
Kelvin Serverraum Temperaturüberwachung Hochschule München

Version

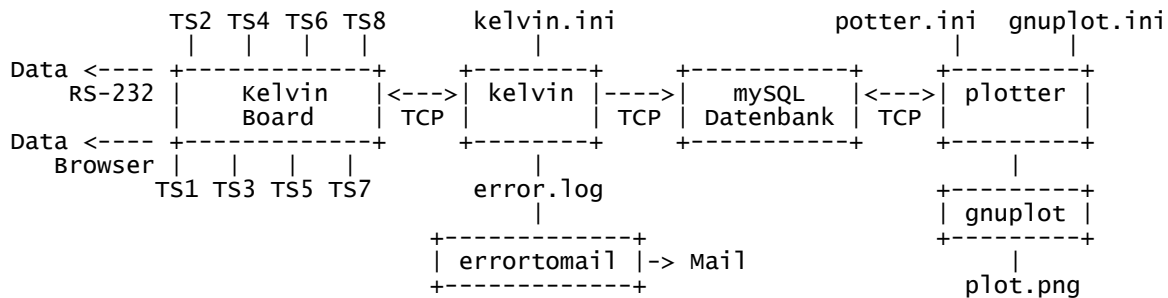
1.4

Autor

Mike Niedermayr

Datum

27.03.2012



Kurzbeschreibung:

Dieses Projekt ermöglicht die Überwachung eines Serverraums mithilfe von bis zu acht Temperatursensoren, deren Daten in eine MySQL-Datenbank abgespeichert werden können. Aus den Messwerten lassen sich übersichtliche Grafiken anfertigen, die den Temperaturverlauf der Sensoren über der Zeit wiedergeben.

Hauptbestandteil ist das "Kelvin Board", an das die acht Temperatursensoren angeschlossen werden können und somit im Serverraum zu installieren ist. Das Board liest die Temperaturen ein und bietet diese über Netzwerk via TCP/IP in Textform an oder als Webpage für die Ansicht in einem Browser.

Das Kommandozeilen basierte Linux-Programm "kelvin" fragt die Daten periodisch beim "Kelvin Board" ab und speichert diese mit Timestamp in eine MySQL-Datenbank.

Um den Temperaturverlauf in eine Grafik zu überführen kann das ebenfalls Kommandozeilen basierte Linux-Programm "plotter" verwendet werden, dass als Plotterengine das Programm "gnuplot" verwendet.

Das Shellscript "errortomail" schickt die Fehlerausgabe von "kelvin" gegebenenfalls an eine E-Mailadresse. Damit können Temperaturüberschreitungen und Fehler rechtzeitig erkannt werden.

Da das "kelvin Board" und die Programme "kelvin", "plotter" über TCP/IP kommunizieren, können diese beliebig über mehrere Rechner verteilt sein. Voraussetzung ist lediglich das auf dem Linux-Rechner auf dem "plotter" betrieben wird, das Programm "gnuplot" installiert ist. Informationen zur Installation finden sich unter www.gnuplot.info.

Um die Daten zu speichern wird die MySQL-Datenbank benötigt, die unter www.mysql.de dokumentiert ist. Die Datenbank kann auch auf einem anderen Rechner laufen als die restlichen Programme, aber es muss ein Login, bestehend aus Nutzernamen und Passwort eingerichtet werden, damit sich "kelvin" und "plotter" mit der Datenbank verbinden können.

Mit entsprechenden Einträgen im "Cronjob" können die Temperaturen periodisch abgefragt werden.

Wichtige Dateien und Verzeichnisstruktur:

Kelvin Board Firmware
Code # Ordner mit dem Sourcecode vom "Kelvin-Board"
Config # Ordner mit Konfigurationsdateien
webpage # Webpage vom "Kelvin-Board"

Plotter and Kelvin
plots # Ordner in den die fertigen Plots gespeichert werden
scr # Ordner mit dem Sourcecode von "kelvin" und "plotter"
temp # Ordner für temporäre Daten

kelvin # Programm zum speichern der Temperaturdaten in MySQL
kelvin.ini # Konfigurationsdatei von "kelvin"
error.log # Fehler-Log-Datei von "kelvin"
errortomail # Shellscript um "error.log" an ein E-Mail zu schicken
plotter # Programm zur Erzeugung von Grafiken
plotter.ini # Konfigurationsdatei von "plotter"
gnuplot.ini # Zusätzliche Einstellungen von "gnuplot"

Kelvin Board:

Funktionsweise:

Die Temperaturwerte der acht Sensoren werden jeweils in kurzen Zeitabständen mit einer Auflösung von 0.1 C eingelesen und können über eine TCP/IP, RS-232 oder einer Webpage abgerufen werden. Während des Betriebs können neue Sensoren angesteckt oder entfernt werden, die selbstständig erkannt werden. Da die Temperatur digital vom Sensor zum Board übertragen wird, verfälscht die Länge des Kabels den Messwert nicht.

Anzeige über die Webpage:

Um die Daten über die Board-interne Webpage abzufragen, kann die IP-Adresse des Boards in die Adresszeile eines Browsers eingegeben werden. Diese Webseite wird direkt vom Board erzeugt und zeigt die Temperaturwerte in Echtzeit an.

Datenabfrage über TCP/IP:

Das Board lauscht auf einem einstellbaren Port auf Verbindungsanfragen über TCP/IP. Geht eine SYN Paket ein, so werden die Temperaturdaten in folgendem Format zurückgeschickt. Danach wird die Verbindung wieder geschlossen.

Beispiel einer Abfrage über TCP:

TS1: 50.2 C\r\nTS2: 83.5 C\r\nTS3: ---.- C\r\nTS4: ---.- C\r\nTS5: 23.0 C\r\nTS6: 10.2 C\r\nTS7: ---.- C\r\nTS8: 123.0 C\r\nnda8236889a61139e3da\r\n'

Die Signatur am Ende der Nachricht ist ein MD5-Hashwert über die gesendeten Daten und ein einstellbares Passwort. Dieser hexadezimale Hashwert ist immer 16 Zeichen lang und erlaubt es die Integrität der Nachricht und die Authentizität des Absenders zu überprüfen.

Um eine Verbindungsanfrage an das Board zu schicken kann der Befehl "telnet <Board IP-Adresse> <Port>" aus der Kommandozeile aus verwendet werden.

Datenabfrage über RS-232:

Wenn über die serielle Schnittstelle RS-232 der Kelvin-Boards ein beliebiges Zeichen empfangen wird, dann werden die Netzwerkadresse, der Listenport, die Subnet Mask und das Gateway, sowie alle Temperaturen der Sensoren zurückgesendet. Die einstellbare Baudrate beträgt standardmäßig 19200; 8 Datenbits, 1 Stopbit und keine Flusssteuerung.
Zur Abfrage eignet sich beispielsweise das Programm "Putty".

Beispiel einer Abfrage über RS-232:

```
'\r\nIP Address: 192.168.1.187 : 666
\r\nSubnetMask: 255.255.255.0
\r\nGateway   : 192.168.1.1
\r\n\r\n
TS1: 24.0 C\r\n
TS2: 21.7 C\r\n
TS3: 23.9 C\r\n
TS4: ---.- C\r\n
TS5: 27.6 C\r\n
TS6: 28.2 C\r\n
TS7: 27.6 C\r\n
TS8: ---.- C\r\n\r\n\r\n'
```

Anbindung an ein Netzwerk:

Die IP Adresse vom Kelvin-Board kann entweder statisch vergeben werden oder es kann ein DHCP Client aktiviert werden, um die Adresse dynamisch zu erhalten. Die Wahl wird in der Datei "Kelvin Board Firmware/Config/ProjectConfig.h" getroffen und muss extra kompiliert und ins Board geflasht werden. Außerdem kann festgelegt werden, ob das Board auf Ping-Anfragen reagiert. Die MAC Adresse kann beliebig eingestellt werden, wobei darauf zu achten ist, dass eine einmalige Adresse verwendet wird.

Hardware des Boards:

Als Hardware des Sensor-Boards kommt das Development-Board "PIC-WEB REVISION-B" der Firma Olimex (www.olimex.com) zum Einsatz, das den Mikrocontroller "PIC18F67J60" von Microchip verwendet. Das Projekt kann auf andere Hardware portiert werden, wenn die entsprechenden Einstellungen in der Datei "Kelvin Board Firmware/Config/HardwareProfile.h" vorgenommen werden.

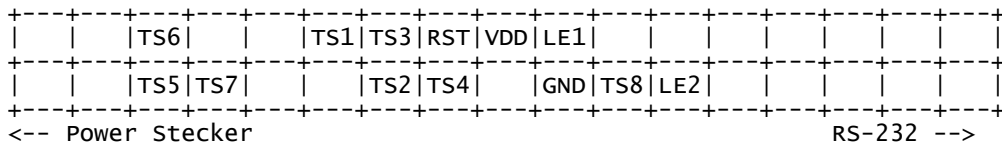
Alle Konfigurationen vom "Kelvin-Board" können in der Datei "kelvin Board Firmware/Config/ProjectConfig.h" getätigt werden.

Um das Projekt zu kompilieren und ins Board flashen zu können, eignet sich die kostenlose Entwicklungsumgebung "MPLAB IDE" von Microchip, die von www.microchip.com heruntergeladen werden kann. Dort findet sich ebenfalls der benötigte Compiler "mplabc18" der in einer Testversion bezogen werden kann, die nach 60 Tagen lediglich die Optimierung des Codes abschaltet. Nachdem das Projekt kompiliert wurde kann es direkt mit einem Programmer ins Board übertragen werden. Dazu kann beispielsweise das "ICD 3" oder "PICKit 3" benutzt werden. Die nötigen Anleitungen sind unter www.microchip.com zu finden.

Hochladen einer eigenen Webpage:

Eine eigene Webpage kann individuell gestaltet werden, und in das EEPROM des Boards geladen werden. Die Templates im HTML Code der Form "~TS(1)~" von 1 bis 8 werden durch die jeweiligen Temperaturwerte ersetzt. Wenn das JavaScript von Microchip verwendet wird, können auch nur bestimmte Teile nachgeladen werden die in einer XML Datei aktualisiert werden. Die fertige Webpage muss anschließend in das MPFS Filesystem der Firma Microchip gebracht werden. Dazu kann das mitgelieferte Tool "kelvin Board Firmware/Tools/MPFS2.exe" verwendet werden. Wählen Sie als Source den Ordner aus, der die von Ihnen gestaltete Webpage und sonstige Daten enthält. Als Output markieren Sie "BIN Image" und geben den Pfad ".../Server Temperature/Code" an. Der Name des Images kann beliebig gewählt werden. Der Upload ins Board erfolgt entweder direkt aus dem Tool oder per http. Geben Sie dazu die IP Adresse des Boards in eine Browser ein, gefolgt von "/httpupload". Sie werden nach einem einstellbaren Nutzernamen und Passwort gefragt und können dann die zuvor erstellte *.bin Datei ins Board laden. Diese können in der Datei "Kelvin Board Firmware/Config/ProjectConfig.h" geändert werden.

Belegung der Pinleiste des PIC-WEB-Boards:

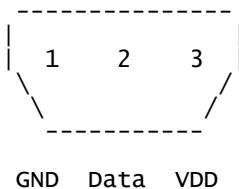


TSx = Datenpins der einzelnen Temperatur Sensoren (1-8).
VDD = Versorgungsspannung 3,3 V der Sensoren.
GND = Masse
RST = Low-aktiver Anschluss für einen Resetknopf.
LE1 = LED1 Pin, der durch blinken den Normalbetrieb anzeigt.
LE2 = LED2 Pin, der aufblinkt, wenn Daten über TCP abgefragt werden.

Hardware und Anschluss der Temperatur-Sensoren:

Es werden bis zu acht digitale Temperatursensoren "DS18S20" der Firma Maxim unterstützt, die über das 1-wire Protokoll mit dem Kelvin Board kommunizieren. Diese werden jeweils über einen eigenen Datenpin angesprochen und benötigen zudem jeweils eine Versorgungsspannung VDD von 3,3 V sowie einen Masseanschluss GND.

Ansicht des Temperatursensors DS18S20 von unten:

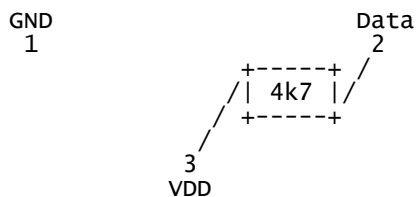


Pinbelegung für XLR Steckverbindung und DS18S20:

PIN 1 = GND
PIN 2 = Data
PIN 3 = VDD

Als Anschluss kommt die dreiadrige XLR-Steckverbindung zum Einsatz, wobei für jeden der acht Sensoren ein eigener XLR-Anschluss verwendet wird. Der weibliche Part des Steckers ist ins Gehäuse vom "Kelvin Board" eingebaut wobei jeweils Pin 1 mit dem GND Anschluss der Pinleiste des Boards verbunden wird, sowie VDD mit Pin 3. Der individuelle Datenpin jedes Sensors wird an den Pin 2 angeschlossen. Zwischen dem Datenpin und der Versorgungsspannung VDD muss zudem jeweils ein 4,7k Ohm Widerstand vorhanden sein, der direkt auf die Rückseite des Steckers gelötet werden kann.

Ansicht der Rückseite XLR-Stecker weiblich:



Der männlich Part des XLR-Steckers ist direkt mit den Temperatursensoren "DS18S20" verbunden, wobei die Pinbezeichnungen übereinstimmen.

```
*****
Programm "kelvin":
*****
```

Mit dem kommandozeilen basierte Linux-Programm "kelvin" können die Temperaturdaten vom Kelvin-Board abgefragt werden, die zusammen mit einem Timestamp in eine MySQL-Datenbank gespeichert werden.

Um eine periodische Abfrage der Daten zu erreichen, kann man entweder einen Cronjob einrichten, der das Programm jeweils einmal aufruft oder man stellt ein Update-Intervall in Sekunden ein. Dann beendet sich das Programm nicht nach der Ausführung sondern wartet die Intervall-Zeit ab und führt danach das Programm in einer Endlosschleife aus. Zum Beenden dient die Tastenkombination [Strg] + [C].

Bei der Abfrage der Daten wird auch der vom Kelvin-Board mitgesendete Hashwert der Daten überprüft, um die Integrität der Daten und die Authentizität des Absenders festzustellen. Das Hash-Passwort muss mit dem Hash-Passwort im Kelvin-Board übereinstimmen. Falls das Programm einen anderen abweichenden Hashwert errechnet, dann wird eine Fehlermeldung ausgegeben und die nicht vertrauenswürdigen Daten werden nicht in die MySQL-Datenbank aufgenommen.

Immer wenn ein Fehler auftritt, wird dieser mit einer Zeitangabe auf der Standard-Fehlerausgabe ausgegeben. Es ist zu empfehlen, die Fehlerausgabe mit `"/kelvin 2>>/home/kelvin/error.log"` auf eine Log-Datei umzulenken, um somit alle Fehler zu protokollieren. Die Datei "error.log" wird im Shellscript "errortomail" verwendet, um den Administrator über aufgetretene Fehler zu informieren.

Alle Einstellungen, wie zum Beispiel das Passwort für MySQL, können in der Konfigurationsdatei "kelvin.ini" vorgenommen werden, und/oder als Kommandozeilenparameter übergeben werden.

Beim Programmstart wird im aktuellen Verzeichnis nach der Datei "kelvin.ini" gesucht und gegebenenfalls als Grundeinstellungen geladen. Als erster Kommandozeilenparameter kann mit `"-load Dateiname"` auch eine alternative Konfigurationsdatei mit absolutem Pfad geladen werden. Danach werden die Kommandozeilenparameter ausgewertet, die die in der Konfigurationsdatei getroffene Einstellungen gegebenenfalls überschreiben.

Die Syntax bestehen immer aus zwei Teilen: Dem Befehl, eingeleitet mit einem "-" und dem Übergabewert der vom Befehl durch mindestens ein Leerzeichen getrennt sein muss.

In der Konfigurationsdatei werden leere Zeilen und Zeilen die mit einem "#" -Zeichen anfangen übersprungen. In jeder Zeile darf nur ein Befehl stehen und muss vom Übergabewert durch mindestens ein Leer- oder Tabulatorzeichen abgetrennt sein.

Inhalt der "kelvin.ini" mit Standardwerten:

```
# [-l], [--load]           Lade Config Datei
# [-i], [--interval]      Update Intervall in Sekunden
# [-mu], [--user]         MySQL User
# [-p], [--password]      MySQL Passwort
# [-d], [--database]      MySQL Database
# [-tb], [--table]        MySQL Table
# [-mh], [--sqlhost]      MySQL Host
# [-mp], [--sqlport]      MySQL Port
# [-kh], [--boardhost]    Kelvin Board Hostname
# [-kp], [--boardport]    Kelvin Board Port
# [-hp], [--hashpassword] MD5-Hash Passwort
# [-at], [--alarm]        Alarm Temperatur in C

--interval      5
--user          root
--password      1234
--database      kelvin
--table         Temperatur
--sqlhost       localhost
--sqlport       3306
--boardhost     192.168.1.187
--boardport     666
--hashpassword  1234567
--alarm         50.0
```

Befehlserläuterung für "kelvin":

"-l Dateiname" oder "--load Dateiname"

Die angegebene Datei z.B. "kelvin.ini" wird geladen und die darin befindlichen Einstellungen werden gelesen. Dieser Befehl kann nur als erster Kommandozeilenparameter aufgerufen werden.

"-i Zahl" oder "--interval Zahl"

Wenn eine Zahl größer Null angegeben wird, so wird diese als Update-Intervall in Sekunden interpretiert. Dann wird das Programm nach einem Durchlauf nicht beendet, sondern suspendiert bis die gewählte Wartezeit abgelaufen ist. Wird der Wert "0" übergeben, läuft das Programm genau einmal durch und beendet sich dann selber.

"-mu Name" oder "--user Name"

Um sich mit einer MySQL-Datenbank verbinden zu können muss man sich als User mit Passwort einloggen. Der übergebene Name ist der Username für MySQL.

"-p Zeichenkette" oder "--password Zeichenkette"

Die Zeichenkette ist das Passwort für die MySQL-Datenbank.

"-d Name" oder "--database Name"

Der Name der Datenbank innerhalb von MySQL. Wenn diese Datenbank nicht vorhanden ist, so wird sie neu angelegt.

"-tb Name" oder "--table Name"

Der Name der Tabelle innerhalb von MySQL unter der die Daten gespeichert werden sollen.

"-mh Hostname" oder "--sqlhost Hostname"

Die IP-Adresse unter der die MySQL-Datenbank zu erreichen ist. "localhost" adressiert den lokalen Rechner.

"-mp Nummer" oder "--sqlport Nummer"

Die Portnummer unter der die MySQL-Datenbank zu erreichen ist. 3306 ist der Standardport für MySQL.

"-kh Adresse" oder "--boardhost Adresse"

Die IP-Adresse vom Kelvin-Board. Wenn diese per DHCP zugeteilt wurde, kann die Adresse über RS-232 eingesehen werden.

"-kp Nummer" oder "--boardport Nummer"

Die Portnummer unter der das Kelvin-Board zu erreichen ist.

"-hp Passwort" oder "--hashpassword Passwort"

Das Passwort, dass in den MD5 Hash der Daten bei der Übertragung eingerechnet wird. Dieses muss gleich dem Passwort vom Kelvin-Board sein, dass in der Datei "kelvin Board Firmware/Config/ProjectConfig.h" geändert werden kann.

"-at Zahl" oder "--alarm Zahl"

Die Zahl wird als Temperaturwert interpretiert. Überschreitet ein Sensor diese Temperatur, so wird eine entsprechende Warnung mit Zeitangabe auf der Standard-Fehlerausgabe ausgegeben.

Shellscript "errortomail":

Das Skript "errortomail" prüft, ob die Datei "Error.log" existiert und Inhalt hat. Gegebenenfalls wird dann der Inhalt dieser Datei an eine einstellbare E-Mail Adresse gesendet und danach wird die Log-Datei gelöscht. Sinnvollerweise muss das Skript in regelmäßigen Abständen per "Cronjob" aufgerufen werden, um einen Administrator rechtzeitig über einen Temperaturanstieg oder den Ausfall des Kelvin-Boards informieren zu können. Im Skript selber müssen die folgenden beiden Variablen entsprechend angepasst werden.

```
ERRORLOG="/home/kelvin/error.log"  # Die zu überwachende Log-Datei
MAILADDRESS="kelvin@test.de"       # Die Ziel-Mailadresse
```

```

*****
Programm "plotter":
*****

```

Um den Temperaturverlauf in eine Grafik zu überführen kann das ebenfalls kommandozeilen basierte Linux-Programm "plotter" verwendet werden, dass seine Einstellungen aus der Datei "plotter.ini" oder aus den Kommandozeilenparameter analog zu "kelvin" ausliest. Dabei lassen sich gezielt Zeitbereiche auswählen, die der zu erstellenden Grafik zugrunde liegen. Als Plotterengine wird "gnuplot" verwendet das über die Datei "gnuplot.ini" konfiguriert werden kann. Deshalb muss "gnuplot" auf dem System installiert sein, auf dem das Programm "plotter" ausgeführt werden soll.

Die Temperaturen aus der mysql-Datenbank werden über der Zeitachse für die ausgewählten Sensoren angezeigt.
 Um den Zeitbereich einzuschränken können ein Start- und ein Endzeitpunkt festgelegt werden. Dies geschieht zum Beispiel durch
 "-start '2012-12-24 18:00:00'" und "-end '2013-01-01 00:00:00'". Zusätzlich kann auch ein Zeitbereich mit Range festgelegt werden. Dazu wird eine Zahl und eine Einheit verwendet, die zusammen Range ergeben wie zum Beispiel
 "-range 2 -unit day". Es sind fünf verschiedene Kombinationen zulässig, um den Zeitbereich anzugeben.
 Start, End und Range können nicht alle gleichzeitig definiert werden.

Bedeutung der Kombinationen von Start, End und Range:

1. Start, End		2. Start, Range		3. End, Range		4. Range		5.	
Start	End	Start		End		Last	First	Last	
-----		-----		-----		-----	-----	-----	
		<-Range->		<-Range->		<-Range->			

1. wenn nur Start und End gesetzt sind,
dann wird der Zeitraum dazwischen ausgewählt.
2. wenn nur Start und Range gesetzt sind,
dann wird der Zeitraum ab Start bis Start plus Range ausgewählt.
3. wenn nur End und Range gesetzt sind,
dann wird der Zeitraum von End minus Range bis Range ausgewählt.
4. wenn nur Range gesetzt ist,
dann wird als End Zeitpunkt der letzte bzw aktuellste Datensatz gewählt.
5. wenn keine Angabe gemacht wird,
dann wird der gesamte Datensatz zugrunde gelegt.

Inhalt der "kelvin.ini" mit Standardwerten:

```
# [-l], [--load]      Lade Config Datei
# [-mu], [--user]    MySQL User
# [-p], [--password] MySQL Passwort
# [-d], [--database] MySQL Database
# [-tb], [--table]   MySQL Table
# [-h], [--host]     MySQL Host
# [-mp], [--port]    MySQL Port
# [-mo], [--sqlout]  Zielort der SQL Output Datei
# [-go], [--gnuout]  Zielort der SQL Gnuplot Output Datei
# [-i], [--gnuini]   Ort der Gnuplot.ini Datei, die mit geladen wird
# [-gp], [--gnupic]  Zielort der zu erzeugenden Grafik Datei
# [-s], [--start]    Start Zeitpunkt im SQL Datumsformat
# [-e], [--end]      End Zeitpunkt im SQL Datumsformat
# [-r], [--range]    Anzahl des Umfangs
# [-u], [--unit]     Einheit von Range
# [-x], [--sensors]  Auswahl der Sensoren [all, ts1,ts2,ts3...]
# [-o], [--smooth]   Interpolation "smooth csplines", "smooth bezier",
#                   "smooth acsplines"
# [-t1], [--nameTS1] Name in Grafik fuer TS1 bis TS8
# [-c1], [--colourTS1] Farbe der Linie fuer TS1 bis TS8
# [-w1], [--styleTS1] Darstellung der fuer TS1 bis TS8 z.B "lines",
#                   "points", "linespoints"
```

```
--user      root
--password   1234
--database   kelvin
--table      Temperatur
--host       localhost
--port       3306
--sqlout     /home/kelvin/temp/sqloutput
--gnuout     /home/kelvin/temp/gnuoutput
--gnuini     /home/kelvin/gnuplot.ini
--gnupic     /home/kelvin/plots/plot.png
--sensors    all
--nameTS1    Server 1
--colourTS1  black
--styleTS1   lines
--nameTS2    Server 2
--colourTS2  green
--styleTS2   lines
--nameTS3    Server 3
--colourTS3  blue
--styleTS3   lines
--nameTS4    Server 4
--colourTS4  magenta
--styleTS4   lines
--nameTS5    Server 5
--colourTS5  light-blue
--styleTS5   lines
--nameTS6    Server 6
--colourTS6  red
--styleTS6   lines
--nameTS7    Server 7
--colourTS7  grey
--styleTS7   lines
--nameTS8    Server 8
--colourTS8  orange
--styleTS8   lines
```


Befehlserläuterung für "plotter":

"-l Dateiname" oder "--load Dateiname"

Die angegebene Datei z.B. "plotter.ini" wird geladen und die darin befindlichen Einstellungen werden gelesen. Dieser Befehl kann nur als erster Kommandozeilenparameter aufgerufen werden.

"-mu Name" oder "--user Name"

Um sich mit einer MySQL-Datenbank verbinden zu können muss man sich als User mit Passwort einloggen. Der übergebene Name ist der Username für MySQL.

"-p Zeichenkette" oder "--password Zeichenkette"

Die Zeichenkette ist das Passwort für die MySQL-Datenbank.

"-d Name" oder "--database Name"

Der Name der Datenbank innerhalb von MySQL. Wenn diese Datenbank nicht vorhanden ist, so wird sie neu angelegt.

"-tb Name" oder "--table Name"

Der Name der Tabelle innerhalb von MySQL unter der die Daten gespeichert werden sollen.

"-h Hostname" oder "--host Hostname"

Die IP-Adresse unter der die MySQL-Datenbank zu erreichen ist. "localhost" adressiert den lokalen Rechner.

"-mp Nummer" oder "--port Nummer"

Die Portnummer unter der die MySQL-Datenbank zu erreichen ist. 3306 ist der Standardport für MySQL.

"-mo Dateiname" oder "--sqlout Dateiname"

In diese Datei wird die Ausgabe der MySQL-Datenbank nach der individuellen Abfrage geschrieben. Sie enthält die Daten, aus denen die Grafik erzeugt wird, danach kann sie gelöscht werden.

"-go Dateiname" oder "--gnuout Dateiname"

In diese Datei werden die Anweisungen geschrieben, die dann von gnuplot ausgeführt werden. Innerhalb dieser Anweisungen wird die "gnuplot.ini" geladen.

"-i Dateiname" oder "--gnuini Dateiname"

In dieser Datei können Einstellungen für gnuplot vorgenommen werden, die bei der Erstellung der Grafik mit berücksichtigt werden.

"-gp Dateiname" oder "--gnupic Dateiname"

Der Zielort und der Name der zu erzeugenden Gnuplot-Grafik

"-s Datum/Uhrzeit" oder "--start Datum/Uhrzeit"

Der Startzeitpunkt beispielsweise im Format "YYYY-MM-DD hh:mm:ss" ab dem die Daten für die Erzeugung der Grafik verwendet werden. Jedes MySQL Datumsformat ist gültig.

"-e Datum/Uhrzeit" oder "--end Datum/Uhrzeit"

Der Endzeitpunkt beispielsweise im Format "YYYY-MM-DD hh:mm:ss" bis zu dem die Daten für die Erzeugung der Grafik verwendet werden. Jedes MySQL Datumsformat ist gültig.

"-r Zahl" oder "--range Zahl"

Die Zahl ergibt zusammen mit der Einheit einen Zeitraum wie zum Beispiel "-r 1 -u day"

"-u Einheit" oder "--unit Einheit"

Die Einheit in der der Zeitraum definiert werden soll. Gültig sind alle MySQL-Einheiten wie zum Beispiel: "MINUTE", "HOUR", "DAY", "WEEK", "MONTH" oder "YEAR"

"-x Auswahl" oder "--sensors"

Die Auswahl der Sensoren, die angezeigt werden sollen. Es ist eine beliebige Kombination aus den Sensoren "ts1" bis "ts8" möglich, die durch ein Komma aber ohne Leerzeichen getrennt werden. Bsp: "ts1,ts4,ts3" Mit "all" werden alle Sensoren ausgewählt.

"-o Text" oder "--smooth Text"

Mit Text kann eine Interpolations- oder Approximationsfunktion ausgewählt werden, die für alle Sensoren gilt. Dadurch können sehr zackige Kurve geglättet dargestellt werden. Es sind alle gnuplot Funktionen gültig wie zum Beispiel: "smooth csplines", "smooth bezier" oder "smooth acsplines".

"-t1 Name" oder "--nameTS1 Name"

Der Name des jeweiligen Sensors, wie er in der Legende der Grafik angezeigt werden soll. Gültig analog für alle Sensoren TS1 bis TS8.

"-c1 Farbe" oder "--colourTS1 Farbe"

Die Farbe des jeweiligen Graphen. Gültig sind alle Farbbezeichnungen von gnuplot als Text oder als RGB Farbcode.

Bsp: "#CCCCC". Analog für alle Sensoren TS1 bis TS8.

"-w1 Text" oder "--styleTS1 Text"

Mit Text kann für jeden Sensor eine individuelle Darstellung gewählt werden.

Gültig sind alle Bezeichnungen von gnuplot wie "lines", "points"

oder "linespoints". Analog für alle Sensoren TS1 bis TS8.

Cronjob Einträge:

Vorausgesetzt der Inhalt des Ordners "Plotter and kelvin" befindet sich unter "/home/kelvin" kann ein Cronjob folgendermaßen aussehen und mit "crontab -e" hinzugefügt werden. Wenn für "kelvin.ini" und "plotter.ini" nur der Besitzer Leserechte besitzt, werden die Passwörter für andere Nutzer verborgen.

Beispiel für einen Cronjob Eintrag:

```
# Minute Stunde Tag Monat Wochentag
# 0-59 0-23 1-31 1-12 0-6 (0=So)
```

```
# Alle 10 Minuten werden die Temperaturwerte der Server im Serverraum abgerufen
# und in die mySQL Datenbank gespeichert.
```

```
0,10,20,30,40,50 * * * * /home/kelvin/kelvin -l /home/kelvin/kelvin.ini
>/dev/null 2>>/home/kelvin/error.log
```

```
# Immer um ca Mitternacht wird ein Plot über den letzten Tag erstellt
```

```
55 23 * * * /home/kelvin/plotter -l /home/kelvin/plotter.ini
-r 1 -u day -gp /home/kelvin/plots/lastday.png >/dev/null 2>&1
```

```
# Immer Sonntags um ca Mitternacht wird ein Plot über die letzte Woche erstellt
```

```
56 23 * * 0 /home/kelvin/plotter -l /home/kelvin/plotter.ini
-r 1 -u week -gp /home/kelvin/plots/lastweek.png >/dev/null 2>&1
```

```
# Immer am ersten des Monats um ca Mitternacht wird ein Plot über
# den letzten Monat erstellt
```

```
57 23 1 * * /home/kelvin/plotter -l /home/kelvin/plotter.ini
-r 1 -u month -gp /home/kelvin/plots/lastmonth.png -o "smooth bezier"
>/dev/null 2>&1
```

```
# Jede Stunde wird ein Skript gestartet, dass ueberprueft ob Daten in der
# Error.log vorhanden sind und diese dann an eine Mailadresse schickt.
```

```
5 * * * * /home/kelvin/errortomail
```