

external Interface for the Open-Source Dive-Computer

Date: 01/12/08

Last Update: 11/04/08

Written by: Matthias Heinrichs (info@heinrichsweikamp.com) and Pascal Pellmont

Hardware

The USB interface is equipped with an USB Chip (FT232 RL) from FTDI. You can get Windows/

Linux drivers from here: <http://www.ftdichip.com/Drivers/VCP.htm> (current Version is 2.02.04)

With Linux Kernel >2.6.9, these drivers are included in the Kernel.

After installation a virtual COM port can be found in your windows hardware control panel.

You can change the COM port number there if required.

Software

The data rate is 115200 Baud, 8 Databits, 1 Stopbit, no parity, (115200, N, 8, 1)

16bit Values are transmitted in little Endian format (LOW:HIGH)

Transfer protocol

Transfer is bi-directional and is initiated from the PC software by sending a command byte to the OSTC

command 0x61 („a“): download status and dive profiles from OSTC

Output OSTC:

Byte1-5:

0xAA 0xAA 0xAA 0xAA 0xAA (Preamble)

Byte6:

0x55 (Preamble)

Byte7-262:

Internal EEPROM BANK 0 (First 256Bytes) **See Command „g“**

Byte263-264:

current battery voltage in mV

Byte265-266:

2 Byte current firmware version , 1st Byte.2nd Byte (e.g. „0.19“)

Byte267-33034 (32768 Byte):

logbook with profile data

The complete profile memory will be send, no matter how many dives are stored.

One dive consists of a header and the profile block. First, the header is send with a fixed length of 46 bytes:

Byte 0-1:

0xFA 0xFA (start Header)

Byte 2:
Logbook-Profile version (0x20 for this 46 Byte version)

Byte 3:
Date of dive, month

Byte 4:
Date of dive, day

Byte 5:
Date of dive, year (20xx)

Byte 6:
Time at end of dive, hours

Byte 7:
Time at end of dive, minutes

Byte 8-9:
max. depth in mBar

Byte 10-11:
divetime (minutes)

Byte 12:
divetime (seconds)

Byte 13-14:
min. watertemperature in 0.1 °C

Byte 15-16:
Surface air pressure before dive

Byte 17-18:
Desaturationtime in minutes after the dive

Byte 19-20:
Gas 1 current (%O2, %He)

Byte 21-22:
Gas 2 current (%O2, %He)

Byte 23-24:
Gas 3 current (%O2, %He)

Byte 25-26:
Gas 4 current (%O2, %He)

Byte 27-28:
Gas 5 current (%O2, %He)

Byte 29-30:
Gas 6 current (%O2, %He) (Gas 6 is a manual gas that can be configured during the dive),
value at Beginning of dive.

Byte 31:
Selected Gas# at beginning of dive

Byte 32-33:
2 Byte firmware version , 1st Byte.2nd Byte (e.g. „0.19“)

Byte 34-35:
2 Byte battery voltage in mV after the dive

Byte 36:
Sampling Rate in Seconds

Byte 37:

Sampling Divisor Temperature

Bit 0-3: Divisor, 0=never, 1=every Sample, 2=every 2nd Sample, ...

Bit 4-7: Information size (currently 2)

Byte 38:

Sampling Divisor Deco/NDL Status

Bit 0-3: Divisor, 0=never, 1=every Sample, 2=every 2nd Sample, ...

Bit 4-7: Information size (currently 2)

Byte 39:

Sampling Divisor Tank Pressure

Bit 0-3: Divisor, 0=never, 1=every Sample, 2=every 2nd Sample, ...

Bit 4-7: Information size (currently 2)

Byte 40:

Sampling Divisor ppO2 Sensors

Bit 0-3: Divisor, 0=never, 1=every Sample, 2=every 2nd Sample, ...

Bit 4-7: Information size (currently 3)

Byte 41:

Sampling Divisor "not yet used 1"

Bit 0-3: Divisor, 0=never, 1=every Sample, 2=every 2nd Sample, ...

Bit 4-7: Information size (currently 0)

Byte 42:

Sampling Divisor "not yet used 2"

Bit 0-3: Divisor, 0=never, 1=every Sample, 2=every 2nd Sample, ...

Bit 4-7: Information size (currently 0)

Byte 43-44:

2 Byte spare for custom functions/future features (0x00 if not used)

Byte 45-46:

0xFB 0xFB (end of header)

After the header, the profile is send. It has a variable length:

Depth in mBar (LOW:HIGH), **ProfileFlagByte**, **extended Information**

...

2 Byte:

0xFD 0xFD (Profile and dive end)

ProfileFlagByte:

Bit0-7: amount of additional bytes after the ProfileFlagByte (0-127)

Bit8: **EventByte Flag** (Event Byte is available)

EventByte:

Bit0-3: Alarm type

0 =No Alarm

1=SLOW

2=Deco Stop missed

3=Deep Stop missed
4=ppO Low Warning
5=ppO High Warning
6=Manual Marker
7=Low Battery

...

Bit4: Manual Gas Set
Bit5: Gas Change
Bit6: unused
Bit7: unused

Extended Information will be stored strictly in the following order (future extensions will append at the end of the list for compatibility reasons!):

- EventByte (1 byte)
- ManualGasSet: new Values for Gas#6 (2 bytes: byte0=O2%, byte1=He%)
- Gas Change (1 byte: current Gas#)
- Temperature in 0.1°C (2 bytes: LOW:HIGH)
- Deco/NDL Status (2 bytes: byte0: =0 if within no deco limit, if in deco mode: depth of 1st stop; byte1: NDL in minutes or length of first stop in minutes)
- Tank pressure (according to header information)
- ppO2 Sensor Values (according to header information, t.b.d)
- "Not yet used 1" (according to header information, t.b.d)
- "Not yet used 2" (according to header information, t.b.d)

For calculation of the available the following Rules apply:

- **First sample# is 1**
- **if divisor == 0 the information will never appear**
- **dataAvailable = (sample# modulo divisor) == 0**

Sampling rate can vary! Always use sample rate from Dive Header.

Profile memory is a „first-in first-out” memory, so older dives will be overwritten when there is no free space left. This can result in „half” or broken dives in the memory. A complete dive always has a complete Header (0xFA 0xFA to 0xFB 0xFB) and a complete profile block (ending with 0xFD 0xFD)

The latest dive has an additional 0xFE after the profile block. This byte is required to find the latest dive after a firmware update, etc.

Examples:

Assume the following configuration:

Divisor Temperature=3
Divisor Deco/NDL=1
Divisor Tank Pressure=0
Divisor ppO2=0
Divisor Unused1+2=0.

Sample 1:

0x64 0x00 0x02 0x00 0xA0

Depth = 1m

ProfileFlagByte = 2 bytes will follow, No Event

Deco/NDL = NDL 160 minutes

Sample 2:

0xC8 0x00 0x02 0x00 0xA0

Depth = 2m

ProfileFlagByte = 2 bytes will follow, No Event

Deco/NDL = NDL 160 minutes

Sample 3:

0x2C 0x01 0x04 0x32 0x00 0x00 0xA0

Depth = 3m

ProfileFlagByte = 4 bytes will follow, No Event

Temperature = 5°C

Deco/NDL = NDL 160 minutes

Sample 4:

0x5E 0x01 0x84 0x21 0x02 0x00 0x9f

Depth = 3.5m

ProfileFlagByte = 6 bytes will follow, Event available (Bit7=1)

EventByte = Alarm SLOW, Gas Change

GasChange = Gas#2

Deco/NDL = NDL 159 minutes

...

command 0x62 („b“): Sync clock and date with PC

Output OSTC: 0x62

Now, 6 bytes must be send to the OSTC

Byte1: Hours

Byte2: Minutes

Byte3: Seconds

Byte4: Month

Byte5: Day

Byte6: Year (20xx)

All bytes will be checked for plausibility and the clock is set at once. After a timeout of about 30ms, the transmission will be aborted

command 0x63 („c“): Starts simulator mode

This command sets the OSTC into simulator mode. Every byte following this command will set

ambient_pressure to [byte]*100mBar and the OSTC will operate as if the unit is actually in use

underwater

e.g.: 0x1E (dec 30) simulates a pressure of $30 \times 100\text{mBar} = 3\text{Bar} = 20\text{m}$ depth

Simulator mode will be aborted if the device leaves the divemode after the simulated dive.

(ascend

to 0m by sending 0x00 and wait)

command 0x64 (.,d“): Access to internal EEPROM BANK 0

This command allows write access to the internal EEPROM from address 0x004 to 0xFF.

Serial

number and total dives can not be written.

Output OSTC: 0x64

Now, every incoming byte will be written to the internal EEPROM starting with address 0x004. Each byte will be echoed if the write was completed.

Important: Saving one byte takes about 4ms, so the next byte must be send AFTER the echo of the current byte was received from the software.

A total of 251Byte can be send. After a timeout of about 30ms, the current byte transmission will be aborted and 0xFF is send instead of the received value

command 0x65 (.,e“): Send MD2 Hash fingerprint to PC

Byte1-2:

2 Byte firmware version , 1st Byte.2nd Byte (e.g. „0.19“)

Byte3-18:

16 Byte MD2 hash fingerprint (0x00 for all bytes if hash was not built yet)

command 0x66 (.,f“): Adjust temperature calibration

Output OSTC: 0x66

Now, the next two byte (LOW:HIGH) must be the actual/current temperature in $0.1\hat{\text{A}}^{\circ}\text{C}$ steps.

The OSTC will compensate the sensor reading to adjust the calibration. After a timeout of about

30ms, the transmission will be aborted and the compensation remains unchanged.

command 0x67 (.,g“): Send internal EEPROM BANK 0 only

This command can be used to read out the gas setup and custom functions without reading the profile data (faster download)

Byte1-2:

Serial number

Byte3-4:

Total dives

Byte5-6:

Gas 1 default (%O₂=21, %He=0)

Byte7-8:

Gas 1 current (%O₂, %He)

Byte9-10:

Gas 2 default (%O₂=21, %He=0)

Byte11-12:

Gas 2 current (%O2, %He)

Byte13-14:

Gas 3 default (%O2=21, %He=0)

Byte15-16:

Gas 3 current(%O2, %He)

Byte17-18:

Gas 4 default (%O2=21, %He=0)

Byte19-20:

Gas 4 current (%O2, %He)

Byte21-22:

Gas 5 default (%O2=21, %He=0)

Byte23-24:

Gas 5 current (%O2, %He)

Byte25-26:

Gas 6 current (%O2, %He)

Byte27

not used/reserved

Byte28

Active Gas Flag Register

Bit0=Gas1

Bit1=Gas2

Bit2=Gas3

Bit3=Gas4

Bit4=Gas5

Bit5=not used/reserved

Bit6: not used/reserved

Bit7: not used/reserved

Byte29:

Gas1: Change depth in m

Byte30:

Gas2: Change depth in m

Byte31:

Gas3: Change depth in m

Byte32:

Gas4: Change depth in m

Byte33:

Gas5: Change depth in m

Byte34:

First Gas number (1-5)

Byte35:

DecoType (0=ZH-L16, 1=Gauge, 2=const. ppO2, 3=Apnoe 4=Multi Gradient-Factors)

Byte36:

Use O2 Sensor Module in CC Modes (0= OFF, 1= ON)

Byte37:

SetPoint1: ppO2 (x 0.01Bar, e.g. 155 = 1.55Bar ppO2)

Byte38:
SetPoint2: ppO2 (x 0.01Bar, e.g. 155 = 1.55Bar ppO2)
Byte39:
SetPoint3: ppO2 (x 0.01Bar, e.g. 155 = 1.55Bar ppO2)
Byte40:
Debugmode (0= OFF, 1= ON)
Byte41-42:
Lowest Battery voltage seen (in mV)
Byte43:
Lowest Battery voltage seen at (Month)
Byte44:
Lowest Battery voltage seen at (Day)
Byte45:
Lowest Battery voltage seen at (Year)
Byte46-47:
Lowest Battery voltage seen at (Temperature in 0.1 °C)
Byte48:
Last complete charge at (Month)
Byte49:
Last complete charge at (Day)
Byte50:
Last complete charge at (Year)
Byte51-52:
Total charge cycles
Byte53-54:
Total complete charge cycles
Byte55-56:
Temperature Extrema minimum (Temperature in 0.1 °C)
Byte57:
Temperature Extrema minimum at (Month)
Byte58:
Temperature Extrema minimum at (Day)
Byte59:
Temperature Extrema minimum at (Year)
Byte60-61:
Temperature Extrema maximum (Temperature in 0.1 °C)
Byte62:
Temperature Extrema maximum at (Month)
Byte63:
Temperature Extrema maximum at (Day)
Byte64:
Temperature Extrema maximum at (Year)
Byte65:
Custom Text active (=1), Custom Text Disabled (<>1)
Byte66-90:

(25Bytes): Custom Text for Surfacemode (Real text must end with "{")
Example: OSTC Dive Computer} (19 Characters incl. "{") Bytes 85-90 will be ignored.

Byte91:

Dim OLED in Divemode (>0), Normal mode (=0)

Byte92-128

not used/reserved

Byte129-256:

32 custom Functions (CF0-CF31)

The custom functions are stored in the internal EEPROM after 0x080

any custom function occupies 4 byte:

2 byte (low:high) store the default value, reset from menu "reset"

if bit16=1 then the custom function is 15bit value, if not it's a 8bit value

2 byte (low:high) store the actual value

CF#0.....

..

....CF#31:

command 0x68 (,h“): Reset DecoData (e.g. after a Simulated dive)

Output OSTC: 0x68 (command echo)

command 0x69 (,i“): Access to internal EEPROM BANK 1

This command allows write access to the internal EEPROM from address 0x104 to 0x1FF.

The first four bytes of EEPROM Bank1 are not available

Output OSTC: 0x69

Now, every incoming byte will be written to the internal EEPROM starting with address 0x104. Each byte will be echoed if the write was completed.

Important: Saving one byte takes about 4ms, so the next byte must be send AFTER the echo of the current byte was received from the software.

A total of 251Byte can be send. After a timeout of about 30ms, the current byte transmission will be aborted and 0xFF is send instead of the received value

command 0x6A (,j“): Send internal EEPROM BANK 1

This command is required to read out the BANK1 of the EEPROM

Byte1:

Temperature compensation value (Not writable!)

Byte2-3:

Last Firmware installed, 1st Byte.2nd Byte (e.g. „0.19“) (Not writable!)

Byte4:

Not used (Not writable!)

Byte5-128

not used/reserved

Byte129-256:

32 custom Functions (CF 32-63)

The custom functions are stored in the internal EEPROM after 0x180

any custom function occupies 4 byte:

2 byte (low:high) store the default value, reset from menu "reset"

if bit16=1 then the custom function is 15bit value, if not it's a 8bit value

2 byte (low:high) store the actual value

CF#32.....

..

....CF#63: