

chelp | C - Linux - Arduino - Raspberry

Inhaltsverzeichnis

chelp | C - Linux - Arduino - Raspberry

Homepage	1
Allgemeine Beschreibungen	1
Dokumentation: C-Programme und Bibliotheken	1
chelp Projektbrowser für Projekt c/	
Programm chelp einrichten	2
Startcheck	2
Einstellungen	3
Hilfestellungen und Beschreibungen	
Readme's	7
Dokus und Anleitungen	7
Linux	7
Links	7
Stichworte	7
Bibliotheken	7
Funktionstasten	
Farben und Tasten	8
UTF-8 Codes	9
www-Links	9
C Programme	
Programme/Libraries anlegen/compilieren/ausführen	10
Bibliotheksfunktionen	10
chelp konfigurieren	
Programmeinstellungen	12
Stichwortliste ändern	13
Weitere Hilfsprogramme	
Arduino	17
Geany	17
Screen	17
GNU General Public License	

Homepage

Homepage und Downloads: www.projektc.at

Allgemeine Beschreibungen

Die allgemeinen Dokumentationen findet man unter [c/1_Dokus](#) oder im Internet:

Vorwort:	www.projektc.at/programs/c/clar_vorwort.pdf	c/1_Dokus/clar_vorwort.pdf
Projekt c/ einrichten. Die ersten Schritte:	www.projektc.at/programs/c/clar_start.pdf	c/1_Dokus/clar_start.pdf
Projekthilfe und Projektmanager:	www.projektc.at/programs/c/clar_chelp.pdf	c/1_Dokus/clar_chelp.pdf
Ein neues C Programm erstellen:	www.projektc.at/programs/c/clar_projekt.pdf	c/1_Dokus/clar_projekt.pdf
Basisobjekte ohne Terminal In/Output:	www.projektc.at/programs/c/clar_objekte1.pdf	c/1_Dokus/clar_objekte1.pdf
Terminalsteuerung Box-Objekte für In/Output:	www.projektc.at/programs/c/clar_objekte2.pdf	c/1_Dokus/clar_objekte2.pdf
Datenspeicherung:	www.projektc.at/programs/c/clar_objekte3.pdf	c/1_Dokus/clar_objekte3.pdf

Dokumentation: C-Programme und Bibliotheken

- ▷ Die ersten Infos zu den C-Programmen oder Bibliotheken findet man in der Hilfedatei '1_read.me' im jeweiligen Programmordner.
- ▷ Für aufwendige Programme gibt es Beschreibungen im Format *.odt oder *.pdf im Ordner `name/bin/_name/`
- ▷ Die Dokumentation des Programmcodes befindet sich in den C-Headern der Programme oder Bibliotheken.
- ▷ Hilfe zu den fertigen Programmen liefert immer die Startoption '-h'.

Einstiege:

Programm chelp	Menügesteuerter Zugriff auf alle Dokus
Projekt c/ Einstieg und Übersicht	c/1_read.me
Header/Dokus für Bibliotheksfunktionen	c/lib/1_read.me
Testprogramme für Bibliotheksfunktionen	c/libtest/1_read.me

chelp | Projektbrowser für Projekt c/

www.projektc.at/programs/c/clar_chelp.pdf

Der Projektbrowser **chelp** ist eine gut eingerichtete Werkstatt zum Entwickeln von einfachen oder komplexen C-Programmen. Die gesamte Werkstatt befindet sich in einem beliebigen Ordner unter **c/**.

Die Tools aus **chelp** können den ganzen Entwicklungsprozess begleiten: Werkstatt anlegen, IDE geany einstellen, Informationssuche intern und extern, Stichwortverzeichnis, Dokus zu Bibliotheken und Programmen, Versionskontrolle, Programmgerüste anlegen, Programme Compilieren, Startlinks, Projektlinks, Makefiles anlegen, Fehlersuche, UTF-8 Codes anzeigen, usw.

Programm chelp einrichten

Hilfsprogramm chelp bereitstellen:

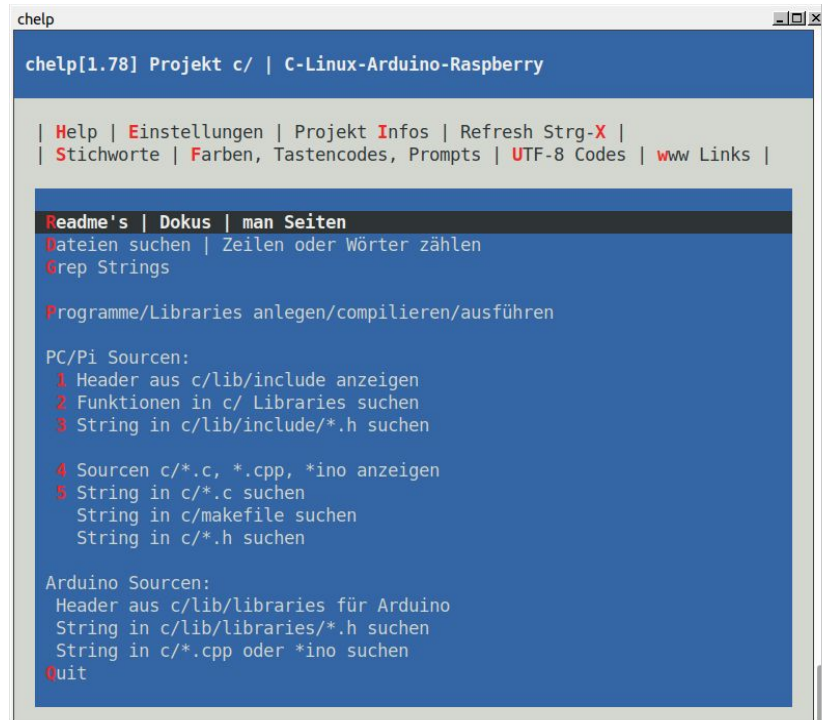
Ins Projektverzeichnis **/home/pi/c** oder **/home/user/c** wechseln.

Mit Befehl **make chelp** wird das Programm kompiliert und automatisch gestartet.

Im **Startcheck** können die Einstellungen geprüft und geändert werden.

Der Startcheck kann auch über den Dialog **Einstellungen** aufgerufen werden.

Danach kann das Programm **chelp** in jedem Ordner ohne Pfadangabe gestartet werden.



Startcheck

Beim Start von **chelp** erfolgt automatisch ein Startcheck zu Überprüfung der Einstellungen.

Im Dialog **Einstellungen** kann dieser Startcheck manuell aufgerufen werden. Dort können die verschiedenen Einstellungen auch angepasst werden.

▷ **Projektumgebung prüfen:** Es wird der aktuelle Projektordner DIRC mit dem im Setup gespeicherten Ordner verglichen.

▷ **Pfad zum Projektordner c/ prüfen:** Schreibzugriff auf den Projektordner und Lesezugriff auf **chelp.c** prüfen.

▷ **Startlinks:** Der Linkordner **~/bin** wird angelegt/geprüft und der Startlink für **chelp** wird angelegt/geprüft. Variable \$PATH wird beim normalerweise beim Anmelden automatisch ergänzt. Mit dem angelegten Startlink kann das Programm nun in jedem Ordner mit **chelp** werden.

▷ **Texteditor für Console und X-Terminal:** Die Verfügbarkeit des eingestellten Editors prüfen.

▷ **Texteditor für X:** Die Verfügbarkeit des eingestellten Editors prüfen..

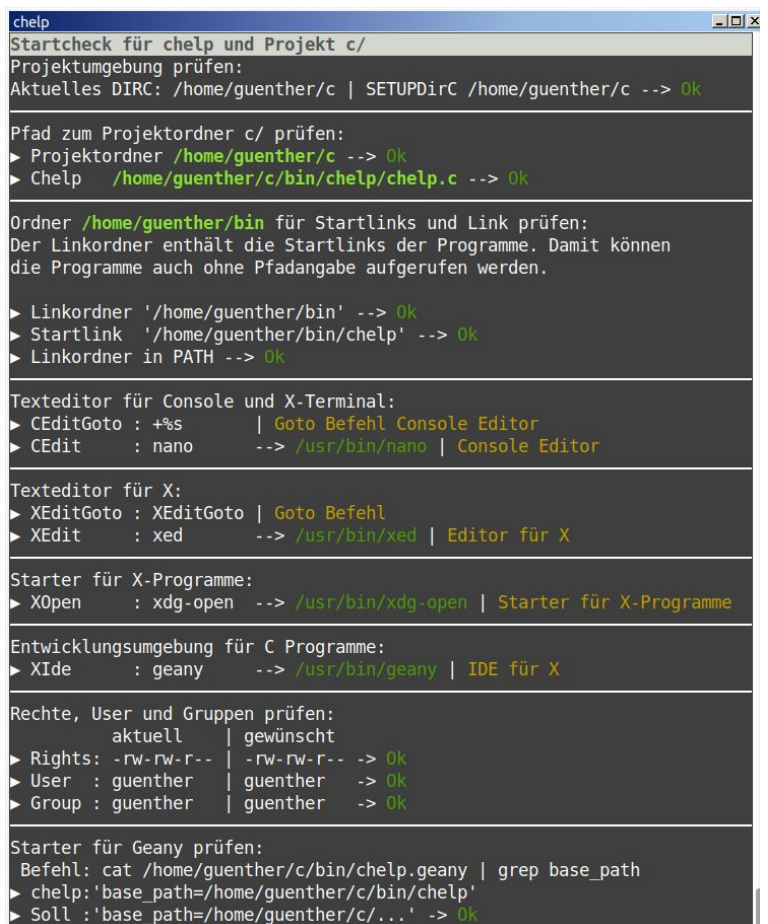
▷ **Starter für X-Programme:** Die Verfügbarkeit des eingestellten Editors prüfen.

▷ **Entwicklungsumgebung für C Programme:** Die Verfügbarkeit des eingestellten Editors prüfen.

▷ **Rechte, User und Gruppen prüfen:** Einstellen unter **Einstellungen > Rechte, User ...**

▷ **Starter für Geany prüfen:** Unter **Einstellungen > Geany Startdateien** können die Startdateien für Geany der aktuellen Umgebung angepasst werden.

Der Startcheck sollte fehlerfrei ablaufen!



Einstellungen

Befehl: **Einstellungen**

Menu zum Überprüfen der Einstellungen von **chelp** und dem gesamten Projekt **/c/**,

Befehl: **Startcheck für chelp und Projekt /c/**
siehe oben: [Startcheck](#)

```
chelp
Einstellungen für Projekt c/

Einstellungen für Projekt c/ und die Projektumgebung prüfen/anpassen.
Der Startcheck sollte fehlerfrei ablaufen!

Startcheck für chelp und Projekt /c/ | Links und Hilfsprogramme für /c/ prüfen
Rechte, User und Gruppe einstellen | Für alle Projektdateien aktualisieren
Geany Startdateien anpassen | Dateipfade zum Projektordner einstellen
Links in ~/bin prüfen | Programmstartlinks prüfen/anlegen
Make Dateien prüfen | Alle makefile's prüfen
Projektlinks prüfen | Alle symbolischen Links unter c/ prüfen

Optionale Projekteinstellungen | Libraries und Hilfsprogramme prüfen
Infos zu Programm chelp | Aktuelle Programmvariablen anzeigen
Versionskontrolle | Programmversionen anzeigen

Konfiguration von chelp bearbeiten | Datei chelp.conf
Stichwortliste für chelp bearbeiten | Datei chelp.keys

Befehl | ESC ?
```

Befehl: **Rechte, User und Gruppe einstellen**

Diese Einstellungen können durch geänderte User notwendig werden.

```
chelp
Rechte und Besitzer
ls -lR --color=always | Projektordner /home/guenther/c
.:
insgesamt 68
drwxrwxr-- 2 guenther guenther 4096 Okt 13 15:10 1_Dokus
-rw-rw-r-- 1 guenther guenther 15679 Okt 8 18:39 1_read.me
-rw-rw-r-- 1 guenther guenther 174 Okt 13 08:55 2_todo.txt
drwxrwxr-- 6 guenther guenther 4096 Nov 19 2021 arduino
drwxrwxr-- 22 guenther guenther 4096 Okt 13 09:32 bin
drwxrwxr-- 7 guenther guenther 4096 Okt 13 09:32 bindemo
drwxrwxr-- 25 guenther guenther 4096 Okt 13 09:32 bsp
-rw-rw-r-- 1 guenther guenther 1450 Okt 8 14:17 GNU_Licence
drwxrwxr-- 8 guenther guenther 4096 Okt 13 10:06 lib
drwxrwxr-- 22 guenther guenther 4096 Okt 13 09:32 libtest
-rw-rw-r-- 1 guenther guenther 5358 Okt 9 08:38 makefile
drwxrwxr-- 6 guenther guenther 4096 Sep 7 20:03 pi
/tmo/less4223 lines 1-16/3275 1%
```

Befehl: **Geany Startdateien anpassen**

Die Geany Entwicklungsumgebung kann für ein bestimmtes Programm mit einer Startdatei aufgerufen werden.

Beispiel: **chelp.geany**

Der Aufruf stellt die zuletzt in **geany** verwendete Entwicklungsumgebung für **chelp** wieder her.

Bei einer Änderung des Projektordners von **c/** funktionieren die Pfade in ***.geany** nicht mehr.

Der Befehl **Alle Starter prüfen/updates** korrigiert alle Pfade.

```
chelp
chelp: geany | Pfade in den Startdateien anpassen

Die Startdateien '*.geany' für Geany enthalten Projektpfade.
Bei geändertem Projektordner c/ können alle Projektpfade in
den Startdateien aktualisiert werden.

Projektordner c/: /home/guenther/c

Check Geany-Starter für chelp
Update Geany-Starter chelp

Startdateien von Projekt c/ anzeigen
Alle Starter prüfen/updates
```

Befehl: **Links in ~/bin prüfen**

Mit Startlinks in **~/bin** können die Programme ohne Pfadangabe gestartet werden.

Nach einer Änderung des Projektordners **c/** funktionieren die Startlinks in **~/bin** nicht mehr.

Der Befehl **Make Startlinks** erzeugt zu allen compilierten Programmen aus **bin/** und **pi/bin** passende Startlinks.

```
chelp
chelp: Startlinks für Programme anlegen

Zum Aufruf der Programme ohne Pfad kann man Startlinks anlegen.
Der Linkordner mit den Startlinks wird in PATH eintragen.
Linkordner: /home/guenther/bin

Linkordner prüfen
Startlinks anzeigen
Programmpfade anzeigen
Make Startlinks
Remove Startlinks

Befehl | ESC ?
```

Befehl: **Make Dateien prüfen**

Alle makefile's anzeigen und prüfen.

Befehl: **Projektlinks prüfen**

Alle symbolischen Links unter **c/** prüfen. Es sind nur relative Links erlaubt.

Befehl: **Optionale Projekteinstellungen**

Zeigt die eventuell benötigten externen Bibliotheken und Linuxprogramme für alle C-Programme.

Bibliotheken müssen nur verfügbar sein, wenn sie vom C-Programm auch benötigt werden!

Fehlende Bibliotheken werden beim Compilieren gemeldet.

Es wird die Verfügbarkeit der konfigurierten Hilfsprogramme und Bibliotheken überprüft.

Die gewünschten Einstellungen zum Projekt c/ werden in der [Konfiguration](#) definiert.

```
chelp
Optional Projekteinstellungen

Programmumgebung von chelp
Projektordner : /home/guenther/c
Startlinks    : /home/guenther/bin
PATH          : /home/guenther/bin:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/

Shell         : /bin/bash
HostName      : pc780mint
PID           : 4223
PPID          : 2215
Terminal      : xterm-256color
TTYName       : pts/0
SSH Client    : (null)
Screen Sitzung : (null)

Konfiguration: Externe Libs für einzelne C Programme
Fehlende Bibliotheken aus 'usr/include' werden beim Compilieren gemeldet!

> Lib[ 0]: alsa/asoundlib.h ->/usr/include/alsa/asoundlib.h
           Lib für ALSA Programme, Raw-Midi | sudo apt-get install libasound2-dev
> Lib[ 1]: libpng/png.h    ->/usr/include/libpng/png.h
           png Bilder lesen und anzeigen | sudo apt-get install libpng-dev
> Lib[ 2]: jpeglib.h       ->/usr/include/jpeglib.h
           jpg Bilder lesen und anzeigen | sudo apt-get install libjpeg-dev
> Lib[ 3]: termios.h       ->/usr/include/termios.h
           The termios functions describe a general terminal interface
> Lib[ 4]: termcap.h       ->'/usr/include/termcap.h' nicht gefunden
           Terminal Infos

Konfiguration: Optionale Hilfsprogramme
Diese Hilfsprogramme werden von einzelnen Programmen benötigt.

> Prog[ 0]: geany          -> /usr/bin/geany | C Entwicklungsumgebung | sudo apt-get
> Prog[ 1]: shasum         -> /usr/bin/shasum | Prüfsummen berechnen für Programm sav
> Prog[ 2]: screen        -> /usr/bin/screen | Multiplexer für ssh Fernbedienungen |
> Prog[ 3]: i2c-tools      -> nicht gefunden | Pi, i2c Schnittstelle | sudo apt-get
> Prog[ 4]: timidity       -> /usr/bin/timidity | MIDI-to-WAVE converter and player |
> Prog[ 5]: qsynth         -> /usr/bin/qsynth | A fluidsynth Qt GUI Interface | sudo

/tmo/less4223 lines 1-41/49 79%
```

Befehl: **Infos zum Programm chelp**

Die aktuellen Laufzeitwerte der Konstanten und Laufzeit-Variablen von [chelp](#).

```
chelp
Infos zu chelp

Programmkonstanten von chelp
Defaulteinstellungen werden von der Konfiguration überschrieben!

CEDITDef      : nano          | Editor Console
CEDITGotoDef  : +%s          | Editor Goto
XIDEDef       : geany         | Entwicklungsumgebung
XOPENDef      : xdg-open      | X Programmstarter

CWD           : /home/guenther/c/bin/chelp/bin
isXRRunning() : true
isPi()        : false
MAXTmpStr     : 300
MALLOCTRACE   : (null)
AbsPathKeys   : /home/guenther/c/bin/chelp/bin/_chelp/chelp.keys
AbsDirHeader  : /home/guenther/c/lib/include
AbsDirDokus   : /home/guenther/c/l_dokus

Var-Gruppe: Programm
Nr | Grp/Sub | Name          | Typ | Wert
000 | 0/ 0    | ProgName      | Str | "chelp"
001 | 0/ 0    | WorkDir       | Str | "/home/guenther/c/bin/chelp/bin"
002 | 0/ 0    | CEdit        | Str | "nano"
003 | 0/ 0    | CEditGoto    | Str | "+%s"
004 | 0/ 0    | XEdit        | Str | "xed"
005 | 0/ 0    | XEditGoto    | Str | "+%s"
006 | 0/ 0    | XOpen        | Str | "xdg-open"
007 | 0/ 0    | XIde         | Str | "geany"
008 | 0/ 0    | DirC         | Str | "/home/guenther/c"
009 | 0/ 0    | DirSrcLast   | Str | "/home/guenther/c"
010 | 0/ 0    | Config       | Str | "/home/guenther/c/bin/chelp/bin/_chelp/chelp.keys"
011 | 0/ 0    | SetupHost    | Str | "pc780mint"
012 | 0/ 0    | SetupUser    | Str | "guenther"
013 | 0/ 0    | SetupDirC    | Str | "/home/guenther/c"
014 | 0/ 0    | PathKeys     | Str | "chelp.keys"
015 | 0/ 0    | SuffixeH     | Str | "*.h"
016 | 0/ 0    | SuffixeC     | Str | "*.h,*.c,*.cpp,*.ino"
017 | 0/ 0    | SuffixeTxt   | Str | "*.h,*.c,*.me,*.txt,*.cpp,*.ino,*.geany,*.c"
018 | 0/ 0    | DirLinHeader | Str | "/usr/include/linux"
019 | 0/ 0    | DirHeader    | Str | "lib/include"
020 | 0/ 0    | DirHeaderArd | Str | "lib/libraries"
021 | 0/ 0    | DirDokus     | Str | "l_dokus"
022 | 0/ 0    | DirPiBinInc  | Str | "pi/bininc"

/tmo/less4223 lines 1-45/102 37%
```

Befehl: Versionskontrolle

Die Versionsnummern und Programmversionen von allen Programmen aus c/ auslesen.

```
chelp
Die Versionsnummern aller Programme anzeigen!
bin/kbdctl/kbdctl.h      [69] 0.53 2024-09-11 Programmversion
bin/infosys/infosys.h   [27] 2.25 2024-10-01 Programmversion
bin/mtrainer/mtrainer.h [37] 0.60 2020-08-09 Programmversion
bin/pshow/pshow.h       [54] 1.25 2022-02-01 Programmversion
bin/xmlextract/xmlextract.h [47] 0.04 2020-03-17 Programmversion
bin/newprg/newprg.h     [41] 0.18 2024-09-05 Programmversion
bin/ffvideo/ffvideo.h   [35] 1.14 2024-09-09 Programmversion
bin/mid2midraw/mid2midraw.h [43] 0.02 2020-01-05 Programmversion
bin/saveit/saveit.h     [37] 1.12 2024-09-28 Programmversion
bin/chelp/chelp.h       [31] 1.72 2024-10-08 Programmversion
bin/midixf/midixf.h     [52] 0.02 2020-03-06 Programmversion
bin/lpctl/lpctl.h       [45] 0.03 2024-09-09 Programmversion
bin/lpctl/lpctl.h       [46] 0.03 2024-09-09 Programmversion
bin/chksums/chksums.h   [86] 0.18 2022-01-30 Programmversion
bin/fbp/fbp.h          [25] 1.32 2023-05-04 Programmversion
bin/usbaddrun/usbaddrun.h [51] 0.13 2024-09-25 Programmversion
bin/dbebau/dbebau.h     [49] 0.22 2024-06-12 Programmversion
libtest/test_system/testsystem.h [39] 0.00 2023-05-04 Programmversion
libtest/test_info/test.h [47] 0.06 2020-01-18 Programmversion
libtest/test_keys/testkeys.h [22] 0.03 2020-03-17 Programmversion
arduino/sun_900/sun_900/sun900.h [12] 0.80 2016-02-16 Programmversion
arduino/sun_900/sun_900a/sun_900.h [14] 0.40 2016-02-15 Programmversion
pi/bin/piclock/piclock.h [68] 0.17 2024-09-08 Programmversion
pi/bin/picamctl/picamctl.h [52] 0.64 2024-09-18 Programmversion
pi/bin/devtest/devtest.h [54] 0.24 2022-06-06 Programmversion
pi/bin/gardenctl/gardenctl.h [59] 0.53 2022-11-03 Programmversion
pi/bin/gardendown/old/gardendown.h [22] 0.08 2021-06-16 Programmversion
pi/bin/gardendown/gardendown.h [24] 0.13 2021-10-26 Programmversion
vorlagen/myprog_con/myprog.h [45] 0.00 2024-09-05 Programmversion
vorlagen/myprog_min/myprog.h [18] 0.00 2020-03-17 Programmversion

Weiter mit Taste
```

Befehl: Konfiguration von chelp bearbeiten

Die Konfigurationsdatei von chelp ist chelp.conf. Das Format ist einfaches C. Diese Format wird von Projekt c/ für alle Setupdateien, Skripte und zur Speicherung von Arrays verwendet. Zur Bearbeitung wird der dort konfigurierte Texteditor aufgerufen.

Die Daten von chelp.conf werden beim Programmstart geparkt und direkt in die Laufzeitvariablen von Objekt Var eingelesen. Die vom Programm geänderten Variablen können im gleichen Format wieder nach geschrieben chelp.conf werden. Kommentare bleiben erhalten.

Es ist daher nicht ganz trivial die Konfiguration im laufenden Programm zu ändern. Linux bietet aber auch dazu eine Lösung an.

Details zu chelp.conf siehe Abschnitt [Programmeinstellungen](#)

```
chelp
/home/guenther/c/bin/chelp/bin/ chelp/chelp.conf
// =====
// 2020-06-15 Konfiguration für chelp für Pi und PC
//
// Aufrufoptionen: chelp -h  Hilfe
//
// =====
// Die Allgemeine Einstellungen werden
// in der Programmgruppe 0/0 abgelegt.
VarSetGrpFlt(0,0);

PathKeys  ="chelp.keys"; // Stichwortliste im gleichen
                        // Ordner wie chelp.conf

// Dateiendungen -----
// Trennzeichen ist ','. Beispiel: "*.c,*.cpp"
//
SuffixeH   ="*.h";
SuffixeC   ="*.h,*.c,*.cpp,*.ino";
SuffixeTxt ="*.h,*.c,*.me,*.txt,*.cpp,*.ino,*.geany,*.conf" ;

// Ordner -----
// Relative Pfadangaben relativ zum Homeordner
//
DirLinHeader  ="/usr/include/linux"; // Linux Header
DirHeader     ="lib/include";        // C Header für PC/Pi
DirHeaderArd  ="lib/libraries";      // Arduino Header
DirDokus      ="1_Dokus";            // Dokumentation
DirPiBinInc   ="pi/bininc";          // Includes für Pi Steuerungen

// Konfigurationsgruppe für verschiedene Systeme -----

^G Hilfe      ^O Speichern  ^W Wo ist      ^K Ausschneide ^J Ausrichten
^X Beenden    ^R Datei öffne ^E Ersetzen   ^U Text einfüg ^T Rechtschr.
```

Befehl: **Stichwortliste für chelp bearbeiten**

Die Stichwortliste **chelp.keys** hat wieder das Format einfaches C. Für komplexe Datenstrukturen wird immer das C-Format **struct** verwendet.

Details zu **chelp.keys** siehe
Abschnitt [Stichwortliste ändern](#)

Beschreibung der Stichwortliste

```
chelp.keys (~c/bin/chelp/bin/_chelp)
Datei Bearbeiten Ansicht Suchen Werkzeuge Dokumente Hilfe
chelp.keys x
// =====
// Stichwortliste für chelp | 2023-02-06
//
// Alle Pfadangaben sind relativ zu c/ oder absolut.
// Für die Items vom Typ h,g und v sind Wildcards erlaubt. Z.B. "*.pdf"
// Die Angaben [...] sind optional.
// Kommentare //... am Ende eines Stichworteintrag werden angezeigt.
//
// Aufbau eines Eintrags der Stichwortliste:
//
// {"Stichwort-Info-Text",      Beschreibung des Eintrags
// {t, "Text"                  }, "Text" anzeigen
// {h, "Datei", ["Suchstring"]}, Header aus Ordner include
// {g, "Pfad",  "Suchstring" }, Dateien mit grep suchen
// {v, "Pfad",  ["Nr"]}       }, Viewer aufrufen. Zeilen "Nr"
// {e, "Pfad",  ["Options"]}   }, Execute Pfad mit Options
// {m, "Datei", ["Nr"]}       }, Manpage Datei. Seite Nr
// ...
// };
// ...
// {"Stichwort-Info-Text",      Beschreibung des Eintrags
// { ... },
// ...
// };

{"Dokus zum Projekt c/",
//{t,"|Dokus zum c Projekt"},
{v,"1_Dokus/clar_start.pdf","1"}, // Projekt c/ einrichten
{v,"1_Dokus/clar_chelp.pdf","1"}, // Hilfe, Programm chelp
{v,"1_Dokus/clar_projekt.pdf","1"}, // Neues C Programm anlegen
{v,"1_Dokus/clar_objekte.pdf","1"}, // Beispiele c/ Objekte
{v,"1_Dokus/clar_alsa.pdf","1"}, // Dokus Alsa
{t,""},
{v,"pi/1_Dokus/clar_raspberry.pdf","1"}, // Beispiele Raspberry
{t,""},
{v,"1_Dokus/c_linux.pdf","1"}, // C Bemerkungen
{v,"1_Dokus/Xterm Control Sequences.pdf","1"}, // Xterm Control
Sequences
};

{"Programme in c/bin/",
{g,"1_read.*","bin|"}, // Readme's aller Programme anzeigen
{t,""},
{t,"....."},
{t,"chelp: Hilfe zum Projekt c/"},
{v,"bin/chelp/bin/_chelp/1_read.me","1"}, // Hilfe anzeigen
{e,"bin/chelp/bin/chelp"}, // Programm
{t,""},
```

Hilfestellungen und Beschreibungen

Beschreibungen und Dokus findet man im Menu: [Readme's](#) | [Dokus](#) | [man Seiten](#)

Readme's

In fast allen Verzeichnissen befinden sich Beschreibungen zum jeweiligen Ordner mit dem fixen Namen [1_read.me](#).
Format: UTF8 Text.

Dokus und Anleitungen

Dokus zum Projekt c/ im Format pdf und odt befinden sich im Ordner [c/1_Dokus](#).

Einstieg ins Projekt c/:

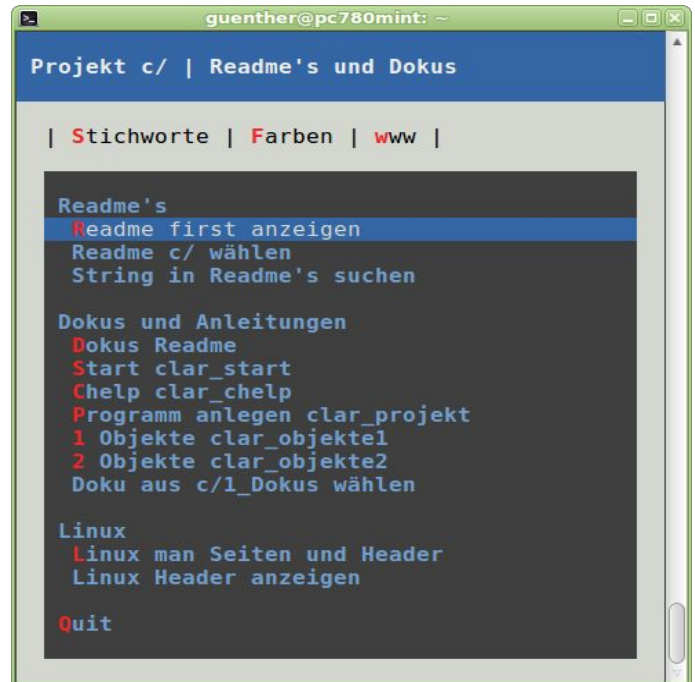
[Start clar_start.pdf](#)

Linux

Linux man Seiten und Header.

Links

Linkliste aus der Konfiguration anzeigen und aufrufen.
Funktionstaste: [www](#)



Stichworte

Funktionstaste: [Stichworte](#)

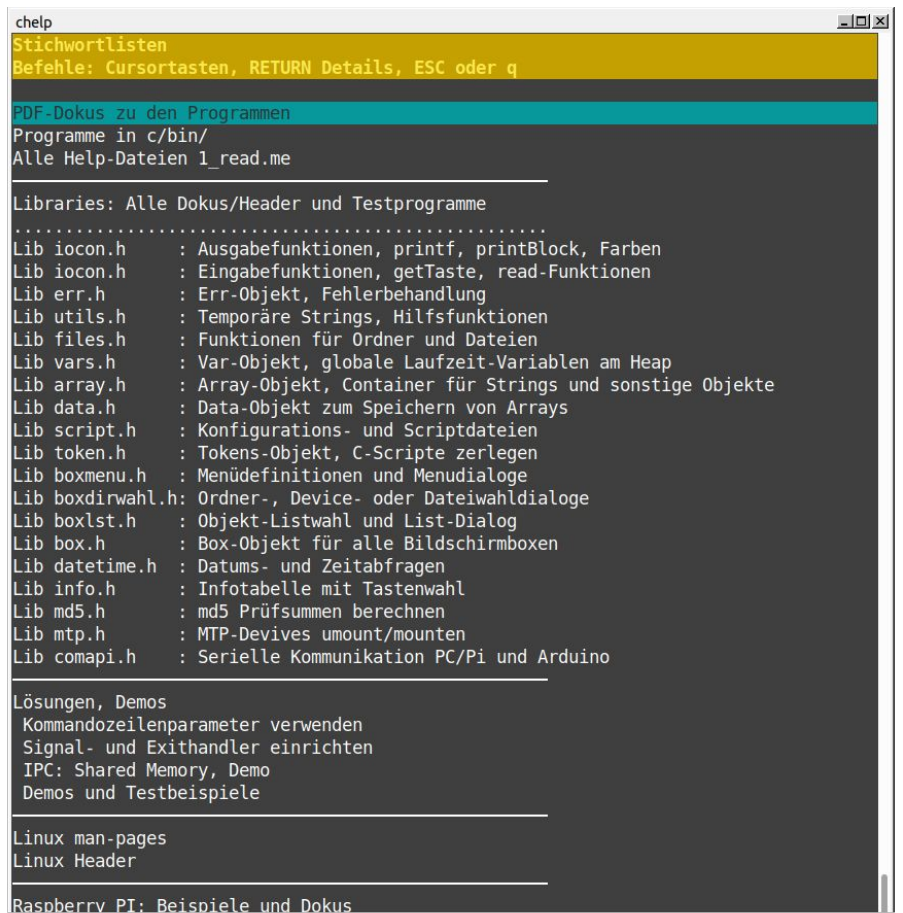
Die Stichwortliste liefert verschiedene Infos zum Projekt c/ und kann auch Programme ausführen.

Die Liste wird beim Aufruf von [chelp](#) aus der Datei [_chelp/chelp.keys](#) neu erstellt.

Die Ergebnisse spiegeln daher immer den aktuellen Stand des Projekts c/ wieder.

Die Einträge der Stichwortliste [_chelp/chelp.keys](#) können angepasst werden.

Siehe [Stichwortliste ändern](#).



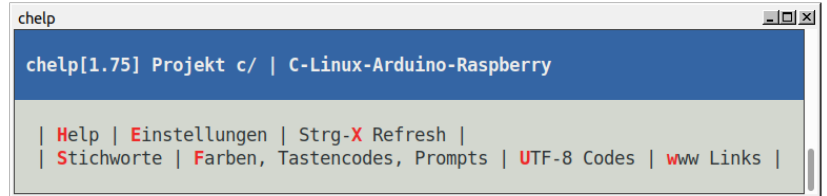
Bibliotheken

Für die Funktionen aus den statischen Bibliotheken gibt es direkte Hilfe im Hauptmenu.
Siehe [Bibliotheksfunktionen](#).

Funktionstasten

Im Hauptmenu:

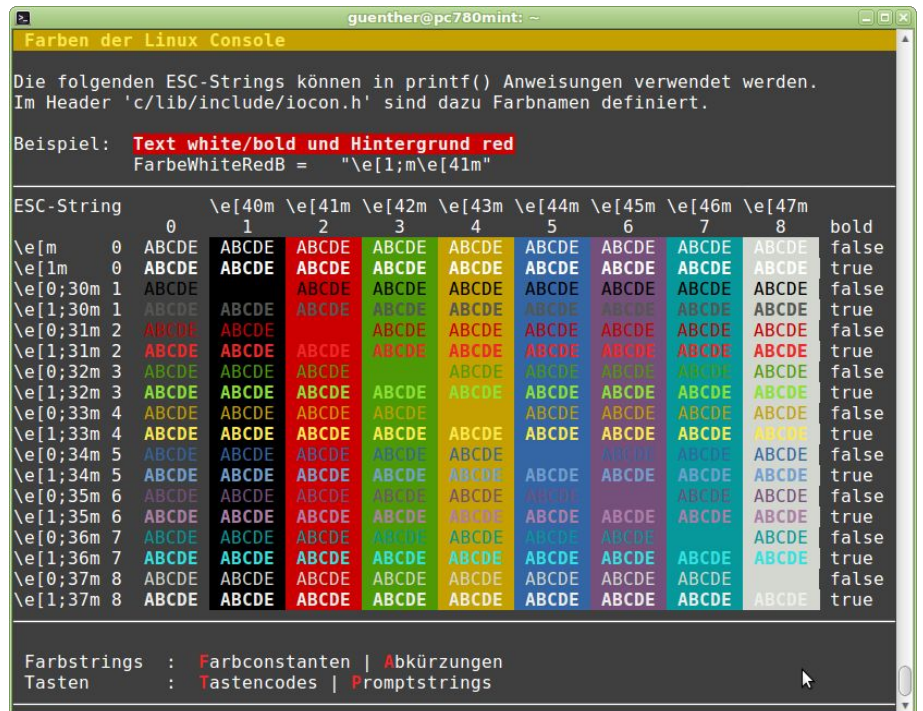
[Help](#) Datei 1_read.me anzeigen
[Einstellungen](#) [Einstellungen](#) prüfen/anpassen
[Stichworte](#) [Stichwortliste](#), Infos anzeigen
[Farben, ...](#) [Farben, Tastencodes und Prompts](#)
[UTF-8 Codes](#) Zeichentabelle, Rahmen Beispiele
[www Links](#) Eigene www-Links



Farben und Tasten

Befehl: **Farben**

Die Farbsteuerung der Terminalausgaben erfolgt mit ESC-Strings.



Befehl: **Farbkonstanten**

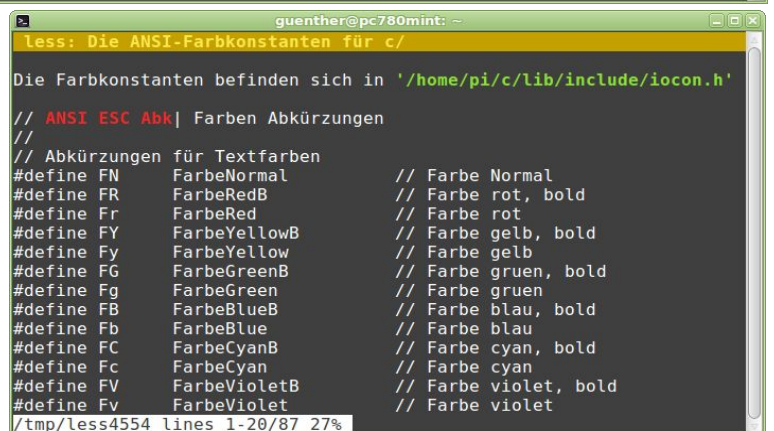
Alle vordefinierten Farbkonstanten.

Befehl: **Abkürzungen**

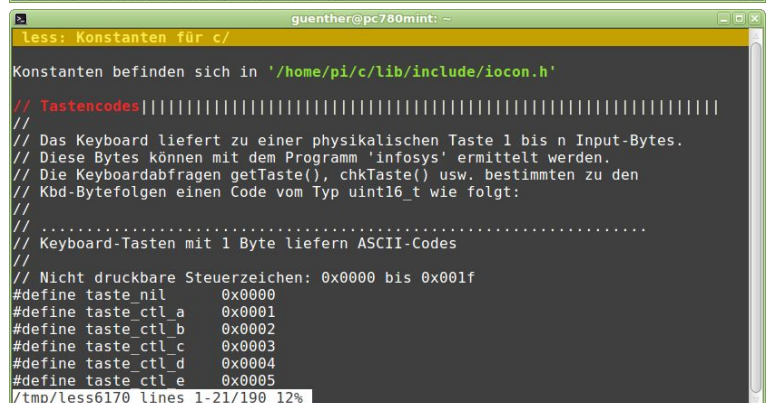
Beispiel für die Farbausgabe:

```
printf( FG" Achtung: "FR" Fehler!\n"FN);
```

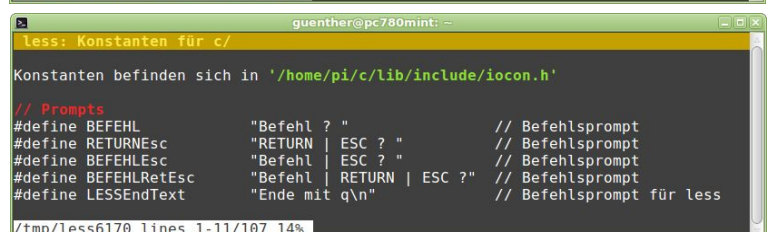
Ausgabe im Terminal: **Achtung: Fehler!**



Befehl: **Tastencodes**



Befehl: **Promptstrings**



C Programme

Programme/Libraries anlegen/compilieren/ausführen

Dialog zum Einstellen der Projektumgebung für C Programme. Programme compilieren, ausführen und editieren. Neue Programme erstellen. Die statischen Bibliotheken editieren.

Befehl: **Programme/Libraries anlegen/compilieren/ausführen**

Optionen für das gewählte Projekt/Programm:

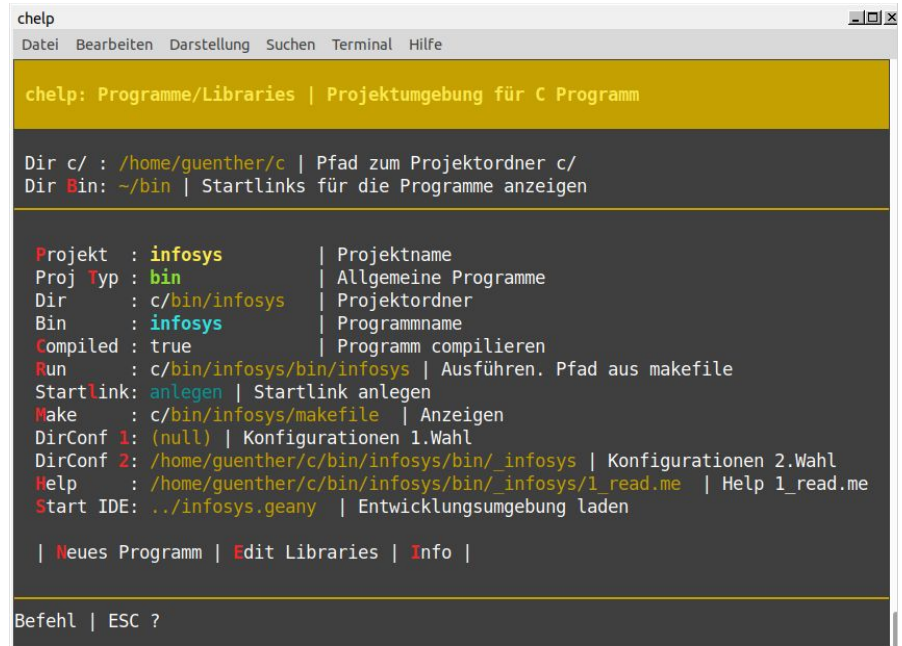
wählen
compilieren
ausführen
Startlink erstellen
Makefile anzeigen
Konfigurationsdateien anzeigen
Helpdatei anzeigen
Entwicklungsumgebung starten

Neue Programme anlegen:

Neues Programm

Bibliotheken bearbeiten:

Edit Libraries

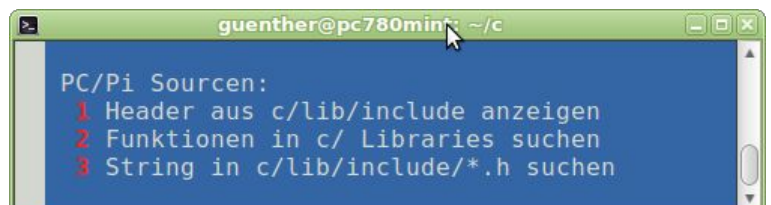


Bibliotheksfunktionen

Projekt c/ stellt über 750 - gut getestete - Library-Funktionen mit stabilen Lösungen zu vielen Programmierproblemen bereit. Die Liste reicht von brauchbaren Input- und Outputfunktionen über Strings, Fehler-Objekt, Dateidialoge, Menüs bis USB-Mount. Alle Funktionen befinden sich in frei zugänglichen und veränderbaren statischen Bibliotheken. Die Funktions-Beschreibungen findet man in den zugehörigen Headern im Ordner **c/lib/include** !

Zum Suchen und Auswählen passender Library-Funktionen dienen Funktionstasten **1, 2** und **3**. im Hauptmenu.

- 1 Header aus c/lib/include anzeigen
- 2 Funktionen in c/ Libraries suchen
- 3 String in c/lib/include/*.h suchen



Option: **1 Header aus c/lib/include**

Die Funktionsbeschreibungen befinden sich in den Headern der Bibliotheken.

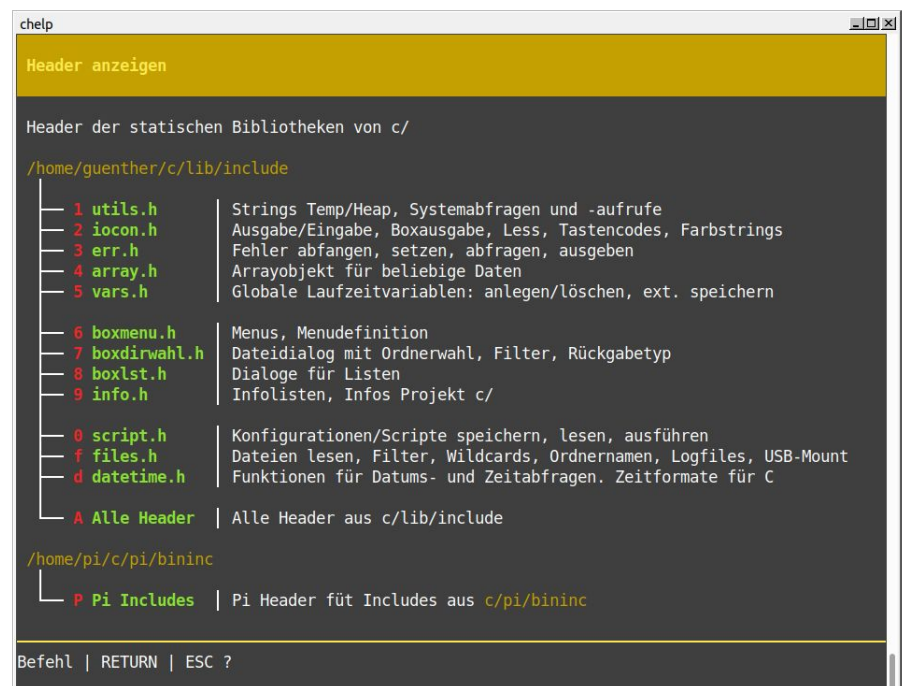
Der Heade-Dialog zeigt die wichtigsten Header mit kurzen Infos aus dem Modul Infos.

Die Header werden mit dem eingestellten Texteditor angezeigt.

Vorsicht: Die Dateien können auch geändert werden!

Beispiel: **1 utils** zeigt **utils.h** im Texteditor

Über **Alle Header** können alle verfügbaren Header gelistet werden.



Option: **2 Funktionen in c/ Libraries suchen**

Die Liste zeigt alle verfügbaren Bibliotheksfunktionen.

Gelistet werden der Funktionsname, das Modul und die Library.

Beispiel:

ArrayData	Funktionsname
array.o	Modulheader array.h
libvars.a	Library libvars

Befehle

. Suche

Öffnet eine Eingabezeile zum Suchen nach Funktionsnamen.

Header

Durchsucht alle Bibliotheks-Header nach dem angegebenen Funktionsnamen. Wildcards erlaubt.

Source

Durchsucht alle Programm-Sourcen von c/ nach dem angegebenen Funktionsnamen. Wildcards erlaubt.

RETURN

Zeigt einige Zeilen aus dem Header in der Umgebung des Suchbegriffs an.

```
chelp
Funktionen in Libraries suchen
Anzahl : 759
Befehle: ESC, RETURN, Cursortasten, . Suche,
Header oder Source durchsuchen, Library anzeigen

AddItem token.o libvars.a
AddWildcard files.o libutils.a
ArrayAdd array.o libvars.a
ArrayAddPtr array.o libvars.a
ArrayAddPtrs array.o libvars.a
ArrayAddStr array.o libvars.a
ArrayAddUInt32 array.o libvars.a
ArrayCompareDirents files.o libutils.a
ArrayCompareFileTime files.o libutils.a
ArrayCompareName files.o libutils.a
ArrayCompareNameOSfx files.o libutils.a
ArrayComparePath files.o libutils.a
ArrayCompareType files.o libutils.a
ArrayData array.o libvars.a
ArrayDelete array.o libvars.a
ArrayDeleteFileInfoItem files.o libutils.a
ArrayDeleteItem array.o libvars.a
ArrayDeleteItemFree array.o libvars.a
ArrayDeleteItemFreePtr array.o libvars.a
ArrayDeleteItemNop array.o libvars.a
ArrayDeleteItems array.o libvars.a
```

```
chelp
173- // Rückgabe: Anzahl der gültigen Datenpointer.
174- // Input NULL liefert 0.
175-
176-
177-void *ArrayData(tArray *Array, uint32_t i);
178- // Datenpointer zum Array-Index i
179- // Rückgabe: if (i<ArraySize()) Datenpointer
180- // else NULL
181-
182-uint32_t ArrayUInt32(tArray *Array, uint32_t i);
183- // nur für Index-Arrays mit uint32_t Items.
184- // Datenpointer zum Array-Index i
185- // Rückgabe: if (i<ArraySize()) uint32_t
/tmp/less11178 lines 18-30/64 44%
```

Option: **3 String in c/lib/include/*.h suchen**

Suchstring mit Wildcards.

```
chelp
String in Header *.h suchen

Include: /home/guenther/c/lib/include

Suchstring oder ESC: ArrayData
```

```
chelp
Dir : /home/guenther/c/lib/include
Suche : grep -nRI --color=always --include='*.h' "ArrayData"
Found : 1
Befehle: Cursortasten, RETURN, ESC,
Alles zeigen, Editor aufrufen
array.h 177 void *ArrayData(tArray *Array, uint32_t i);
```

Befehl: **Editor aufrufen**

```
array.h (-c/lib/include) - pluma
Datei Bearbeiten Ansicht Suchen Werkzeuge Dokumente Hilfe
Öffnen Speichern Rückgängig
array.h
155 //
156 uint32_t ArraySize(tArray *Array);
157 // Rückgabe: Anzahl der gültigen Datenpointer.
158 // Input NULL liefert 0.
159
160
161 void *ArrayData(tArray *Array, uint32_t i);
162 // Datenpointer zum Array-Index i
163 // Rückgabe: if (i<ArraySize()) Datenpointer
164 // else NULL
165
166 uint32_t ArrayUInt32(tArray *Array, uint32_t i);
167 // Datenpointer zum Array-Index i
168 // Rückgabe: if (i<ArraySize()) uint32_t
169 // else NIL
170
```

chelp konfigurieren

Programmeinstellungen

Befehl: **Einstellungen > Konfiguration von chelp bearbeiten**

Die **rot** markierten Einstellungen sollten überprüft und an das verwendete System angepasst werden.
Die Konfigurationsdatei `c/bin/chelp/bin/_chelp/chelp.conf` verwendet eine einfache C-Syntax.

```
// =====
// 2024-10-07 Konfiguration für chelp für Pi und PC
//
// Aufrufoptionen: chelp -h  Hilfe
//
// =====
// Die Allgemeine Einstellungen werden
// in der Programmgruppe 0/0 abgelegt.
VarSetGrpFlt(0,0);

SetupHost ="pc780mint";      // aktuelle Programmumgebung
SetupUser ="guenther";       // aktuelle Programmumgebung
SetupDirC ="home/guenther/c"; // aktuelle Programmumgebung

PathKeys  ="chelp.keys"; // Stichwortliste im gleichen
                        // Ordner wie chelp.conf

// Dateieindungen -----
// Trennzeichen ist ','. Beispiel: "*.c,*.cpp"
//
SuffixeH   ="*.h";
SuffixeC   ="*.h,*.c,*.cpp,*.ino";
SuffixeTxt ="*.h,*.c,*.me,*.txt,*.cpp,*.ino,*.geany,*.conf";

// Ordner -----
// Relative Pfadangaben relativ zum Homeordner
//
DirLinHeader ="usr/include/linux"; // Linux Header
DirHeader    ="lib/include";       // C Header für PC/Pi
DirHeaderArd ="lib/libraries";     // Arduino Header
DirDokus     ="l_Dokus";           // Dokumentationen
DirPiBinInc  ="pi/bininc";         // Includes für Pi Steuerungen

// Konfigurationsgruppe für verschiedene Systeme -----
//
ConfigGrp = 10; // Konfigurationsgruppe
PXSubGrp  = 0; // Subgruppe alle Systeme
PCSubGrp  = 1; // Subgruppe für X Programme am PC
PISubGrp  = 2; // Subgruppe für X Programme am Rasperry Pi

// =====
// Einstellungen für für alle Systeme
// VarSetGrpFlt(ConfigGrp,PXSubGrp);
//
// VarSetGrpFlt(0,0);
// =====

// Editor für Console oder X-Terminal -----
// Option
// CEdit    ="nano"; // Editor
// CEditGoto ="+%s"; // Editor: Option Zeilennummer

// =====
// X Einstellungen für PC
VarSetGrpFlt(ConfigGrp,PCSubGrp);

System      ="PC"; // Bezeichner
XTerminal   ="mate-terminal"; // X Terminal
XEdit       ="pluma"; // X Editor
XEditGoto   ="+%s"; // X Viewer: Option Zeilennummer
XOpen       ="xdg-open"; // X Betrachter für verschiedene Programme
XIde        ="geany"; // X IDE

VarSetGrpFlt(0,0); // Ende der Gruppe PC
// =====

// =====
// X Einstellungen für Rasperry Pi
//
VarSetGrpFlt(ConfigGrp,PISubGrp);

System      ="Rasperry PI"; // Bezeichner
XTerminal    ="lxtterminal"; // X Terminal
XEdit        ="leafpad"; // X Viewer für Texte
XEditGoto    ="--jump=%s"; // X Viewer: Option Zeilennummer
XOpen        ="xdg-open"; // X Betrachter
XIde         ="geany"; // X IDE

VarSetGrpFlt(0,0); // Ende der Gruppe Rasperry Pi
// =====

// =====
// Blockvariablen
//
BlockGrp = 20; // Gruppe Blockvariablen
LibSubGrp = 1;
PrgSubGrp = 2;
LnkSubGrp = 3;

// -----
// Projekt c/ Bibliotheken überprüfen
// Zeilenaufbau: "Include | Beschreibung | Installation"
//
VarSetGrpFlt(BlockGrp,LibSubGrp);
Lib[]=
{
    "alsa/asoundlib.h | Lib für ALSA Programme, Raw-Midi | sudo apt-get install libasound2-dev",
    "libpng/png.h | png Bilder lesen und anzeigen",
    "jpeglib.h | jpg Bilder lesen und anzeigen",
    "termios.h | The termios functions describe a general terminal interface",
    "termcap.h | Terminal Infos"
};
VarSetGrpFlt(0,0);
```

```
// -----
// Projekt c/ Optionale Programme überprüfen
// Zeilenaufbau: "Programm | Beschreibung | Installation"
//
VarSetGrpFlt(BlockGrp,PrgSubGrp);
Prog[]=
{
  "geany | C Entwicklungsumgebung | sudo apt-get install geany",
  "shasum | Prüfsummen berechnen für Programm saveit",
  "screen | Multiplexer für ssh Fernbedienungen | sudo apt-get install screen",
  "i2c-tools | Pi, i2c Schnittstelle | sudo apt-get install i2c-tools",
  "timidity | MIDI-to-WAVE converter and player | sudo apt-get install timidity",
  "qsynth | A fluidsynth Qt GUI Interface | sudo apt-get install qsynth",
  "vmpk | Virtual MIDI Piano Keyboard | sudo apt-get install vmpk",
  "musescore3 | MuseScore 2 sheet music editor | sudo apt-get install musescore",
  "leafpad | Texteditor für X | sudo apt-get install leafpad",
  "timeshift | System snapshots | sudo apt-get install timeshift",
  "grub-customizer | X-Menu für grub | sudo apt-get install grub-customizer";
VarSetGrpFlt(0,0);

// -----
// Projekt c/ Doku Links
// Zeilenaufbau: "LinkBezeicher | Link",
//
VarSetGrpFlt(BlockGrp,LnkSubGrp);
Block[]=
{
  "> Projekt /c Homepage, Dokus, Beispiele |https://www.projektc.at/programs/c",
  "> Projekt /c Beispiele |https://www.projektc.at/programs/c_bsp",
  "The GNU C Reference Manual |https://www.gnu.org/software/gnu-c-manual/gnu-c-manual.html#Keywords",
  "C con A bis Z |https://openbook.rheinwerk-verlag.de/c_von_a_bis_z/",
  "Linux-UNIX-Programmierung |http://openbook.rheinwerk-verlag.de/linux_unix_programmierung/",
  "Makefile |https://cs.colby.edu/maxwell/courses/tutorials/maketutor/",
  "ASCII-Tabelle |http://www.c-howto.de/tutorial/anhang/ascii-tabelle/",
  "UTF8-Tabelle |https://www.key-shortcut.com/schriftsysteme/35-symbole/symbole-typographie",
  "UTF8 Definition |https://en.wikipedia.org/wiki/UTF-8",
  "Xterm Steuerzeichen www |https://www.xfree86.org/current/ctlseqs.html",
  "Xterm Mouse Tracking |https://invisible-island.net/xterm/ctlseqs/ctlseqs.html#h2-Mouse-Tracking",
  "Arduino Sprach-Referenz |https://www.arduino.cc/reference/de/";

VarSetGrpFlt(0,0);
// =====
```

Stichwortliste ändern

Befehl: **Einstellungen > Stichwortliste für chelp bearbeiten**

Die Stichwortliste befindet sich im Konfigurationsverzeichnis: **c/bin/chelp/bin/_chelp/chelp.keys**

Syntax ist einfaches C. Die Liste wird beim Start von **chelp** eingelesen.

Auszug aus einer Stichwortliste:

```
// -----
// Stichwortliste für chelp | 2024-09-08
//
// Mit der Stichwortliste können Infos zum Projekt /c einfach
// angezeigt oder Programme gestartet werden.
//
// Alle Pfadangaben der Liste sind relativ zu c/ oder absolut.
// Die Angaben [...] sind optional.
// Kommentare //... am Ende eines Stichworteintrags werden angezeigt!
// Die Items vom Typ h,g und v erlauben Wildcards.
// Z.B {v, "*.pdf", "1"}, // alle PDF-Dateien zur Wahl
//
// Aufbau eines Eintrags der Stichwortliste:
//
// {"Stichwort-Info-Text",      Beschreibung des Eintrags
// {t, "Text"                  }, "Text" anzeigen
// {h, "Datei", ["Suchstring"]}, Header aus Ordner include
// {g, "Pfad",  ["Suchstring"] }, Dateien mit grep suchen
// {v, "Pfad",  ["Nr"]}         }, Viewer aufrufen. Zeilen "Nr"
// {e, "Pfad",  ["Options"]}    }, Execute Pfad mit Options
// {m, "Datei", ["Nr"]}         }, Manpage Datei. Seite Nr
// ...
// };
// ...
// {"Stichwort-Info-Text",      Beschreibung des Eintrags
// { ... },
// ...
// };
// -----

{"PDF-Dokus zu den Programmen",
{v,"1_Dokus/clar_start.pdf","1"}, // Projekt c/ einrichten
{v,"1_Dokus/clar_chelp.pdf","1"}, // Programm chelp
{v,"1_Dokus/clar_projekt.pdf","1"}, // Neues C Programm anlegen
{v,"1_Dokus/clar_objekte1.pdf","1"}, // Objekte 1, Basisobjekte
{v,"1_Dokus/clar_objekte2.pdf","1"}, // Objekte 2, Input/Output
{t,""},
{v,"pi/bin/picamctl/bin/_picamctl/2_picamctl.pdf","1"}, // picamctl Camerasteuerung
{v,"pi/bin/piclock/1_Dokus/PCF8583_Uhr.pdf","1"}, // Doku Echtzeituhr
{t,""},
{v,"pi/bin/gardenctl/bin/_gardenctl/2_gardenctl.pdf","1"}, // gardenctl Gartensteuerung
{v,"pi/bin/gardenctl/bin/_gardenctl/2_gardenctl2.pdf","1"}, // gardenctl Gartensteuerung
{t,""},
{v,"bin/ffmpeg/bin/_ffmpeg/ffmpeg.pdf","1"}, // Videos mit ffmpeg bearbeiten
{v,"bin/kbdctl/bin/_kbdctl/kbdctl_doku.pdf","1"}, // Songverwaltung für Keyboards
{v,"bin/mtrainer/bin/_mtrainer/mtrainer_doku.pdf","1"}, // Musiktrainer, Setup ALSA/MIDI
{v,"bin/midixf/1_Dokus/midi_usb.pdf","1"}, // USB Dokumentation
{v,"1_Dokus/clar_alsa.pdf","1"}, // Dokus Alsa
{t,""},
{v,"1_Dokus/c_linux.pdf","1"}, // Funktionspointer
{v,"1_Dokus/Xterm Control Sequences.pdf","1"}, // Xterm Control Sequences
};
```

```

{"Programme in c/bin/",
 {g,"l_read.*","bin|"}, // Readme's aller Programme anzeigen
 {t,""},
 {t,"....."},
 {t,"> chelp: Hilfe zum Projekt c/"},
 {v,"bin/chelp/bin/_chelp/l_read.me","1"}, // Hilfe anzeigen
 {e,"bin/chelp/bin/Chelp"}, // Programm
 {t,""},
 {t,"> infosys: Linux Systeminfos ausgeben/ändern"},
 {v,"bin/infosys/bin/_infosys/l_read.me","1"}, // Hilfe anzeigen
 {e,"bin/infosys/bin/infosys"}, // Programm
 {t,""},
 {t,"> pshow: Bild/Video Betrachter für Raspberry Pi"},
 {v,"bin/pshow/bin/_pshow/l_read.me","1"}, // Hilfe anzeigen
 {e,"bin/pshow/bin/pshow"}, // Programm
 {t,""},
 {t,"> kbdctl: Songverwaltung für Keyboard PSR-s975"},
 {v,"bin/kbdctl/bin/_kbdctl/l_read.me","1"}, // Hilfe anzeigen
 {e,"bin/kbdctl/bin/kbdctl"}, // Programm
 {t,""},
 {t,"> mtrainer: Musiktrainer | ALSA-Midi Setup"},
 {v,"bin/mtrainer/bin/_mtrainer/l_read.me","1"}, // Hilfe anzeigen
 {e,"bin/mtrainer/bin/mtrainer"}, // Programm
 {t,""},
 {t,"> dbebau: Datenbank für elektronische Bauteile"},
 {v,"bin/dbebau/bin/_dbebau/l_read.me","1"}, // Hilfe anzeigen
 {e,"bin/dbebau/bin/dbebau"}, // Programm
 {t,""},
 {t,"> fbp: Einfacher Picture Viewer für Framebuffer"},
 {v,"bin/fbp/bin/_fbp/l_read.me","1"}, // Hilfe anzeigen
 {e,"bin/fbp/bin/fbp"}, // Programm
 {t,""},
 {t,"> saveit: Ordner und Dateien mit rsync sichern"},
 {v,"bin/saveit/bin/_saveit/l_read.me","1"}, // Hilfe anzeigen
 {e,"bin/saveit/bin/saveit"}, // Programm
 {t,""},
 {t,"> screenstart: Terminalumschalter screen verwalten"},
 {v,"bin/screenstart/bin/_l_read.me","1"}, // Hilfe anzeigen
 {e,"bin/screenstart/bin/screenstart","-h"}, // Programm
 {t,""},
 {t,"> lpctl: CUPS-Druckerwarteschlangen löschen"},
 {v,"bin/lpctl/bin/_lpctl/l_read.me","1"}, // Hilfe anzeigen
 {e,"bin/lpctl/bin/lpctl"}, // Programm
 {t,""},
 {t,"> archivfoto: Fotos/Videos von USB-Medien archivieren"},
 {v,"bin/archivfoto/bin/_archivfoto/l_read.me","1"}, // Hilfe anzeigen
 {e,"bin/archivfoto/bin/archivfoto"}, // Programm
 {t,""},
 {t,"> chksums: Prüfsummen für Archive, Ordner oder Dateien berechnen"},
 {v,"bin/chksums/bin/_chksums/l_read.me","1"}, // Hilfe anzeigen
 {e,"bin/chksums/bin/chksums"}, // Programm
 {t,""},
 {t,"> ffmpeg: Videobearbeitung mit ffmpeg im Terminal"},
 {v,"bin/ffmpeg/bin/_ffmpeg/l_read.me","1"}, // Hilfe anzeigen
 {v,"bin/ffmpeg/bin/_ffmpeg/ffmpeg.pdf","1"}, // Hilfe anzeigen
 {e,"bin/ffmpeg/bin/ffmpeg"}, // Programm
 {t,""},
 {t,"> newprg: Neues c Programmgerüst anlegen"},
 {v,"bin/newprg/bin/_newprg/l_read.me","1"}, // Hilfe anzeigen
};
{"-----"};

{"Alle Help-Dateien l_read.me",
 {t,"Beschreibungen für (H)elp"},
 {v,"l_read.*"}
};

{"-----"};

{"Libraries: Alle Dokus/Header und Testprogramme",
 {h,"*.h","// | Library"}, // Dokus und Header für Libraries
 {g,"*.c","libTest|"}, // Test- und Doku-Programme für Libraries
 {t,""},
 {h,"*.h","// Debugfunktionen"}, // Funktionen zur Fehlersuche
};
{"-----"};

{"Lib iocon.h : Ausgabefunktionen, Farben",
 {h,"iocon.h", " Library"}, // Header
 {h,"iocon.h", " Terminal"}, // Terminalfunktionen
 {h,"iocon.h", " print"}, // Ausgabefunktionen
 {t,""},
 {h,"iocon.h", "#define Farbe*"}, // Farbdefinitionen
 {h,"iocon.h", "Abk|"}, // Farbe Abkürzungen
 {e,"libtest/testlibiocon"}, // IO-Console
};

{"Lib iocon.h : Eingabefunktionen",
 {t,"Tasten"},
 {h,"iocon.h", " Tastenfunktionen"}, // Keyboard
 {h,"iocon.h", "bool peekTaste"}, // stdin hat Tasten bereit
 {h,"iocon.h", "uint16_t chkTaste"}, // stdin nicht blockierend abfragen
 {h,"iocon.h", "uint16_t getTaste"}, // stdin blockierend abfragen
 {h,"iocon.h", "void Weiter"}, // Weiter mit Taste
 {h,"iocon.h", "uint16_t low("}, // Taste nach lowercase
 {t,""},
 {h,"iocon.h", "Befehlsprompt"}, // Prompts
 {h,"iocon.h", "Tastencodes"}, // Codetabelle
 {h,"iocon.h", "#define taste_"}, // Tastecodes
 {t,""},
 {t,"Daten eingeben"},
 {g,"iocon.h", " read"}, // Komfortable read Funktionen
 {t,""},
 {t,"IO-Multiplexing mit select/pselect"},
 {h,"iocon.h", "Select"}, // Select in Keyboardfunktionen
 {e,"libtest/testlibiocon_select"}
};

{"Lib err.h : Err-Objekt",
 {h,"err.h", " Library"}, // Header
 {h,"err.h", "bool Err"}, // Fehler prüfen/anzeigen
 {h,"err.h", "void Err"}, // Fehler merken

```

```

{t,""},
{t,"Verwendung"},
{g,"*.c","Err"}, // Alle Err Aufrufe
{t,""},
{t,"Tests"},
{e,"libtest/testlibutils"} // Hilfsprogramme
};

{"Lib utils.h : Temporäre Strings, Hilfsfunktionen",
{h,"utils.h"," Library"}, // Header
{h,"utils.h","tmpStr"}, // temporäre Strings
{h,"utils.h","newStr"}, // Strings am Heap
{h,"utils.h","StrFnk"}, // Stringfunktionen
{e,"libtest/testlibutils"}, // Testprogramme
{t,""},
{h,"utils.h","char *"}, // Funktionen mit Strings
{t,""},
{h,"utils.h","SysFnk"}, // Sonstige Funktionen
{h,"utils.h","BitFnk"}, // Bitweise vergleichen
{h,"utils.h","SysCall"}, // Systemaufrufe
{e,"libtest/2libdemo"} // Demo
};

{"Lib files.h : Files",
{h,"files.h"," Library"}, // Header
{h,"files.h","scanDir"}, // Fileinfos aus Ordner sammeln
{h,"files.h","scanFiles"}, // Fileinfos zu Verzeichnisbäumen
{g,"files.h","Files"}, // Funktionen für Files
{g,"files.h","writeLog"}, // Logfiles schreiben
{g,"*.h","infoSys"}, // Dateisysteminformationen
};

{"Lib vars.h : Var-Objekt",
{h,"vars.h"," Library"}, // Header
{g,"vars.h"," VarSet"}, // Var-Filter setzen
{g,"vars.h","*VarSet"}, // Var setzen
{g,"vars.h"," VarGet"}, // Var lesen
{g,"vars.h"," VarClr"}, // Var löschen
{g,"vars.h"," Struct-"}, // Struktur Var
{g,"vars.h"," VarPrint"}, // Var anzeigen
{g,"vars.h"," Find "}, // find Vars's
{t,""},
{e,"libtest/testlibvars"}, // Testprogramm
};

{"Lib array.h : Array-Objekt",
{h,"array.h"," Library"}, // Header
{g,"array.h"," Array New"}, // Neues Array
{g,"array.h"," Array Delete"}, // Array freigeben
{g,"array.h"," Array add"}, // Array add Daten
{g,"array.h"," Array move"}, // Daten verschieben
{g,"array.h"," Array find/sort"}, // suchen sortieren
{g,"array.h"," Array print"}, // Daten anzeigen
{g,"array.h"," Array Debug"}, // Debugfunktion
{t,""},
{e,"libtest/testarray"}, // Testprogramm
};

{"Lib script.h : Konfiguration, Scripte",
{h,"script.h"," Library"}, // Header
{g,"script.h"," cfg"}, // Konfiguration lesen/schreiben
{g,"script.h"," scrp"}, // Scriptdateien
{t,""},
{e,"libtest/testruconfig"}, // Testprogramm Konfiguration
{e,"libtest/testscript"}, // Testprogramm Scripte
};

{"Lib token.h : Tokens-Objekt, C-Scripte zerlegen",
{h,"token.h"," Library"}, // Header
{g,"token.h"," StrTok"}, // C-Strings in Token zerlegen
{g,"token.h"," DebTok"}, // Debugfunktionen
{g,"token.h"," SclptBash"}, // Scripte mit Bashbefehlen
{t,""},
{e,"libtest/testscript"}, // Testprogramm
};

{"Lib boxmenu.h : Menus",
{h,"boxmenu.h"," Library"}, // Header
{g,"boxmenu.h"," MenuDef |"}, // Menue Definitionen
{g,"boxmenu.h"," MenuDlg |"}, // Menufunktion
{g,"boxmenu.h"," Debug |"}, // Debugfunktionen
{t,""},
{e,"libtest/testboxmenu"}, // Testprogramm
};

{"Lib boxdirwahl.h: Dateidialog",
{h,"boxdirwahl.h"," Library"}, // Header
{g,"boxdirwahl.h"," Dialog"}, // Dateidialog
{g,"boxdirwahl.h"," Obj |"}, // Basisobjekt
{t,""},
{e,"libtest/testboxlst","-d"}, // Testprogramm
};

{"Lib boxlst.h : Listwahldialog",
{h,"boxlst.h"," Library"}, // Header
{g,"boxlst.h"," Dialog-"}, // Listwahldialog
{g,"boxlst.h"," Obj |"}, // Basisobjekt
{t,""},
{e,"libtest/testboxlst"}, // Testprogramm
};

//{"Lib boxmsg.h : MessageBox",
// {t,"Nicht verwendet!"},
// {t,""},
// {h,"boxmsg.h"," Library"}, // Header
// {g,"boxmsg.h"," Obj |"}, // Basisobjekt
// {t,""},
// {e,"libtest/testboxmsg"}, // Testprogramm
//};

{"Lib datetime.h : Zeitfunktionen",
{h,"datetime.h"," Library"}, // Header
{g,"datetime.h"," *getDate("}, // Date/Time Bash
{g,"datetime.h"," DateTime |"}, // Date/Time

```

```

{g,"datetime.h"," SysDate |"}, // Date/Time System
{t,""},
{e,"libtest/testlibutils"}, // Testprogramm
{t,""},
{t,"Raspberry"},
{g,"pi/bin/clock8583/*.c", "Echtzeituhr"},// Echtzeituhr am Pi
};

{"Lib box.h      : Box Basisobjekt",
 {t,"Basisobjekt für alle Boxdialoge"},
 {t,""},
 {h,"box.h"}, // Header anzeigen
 {e,"libtest/testbox"}, // Testprogramm
};

{"Lib comapi.h   : Serielle Kommunikation PC/Pi und Arduino",
 {h,"comapi.h"," Library"}, // Header
 {h,"comapi.h"}, // Header anzeigen
 {v,"arduino/1_read.me"}, // Arduino 1_read.me
};

{"-----";
{"Lösungen, Demos";
{" Kommandozeilenparameter verwenden",
 {v,"bsp/bsp_15/progopt.c"}, // Beispiel
 {g,"*.c","*ShortOptions"}, // Doku short options
 {g,"bsp/*.c","getopt"} // short und long
};

{" Signal- und Exithandler einrichten",
 {v,"bsp/bsp_04/bsp04.c"}, /
 / Beispiel
 {e,"bsp/bsp04"} // Testprogramm
};

{" IPC: Shared Memory, Demo",
 {e,"bindemo/ipcclient"}, // Demoprogramm
 {t,""},
 {g,"bindemo/*.me","demo|ipc"}, // Datenserver und Client
 {t,""},
 {v,"bindemo/ipc*.c"}
};

{" Demos und Testbeispiele",
 {g,"bindemo/*.me","demo|"},// 1_read.me Demos
 {t,""},
 {e,"bindemo/ipcclient"}, // Demo Shared Memory
 {e,"bindemo/demo00"}, // Demo Anzeige-Boxen
 {e,"bindemo/demo01"}, // Demo CPU Temperatur
 {e,"bindemo/demo02"}, // Demo Libraries
 {t,""},
 {g,"bsp/bsp*.c","bsp"}, // Testprogramme allgemein
 {g,"bsp/*.c","test|"}, // Testprogramme zu Linux
 {g,"bsp/*.c","versuch|"}, // Tests nicht verwendet
};

{"-----";
{"Linux man-pages",
 {m,"string"},// Stringfunktionen
 {m,"strstr"},// Stringvergleich
 {m,"strcmp"},// Stringvergleich
 {m,"scandir", "3"},// Ordner durchsuchen
 {m,"readdir", "3"},// scandir 'struct dirent'
 {m,"fnmatch","3"},// Wildcard match
 {m,"select"},// I/O Multiplexing
 {m,"exit"},// Exitfunktion
 {m,"stdio"},// Standard input/output library functions
 {m,"utf-8"},// ASCII compatible multibyte Unicode encoding
 {m,"terminfo","5"},// Terminalbeschreibung
 {m,"console_codes"},// Linux console escape and control sequences
 {m,"ioctl"},// Control Device
 {m,"ioctl_list"},// list of ioctl calls in Linux/i386 kernel
 {m,"console_ioctl"},// ioctls for console terminal and virtual consoles
 {m,"tty_ioctl"},// ioctls for terminals and serial lines
 {m,"fbset"},// Framebuffer Einstellung
 {m,"setbuf"},// stream buffering operations
};

{"Linux Header",
 {v,"usr/include/stdint.h"}, // ISO C99: Integer types
 {v,"usr/include/dirent.h"}, // Typen für scandir 'struct dirent' Struktur
 {v,"usr/include/linux/fb.h"}, // Framebuffer Strukturen
 {v,"usr/include/linux/videodev2.h"}, // Framebuffer Video Formate
};

{"-----";

{"Raspberry PI: Beispiele und Dokus",
 {g,"pi/*.me","Raspberry|"}, // 1_read.me
 {g,"pi/*.c","Raspberry|"}, // Sourcen
 {e,"bin/infosys/bin/infosys"},// Systeminfos
 {t,""},
 {v,"1_Dokus/clar_raspberry.pdf","1"}, // Beispiele Raspberry
 {v,"1_Dokus/clar_wiring_pi.pdf","1"}, // Doku Wiring Interface
 {t,""},
 {t,"Dokus zum Raspberry"},
 {v,"../pi/1_Dokus/*.pdf"}
};

{"Arduino: Beispiele und Dokus",
 {v,"1_Dokus/clar_arduino.pdf","1"}, // Doku Arduino
 {g,"arduino/*.me","Arduino|"}, // 1_read.me
 {g,"arduino/*.ino","Arduino|"}, // Sourcen
 {t,""},
 {v,"arduino/*.c"},
};

{"-----";
{"Check geany",
 {g,"*.geany","base_path="},// Basisverzeichnisse prüfen
 {g,"*.geany","last_dir="}, // Letzte Ordner
};

// =====...

```

Weitere Hilfsprogramme

Arduino

The open-source Arduino Software (IDE)
www.arduino.cc

Geany

Texteditor mit IDE-Funktionalität. Funktioniert gut über ssh-Verbindungen. Die Konfiguration wird in [clar_projekt.pdf](#) genau beschrieben.
wiki.ubuntuusers.de/Geany/

Screen

Fenstermanager zur Verwendung mit textbasierten Eingabefenstern (Textkonsole).
wiki.ubuntuusers.de/Screen/

GNU General Public License

/*

Copyright 2022-2025 Günther Schardinger <schardinger@projektc.at>

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

Dieses Programm ist Freie Software: Sie können es unter den Bedingungen der GNU General Public License, wie von der Free Software Foundation, Version 3 der Lizenz oder (nach Ihrer Wahl) jeder neueren veröffentlichten Version, weiter verteilen und/oder modifizieren.

Dieses Programm wird in der Hoffnung bereitgestellt, dass es nützlich sein wird, jedoch OHNE JEDE GEWÄHR,; sogar ohne die implizite Gewähr der MARKTFÄHIGKEIT oder EIGNUNG FÜR EINEN BESTIMMTEN ZWECK. Siehe die GNU General Public License für weitere Einzelheiten.

Sie sollten eine Kopie der GNU General Public License zusammen mit diesem Programm erhalten haben. Wenn nicht, siehe <<https://www.gnu.org/licenses/>>.

*/