

Package lcd4linux

Version 3.6.0

Frank Meyer
email: frank@fli4l.de

Das fli4l-Team
email: team@fli4l.de

29. April 2011

Inhaltsverzeichnis

1	Dokumentation des Paketes LCD4Linux	3
1.1	LCD4Linux - Anzeige von Statusinformationen über LC-Display	3
1.1.1	Einleitung	3
1.1.2	Konfiguration des Displaytreibers	3
1.1.3	Konfiguration der Displayanzeige	8
1.1.4	Anschlussbelegung eines LCD-Moduls am Paralellport	11
1.1.5	Anschluss eines 4x40 Displays	12
1.1.6	Danksagung	13
A	Anhang zum Paket LCD4Linux	14
	Index	15

1 Dokumentation des Paketes LCD4Linux

1.1 LCD4Linux - Anzeige von Statusinformationen über LC-Display

1.1.1 Einleitung

Mit diesem Paket ist es möglich, ein LCD-Modul über verschiedene Schnittstellen an den Fli4l Rechner anzuschließen (parallel und USB sind direkt über die Fli4l-Konfiguration möglich, seriell muss über den 'Experten'-Modus konfiguriert werden, siehe entsprechenden Abschnitt weiter unten).

Auf diesem Display werden Informationen wie Datum, externe IP-Adresse, die aktuellen Loadwerte und natürlich auch der ISDN- oder DSL-Durchsatz für Up- und Download in kb/s und ein Balken angegeben.

Es können durch den Benutzer Icon's frei definiert werden, die animiert sein können, oder auch erst bei Erreichen von definierbaren Schwellenwerten erscheinen (z.B. ein Blitz bei hoher CPU Auslastung).

Es ist möglich Balken darzustellen (auch zwei Balken in einer Zeile), die sich in jegliche Richtung bewegen können (links, rechts, von oben oder unten).

Den Gestaltungsmöglichkeiten sind kaum Grenzen gesetzt - die meisten Grenzen setzt die Größe des Display's ;-)

Zudem können die Anzeigemöglichkeiten durch Plugins nahezu grenzenlos erweitert werden.

Es ist auch ein IMON-Plugin vorhanden, mit dem Statusinformationen abgefragt werden können - d.h. Dauer Onlineverbindung, Übertragungsrate und Menge, usw.

Zum Einsatz kommt dabei eine Software namens lcd4linux in der Version 0.10.1-CVS (vom 21.02.2007).

Auf der Homepage des Projektes (<https://ssl.bulix.org/projects/lcd4linux>) sind einige Bilder der unterstützten Displays und detaillierte Informationen zu den jeweiligen Konfigurationsmöglichkeiten im 'Experten'-Modus.

1.1.2 Konfiguration des Displaytreibers

Möchte man das LCD4Linux-Paket einsetzen, sind noch folgende Variablen zu setzen:

```
OPT_LCD4LINUX='yes'   (Standard-Einstellung: OPT_LCD4LINUX='no')
```

Zuerst wird das gewünschte Display mit seinen jeweiligen möglichen Zusatzoptionen und, sofern möglich, die Anschlussart konfiguriert. Danach dann, was wo auf dem Display erscheinen soll.

Um die Konfiguration für den Benutzer möglichst einfach zu halten und nicht Zusatzoptionen für ein Display gesetzt werden können, die das Display gar nicht kann, habe ich die häufigsten Displays, bzw. die wofür Supportanfragen kamen, per Default in der Fli4l-Konfiguration vordefiniert.

Sollten Sie ein Display haben welches nicht vordefiniert ist und es wird von lcd4linux unterstützt (siehe dazu obige WebSeite des Projektes), so können Sie den 'Experten'-Modus aktivieren und direkt selbst an der eigentlichen lcd4linux-Konfiguration die gewünschten Änderungen vornehmen.

Zur Zeit habe ich folgende Displays vorkonfiguriert:

- Hitachi HD44780 (parallel und i2c, konfigurierbar)
- LCD2USB, ein USB-Anschluss für das Hitachi HD44780
- HYUNDAI HP12542R-DY0 (funktioniert nur parallel)
- c't includ usb display (USB, das Display wurde im Rahmen eine C't Projektes in einer der Ausgaben Mitte 2006 vorgestellt)

Beachten sie bitte, dass Sie nur EINEN Treiber auswählen können/dürfen!

LCD4LINUX_HD44780_DRIVER - Aktivierung des Treibers für Hitachi HD44780 LC-Displays.

Dies ist das wohl am meisten verbreiteste Display. Dieses Display wird auch vom LCD Paket angesprochen.

(Standard-Einstellung: LCD4LINUX_HD44780_DRIVER='yes')

LCD4LINUX_HD44780_MODEL - Das verwendete Model.

Mögliche Werte:

generic Standard HD44780 Display
Noritake wie 'generic', zusätzlich mit softwaregesteuertem Hintergrundlicht
Soekris spezielle Schnittstelle mit BusyFlag Check im 4 Bit Mode
HD66712 wie 'generic' nur mit etwas anderen RAM Aufbau, auch für 'KS0073'
LCM-162 spezielles Display, häufig in Nexcom Blade Servers

(Standard-Einstellung: LCD4LINUX_HD44780_MODEL='generic')

LCD4LINUX_HD44780_BUS - Wie das Display angeschlossen ist. Standardmäßig ist das Display über den Druckeranschluss (parallel = parport = LPT) angeschlossen.

Für sogenannte Soekris-Rechner (spezielle Motherboards des Herstellers Soekris) und WRAP-Rechner gibt es eine weitere Möglichkeit: 'i2c'. Dies ist ein spezieller Bus mit einer besonderen Anschluss-Verkabelung und Steuerung für das Display.

Bezüglich Soekris und WRAP siehe auch:

Soekris: <http://soekris.hejl.de/>

WRAP: <http://wiki.port23.de/index.php/Creative.chaos.de>

Mögliche Werte: parport oder i2c.

(Standard-Einstellung: LCD4LINUX_HD44780_BUS='parport')

LCD4LINUX_HD44780_PORT - Wenn das Display über den Parallel-Port angeschlossen ist, wird hier der genaue Anschluss-Port bestimmt. Wenn nur ein Drucker-Port vorhanden ist (wie wohl in den meisten Fällen) so lautet der Eintrag `'/dev/parports/0'`, sollten mehrere Druckerport vorhanden sein, so ist der Wert am Ende entsprechend hochzusetzen (z.B.: `'/dev/parports/1'`).

Einträge mit `'/dev/parports/'` beginnend, nennen sich 'ppdev' Konfiguration (ParPort-DEvice). Es ist auch eine Konfiguration, wie im LCD Paket, mittels der hexadezimalen Adresse möglich (z.B. `'0x378'`), dies wird aber seitens der Entwickler nicht empfohlen.

(Standard-Einstellung: `LCD4LINUX_HD44780_PORT='/dev/parports/0'`)

LCD4LINUX_HD44780_I2C_DEV - Wenn das Display über den i2c Bus angeschlossen wird, wird hier der entsprechende GPIO Anschluss definiert. Dieser Anschluss steht nur zur Verfügung wenn die entsprechenden speziellen Soekris Module geladen wurden.

Diese Option ist standardmäßig deaktiviert (sprich: durch die Raute auskommentiert).

(Standard-Einstellung: `LCD4LINUX_HD44780_I2C_DEV='/dev/gpio0'`)

LCD4LINUX_HD44780_WIRE_TYPE - Wie sind die Anschlusspins des Displays mit der Schnittstelle des Rechners verbunden?

Es gibt leider keine allgemeingültige Regel, wie ein Display angeschlossen werden muss. Daher bietet lcd4linux die Möglichkeit eines sogenannten SoftWiring, d.h. über die Konfigurationsdatei kann jede einzelne Steuerleitung konfiguriert werden.

Da dies für die meisten StandardUser zu kompliziert (da dafür ein hohes technisches Wissen benötigt wird) und zu fehleranfällig wäre, habe ich die am häufigstens vorkommenden Beschaltung ebenfalls vorkonfiguriert. Sollte keine der Optionen auf Ihr Display zutreffen, so können Sie Ihre Beschaltung über den Experten-Modus direkt vornehmen.

Neuere fertig bestückte Displays von Kernel-Concepts benutzen meist die winamp-Variante, die älteren meist die generic-Version.

Mögliche Werte:

`generic` # auch LCD4Linux Wiring genannt (sehr häufig)
`winamp` # ebenfalls eine sehr häufige Beschaltung
`soekris` # spezielle Beschaltung für die Soekris-Boards

(Standard-Einstellung: `LCD4LINUX_HD44780_WIRE_TYPE='generic'`)

LCD4LINUX_HD44780_ASC255BUG - sollten Sie in den Laufbalken (bars) keine 'stabilen' Blöcke, sondern invertierte P's haben, so hat Ihr Display das 'Portishead Syndrom'. In diesem Fall setzen Sie die Option auf 'yes'.

(Standard-Einstellung: `LCD4LINUX_HD44780_ASC255BUG='yes'`)

Nun folgen einige optionale Konfigurationsmöglichkeiten. Es hängt sehr vom jeweiligen Einzelfall/Display ab, ob die Option unterstützt, bzw. benötigt wird.

LCD4LINUX_HD44780_CONTROLLERS - Display's größer als 4x40 besitzen einen zweiten Controller um die Zeichen jenseits von Zeichen 21 anzusteuern.

Zusätzlich muss allerdings auch der entsprechende Steuerkanal angeschlossen sein. Wenn zwei Controller definiert sind, so wird der ENABLE2 auf SLCTIN des Display's konfiguriert.

LCD4LINUX_HD44780_CONTRAST - bei einigen Displays lässt sich der Kontrast einstellen. Mögliche Werte sind von 0 bis 15.

LCD4LINUX_HD44780_BACKLIGHT - bei einigen Displays lässt sich die Hintergrundbeleuchtung beim Start des Programmes an und abstellen. Ist die Option nicht gesetzt, geht lcd4linux davon aus, dass die Hintergrundbeleuchtung an sein soll.

LCD4LINUX_HD44780_BRIGHTNESS - bei Displays von Noritake lässt sich softwaregesteuert die Hintergrundhelligkeit konfigurieren. Mögliche Werte sind von 0 bis 8.

LCD4LINUX_HD44780_BITS - die meisten Displays benutzen den 8 Bit Modus. Das bezieht sich auf die Art der Verkabelung. Sind alle 8 Steuerleitungen der LPT-Schnittstelle mit den 8 Steuerleitungen des Displays verbunden (der Rest sind Versorgungsleitungen), so wird vom 8 Bit Modus gesprochen. Sind nur 4 Leitungen verbunden (wird meist benutzt, wenn mehrere Displays angeschlossen werden sollen) dann den 4 Bit Modus konfigurieren.

LCD4LINUX_HD44780_USEBUSY - HD44780 Displays sind etwas Zeitkritisch in Ihrem Verhalten und der Darstellung. In einigen Fällen kann es sinnvoll sein, ähnlich wie bei einem Drucker der gerade etwas ausdruckt, über die Steuerleitung (sofern diese den verbunden ist) 'nachzufragen', ob das Display noch beschäftigt ist oder ob die nächsten Anweisungen geschickt werden können.

Auch hier hängt es sehr vom benutzten Display/Controller ab, ob diese Option benötigt wird oder nicht.

LCD4LINUX_LCD2USB_DRIVER - Es gibt ein OpenSource/Open Hardware Projekt, welches es sich zur Aufgabe gemacht hat, HD44780-Displays über den USB-Port ansprechen zu können. Der Name des Projektes lautet LCD2USB und die Homepage ist unter <http://www.harbaum.org/till/lcd2usb/> zu erreichen. Das Projekt vertreibt keine fertige Hardware, sondern listet dort die benötigten Bauteile und eine Aufbauanleitung. Lediglich fertig geätzte Platinen können dort bezogen werden. Aber es wird nicht schaden, auch nach einem fertig programmierten ATMEL-Controller zu fragen, dann kann man sich einige Bauteile und vor allen Dingen Arbeit sparen.

Möchten Sie ein solches Display benutzen, so setzen Sie die Konfiguration für diesen Treiber auf 'yes'. Die Optionen von LCD4LINUX_HD44780 gelten NUR für über parallel oder i2c angeschlossene Displays.

LCD4LINUX_LCD2USB_CONTRAST - eine optionale Konfiguration zur Einstellung des Kontrastes, einstellbar von 0 bis 255.

LCD4LINUX_LCD2USB_BRIGHTNESS - eine optionale Konfiguration zur Einstellung der Hintergrundhelligkeit, einstellbar von 0 bis 255.

LCD4LINUX_GLCD2USB_DRIVER - seit kurzem gibt es ein OpenSource/Open Hardware Projekt, welches es sich zur Aufgabe gemacht hat, KS0108-Graphik-Displays über den USB-Port ansprechen zu können. Der Name des Projektes lautet GLCD2USB und die Homepage ist unter <http://www.harbaum.org/till/glcd2usb/> zu erreichen. Das Projekt vertreibt keine fertige Hardware, sondern listet dort die benötigten Bauteile und eine Aufbauanleitung. Lediglich fertig geätzte Platinen können dort bezogen werden. Aber es wird nicht schaden, auch nach einem fertig programmierten ATMEL-Controller zu fragen, dann kann man sich einige Bauteile und vor allen Dingen Arbeit sparen.

Möchten Sie ein solches Display benutzen, so setzen Sie die Konfiguration für diesen Treiber auf 'yes'. Die Optionen von LCD4LINUX_HD44780 gelten NUR für über parallel oder i2c angeschlossene Displays.

LCD4LINUX_GLCD2USB_BRIGHTNESS - eine optionale Konfiguration zur Einstellung der Hintergrundhelligkeit, einstellbar von 0 bis 255.

LCD4LINUX_HP12542R_DRIVER - die Firma Pollin (<http://www.pollin.de>) vertreibt zur Zeit (Stand: September 2006) ein sehr großes und günstiges Display von HYUNDAI, dass 'HP12542R-DYO'. Es wird in zwei Teilen vertrieben, zum einem das Display selbst und zusätzlich eine Anschlussplatine für den parallelen Port.

LCD4LINUX_HP12542R_PORT - auch dieses Display wird wie das HD44780 über den Druckeranschluss angesteuert. Deshalb muss auch hier der Anschlussport definiert werden. Sollte Ihr Rechner nur über einen Druckeranschluss verfügen (Standard), so müssen Sie hier nichts ändern. Beachten Sie bitte, dass es bei diesem Display NICHT möglich ist, im Gegensatz zum HD44780-Display, eine hexadezimale Hardwareadresse einzugeben.

LCD4LINUX_HP12542R_OPTIONS - dieses Display wird nicht von lcd4linux direkt angesteuert, sondern von einer zusätzlichen Bibliothek, der 'serdisplib' (<http://serdisplib.sourceforge.net/>). Diese Bibliothek kennt ein paar weitere Konfigurationsmöglichkeiten, abhängig vom benutztem Display, die lcd4linux über diese Optionen weitergibt. Zur Zeit ist das für dieses Display lediglich die Option, ob die Anzeige invertiert werden soll.

LCD4LINUX_HP12542R_CONTRAST - auch bei diesem Display ist es möglich den Kontrast zu konfigurieren. Die möglichen Werte sind von 0 bis 10.

LCD4LINUX_HP12542R_BACKLIGHT - ob die Hintergrundbeleuchtung beim Start des Display eingeschaltet werden soll.

LCD4LINUX_HP12542R_ROTATE - die 'serdisplib' hat noch ein interessantes Feature, nämlich die Möglichkeit die Anzeige auf dem Display zu drehen.

Dies ist in vier Schritten möglich:

```
0  normal
1  90 Grad
2  180 Grad
3  270 Grad
```

LCD4LINUX_CTINCLUD_DRIVER - die Zeitschrift C't veröffentlichte unter dem Namen 'Mäusekino' im April 2006 ein Projekt zur Ansteuerung eines USB-Displays. Dies erfolgte in Zusammenarbeit mit der Firma INCLUD (<http://www.includ.de/>). Im dortigen WebShop können Einzelteile, als auch komplette Displays bestellt werden. Die Ansteuerung des Displays erfolgt über die Zusatzbibliothek 'serdisplib'.

LCD4LINUX_CTINCLUD_OPTIONS - über diese Konfiguration gibt lcd4linux Optionen an die Zusatzbibliothek weiter die lcd4linux selbst nicht kennt. Zur Zeit ist das für dieses Display lediglich die Option, ob die Anzeige invertiert werden soll.

LCD4LINUX_CTINCLUD_ROTATE - auch bei diesem Display ist es durch die 'serdisplib' möglich (so wie bei dem HP12542R) die Anzeige auf dem Display zu drehen.

Dies ist in vier Schritten möglich:

```
0 normal
1 90 Grad
2 180 Grad
3 270 Grad
```

LCD4LINUX_FW8888_DRIVER Es gibt von Allnet ein relativ stromsparendes 19 Zoll 1HE Gerät mit dem Namen FW8888. Das Gerät verfügt über ein 16x2 LC Display mit 4 Tasten.

([http://www.allnet.de/nas-utility-server.html?%26tx_mmallnetproductplugin_pi1\[showUid\]=421433%26](http://www.allnet.de/nas-utility-server.html?%26tx_mmallnetproductplugin_pi1[showUid]=421433%26)

1.1.3 Konfiguration der Displayanzeige

LCD4LINUX_ICONS - LCD4Linux bietet die Möglichkeit von animierten Icons, z.B. eines schlagenden Herzens oder auch einen Blitz. Diese Icons können über das Layout konfiguriert werden, allerdings benötigt das Programm dazu etwas Speicherplatz, den es sich beim Programmstart reservieren muss, um die Icons zu laden. Geben Sie hier die Anzahl der gewünschten Icons an (max. 8). Sollten Sie im Layout mehr Icons konfigurieren, als Sie hier Speicherplatz reservieren, werden die 'zusätzlichen' nicht angezeigt.

LCD4LINUX_DISPLAY_SIZE - die Größe des Displays. Anzugeben in Länge (Zeichen) x Höhe (Zeilen). Wenn Ihr Display 20 Zeichen in 4 Zeilen darstellen kann, so tragen Sie hier bitte '20x4' ein.

LCD4LINUX_DSL_SPEED_IN - Die Download-Geschwindigkeit des DSL Anschlusses. Wird benutzt um die Anzeige des Download-Balkens zu kalibrieren.

LCD4LINUX_DSL_SPEED_OUT - Die Upload-Geschwindigkeit des DSL Anschlusses. Wird benutzt um die Anzeige des Upload-Balkens zu kalibrieren.

LCD4LINUX_LAYOUT_N - Der interessanteste Teil des Paketes, denn hier geht es um die eigentliche Darstellung - also was auf dem Display wo erscheinen soll. Geben Sie hier die Anzahl der Layout-Konfigurations-Zeilen an.

LCD4Linux benutzt ein Konzept namens 'Widgets'. Das sind im Grunde genommen kleine eigenständige Funktionen, die dann irgendetwas machen.

Über 'LCD4LINUX_LAYOUT_x' werden dann diese Funktionen an die ihnen zugedachte Position verwiesen. 'Row1.Col1 :Info:' bedeutet das der Info-Lauftext (die Erklärung welches Widget was macht und wie lang es ist erfolgt weiter unten im Text) an der ersten Position der ersten Zeile startet, und da der Info-Text die ganze Zeile belegt passt da auch sonst nichts mehr hin. Möchten Sie jetzt aber trotzdem, dass vor dem Info-Text noch ein kleines Icon dargestellt werden soll, so müssen Sie die Konfiguration ein wenig ändern.

z.B. in:

```
LCD4LINUX_LAYOUT_1='Row1.Col1 :Lightning:'
LCD4LINUX_LAYOUT_2='Row1.Col2 :Info:'
```

Beachten Sie bitte, dass dann in diesem Beispiel das letzte Zeichen des Lauftextes 'abgeschnitten' wird, d.h. es kann nicht dargestellt werden, weil das Display nicht groß genug ist. Das ist in dem Fall des Lauftextes nicht weiter schlimm, da ja ohnehin die Zeichen 'weiterwandern' und dann der Text auch erscheint, nur bei anderen Widgets die statisch sind und nicht weiterlaufen, könnte dann das ein oder andere Zeichen fehlen.

Die zur Zeit möglichen Widgets sind in Table 1.1 aufgeführt. (die Info_Tel_x Widgets haben eine wechselnde Anzeige, in Abhängigkeit ob eine 'letzte Rufnummer' vorliegt, darum steht dort ein '-ORDER-')

Typ	Information	Zeichenbreite
:Info:	Lauftext mit Name und IP	20
:Info2:	Lauftext mit Name, Version, Ram und CPU	20
:Date_dd_mm_yyyy:	Datum mit vier Stellen Jahr	10
:Date_dd_mm_yy:	Datum mit zwei Stellen Jahr	8
:Time_hh_mm_ss:	Uhrzeit	8
:ImonDSLQuantity_In:	DSL Eingangs Volumen	10
:ImonDSLQuantity_Out:	DSL Ausgangs Volumen	9
:ImonDSLRate_In:	DSL Eingangs Rate	9
:ImonDSLRate_Out:	DSL Ausgangs Rate	9
:ImonDSLRate_Bar:	DSL Rate Balken	20
:ImonDSLOnline_Time:	DSL Onlinezeit	20
:ImonDSL_IP:	DSL IP oder Offline	19
:ImonDSL_IP_2:	DSL IP oder Datum, Uhrzeit	20
:ImonDSLName:	DSL Circuit Name	20
:ImonDSLCharge:	DSL Online-Kosten	17
:ImonISDN1Status:	ISDN Circuit Name	20
:ImonISDN2Status:	ISDN Circuit Name	20
:ImonISDN1Name:	ISDN Circuit Name	20
:ImonISDN2Name:	ISDN Circuit Name	20
:ImonISDN1Rate_Bar:	ISDN Circuit Name	20
:ImonISDN2Rate_Bar:	ISDN Circuit Name	20
:ImonISDN1Rate_In:	ISDN Circuit Name	20
:ImonISDN2Rate_In:	ISDN Circuit Name	20
:ImonISDN1Rate_Out:	ISDN Circuit Name	20
:ImonISDN2Rate_Out:	ISDN Circuit Name	20
:ImonISDN1Charge:	ISDN Circuit Name	20

:ImonISDN2Charge:	ISDN Circuit Name	20
:ImonISDN1_IP:	ISDN Circuit Name	19
:ImonISDN2_IP:	ISDN Circuit Name	19
:ImonISDN1Online_Time:	ISDN Circuit Name	20
:ImonISDN2Online_Time:	ISDN Circuit Name	20
:ImonISDN1Quantity_In:	ISDN Circuit Name	15
:ImonISDN2Quantity_In:	ISDN Circuit Name	15
:ImonISDN1Quantity_Out:	ISDN Circuit Name	15
:ImonISDN2Quantity_Out:	ISDN Circuit Name	15
:Info_Tel_1:	Kein Anruf! -ODER- Nummer, Datum, Uhrzeit	20
:Info_Tel_2:	Kein Anruf, DSL Onlinezeit -ODER- Nummer, Datum, Uhrzeit, DSL Onlinezeit	20
:Info_Tel_3:	DSL Onl.Zeit -ODER- Rufnummer	20
:TelNr:	TelNummer letzter Anruf	20
:TelDate:	Datum letzter Anruf	8
:TelTime:	Uhrzeit letzter Anruf	8
:OS:	Anzeige Betriebssystem	20
:CPU:	Anzeige CPU Version	9
:RAM:	Anzeige gesamt RAM Speicher	11
:RAM_FREE:	Anzeige freien RAM Speicher	16
:Busy:	CPU Auslastung (cpu usage)	9
:BusyBar:	CPU Auslastungs Balken	10
:Load:	Rechner Auslastung	10
:LoadBar:	Rechner Auslastungs Balken	10
:Eth0:	Volumen von Eth0	10
:Eth0Bar:	Rate von Eth0	14
:PPP:	Volumen über PPP-Verbindung	9
:Uptime:	Zeit seit letztem Neustart	20
:VarText1:	text -> /etc/lcd_text1.txt	max 20
:VarText2:	text -> /etc/lcd_text2.txt	max 20
:VarText3:	text -> /etc/lcd_text3.txt	max 20
:VarText4:	text -> /etc/lcd_text4.txt	max 20
:Lightning:	Icon: Blitz	1
:Heartbeat:	Icon: Herzschlag	1
:Heart:	Icon: Herz	1
:Blob:	Icon: (Luft)Blase	1
:Wave:	Icon: Welle	1
:Squirrel:	Icon: Wirbel	1
:Rain:	Icon: Regen(tropfen)	1

Tabelle 1.1: Übersicht über mögliche Widgets

LCD4LINUX_EXPERT_MODE - wenn diese Option auf 'yes' gesetzt wird, so wird die komplette Konfiguration ignoriert und stattdessen eine 'lcd4linux.conf' aus dem Ordner 'config/etc/lcd4linux\' (Ordner muss selbst erstellt werden, eine Beispiel-Konfig findet sich unter 'opt/etc/lcd4linux') mit auf den Router kopiert. Diese Datei muss entsprechend den eigenen Bedürfnissen angepasst werden.

LCD4LINUX_TEST - zum Testen der Konfiguration der 'lcd4linux.conf' kann man diese Option einschalten.

Der LCD4LINUX-Daemon wird dann nicht automatisch gestartet, sondern man muss sich auf die Console verbinden (per ssh oder direkt) und den Daemon im DebugModus starten - man sieht dann direkt evtl. Fehlermeldungen und der Daemon läuft dann nicht als Hintergrunddienst, sondern kann über 'strg+c' abgebrochen werden.

Wenn man nun einen Editor auf dem Fli4l mitinstalliert hat (z.B. den e3) kann man nun unter /etc/lcd4linux/lcd4linux.conf die Konfiguration korrigieren.

Der Aufruf für den LCD4Linux-DebugModus lautet:

```
'lcd4linux -f /etc/lcd4linux/lcd4linux.conf -Fvv'
```

1.1.4 Anschlussbelegung eines LCD-Moduls am Paralellport

```

13 ----- 1 Draufsicht auf den
   \ o o o o o o o o o o o o /   Paralellport, Rück-
   \ o o o o o o o o o o o o /   seite PC
25 ----- 14

```

Der Anschluss eines LCD-Moduls an den Router wird folgendermaßen aufgetrennt:

Paralellport-Pin	Beschreibung	LCD-Modul	LCD-Pin
18-25	GND		--
	GND		1 -- - Brücke
	R/W		5 --
	+5V		2
1	STROBE	EN(1)	6
2	D0	D0	7
3	D1	D1	8
4	D2	D2	9
5	D3	D3	10
6	D4	D4	11
7	D5	D5	12
8	D6	D6	13
9	D7	D7	14
14	Autofeed	RS	4
17	Select In	EN(2)	? (für LCDs mit 2 Controller)

Bei Display mit Hintergrundbeleuchtung:

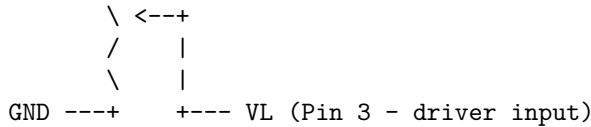
HG+	15 (mit Vorwiderstand ca. 200hm)
GND	16

An Pin 3 kann der Abgriff eines $\geq 20k\Omega$ Potis zwischen +5V und GND geschaltet werden. Damit kann der Kontrast des Displays reguliert werden. Bei meinem Display (Conrad) liegt Pin 3 direkt an Masse und man kann alles einwandfrei erkennen.

```

+5V ----+
        /

```

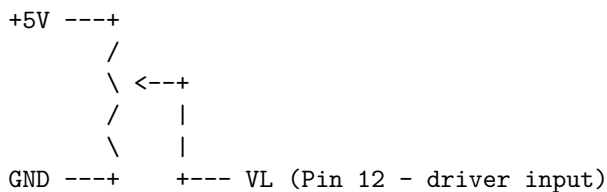


1.1.5 Anschluss eines 4x40 Displays

Da sich der Anschluss eines 4x40 Displays stark von anderen Displays unterscheidet, hier ein Beispiel (Conrad - NLC-40x4x05):

Paralellport-Pin	Beschreibung	LCD-Modul	LCD-Pin
18-25			--
	GND		13 -- - Brücke
	R/W		10 --
	+5V		14
1	STROBE	EU (Enable-Upper)	9
2	D0	D0	8
3	D1	D1	7
4	D2	D2	6
5	D3	D3	5
6	D4	D4	4
7	D5	D5	3
8	D6	D6	2
9	D7	D7	1
14	Autofeed	RS	11
17	Select In	ED (Enable-Down)	15

An Pin 12 kann der Abgriff eines $\geq 20k\Omega$ Potis zwischen +5V und GND geschaltet werden. Damit kann der Kontrast des Displays reguliert werden. Es kann aber auch reichen, Pin 12 direkt an Masse zu legen um alles einwandfrei erkennen zu können.



- Die Leitung ED wird an Pin 17 des parallelen Ports angeschlossen.
- Das Display wird in der lcd.txt als 2x40 Display definiert.
- Bei den Typdefinitionen für isdn_rate wird aber die 4x40 als Zeilen-/Spaltengröße angesehen.

Leider gibt es keinen Standard, was die Pinbelegung des Parallelports auf dem Motherboard betrifft. Für die interne Verwendung von LCD-Modulen muß man also die Anschlüsse anhand des zum Motherboard mitgelieferten Slotblechadapters durchmessen.

Die erforderliche Spannungsversorgung kann man leider nicht dem Parallelport entnehmen, da die Stromaufnahme eines LCD-Modules zu hoch ist. Geeignet dafür sind die Anschlüsse

für Maus (PS/2), Tastatur (DIN, PS/2), Gameport, USB oder ein freier Anschluss vom PC-Netzteil. Da einige Soundkartenhersteller am Gameport spezielle Signale generieren, kann keine Garantie übernommen, daß es in jeder Kombination funktioniert. Daher gilt hier: Immer vorher messen!

1.1.6 Danksagung

Dank geht an:

- Frank Meyer für das imond Interface und Fli4l :)
- Gernot Miksch für das Paket LCD
- Michael Reinelt (<https://ssl.bulix.org/projects/lcd4linux>) für das LCD4Linux Programm

Für Fragen, Anregungen Kritik usw.:

Schreibt eine Email an Christian Brüggemann

A Anhang zum Paket LCD4Linux

Index

LCD4LINUX_CTINCLUD_DRIVER, [7](#)
LCD4LINUX_CTINCLUD_OPTIONS, [8](#)
LCD4LINUX_CTINCLUD_ROTATE, [8](#)
LCD4LINUX_DISPLAY_SIZE, [8](#)
LCD4LINUX_DSL_SPEED_IN, [8](#)
LCD4LINUX_DSL_SPEED_OUT, [8](#)
LCD4LINUX_EXPERT_MODE, [10](#)
LCD4LINUX_FW8888_DRIVER, [8](#)
LCD4LINUX_GLCD2USB_BRIGHTNESS,
[7](#)
LCD4LINUX_GLCD2USB_DRIVER, [6](#)
LCD4LINUX_HD44780_ASC255BUG, [5](#)
LCD4LINUX_HD44780_BACKLIGHT, [6](#)
LCD4LINUX_HD44780_BITS, [6](#)
LCD4LINUX_HD44780_BRIGHTNESS, [6](#)
LCD4LINUX_HD44780_BUS, [4](#)
LCD4LINUX_HD44780_CONTRAST, [6](#)
LCD4LINUX_HD44780_CONTROLLERS,
[5](#)
LCD4LINUX_HD44780_DRIVER, [4](#)
LCD4LINUX_HD44780_I2C_DEV, [5](#)
LCD4LINUX_HD44780_MODEL, [4](#)
LCD4LINUX_HD44780_PORT, [4](#)
LCD4LINUX_HD44780_USEBUSY, [6](#)
LCD4LINUX_HD44780_WIRE_TYPE, [5](#)
LCD4LINUX_HP12542R_BACKLIGHT, [7](#)
LCD4LINUX_HP12542R_CONTRAST, [7](#)
LCD4LINUX_HP12542R_DRIVER, [7](#)
LCD4LINUX_HP12542R_OPTIONS, [7](#)
LCD4LINUX_HP12542R_PORT, [7](#)
LCD4LINUX_HP12542R_ROTATE, [7](#)
LCD4LINUX_ICONS, [8](#)
LCD4LINUX_LAYOUT_N, [8](#)
LCD4LINUX_LCD2USB_BRIGHTNESS, [6](#)
LCD4LINUX_LCD2USB_CONTRAST, [6](#)
LCD4LINUX_LCD2USB_DRIVER, [6](#)
LCD4LINUX_TEST, [10](#)

OPT_LCD4LINUX, [3](#)