

# OpenSeaMap Hardware Logger

Durch einen Kontakt über den Franzis Verlag bin ich auf dieses schöne Projekt gestossen.

[OpenSeaMap](#)

Ziel von OpenSeaMap ist es, für den Seemann interessante nautische und touristische Information in die freie Weltkarte einzutragen und in nautisch ansprechender Weise darzustellen.

Dazu gehören auch die Tiefendaten.

Da diese aber nicht immer frei zugänglich bzw. auch manchmal recht veraltet sind, kam man auf die Idee mithilfe von Crowdsourcing die Daten durch Hobbyisten aufbessern zu lassen.

Es gibt mittlerweile schon einige recht spannende Projekte rund um das Projekt.

Wassertiefenmessung per WLAN Router (der dazu eine neue Firmware erhält), per Handy, oder auch per PC. Für einige Hobbyskipper ist die Hürde aber zu hoch. Um mit helfen zu können muss also ein Logger zur Verfügung stehen der



- einfach zu installieren und zu bedienen ist
- die Daten von GPS und Echolot aufnehmen kann
- und diese dann auf ein einfaches Speichermedium bringt.

Zusammen mit verschiedenen Leuten vom OpenSeaMap Projekt habe ich einen Datenlogger entwickelt, der genau die Vorgaben erfüllt.

Features

- Datenlogger für GPS und Echolot Daten.
- Eingebauter Lagesensor
- Daten auf normaler SD Karte im NMEA 0183 Format
- Einfachste Bedienung.
- LED für Betrieb, Datenempfang und SD Zugriff
- 2 NMEA0183 Eingänge
- 1 Eingang kann für SeaTalk mit entsprechender Platine benutzt werden. (Experimentell)
- einfacher Anschluss per RJ45.

Der Logger wird einfach an geeigneter Stelle ins Boot eingebaut. Zum Anschluss dient ein einfaches Netzkabel. (Der Logger hat keinen LAN Eingang, sondern wir benutzen nur den elektrischen Anschluss für unsere Zwecke.)

[Hier findest du die aktuelle Bedienungsanleitung.](#)

## Hardware

Die Hardware selber besteht aus einer Handvoll Bauteile:

- ATmega328P als Prozessor
- MPU 6050 als Lage und Beschleunigungssensor
- 3 Optokoppler für die Eingänge, damit Einflüsse auf die sonstige Boardelektrik direkt ausgeschlossen werden können.
- SD Karten Einschub

- Goldcap als Backup, damit das Dateisystem der SD Karte beim Ausschalten nicht zerstört wird

So sieht die Platine aus:



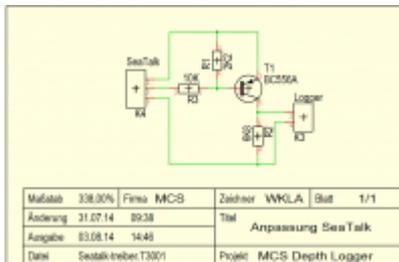
Zusätzlich bekommt der Logger einen eigenen Bootloader, sodass später im Bootseinsatz die Firmware per SD-Karte upgedatet werden kann.

Hier eine genauere Beschreibung: [Beschreibung des OpenSeaMap Loggers](#)

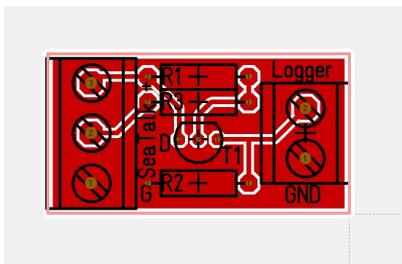
Zu Beziehen ist die Hardware über [Küstenschmiede GmbH](#)

## Seatalk

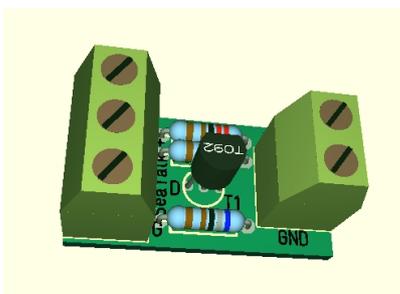
Bedingt durch den Aufbau des Seataalk-Buses ist der direkte Anschluss des Loggers an diesen Bus nicht möglich. Eine Lösung habe ich bereits erarbeitet. Um den Logger an Seataalk anzuschliessen muss eine kleine Extrahardware benutzt werden. Diese wird zwischen Seataalk Bus und Logger geschaltet.



Schaltplan:



Platine:

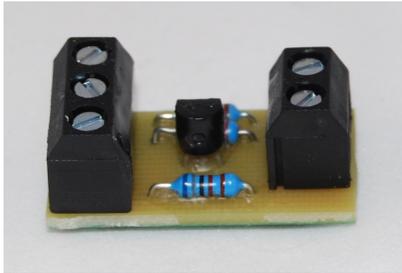


3D Modell:

Auf der linken Seite wird der Seataalk Bus mit den 3 Leitung +12V, Data, GND angeschlossen. Auf der Rechten Seite geht's dann zum Logger. Dort wird der NMEA A Anschluss verwendet (nicht den Seataalkanschluss). Ein Umstecken des Jumpers entfällt nun, da die kleine Platine die elek. Konvertierung der Signal übernimmt. Allerdings muss die Firmware weiterhin mittels

Konfigurationsdatei auf Seataalk umgeschaltet werden. Die Platine wird ab Anfang September zur Verfügung stehen.

### 6.10.2014 fertige Platine.



Hier auch die Anleitung: [SeaTalk treiber.pdf](#)

## Firmware

aktuelle Firmware und Handbuch findet man hier: [Firmware](#)

Eigentlich hört sich die Software für den Logger (Firmware) recht einfach an. 2 Serielle Schnittstellen mit gerade mal 4800 Baud auslesen und auf eine SD Karte schreiben. Und nebenher jede Sekunde einmal die Daten von dem Acc/Gyro Chip auslesen und auch dazu schreiben. Der Teufel steckt dabei aber im Detail.

- Die SD Karte

Für einen kleinen Microcontroller ist das Schreiben und Lesen von SD Karten eh schon eine eigene Herausforderung. Noch dazu muss FAT32 unterstützt werden, damit auch größere Karten benutzt werden können. Alleine die Bibliothek dafür frißt schon fast den halben Flashspeicher. Kleinere Probleme gibt es immer noch mit den SD Karten. Diese sind zwar mittlerweile recht schnell geworden, haben aber ab und zu mal eine Latenzzeit von bis zu 250ms, wo sie den kompletten Programmablauf blockieren. Dabei können, wenn man nicht aufpasst, die serielle Puffer überlaufen.

- serielle Schnittstellen

Dann müssen die beiden Schnittstellen gleichzeitig ausgelesen, aber nacheinander geschrieben werden. Damit man später die Datensätze wieder in die richtige Reihenfolge bringen kann, muss ein zus. Zeitstempel vor jeden Datensatz geschrieben werden.

Weiterhin ist das SeaTalk 1 Protokoll ein 9Bit Protokoll für das es erstmal im Standartarduino Umfeld keine Unterstützung gibt.

Also habe ich basierend auf der vorhandenen HardwareSerial Schnittstelle eine Erweiterung implementiert, die sowohl 9N1 kann, wie auch die Empfangs- und Sendepuffer getrennt einstellbar macht.

- interne Daten

Für den Acc/Gyro mussten auch eigene NMEA 0183 Nachrichten erzeugt werden.

Der Einfachheit halber habe ich für das Board ein eigene Arduino Hardware Definition erzeugt. Das ganze Paket findet man in meinem [Github repository](#).

### 13.10.2013 WKLA

Neues vom Logger.

Die Karten waren gar nicht das Problem, sondern die für die 2. Schnittstelle benötigte SoftwareSerial. Die ist leider, na sagen wir es vorsichtig, etwas suboptimal implementiert. Bei niedrigen Baudraten bremst Sie gehörig das System aus, was dann zum Pufferüberlauf führt. Deswegen habe ich SoftwareSerial durch eine bessere Implementierung [AltSoftSerial](#) ersetzt. Zwar kann die nur auf bestimmten Pins hören, aber aus welchen Gründen auch immer, hab ich im Layout genau die richtigen Pins verwendet.

Statt bisher 10-15% Fehlern, macht der Logger jetzt nur noch 0,1% Fehler. Und das sind wahrscheinlich Leitungsfehler...

### **26.03.2014 WKLA**

Wieder neues vom Logger.

Seit ein paar Tagen beschäftige ich mich intensiv mit einem neuen Bootloader für den Logger. Der „alte“ kann nur FAT16 formatiert SD Karten. Nun wird es aber langsam schwierig an diesen Kartentyp heran zu kommen. Außerdem verweigert er bei SDHC Karten leider völlig den Dienst. Deshalb ist es mir wichtig einen neuen besseren Bootloader für den Logger zu haben.

Heute ist es endlich soweit. Der neue Bootloader kann getestet werden. Das Problem, um ihn zu testen muss man ihn erst einmal auf die Loggerhardware bekommen. Und das geht leider nur mit einem ISP und dem entsprechenden Brennprogramm. Wer sowas hat, kann sich bei mir melden. Ich hab im Repository bereits alles für ein einfaches Update vorgesehen. Aber bitte vorher melden. Man kann auch einiges falsch machen. Der neue Bootloader kann jetzt endlich FAT32 Karten verarbeiten und dated auch nur ab, wenn eine neuere Firmware-Version auf der Karte ist.

Also Schluss mit 2 verschiedenen Karten für's Loggen und Updaten. Allerdings ist dazu auch eine neue Firmware nötig. Somit hab ich die Firmware Version 12 auch gleich als Beta bereit gestellt.

### **30.03.2014 WKLA**

neue Testdaten hab ich jetzt auch.

[Download](#)

### **17.04.2014 WKLA**

Bootloader 2 und Firmware 12 stehen nun offiziell zur Verfügung.

### **27.06.2014 WKLA**

Die Probleme mit Seataalk sind erkannt und eine Lösung wird gerade erarbeitet. Wir bitten noch um etwas Geduld.

### **6.10.2014 WKLA**

Das SeaTalk-Modul ist nun endlich fertig. Die Platinen können bei AK Modulbus erworben. Um SeaTalk zu nutzen ist zwingend ein Update auf die Firmware Version 0.1.15 nötig. Dieses steht zeitgleich zur Verfügung.

## **Konfiguration**

Für den Logger gibt es eine eigene Konfigurationsseite.

Die findet man hier: [MCS Depth Logger Konfiguration](#)

aktuelle Firmware und Handbuch findet man hier: [Firmware](#)

### **Nur für Logger mit Firmware Version 0.9 und ohne Seataalk Modul.**

Wichtig: Für Seataalk muss im Gerät ein Jumper umgesteckt werden.

Hier die beiden Einstellungen:



NMEA



Seatalk

## technische Daten

- Betriebsspannung: 9V-15V DC
- Stromaufnahme: max. 100mA
- unterstütze SD Karten: SD/MMC/SDHC Karten mit FAT16 oder FAT32 Dateisystem. <sup>1)</sup>
- Anschlüsse: 2x seriell RS422 Protokoll (RS232 kompatibel), 1x SeaTalk, galvanisch getrennt, 1x Stromversorgung, über RJ45
- 3 LED's zur Statusanzeige
- interner Stromspeicher für kurzfristigen Spannungsstörungen
- Lage- und Beschleunigungssensor MPU 6050
- Prozessor: ATmega328P, 2kb RAM, 1KB EEPROM, 32KB Flash, 16MHz
- 6-poliger ISP Anschluss
- robustes Metallgehäuse

## PC Software

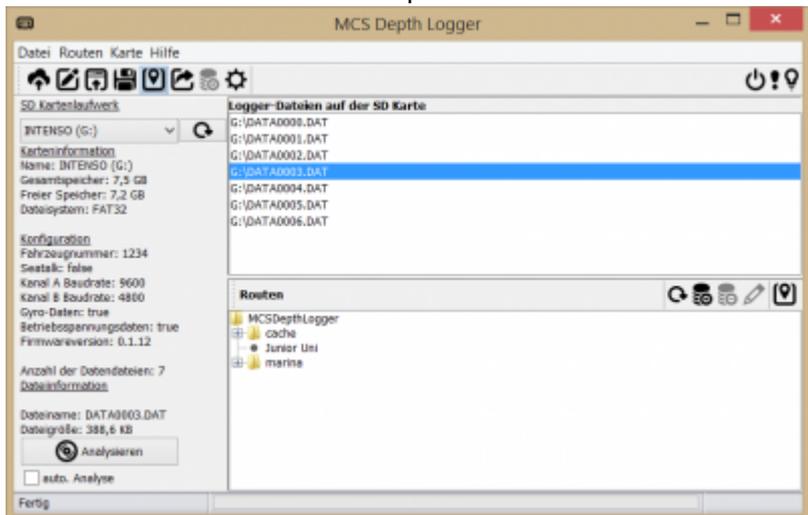
Speziell für diesen Logger habe ich eine PC Software geschrieben, die speziell für diesen Logger abgestimmt ist.

Die Software enthält folgende Features:

- Verwaltung/Backup/Restore der SD Karten
- Verwaltung der Firmware inkl. Vorbereitung der SD Karte für ein evt. Firmware Update
- Überprüfung/Korrektur der geloggtten Daten
- Export ins NMEA Format
- Upload auf einen Server
- Darstellung einzelner Daten auf Karten
- Verwaltung von Routen.
  - Hinzufügen/Löschen von einzelnen Dateien
  - Darstellung auf Karte
- Kartendarstellung
  - verschiedene Kartenprovider: OpenStreetMap, OpenSeaMap, Virtual Earth
  - Möglichkeit verschiedener Overlays wie Sportstätten, Seezeichen. (WEitere sind in Vorbereitung)

- Darstellung verschiedener Daten als Diagramm: Tiefe, Höhe, Beschleunigung und Lage, Geschwindigkeit
- Internationalisierung: Sprachen Deutsch und Englisch, Maßsysteme: metrisch, imperial, nautisch

Hier mal ein Blick auf das Hauptfenster:



Links sieht man die Angabe zu der verwendeten SD Karte sowie Angaben zu dem verwendeten Logger. Hier kann man auch eine Analyse der Daten starten.

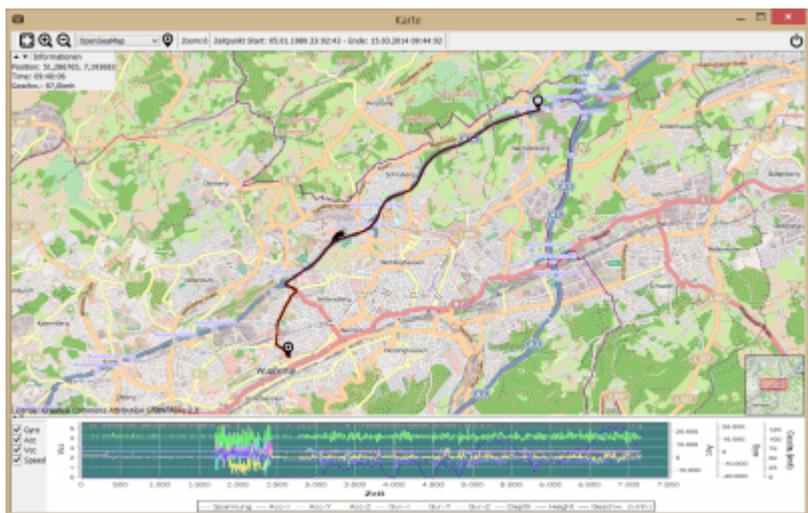
Mitte oben sieht man die Datendateien auf der SD Karte. Hier kann man die Daten selektieren. Mitte unten sieht man die Routenverwaltung. Mehrere Datendateien können hier zu einer Route zusammengefasst werden.

Jetzt mal ein Blick auf die Kartendarstellung.

Oben sieht man die Karte mit der eingezeichneten Route. Die Route wird mit Start und Endpunkt eingezeichnet.

Darunter sieht man eine Chartansicht mit den verschiedenen Daten, wie Beschleunigung, Lage, Höhe, Geschwindigkeit.

Klickt man in dieses Chart wird in der Karte der aktuelle Punkt gekennzeichnet und man sieht oben links im Fenster die Daten zu diesem Punkt.



Die aktuelle Version kann man hier herunter laden:

[MCSDepthLogger](#)

Java 1.7 ist allerdings Voraussetzung für die Benutzung.

Testdaten hab ich auch: [Testdata](#)

Das ganze Projekt ist OpenSource und wird unter [MCS Depth Logger @ sourceforge](#) gehostet. Viel Spass damit.

## Datenformat

Der Logger schreibt alle Datensätze, die er empfängt auf die SD Karte. Dabei wird keinerlei Analyse oder Filterung vorgenommen. Beim Empfang des 1 Bytes eines Satzes wird ein interner Zeitstempel generiert. Dieser wird dann beim Schreiben vor dem Satz eingefügt. Weiterhin kommt noch eine Kanalkennzeichnung hinzu. Zusammen mit dem Zeitstempel des GPS kann man den tatsächlichen Zeitpunkt berechnen.

Bei Seataalk werden die Daten in ein NMEA Datensatz verpackt. Da die Seataalk Datagramme selber binär sind, werden Sie in hexadezimalen Format und dann mit entsprechender Prüfsumme als eigener NMEA Datensatz auf die SD-Karte geschrieben.

Für den Lagesensor, die Betriebsspannung sowie für den Start bzw. den Stopp des Loggers werden eigene Messages verwendet.

Das Datenvolumen hängt natürlich stark von den verwendeten Geräten ab. Die technisch bedingte Obergrenze liegt bei 3,3MB die Stunde bzw. 80Mb pro Tag. Damit passen auf eine 2GB SD-Karte bei voller Last ca. 25 Tage.

Tests haben jedoch ergeben, daß ein Wert von ca. 2MB pro Stunde selten überschritten wird, was bei einer 2GB Karte einen Wert von ca. 40 Tagen Daueraufzeichnung ergibt.

## Start/Stopmeldung

Zunächst gibt es Start- und Stopmeldungen für den Logger

```
00:00:11.165;I;$POSMST,Start NMEA Logger,V 0.1.2*30
```

```
00:00:27.651;I;$POSMSO,Stop NMEA Logger*3A
```

Die vorderen Zeichen, bis zum „;“ sind der Zeitstempel.

Dann folgt die Kanalkennzeichnung, I bedeutet intern, A für Kanal A, B entsprechend.

Seataalk wird auf Kanal A empfangen.

Als letzter Teil folgt dann die eigentliche Meldung.

### ab Version 12

Am Ende einer jeden Datei steht eine Stopmeldung, die den Grund für das Dateiende angibt.

```
00:00:27.651;I;$POSMSO,Reason: times up*3A
```

Automatischer Wechsel aufgrund der Zeiteinstellung. Jede Stunde wird eine neue Datei geschrieben.

```
00:00:27.651;I;$POSMSO,Reason: supply low*3A
```

Ende der Aufzeichnung, da die Spannung des Loggers zu gering ist. (Normale Meldung beim Ausgeschalten)

```
00:00:27.651;I;$POSMSO,Reason: stop switch*3A
```

Es wurde der Stopschalter gedrückt. (Nur bei Geräten mit herausgeführtem Stopschalter)

## Gyroscope

Das Gyroscope basiert auf dem MPU6050 Baustein von InvenSense ([MPU 6050](#)). Als Einstellungen verwende ich 250°/sec, was ich für ein Boot als völlig ausreichend betrachte. Im Datensatz erscheinen dann die 16-Bit Werte der X/Y/Z Achse ohne weitere Berechnung. Um die tatsächliche Achsenbewegung zu ermitteln, müssen die Werte noch entsprechend verarbeitet werden.

```
00:00:11.179;I;$POSMGYR,1800,7772,13164*5C
```

## Accelerometer

Das Accelerometer ist ebenfalls auf dem MPU6050 Baustein vorhanden. Als Einstellungen verwende ich +-2g. Im Datensatz erscheinen dann die 16-Bit Werte der X/Y/Z Achse ohne weitere Berechnung. Um die tatsächliche Beschleunigung zu ermitteln, müssen die Werte noch entsprechend verarbeitet werden.

```
00:00:11.179;I;$POSMACC,-684,-2145,-7012*7D
```

## Betriebsspannung

Die Betriebsspannung ist nur für Tests interessant und sollte im realen Betrieb über die Konfiguration abgeschaltet werden. (Ist auch der Default) Die angegebene Spannung ist die Spannung in mV.

```
00:00:11.151;I;$POSMVCC,4962*72
```

### ab Version 12

```
00:00:11.151;I;$POSMVCC,4962, 4792*72
```

Hier wird nun zusätzlich als zweiten Wert die Abschaltspannung mit ausgegeben.

## Seatalk-1

Da Seataalk ein binäres Format ist, werden die einzelnen Bytes als HEX ausgegeben. NMEA Kennzeichnung ist \$POSMSK. Danach folgt eine Kolonne aus HEX Zahlen. Jeweils 2 Ziffern bilden ein Byte.

```
00:00:11.253;A;$POSMSK,4AEF4245265E62*72
```

## FAQ Fragen und Antworten

### ? Bei meinem Logger funktioniert SeaTalk nicht.

Sobald ich den Logger an den SeaTalk-Bus anschliesse melden die anderen Geräte einen Busfehler.

! SeaTalk ist ein Bussystem, welches versucht mit sehr geringen Strömen auszukommen. Leider reicht die vorhandene Stromstärke auf dem Bus nicht in allen Fällen aus, um die Daten in den Logger zu schicken.

Eine Lösung für das Problem ist in Arbeit und wird voraussichtlich ab September zur Verfügung stehen.

### ? Kanal A und B vertauscht?

Gebe ich laut beiliegender Anleitung ein GPS Signal auf den Kanal A des Logger, wird in der Datei dieses Signal mit Kanal B gekennzeichnet.

! Ja, in der beiliegenden Anleitung sind die beiden Kanäle vertauscht. Der Kanal A liegt auf den beiden

inneren Leitungen (4,5) des RJ45 Steckers, Kanal B auf den beiden äußeren Leitungen (3,6).

### **? Seotalk und Kanal B nicht gleichzeitig nutzbar?**

Gebe ich laut beiliegender Anleitung ein Signal auf den Seotalk Kanal des Logger, und ein weiteres Signal auf Kanal B, wird nur das Seotalk Signal aufgezeichnet.

! Ja, in der beiliegenden Anleitung sind die Kanäle A und B vertauscht. Der Kanal A liegt auf den beiden inneren Leitungen (4,5) des RJ45 Steckers, Kanal B auf den beiden äußeren Leitungen (3,6). Da der Kanal A auch für Seotalk zuständig ist, müssen die Geräte auf Seotalk und Kanal B angeschlossen werden.

### **? Mein Logger zeichnet keine Daten auf.**

Eigentlich funktioniert alles einwandfrei. Die LED's leuchten richtig, aber nach kurzer Laufzeit ist zwar eine Datendatei auf der SD Karte, aber diese ist leer.

! Logger, die auf der Boot 2014 verkauft wurden, haben ein kleines Problem, daß die Daten erst nach 1 Stunde korrekt sichtbar werden. Bitte warten Sie einfach die entsprechende Zeit ab.

Eine weitere Möglichkeit ist, die Zusendung des Loggers an mich. Ich werde dann den Logger auf den aktuellen Bootloader und die aktuelle Firmware updaten. Leider muss ich zumindest die Portokosten (innerhalb Deutschlands 3.90€) in Rechnung stellen.

### **? Mein Logger nimmt meine neue 4GB SDHC Karte nicht.**

Mit kleineren Karten funktioniert der Logger einwandfrei, aber größere Karten nach dem SDHC Standard werden nicht angenommen. Es sieht so aus, als ob der Logger hängt.

! Es gibt ein paar Logger, die mit den aktuellen SDHC Karten nicht funktionieren. Ursache ist ein Timing-Problem des Bootloaders mit einigen SDHC Karten. Sie können diese SD Karten aber nach dem Bootloader-Start einlegen. (Wenn die rote LED blinkt) Dann wird die Karte verwendet.

Eine weitere Möglichkeit ist, die Zusendung des Loggers an mich. Ich werde dann den Logger auf den aktuellen Bootloader und die aktuelle Firmware updaten. Dieser Service ist für alle Logger, die auf der Boot verkauft wurden, kostenlos. Leider muss ich aber zumindest die Portokosten (innerhalb Deutschlands 2,40€) in Rechnung stellen.

### **? Alle Dateien des Loggers haben das Datum 1.1.2000.**

Der Logger funktioniert, aber alle Dateien auf der SD Karte haben das gleiche Datum 1.1.2000. Warum?

! Ja, das ist so korrekt. Der Logger besitzt keine Echtzeituhr. Und auch auf eine Auswertung der GPS Daten wurde bewußt verzichtet, weil eine solche Auswertung zusätzlich Rechenleistung und Speicher kostet. Und beides ist in der Loggerhardware Mangelware.

Im Auswerteprogramm MCSDepthLogger wurde jedoch eine Funktion in der Dateiliste integriert, mit dessen Hilfe man die Dateien mit dem richtigen Zeitstempel versehen kann. Dazu werden alle Dateien, die ein Datum < 2.1.2000 haben, eingelesen und der letzte GPS Datensatz (GPRMC) wird dann als letztes Änderungsdatum für die Datei verwendet. Enthält die Datei keine bzw. keine gültigen GPS Datensätze, wird die Datei ignoriert und nicht verändert.

Diese Funktion steht ab der Build-Version 292 zur Verfügung.

## **Errata**

Leider hat sich in die Anleitung ein kleiner Fehler eingeschlichen. (Siehe auch FAQ Frage 1 und 2) Hier

die korrektur als PDF: [Errata V1](#)

1)

ACHTUNG: Durch einen Bug im aktuellen Bootloader ist es nicht möglich, einen automatischen Betrieb mit SDHC (FAT32) Karten zu ermöglichen. Bitte verwenden Sie dafür 2GB SD (MMC) Karten mit FAT16 Dateisystem. Eine Lösung steht bereits zur Verfügung. Wenn der automatische Start mit SDHC Karten dringend benötigt wird, kontaktieren Sie mich bitte.

From:

<https://wkla.no-ip.biz/ArduinoWiki/> - **Arduino im Modellbau**

Permanent link:

<https://wkla.no-ip.biz/ArduinoWiki/doku.php?id=arduino:oseam>

Last update: **2018/11/04 10:51**

