

Bike Owl



Oder: Luxus Lowcost Lampensteuerung – ganz einfach selbstgebaut

Teil2

Anleitung Version 1.1/14.10.2011

Zu Software Version 1.1

Siehe auch: <http://www.mtb-news.de/forum/showthread.php?t=378434>

1. Überblick und Einführung

Ich möchte hier mein zweites Lampensteuerungsprojekt zum Nachbau vorstellen, wieder einmal eine Lampensteuerung der besonderen Art.

Die alte Basishardware war ja leider kurz nach der Fertigstellung der SW nicht mehr verfügbar:o(

Die neue LLL sieht schicker aus, ist dabei aber immer noch relativ einfach und für (fast) jedermann nachzubauen, günstig und mit einigen Features, die man sonst nicht so findet!

1.1 Die Grundschaltung

Basis ist der CellLog in seiner einfachsten Ausführung (**Cell-Log 8M**), z.B. vom HobbyKing.com:

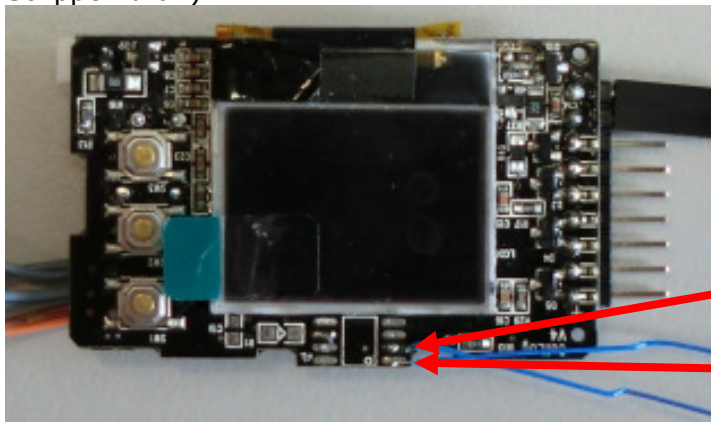
Bei dieser Seite einfach die Seite ein paar Minuten offen lassen, dann gibt es sogar noch Rabatt!

Der Preis liegt dort bei ca. 12-14€, je nach Dollarkurs;o) Lieferung, wenn verfügbar, ca. 2 Wochen. Schneller (und teurer) geht's natürlich bei einem Händler der Wahl aus Deutschland.

Ein kurzer Blick in das Gehäuse offenbart:

Das Gehäuse ist zusammengeschnappt, am Besten zu öffnen, wenn man am Steckeranschluß das Gehäuse einfach auseinanderdrückt (Handkraft sollte reichen). Vorsicht, daß das Display nicht beschädigt wird!

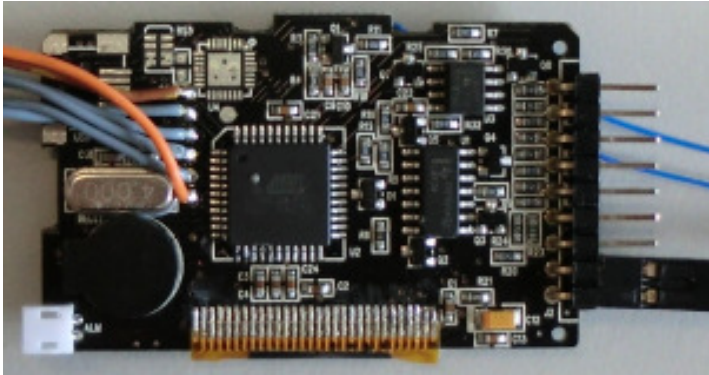
So schaut's von vorne aus (Displayschutzfolie ist noch drauf und schon ein paar Strippen dran):



Lampe 1 PWM 3Vss

Lampe 1 PWM 3Vss

Und so von hinten:



Im Cellog steckt ein ATmega32 drin mit 4MHz Quarz und einem transflectiven Graphikdisplay mit 128*64 Pixeln (auch bei Sonnenschein lesbar->gut für Siam;o), vertikal aufgeteilt in 16+48 Pixel. Außerdem ein Lautsprecher und ein wenig Analoggeräffel zur Spannungsmessung. Die Programmieranschlüsse sind auf Lötunkte herausgeführt!

Im Display steckt nach eingehender Analyse ein SED1565 Derivat als COG mit reduziertem Pinout (daher nur 30(-2) Anschlüsse).

Eine schöne Basis für alles mögliche, wenngleich das Display so seine Tücken hat, die fallen hier aber nicht ins Gewicht.

Leider ist der 3V Spannungsregler als Shuntregler ausgeführt, d.h. im Controller Standbymode ist die Stromaufnahme relativ „hoch“ (noch nicht ausgetestet, aber abzusehen).

Aber angesichts der verwendeten Akkukapazitäten ist das wohl noch vertretbar, wenn die Einheit ein paar Tage dran bleibt. Für Low Power muß man da aber nachhelfen.

1.2. Funktionalität der Lampensteuerung

Die Software dazu ist immer noch im entstehen (wann ist Software je fertig?!?). Eine erste Version 1.1 ist nun fertig.

Die aktuellen Features der sind:

- fast komplett frei konfigurierbare Funktionen
- selbstständige Zellenerkennung (LiIon/LiPo 1-8Zellen), bislang nur Summenmessung, Einzelmessung in Vorbereitung.
- Spannungs-/Kapazitätsanzeige
- 1-2 Lampenausgänge
- 1-4 frei programmierbare Dimmstufen je Lampe
- programmierbares Startverhalten (hart, soft, Xenon-Effekt)
- wählbare PWM Frequenzen (falls es z.B. Probleme mit dem Tacho gibt)
- 1+2 Unterspannungsschwellen (1*Vorwarnung, 2*Leistungsbegrenzung)
- Batterieschutz (Lampe aus bei 3V/Zelle) mit Vorwarnung bei 2% Kapazität
- Verschiedene Anzeigeoptionen
- Laufzeitmessung aktuell, Lampenbrenndauer und Summe
- akustisches Feedback
- ...

2. Stückliste

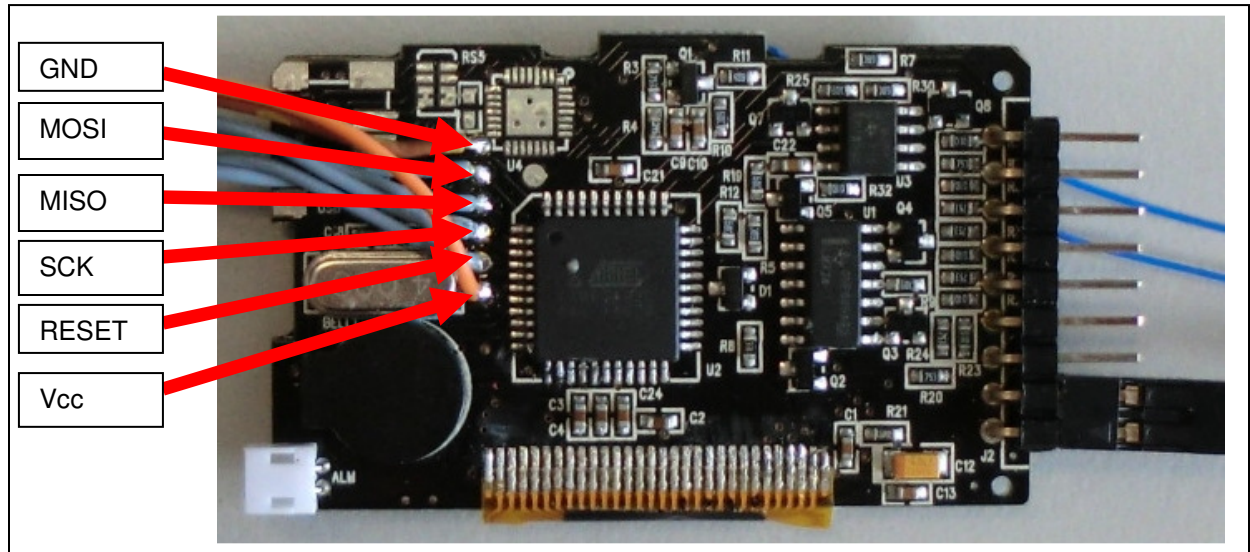
Was braucht man nun genau?

- den CellLog natürlich
- Schrumpfschlauch oder eine andere Idee, um den Cellog wasserdicht zu machen
- Kabel nach Bedarf
- Endstufe für die Lampe (z.B. IRF 3708 bei Reichelt, Artikel-Nr.: IRF 3708, 1,10€, o.ä. mit $V_{th} \leq 2V$) oder einen direkten Dimmeingang an der Stromquelle

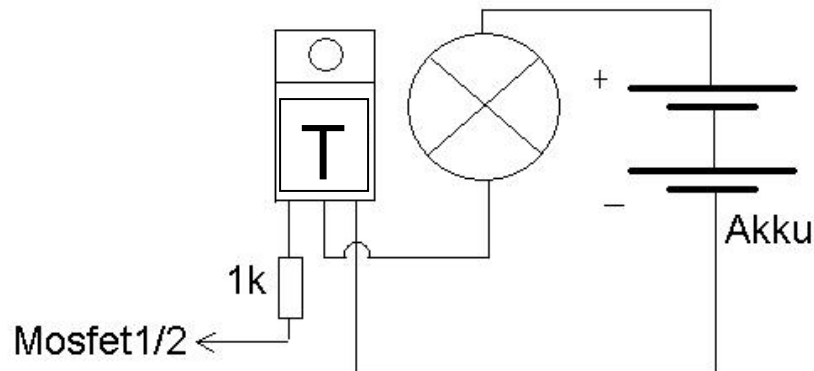
Zum Programmieren benötigt man z.B. das Programm PONYPROG und muß sich einen Programmieradapter bauen (keine Angst, gaaaanz einfach, siehe 4.1.). Alternativ gibt es diverse USB Programmer für ca. 15€. (z.B. unter www.myAVR.de, oder auch <http://www.ullihome.de>, oder ... im Netz und in der Bucht) mit komfortablen Programmen.

3. Der Aufbau:

- 3.1. den CellLog an eine Stromquelle >5V / < 40V anschließen und gucken, ob er geht. Passendes Kabel liegt bei. Er ist verpolgeschützt:o)
- 3.2. Programmieradapter aufbauen (wie unter 4.1 beschrieben)
- 3.3. Programmieranschluß an die Platine anbringen, z.B. so:



- 3.4. z.B. PONYPROG runterladen und den Controller programmieren (Siehe Kapitel 4.2) oder eine alternative Programmierlösung
- 3.5. Fuses des ATmega kontrollieren/setzen
- 3.6. Versorgungs- und PWM-Kabel anlöten:
- 3.7. Mosfets auf einem Stück Lochrasterplatine verlöten, in etwa so:



- 3.8. Steuerung an Akku anschließen und testen
- 3.9. Lampe anschließen und die gesamte Einheit testen
- 3.10. Programmierkabel entfernen
- 3.11. Wasserdicht machen, z.B. Einschrumpfen nach belieben, fertig!

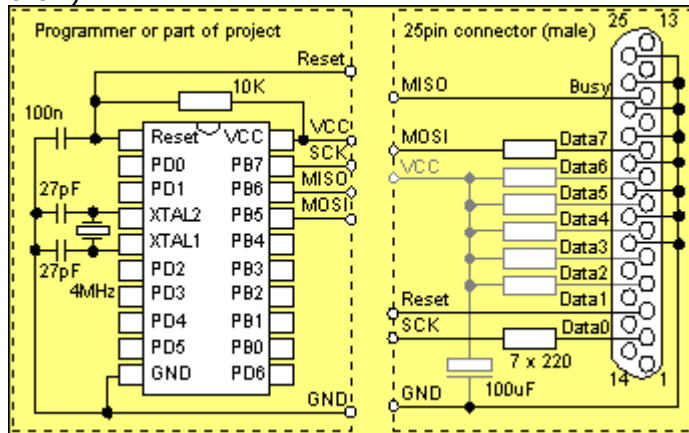
Danach muß die Steuerung nur noch an die persönlichen Anforderungen "personalisiert" werden, wie unter 5.2 beschrieben.

4. Programmieren

4.1 Aufbau des Programmieradapters

Es gibt diverse Programmieradapter. Ich arbeite mit dem Kanda und dem SP12 Dongle problemlos. Die Schaltungen sind recht simpel und stellen keine großen Anforderungen an das Geschick, man kann die Widerstände fliegend an den Stecker löten und daran die Kabel. Damit es keinen Kurzschluss gibt z.B. mit Klebeband einwickeln und das Gehäuse drumrum.

Hier als Beispiel der SP12 Dongle (nur die rechte Seite! Links käme dann der CellLog dran):



Weitere Info unter: <http://www.rowalt.de/mc/avr/progd.htm>

Oder auch hier: <http://s-huehn.de/elektronik/avr-prog/avr-prog.htm>.

Ich nutze den parallelen Adapter. Der Serielle geht auch, ist nur aufwendiger.

Für Rechner ohne Parallelen Port gibt es auch günstige USB Lösungen (ca. 15€) mit komfortablerer Software.

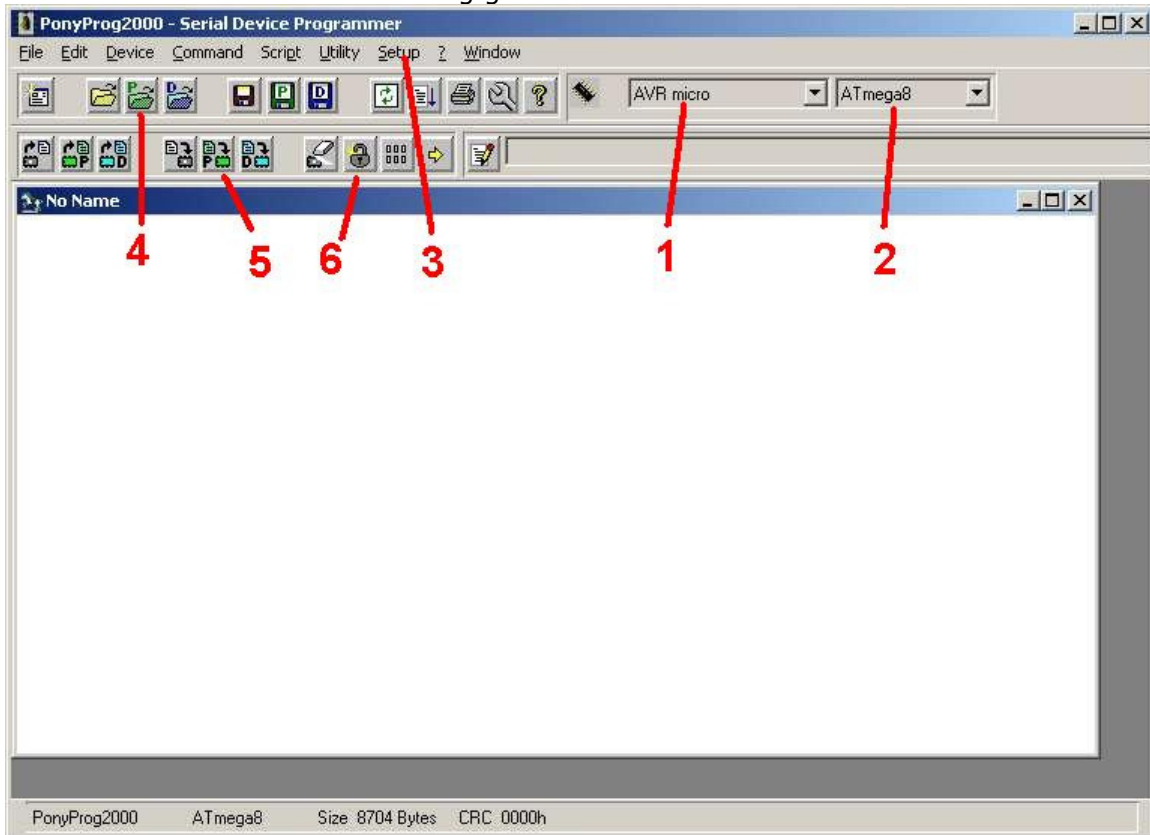
Ich benutze eine Stiftleiste zur Verbindung, direktes Anlöten der Kabel ist aber auch möglich, denn nach dem Programmieren ist der Adapter nicht mehr notwendig und kann wieder entfernt werden.

4.2 Programmiersoftware

Ich benutze u.a. die freie Programmiersoftware "PONYPROG" und habe gute Erfahrungen damit. Ebenso kann man TWINAVR nutzen. Ich werde mich hier in der Beschreibung an "PONYPROG" halten.

4.2.1 Elemente der Ponyprog Oberfläche

Hier sind die wichtigsten Bedienelemente von Ponyprog numeriert. Auf diese Nummern wird weiter unten Bezug genommen:



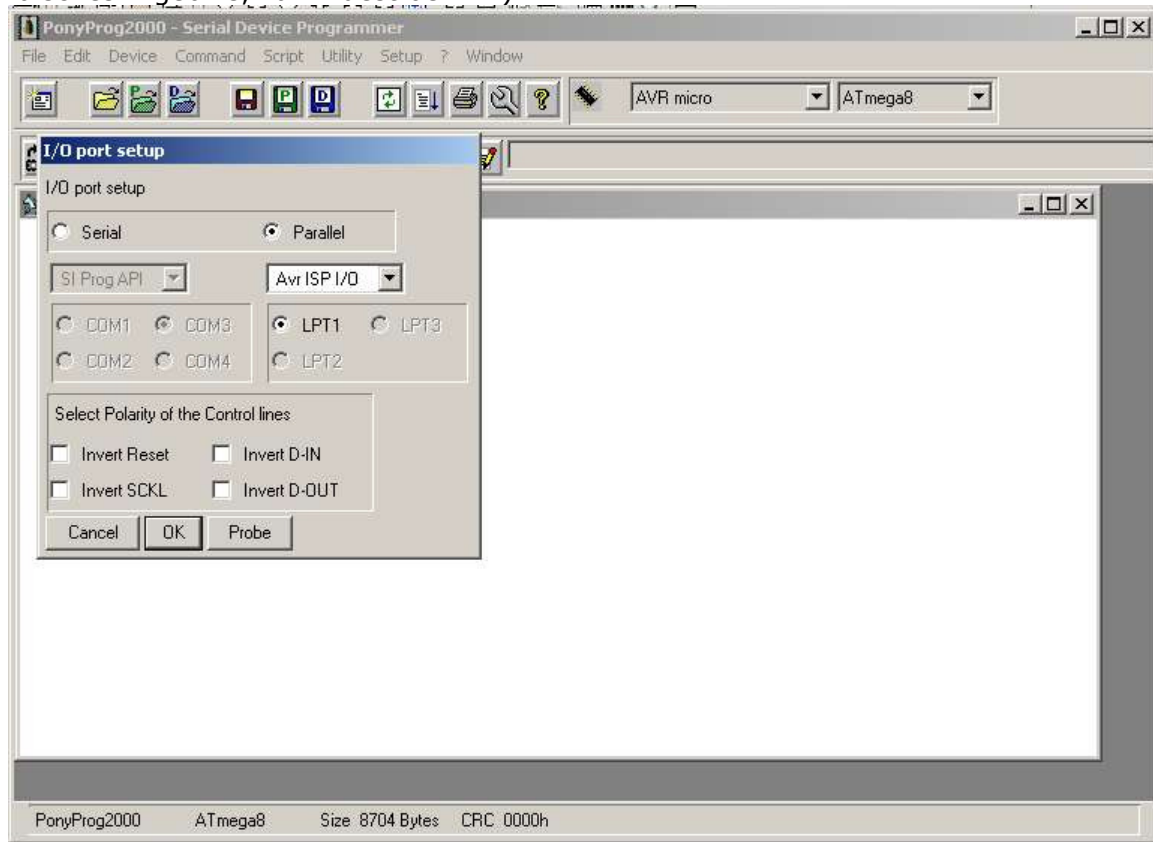
4.2.2 Einstellen des Controller Typs

Um dem Programm den genutzten Controllertyp (ATmega32) mitzuteilen, muß man bei (1) auf "AVR micro" stellen und bei (2) auf "ATmega32", damit weiß daß Programm, was auf es zukommt.

4.2.3 Einstellen des Interface: Menu Setup/Interface setup...

Unter "setup" (3) "interface setup..." wählen.

Hier auf parallel stellen, typischerweise "LPT1" auswählen und den entsprechenden Dongle wählen. Da die Namen nicht direkt zu den auf dem Markt üblichen Donglenamen passen, hilft nur ausprobieren, d.h. Dongle anschließen, Dongle an das Board anschließen, Spannung an das Board anschließen (Minus an GND und Plus an "Zelle1" mit dem beim Cellog mitgelieferten Kabelbaum mit Krokoklemmen) und den nächsten Schritt probieren. (Der Probe Button liefert hin und wieder auch ein falsches Ergebnis, zumindest bei mir)



4.2.4 Programmfile laden

Unter File/"Open program (flash) file..." oder Button (4) wird die Datei geladen. Im Fenster erscheint der Inhalt des Programmfiles. Keine Angst, das gehört so unleserlich für die Meisten;o)

4.2.5 File programmieren

Über Command/"Write Program (Flash)" oder Button (5) wird der Controller dann programmiert.

Nach dem geglückten Programmiervorgang sollte die Anzeige des Boards erstmal leuchten, ein großer Schritt ist getan!

Kommt eine Fehlermeldung: Programmieradapter und/oder Verkabelung überprüfen und/oder anderen Adapter unter 4.2.3 auswählen.

4.2.6 Konfiguration des Controllers (Fuse Bits)

Die sogenannten "Fuse Bits" der Atmel Controller ermöglichen eine gewisse Voreinstellung. Wenn man auf (6) drückt, werden die Fuse bits geladen und angezeigt. Hier ist nun folgende Konfiguration vorzunehmen:

Configuration and Security bits

☐ 7 ☐ 6 ☐ BootLock12 ☐ BootLock11 ☐ BootLock02 ☐ BootLock01 ☐ Lock2 ☐ Lock1

☐ OCDEN ☐ JTAGEN ☒ SPIEN ☐ CKOPT ☒ EESAVE ☐ BOOTSZ1 ☒ BOOTSZ0 ☒ BOOTRST

☐ BODLEVEL ☒ BODEN ☐ SUT1 ☐ SUT0 ☐ CKSEL3 ☐ CKSEL2 ☐ CKSEL1 ☐ CKSEL0

☒ Checked items means programmed (bit = 0) ☐ UnChecked items means unprogrammed (bit = 1)

Refer to device datasheet, please

Cancel OK Write Read

Damit wird u.a. die Taktfrequenz auf 4MHz extern und der Reset eingestellt.

Mit den restlichen Bits bitte nicht rumspielen, man kann den Controller damit kaputt machen!

5. Bedienung

Die Bedienung erfolgt über die drei Tasten Hoch/Runter/Menu. Es werden kurze (<1s) und lange (>1s) Tastendrücke unterschieden.

5.1. Normaler Betrieb

5.1.1 Initialisierung

Nach dem Einschalten meldet sich das Startbild mit „Bike Owl“ und danach erscheint kurz der Statusbildschirm. Dieser gibt Auskunft über den SW Stand und die erkannte Zellenzahl.

Danach erscheint die gewählte Hauptseite. (Aktuell sind nur zwei definiert.)

Diese zeigt im Ausgangszustand die Batteriekapazität ganz oben als Balken an. Links und rechts sind jeweils ein Bargraph für die aktuelle Lampenleistung zu sehen. In der Mitte wird die aktuelle Lampenlaufzeit angezeigt. Diese wird für beide Lampen getrennt angezeigt und entspricht 100% Leistung, d.h. bei 50% Leistung zählt die Uhr nur halb so schnell hoch.

5.1.2 Ein-/Ausschalten:

Mit den Tasten Hoch/Runter kann nun die Leistung der Lampe 1, sichtbar im linken Bargraphen, entsprechend den programmierten Stufen geschaltet werden.

Mit der Menu Taste kann durch einen kurzen Klick zwischen den beiden Lampen umgeschaltet werden (inaktiver Bargraph ist ausgegraut).

Die aktuelle Leistung der Lampe 2 wird im rechten Bargraph angezeigt.

5.1.3. Batterieüberwachung

Um eine Beschädigung der Batterie zu verhindern, schalten sich die Lampen bei Erreichen der 0% Grenze (3V/Zelle) automatisch ab.

Die Lampe kann erst durch Trennen der Versorgung wieder aktiviert werden!

5.2 Konfiguration

Mit der Konfiguration läßt sich die Lampensteuerung an die persönlichen Bedürfnisse und die angeschlossene Hardware anpassen.

Wichtig ist hierbei: man kann durch die falsche Konfiguration auch die Lampen beschädigen, z.B. wenn man versucht, eine Xenonlampe zu dimmen, die nicht dafür vorgesehen ist! Oder man kann die Schutzschaltung des Akkus zerstören, indem man eine Halogenlampe hart an zwei BP945 Akkus in Serie zuschaltet.

Es gibt aber für fast jede Lampe und auch jeden Geschmack die passende Konfiguration;o)

5.2.1. Einstieg in den Konfigurationsmodus

Zum Aktivieren des Konfigurationsmodus muß man den Taster Menu länger als 1s gedrückt halten, dann loslassen.

Es erscheint das Konfigurationsmenu:

```
*** CONFIGURATION ***
```

```
* SYSTEM..
  LAMP1..
  LAMP2..
  STATISTIC..
  DISPLAY..
  AUXILIARY..
```

Mit Hoch/Runter kann man nun das Sternchen an der Linken Seite an den gewünschten Menüpunkt bewegen. Ausgewählt wird das Menu mit einem kurzen Klick auf den Menu Taster.

Einen Schritt zurück im Menu kommt man mit einem langen Menu Tastendruck.

5.2.2 Menu „SYSTEM“

```
*** CONFIGURATION ***
***      SYSTEM      ***
* BRIGHT  0000000
  CONTRAST 0000000
  LOWBAT WARNING xxx%
  SOUND   choice
```

5.2.2.1 Menupunkt BRIGHT

Einstellung der Helligkeit.

5.2.2.2 Menupunkt CONTRAST

Einstellung des Kontrasts.

5.2.2.3 Menupunkt LOWBAT WARNING

Einstellung der Kapazitätsschwelle in %, bei der eine (akustische) Warnung erfolgt.
Die Kapazitätsanzeige blinkt danach im Sekundenrhythmus.

5.2.2.2 Menupunkt SOUND

Einstellung der Tonausgabe. Auswahl: SILENT, WARNINGS, ALL

5.2.3 Menus „LAMP1“/“LAMP2“

```
*** CONFIGURATION ***
***      LAMPx      ***
* STEP#x PWMxxx%
  STARTUP choice
  LOWPOWER xxx%
  LOWPWMMAX xxx%
  PWM FREQUENCY xxxHz
```

5.2.3.1 Menüpunkt STEP

Einstellung der Dimmstufen. Ein kurzer Menu Tastendruck geht durch die Schritte (STEP#), dort kann man dann jeweils die PWM Helligkeit in % angeben. Erhöht man das PWM über 100% hinaus, ist dieser Punkt die letzte der Dimmstufen. Will man eine Dimmstufe mehr einfügen, muß man also einfach bei der Anzeige PWM END einmal Runter drücken.

Mit diesem Menüpunkt kann man (ungeeignete) Lampen u.U. zerstören! Xenon verträgt z.B. normalerweise keine Dimmung.

5.2.3.2 Menüpunkt STARTUP

Einstellung des Lampenstarts. Zur Auswahl stehen

- STEP=harte Ansteuerung der einzelnen Dimmstufen
- SOFT=fließende Ansteuerung (0..100%=2s)
- SPARK="Xenonblitz", sieht z.B. bei LEDs ganz witzig aus;o)

Mit diesem Menüpunkt kann man ungeeignete Lampen u.U. zerstören!

5.2.3.3 Menüpunkt LOWPOWER

Einstellung der jeweiligen Kapazitätsschwelle in %, bei der eine Begrenzung des Lampen PWMs erfolgt, damit man nicht schlagartig im Dunkeln steht.

5.2.3.2 Menüpunkt LOWPWMMAX

Einstellung der PWM Begrenzung der Lampe 1 oder 2 bei zu geringer Kapazität.

5.2.4 Menu „STATISTIC“

```
*** CONFIGURATION ***  
*** STATISTIC ***  
* LAMP1  
  LAMP2  
  CLEAR1  
  CLEAR2  
  TOTAL
```

5.2.4.1 Menüpunkt LAMP1/2

Hier wird die gewertete Brennzeit der einzelnen Lampen über Lebensdauer angezeigt. D.h. bei 50% Helligkeit zählt der Zähler um eins in zwei Sekunden hoch, bei 100% um eins in einer Sekunde, etc.
Das Anzeigeformat ist TTT.HH.MM.SS. Es werden somit max. 255 Tage aufsummiert (=6144h). Wird ev. nochmal upgedatet, das ist ja Ruckzuck rum;o)

5.2.4.2 Menüpunkt CLEAR1/2

Die akkumulierten Lebensdauerzeiten der Lampen können hiermit genullt werden, z.B. beim Wechsel der Lampen.

5.2.4.3 Menüpunkt TOTAL

Hier wird die gesamte absolute Leuchtdauer akkumuliert, sobald eine Lampe an ist. Man kann also analog zum Winterpokal einen Nightridepokal machen;oD

5.2.5 Menu „DISPLAY“

```
*** CONFIGURATION ***  
***   DISPLAY   ***  
* CAPACITY xxxxxx  
  STYLE      x
```

5.2.5.1 Menüpunkt CAPACITY

Hier kann die Anzeige des Akkustatus im Hauptbild eingestellt werden.
Es stehen zur Auswahl BARGRAPH, VOLT, PERCENT.
Die Prozentrechnung wird von 3.0..4.2V/Zelle ausgehen berechnet.

5.2.5.2 Menüpunkt STYLE

Hier kann in Zukunft der Stil des Hauptbildschirm umgestellt werden.

5.2.6 Menu „AUXILIARY“

```
*** CONFIGURATION ***  
***   AUXILIARY   ***
```

Hier ist noch eine Baustelle, da kommen noch Sonderfunktionen rein, z.B. Voltmeter mit Einzelspannungsmessung, etc.

6. Ausblick

Was kommt noch?

Geplant habe ich im Moment:

- Einzelzellenüberwachung (2..8Zellen)
- verschiedene „Styles“, d.h. Anzeigeoptionen zur Auswahl
- Mehrwertfunktionen, z.B. LiIon Spannungsscheck (also die alte Funktion des CellLog;o)
- Vorschläge willkommen

Viel Spaß in der Nacht wünscht Euch

Der Nikolauzi