programmieren.

1. Starte **Eclipse** und öffne das Menü *Run* → *External Tools* → *External Tools Configuration...*
2. In dem erscheinenden Dialogfeld drücke Launch New Configuration (das ist das linke Icon) um eine neue Konfiguration hinzuzufügen. Trage alles exakt so ein wie auf den Bildern aufgeführt.

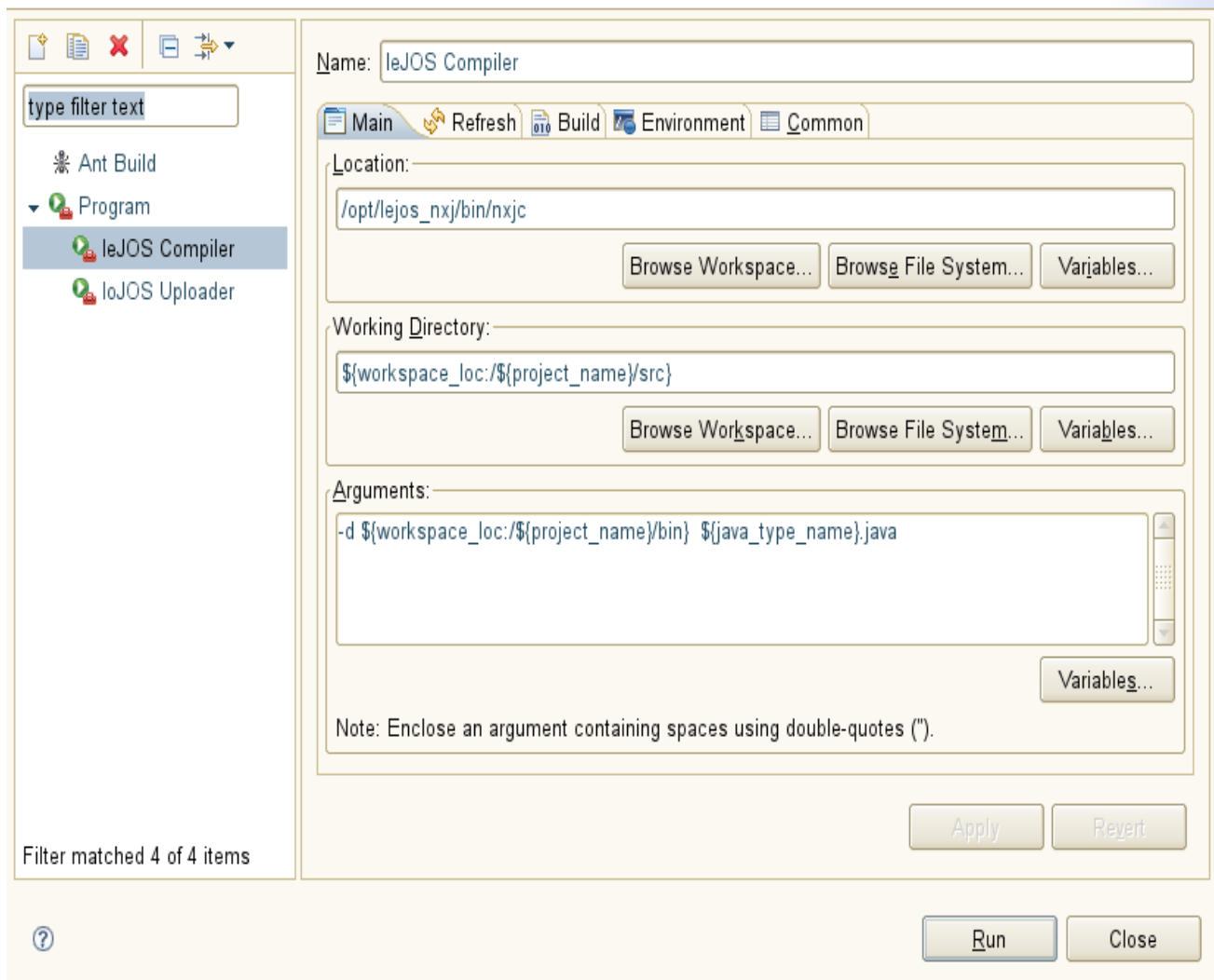
4.1 leJOS Compiler

Working Directory: **`${workspace_loc}/${project_name}/src`**

Arguments: **`-d ${workspace_loc}/${project_name}/bin} ${java_type_name}.java`**

Create, manage, and run configurations

Run a program



4.2 leJOS Uploader

Working Directory: `${workspace_loc}/${project_name}/bin}`

Arguments: `${java_type_name}`

Create, manage, and run configurations

Run a program



The screenshot shows the 'Run Configuration' dialog in Eclipse. The left sidebar lists the configuration types: Ant Build, Program, leJOS Compiler, and leJOS Uploader. The 'leJOS Uploader' configuration is selected. The main area shows the configuration details for 'leJOS Uploader'. The 'Name' field is 'leJOS Uploader'. The 'Location' field is '/opt/lejos_nxj/bin/nxj'. The 'Working Directory' field is '\${workspace_loc}/\${project_name}/bin}'. The 'Arguments' field is '\${java_type_name}'. There are buttons for 'Browse Workspace...', 'Browse File System...', and 'Variables...' for each of the Location, Working Directory, and Arguments fields. At the bottom, there are 'Apply', 'Revert', 'Run', and 'Close' buttons. A note at the bottom states: 'Note: Enclose an argument containing spaces using double-quotes ("").'

Name: leJOS Uploader

Location: /opt/lejos_nxj/bin/nxj

Working Directory: \${workspace_loc}/\${project_name}/bin}

Arguments: \${java_type_name}

Note: Enclose an argument containing spaces using double-quotes ("").

Filter matched 4 of 4 items

Run Close

4.3 External Tools Menu

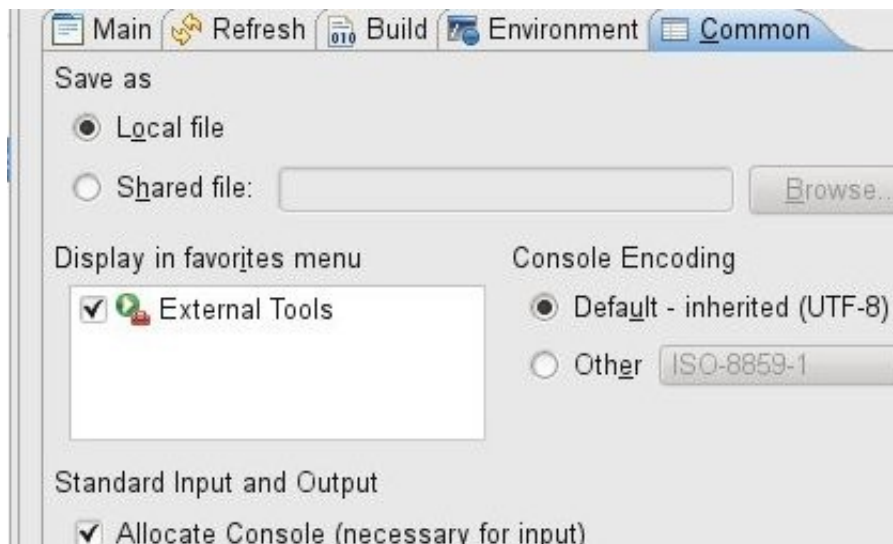


Abbildung 9: Reiter Common: Aktiviere das Kontrollkästchen External Tools (einzeln bei beidem [uploader und compiler])

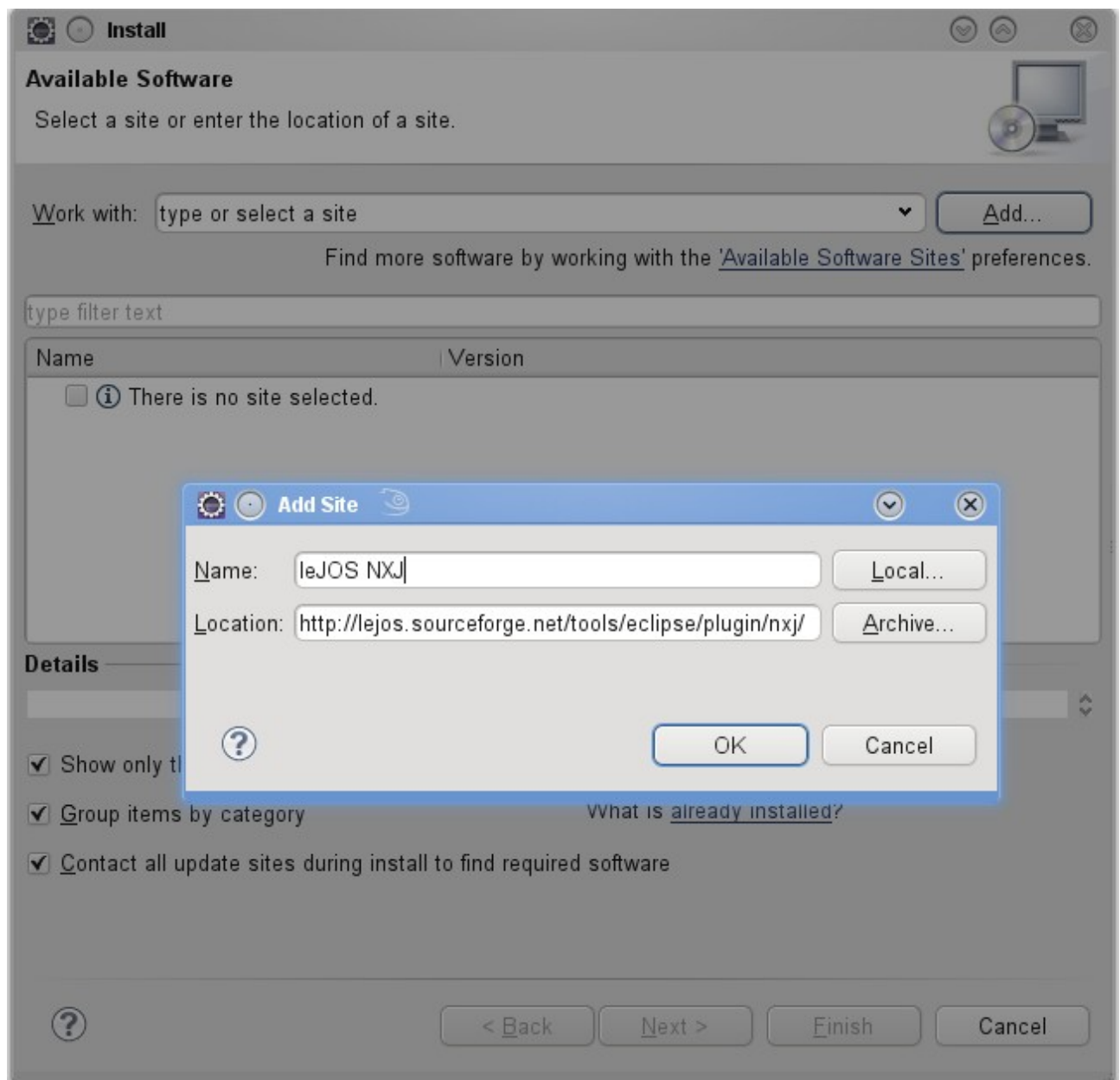
5 Eclipse-Plugin

Zusätzlich kann man das Eclipse-Plugin (Version 0.8) installieren, das hat den Vorteil, das man dann ein Java Projekt per Mausklick in ein leJOS Projekt konvertieren kann. D.h. es ist dann nicht mehr nötig die erforderlichen leJOS Bibliotheken Manuell einzubinden.

5.1 Installation des Plugins

Um das Plugin zu installieren verwenden wir den Update Manager.

3. Starte jetzt **Eclipse** und öffne das Menü *Help* → *Install New Software...* In dem erscheinenden Dialogfeld drücke auf den Link *Available Software Sites*



- Name: leJOS NXJ (Hier kannst Du eingeben was Du möchtest)
 - URL: <http://lejos.sourceforge.net/tools/eclipse/plugin/nxj/>
4. Wir haben jetzt leJOS NXJ hinzugefügt und können es installieren. Wähle in der Dropdownliste den von Dir hinzugefügten Eintrag aus. Entferne das Häkchen bei **Group items by category**.
 5. Bestätige *Next*, akzeptiere die Lizenzbedingungen und drücke *Finish*. Nachdem das Plugin installiert ist erfordert Eclipse einen Neustart.

Available Software

Check the items that you wish to install.




Work with: leJOS NXJ - <http://lejos.sourceforge.net/tools/eclipse/plugin/nxj/> ▼

Add...

Find more software by working with the '[Available Software Sites](#)' preferences.

type filter text

Name	Version
<input checked="" type="checkbox"/>  leJOS NXJ Eclipse features	0.8.5.200909022138

Details

Eclipse features for leJOS NXJ

☒ Show only the latest versions of available software ☐ Hide items that are already installed

☐ Group items by category

What is [already installed](#)?

☒ Contact all update sites during install to find required software



< Back

Next >

Finish

Cancel

5.2 Konfiguration des Eclipse leJOS Plugin

Die Konfiguration beschränkt sich darauf, den Bibliothekspfad einzustellen.

Gehe in das Menü *Window* → *Preferences* → *JOS NXJ* .

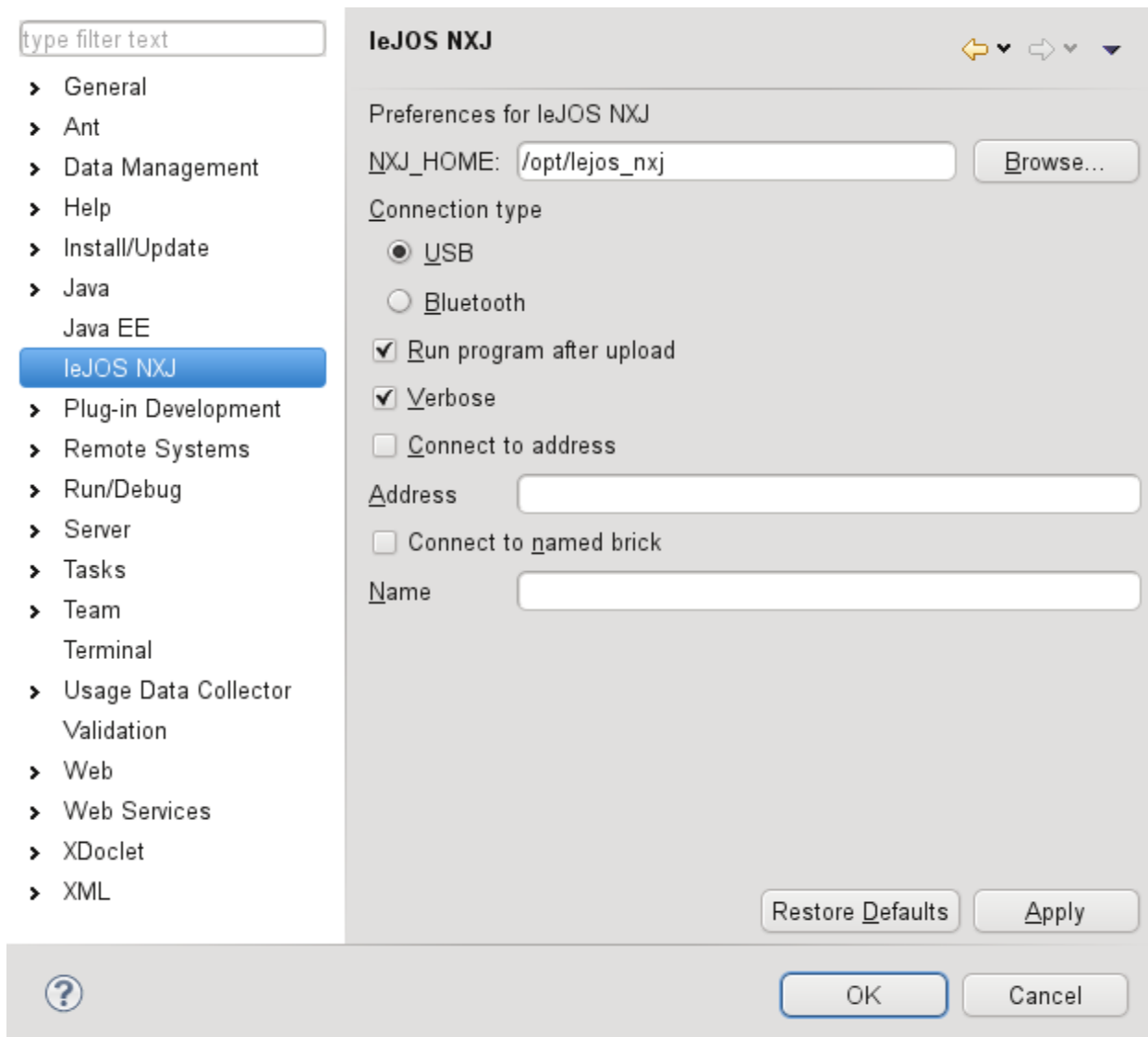


Abbildung 10: Setze `NXJ_HOME`: auf `/opt/lejos_nxj` und den Verbindungstyp auf `USB`.

6 Die Firmware updaten

Das Updaten des Mindstorm Bausteins geschieht auf eigene Gefahr. Für eventuell auftretenden Schäden ist www.informatics4kids.de nicht verantwortlich.

HINWEIS

Das flaschen des Mindstorm Bausteins ist natürlich nur erforderlich, falls noch keine leJOS Firmware auf dem Baustein installiert ist. Soll lediglich eine Eclipse-Entwicklungsumgebung eingerichtet werden kann dieser Schritt entfallen.

6.1 Mindstorm für das Flaschen vorbereiten

Um den Mindstorm mit Java programmieren zu können ist es nötig die leJOSJXJ Firmware auf dem NXT aufzuspielen. Damit wird allerdings alles auf dem NXT gelöscht.

In 3 Schritten in den Updatemode:

1. Versichere Dich das der NXT mit dem USB Kabel mit dem PC verbunden ist.
2. Schalte den NXT an. (Drücken des Orangen Knopfes)
3. Setze den NXT in den Firmware Update-Mode. Dazu muss Du den Resetbutton (am besten geht das mit einer Büroklammer) länger als 4 Sekunden gedrückt werden. Der Resetbutton befindet sich an der Unterseite des Gerätes unter dem USB-Anschluß.
4. Wenn Du ein regelmäßiges Ticken hörst befindet sich das Gerät im Update-Mode¹.

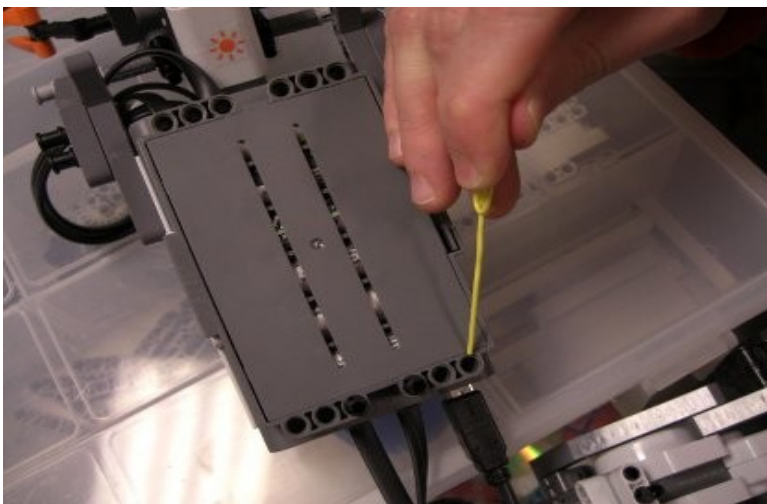


Abbildung 11: Der Resetbutton befindet sich unter dem USB Anschluss in ersten Loch.

¹ Ein wenig Hackerfeeling sollte aufkommen ;-)

6.2 Flaschen

Zum Flaschen des Mindstorm gehe nun in das Verzeichnis `/opt/lejos_nxj/bin` und gebe das Kommando `./nxjflash` ein.

```
demokrit:/opt/lejos_nxj/bin # ./nxjflash
Building firmware image.
VM file: /opt/lejos_nxj/bin/lejos_nxt_rom.bin
Menu file: /opt/lejos_nxj/bin/StartUpText.bin
VM size: 52752 bytes.
Menu size: 38016 bytes.
Total image size 91008/94208 bytes.
Locating device in firmware update mode.
Found NXT: %%NXT-SAMBA%% 1
Connected to SAM-BA v1.4
Opened device in firmware update mode.
Unlocking pages.
Writing firmware image.
Verifying firmware.
Verified 94208 bytes ok.
Restarting the device.
demokrit:/opt/lejos_nxj/bin #
```

Wenn alles geklappt hat begrüßt leJOS Dich mit dieser Grafik. Das was vorher einmal auf dem Baustein war, ist für immer verloren.



Abbildung 12: Lejos Startbildschirm

7 Anhang

7.1 A: Quellen

1. Lejos Mindstorms Project von Sourceforge
<http://lejos.sourceforge.net/>

7.2 B: Über dieses Dokument

- Getestet wurde mit OpenSuse 11.2 64Bit AMD System.
- Release 0.1 – 20.01.2010
- Reversion 0.2 – *Schüler Proofreading*
- Reversion 0.3 – 04.02.2011 - *Schüler Proofreading*