

Delta-Tools Sammlung

Version 1.2

19.04.2011

Bayreuther Zentrum für Ökologie und Umweltforschung

Abteilung EDV und Datenbanken

oliver.archner@uni-bayreuth.de

Über diese Software-Sammlung

[Delta-T](#) Logger haben sich seit etlichen Jahren in verschiedenen Forschungseinsätzen zur Aufzeichnung von Sensordaten bewährt. Obwohl mittlerweile weitaus modernere System existieren, sind gerade die [DL2](#) Serien wegen ihrer Robustheit immer noch weit verbreitet. Die vom Hersteller mit den Geräten ausgelieferte Windows [LS2Win](#) Software bietet eine komfortable Möglichkeit zur Konfiguration und dem Betrieb der Hardware, lässt allerdings einige Funktionen wie z. B. Statusüberwachung oder Interpolationsmöglichkeiten und Plattformunabhängigkeit vermissen. Diese Lücke versucht die Software Sammlung Delta-Tools zu schließen.

Inhalt

Über diese Software-Sammlung.....	1
Installation.....	2
DLStatusCheck.....	3
DLDump.....	4
Bin2Csv.....	5
CsvInterpolator.....	6
Lizenz.....	7

Installation

1. Entpacken Sie die ZIP Datei in einen lokalen Ordner z. B. unter C:\Programme\DeltaT-Tools
2. Ergänzen Sie die PATH Variable in ihrem System um das Installationsverzeichnis.
3. Installieren Sie eine aktuelle Version der [JRE](#) (> 1.6)
4. Konfigurieren sie die Java Ergänzung zur Kommunikation mit der seriellen Schnittstelle.
Kopieren Sie dazu die beiden Bibliotheken aus dem RXTX Verzeichnis in die Zielverzeichnisse:

Windows

RXTXcomm.jar -> [JAVA_HOME]\lib\ext (under java)
rxtxSerial.dll -> [JAVA_HOME]\bin

Linux (x86, x86_64, ia64)

RXTXcomm.jar -> [JAVA_HOME]/lib/ext (under java)
librxtxSerial.so -> [JAVA_HOME]/lib/[machine type] (i386 for instance)

5. Schließen Sie einen Logger über die serielle Schnittstelle an ihren PC an und kontrollieren Sie ob Sie den Status des Loggers abfragen können:
 - a. Öffnen Sie eine Windows Konsole (Start/Ausführen/CMD)
 - b. Rufen Sie das Tool DLStatusCheck auf: DLStatusCheck.cmd <device> <baudrate>
z. B. DLStatusCheck.cmd COM1 9600
 - c. In der Konsole müssen jetzt Informationen zum aktuellen Status des Loggers angezeigt werden.

DLStatusCheck

Das Tool stellt über die serielle Schnittstelle eine Verbindung mit dem Logger her und gibt die Informationen über die Console aus.

Aufruf:

```
DLStatusCheck.cmd [device] [baudrate]
```

Argumente:

[device] Serieller Port z. B. COM1 oder /dev/ttyS0

[baudrate] Baudrate zur Kommunikation mit dem Logger

Beispiel Ausgabe:

```
PROM version: 5
PROM revision: 3
battery voltage: 4.9E-324V
logging status: logging
minimum sampling interval: 1h
battery failed flag: ok
memory full flag: ok
experiment name: e4_oebg
password:
started logging: 14.03 11:04:49
stopped logging: 25.02 18:00:07
first stored timed data: 14.03 11:04:52
next timed data to be output: 19.04 09:04:52
date-time format: European
overwrite mode: disabled
next timed data: 09:54:52
current time: 19.04 09:45:17
Number of stored records: 53506
Date of 1st-stored record: 14.03 11:04:52
Number of records since last retrieved: 62
Date of next record to-be-output: 19.04 09:04:52
```

DLDump

Tool um die Messdaten des Loggers in eine BIN Datei abzuspeichern. Die Kommunikation erfolgt über die serielle Schnittstelle. BIN Dateien können mit dem Delta-T Programm DTViewer betrachtet werden. Zur Weiterverarbeitung in EXCEL kann die BIN Datei mit dem Tool Bin2Csv in eine CSV Datei konvertiert werden.

Bitte beachten Sie, dass in der BIN Datei keine Jahresangaben enthalten sind. Eine mögliche Konvention ist es daher die Jahreszahl des ersten Datensatzes im Dateinamen mitzuführen.

Aufruf:

```
DLDump.cmd [device] [baudrate] [file] [new|full]
```

Argumente:

[device] Serieller Port

[baudrate] Baudrate zur Kommunikation mit dem Logger

[file] Ausgabe Pfad des BIN Files

[new|full] Flag zur Auswahl

new: alle Messungen seit dem letzten Auslesevorgang

full: alle im Logger gespeicherten Messungen

Bin2Csv

Konvertiert eine BIN Datei in eine CSV Datei im folgenden Format:

Channel	1	2	3	4
Label	temp1	temp2	temp3	temp4
Sensor Type	RR3	RR3	RR3	RR3
Units	KOhms	KOhms	KOhms	KOhms
23.12.2011 06:00	9.408	9.1936	8.4384	9.9904
23.12.2011 07:00	9.408	9.1936	8.4384	9.9904
23.12.2011 08:00	9.408	9.1936	8.4384	9.9904
23.12.2011 09:00	9.408	9.1936	8.4384	9.9904
23.12.2011 10:00	9.408	9.1936	8.4416	9.9904

Als Spaltentrennzeichen wird das Semikolon verwendet, als Dezimaltrennzeichen kommt ein Punkt zum Einsatz. Die Jahresangabe in der Datumsspalte wird aus dem Übergabeargument ermittelt.

Aufruf:

```
Bin2Csv.cmd [BIN File] [CSV File] [YEAR]
```

Argumente:

[BIN File] Input Binär Datei

[CSV File] Ausgabe CSV Datei

[YEAR] Jahr des ersten Messwerts formatiert als 'YYYY'

CsvInterpolator

Werden Messwerte im Logger in sog. Base Units wie z. B. kohm abgespeichert und soll in Excel mit Engineering Units wie z. B. Grad Celsius weiter gerechnet werden, kann mit Hilfe dieses Tools eine Umrechnung der Messergebnisse in der CSV-Datei erfolgen. Die Umrechnung und die verwendeten Sensor Channels sind frei konfigurierbar. Als Umrechnungsfunktion kann derzeit nur einen Spline Funktion¹ eingesetzt werden.

Aufruf:

```
CsvInterpolator.cmd [Input File] [Output File]
```

Argumente:

[Input File] Input CSV Datei im Format wie es das Tool Bin2Csv erzeugt

[CSV File] Ausgabe CSV Datei mit interpolierten Messwerten im Format entsprechend des Input Files

Konfigurationsdateien

Verzeichnis Functions

Das Verzeichnis dient als Ablageort für Funktionen. Alle Dateien mit der Endung xml werden als Funktion eingelesen und stellen die Interpolationsfunktion über die ID zur Verfügung. Derzeit können nur Spline Funktionen im folgenden XML Format definiert werden:

```
<?xml version="1.0" encoding="UTF-8"?>
<spline name="Thermistor Siemens S863" id="S863">
<data x="0.204" y="100"/>
<data x="0.2164" y="98"/>
<data x="0.2296" y="96"/>
</spline>
```

Datei interpolations.xml

In dieser Datei wird gespeichert für welche Sensorcode (z. B. RR3, TM1) und für welche Einheit eine Umrechnung mit einer Interpolationsfunktion durchgeführt wird. Die Datei hat den folgenden Aufbau:

```
<?xml version="1.0" encoding="UTF-8"?>
<interpolations>
<interpolation code="RR3" unitIn="kohm" unitOut="deg C" function="S863"/>
<interpolation code="RR3" unitIn="KOhms" unitOut="deg C" function="S863"/>
</interpolations>
```

¹ The cubic spline interpolation algorithm implemented is as described in R.L. Burden, J.D. Faires, Numerical Analysis, 4th Ed., 1989, PWS-Kent, ISBN 0-53491-585-X , pp 126-131.

Lizenz

Copyright 2011 Bayreuth Center of Ecology and Environmental Research

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Die Software verwendet folgende Komponenten:

[RXTX](#)

[Apache Commons Math](#)

[OpenCSV](#)

[Log4j](#)