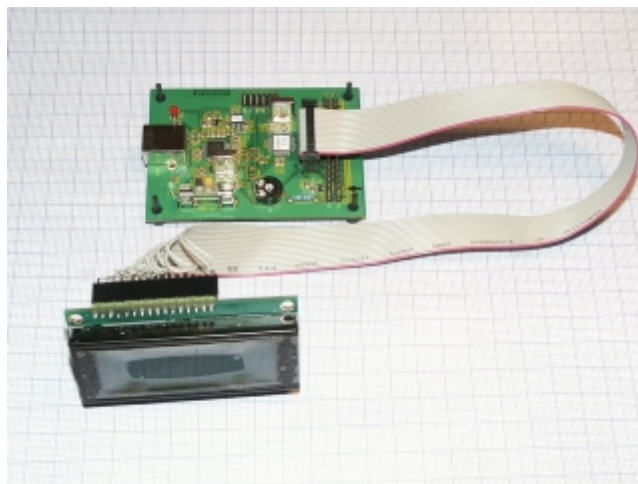


Statusdisplay

Diese Anleitung beschreibt die Installation und Konfiguration verschiedener Programme und Treiber um mit Hilfe der LCD Adapterleiterplatte ULA200 von ELV. So sieht die fertig montierte Leiterplatte mit angeschlossenem LC-Display aus.:



Bevor es los geht sollte überprüft werden, ob die folgenden Pakete installiert sind:

- doxygen
- cvs
- mc
- autoconf, automake, gcc
- kernel-source, kernel-docs, linux-kernel-headers
- libusb-devel
- patch

!! - **Hinweis** - !!

Alle Aktionen **müssen** als Administrator **root** durchgeführt werden !!

Erstes Anschließen der ULA200 Leiterplatte an den PC. Im Syslog sollten jetzt folgende Ausgaben zu sehen sein:

```
# tail -f /var/log/messages
...
```

```
kernel: usb 1-1.4: new full speed USB device using ohci_hcd and address 4
kernel: usb 1-1.4: new device found, idVendor=0403, idProduct=f06d
kernel: usb 1-1.4: new device strings: Mfr=1, Product=2, SerialNumber=3
kernel: usb 1-1.4: Product: ELV USB-LCD-Ansteuerung
kernel: usb 1-1.4: Manufacturer: ELV AG
kernel: usb 1-1.4: SerialNumber: ELE19UA3
kernel: usb 1-1.4: configuration #1 chosen from 1 choice
```

Das bedeutet, die Adapterleiterplatte funktioniert schon mal und wird vom Kernel erkannt.

Installation der "libftdi"

Die Bibliothek libftdi (Vers. 0.14.1) ermöglicht die direkte Kommunikation über USB mit einem FTDI Chip. Auf der ULA200 Leiterplatte wird ein solcher Chip zur Kommunikation mit dem PC eingesetzt. Um also mit der Adapterleiterplatte kommunizieren zu können muß die libftdi heruntergeladen werden von [Intra2Net.com](http://www.intra2net.com) und folgendermaßen übersetzt werden:

```
# su
# wget
http://www.intra2net.com/de/produkte/opensource/ftdi/SRPMs/libftdi-0.14-1.src.rpm
# rpmbuild --rebuild libftdi-0.14-1.src.rpm
```

Jetzt sollte unter /usr/src/packages/RPMS/i586 das neu erstellte RPM libftdi-devel-0.14.1.i586.rpm liegen.

```
# rpm -i /usr/src/packages/RPMS/i586/libftdi-0.14-1.i586.rpm
# rpm -i /usr/src/packages/RPMS/i586/libftdi-devel-0.14-1.i586.rpm
```

Installation von "lcdproc"

Der Zugriff auf das LC-Display erfolgt über den Daemon LCDd.
Entwicklerversion von [LCDproc](http://www.lcdproc.org) per CVS auschecken:

```
# cvs -d:pserver:anonymous@lcdproc.cvs.sourceforge.net:/cvsroot/lcdproc
login
```

Die meldung CVS password: kann problemlos mit der RETURN-Taste bestätigt werden. Eine anschließende Warnmeldung kann ignoriert werden.

```
# cvs -z3 -d:pserver:anonymous@lcdproc.cvs.sourceforge.net:/cvsroot/lcdproc
co -P lcdproc
```

In das neue Verzeichnis lcdproc wechseln:

```
# cd lcdproc
```

Damit der Text im LCD nicht verschoben dargestellt wird muß noch eine Zeile in `client/lcdvc/lcd_link.c` geändert werden so daß die Scrollfunktion permanent ausgeschaltet ist.

```
# cd clients/lcdvc/  
# sed 's/short autoscroll = 1/short autoscroll = 0/g' lcd_link.c > tmp.c  
# mv tmp.c lcd_link.c
```

Jetzt wieder zurück in Basisverzeichnis von `lcdproc` wechseln und `./autogen.sh` ausführen. Dadurch wird das configure Script erzeugt welches zur Erstellung des Makefile benötigt wird.

```
# cd -  
# chmod +x autogen.sh  
# ./autogen.sh
```

Ist `autogen.sh` erfolgreich gewesen sollte die Datei „`configure.sh`“ erstellt worden sein. Diese nun ausführen mit dem Parameter `--enable-drivers=ula200` oder um alle Treiber zu übersetzen `--enable-drivers=all`:

```
# ./configure --enable-drivers=ula200  
# make  
# make install  
# cd ..
```

Die Installation erfolgt ins Verzeichnis `/usr/local/(s)bin/` und eine standard Konfigurationsdatei liegt danach unter `/usr/local/etc/`. Damit auch „normale“ Benutzer den LCDd starten und `lcdvc` verwenden können muß noch das „s“ Bit gesetzt werden:

```
# chmod +s /usr/local/sbin/LCDd  
# chmod +s /usr/local/sbin/lcdvc
```

Installation von "isdn_rate"

Als nächstes muß das Quellcode-Paket `src.tar.gz` von www.fli4l.de heruntergeladen und entpackt werden:

```
# wget http://download.fli4l.de/3.2.2/src.tar.gz  
# tar xzf src.tar.gz
```

Zum Auslesen der Informationen des `imond` auf dem `fli4l` Router ist die Authentifizierung mittels Passwort nötig welches bei der Installation des Routers vergeben wurde. Ohne Passworteingabe können lediglich das Datum, die Uhrzeit und die Uptime des Router dargestellt werden, nicht aber die Upload/Download Informationen oder die öffentliche IP Adresse. Da `isdn_rate` im Original keine Möglichkeit bietet, ein Passwort an den Router zu senden, habe ich einen Patch zur Behebung dieses Problems geschrieben.

[Patch](#) herunterladen und entpacken:

```
# tar xfz isdn_rate_patch.tar.gz
# cp isdn_rate.diff src/fli4l/lcd/isdn_rate/
# cd src/fli4l/lcd/isdn_rate/
# patch -p0 < isdn_rate.diff
# make
# cp isdn_rate /usr/local/bin/isdn_rate
```

Installation der Konfigurationsdateien

Jetzt noch die [Konfigurationsdateien](#) herunterladen, entpacken und installieren. Dazu liegt den Konfigurationsdateien ein kleines Installations-Script bei:

```
# tar xfz install_config_files.tar.gz
# ./install.sh
```

Auf dem Desktop sollte jetzt ein Icon zu starten des Scripte `display_dsl_rate.sh` liegen. Jetzt nur noch im Shell-Script `display_dsl_rate.sh` die IP und das Admin Passwort für den fli4l Router eintragen und abspeichern.

```
# cd /home/<USER>/bin
```

Ein erster Test zeigt, ob alles funktioniert hat:

```
# ./display_dsl_rate.sh
```



!! - Wichtig - !!

Damit man als normaler „User“ auch schreibend auf ein ttyx Gerät zugreifen kann muß jeder User Mitglied in der Gruppe tty sein! Zu erkennen ist dies in der Datei `/etc/group`

```
# cat /etc/group
# ...
# tty:x:5:<LIST of USERS>
```

From:
<https://www.von-thuelen.de/> - **Christophs DokuWiki**

Permanent link:
https://www.von-thuelen.de/doku.php/wiki/projekte/dsl_rate_display/uebersicht

Last update: **2020/04/15 19:22**

