
OBP60 V2 Dokumentation

Norbert Walter

08.07.2025

Einführung

1 Historie	3
2 Technische Daten	5
2.1 Funktionen	5
2.2 Aufbau	6
2.3 Spezifikation	7
2.4 Anschlussbelegung	8
2.5 Schaltplan	8
2.6 Maßbilder	9
2.7 Nutzbare und konvertierbare Telegramme	9
2.8 Nutzbare I2C-Sensorik	9
2.9 Nutzbare 1Wire-Sensorik	9
3 Funktionsweise	11
3.1 Bussysteme	11
3.2 Gateway	12
3.3 WiFi-Verbindung	12
3.4 Konfiguration	12
3.5 Anzeige und Bedienung	12
4 Bedienelemente	15
4.1 Statuszeile	17
4.2 Anzeigebereich	17
4.3 Fußzeile	18
4.4 Sensor-Tasten	18
4.5 Flash LED	19
4.6 Hintergrundbeleuchtung	19
4.7 Buzzer	20
4.8 Reset-Taster	20
5 Inbetriebnahme	21
5.1 Schutzkonzept	21
5.2 Stromversorgung 12V/24V	23
5.3 Stromversorgung USB-C	25
5.4 Einbau	25
6 Konfiguration	27

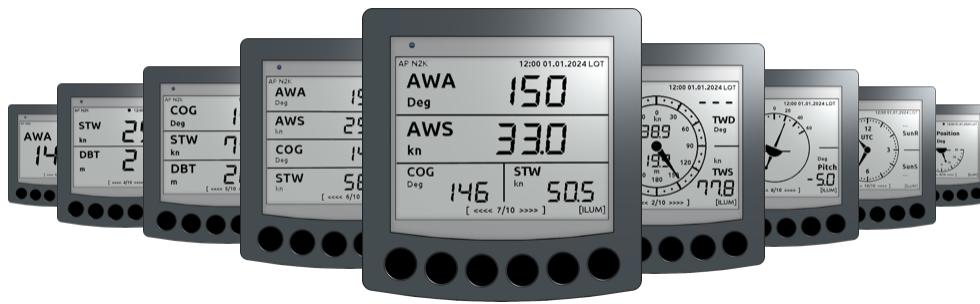
6.1	Status	30
6.2	Config	31
6.3	Config - System	33
6.4	Config - Converter	35
6.5	Config - USB Port	35
6.6	Config - Serial Port	37
6.7	Config - TCP Server	38
6.8	Config - TCP Client	40
6.9	Config - WiFi Client	42
6.10	Config - OBP Settings	43
6.11	Config - OBP Units	44
6.12	Config - OBP Hardware	45
6.13	Config - OBP Calibrations	50
6.14	Config - OBP Display	51
6.15	Config - OBP Buzzer	54
6.16	Config - OBP Pages	55
6.17	Config - OBP Page X	55
6.17.1	OneValue	58
6.17.2	TwoValue	59
6.17.3	ThreeValue	60
6.17.4	FourValue	61
6.17.5	FourValue2	62
6.17.6	Voltage	63
6.17.7	WindRose	64
6.17.8	WindRoseFlex	65
6.17.9	DST810	66
6.17.10	Clock	67
6.17.11	WhitePage	68
6.17.12	BME280	69
6.17.13	Rudder	71
6.17.14	Keel	72
6.17.15	Battery	73
6.17.16	Battery2	76
6.17.17	RollPitch	79
6.17.18	Solar	79
6.17.19	Generator	81
6.18	XDR	83
6.19	Data	88
6.20	Update	89
6.21	Help	90
7	Bussysteme	91
7.1	NMEA2000	92
7.2	NMEA0183	99
7.3	I2C	104
7.4	1Wire	106
7.5	USB	108
8	Datenaustausch	111
8.1	Interne Datenverarbeitung	111
8.2	Übertragungsarten	113
8.3	Übertragungswwege	113
8.4	Datenquellen	113
8.5	Datensenken	114

8.6	NMEA2000 - Kabelgebunden	114
8.7	NMEA2000 - WiFi via SeaSmart	115
8.8	NMEA0183 - Kabelgebunden	117
8.9	NMEA0183 - USB	118
8.10	NMEA0183 - WLAN	119
8.11	Konfigurationsbeispiele	120
8.11.1	Beispiel AVnav auf Raspberry Pi	120
8.11.2	Beispiel AvNav auf Android-Autoradio	125
8.11.3	Beispiel SignalK auf Raspberry Pi	129
8.11.4	Beispiel OpenPlotter auf Raspberry Pi	130
8.11.5	Beispiel Navionics auf Android-Autoradio	130
8.12	I2C-Bus	133
8.12.1	Beispiel I2C Ruderlagensor	133
8.13	1Wire-Bus	134
8.13.1	Konfigurationsbeispiel 1Wire	135
9	Erweiterte Sensorik	137
10	Beispielkonfiguration	141
10.1	OBP60 Yachta AvNav	141
10.1.1	LTE-Router einrichten	142
10.1.2	Datenübertragung	144
10.1.3	Konfiguration OBP60	145
10.1.4	Konfiguration Yachta	146
10.1.5	Tablett Konfiguration	147
11	Sicherheitshinweise	151
11.1	Allgemeine Hinweise	151
11.2	Sicherheit im WiFi-Netzwerk	152
12	Geräteaufbau	153
12.1	Mechanischer Aufbau	153
12.2	Main Board	155
12.3	Platinen	157
12.4	Schaltplan und Fertigungsdaten	159
12.5	Schaltungsbeschreibung	159
13	Vorbereitung	161
14	Bauteilliste	163
15	Durchführung	165
16	Funktionstests	167
17	Programmierumgebung	169
18	Seitenerstellung	171
19	Kompilieren und Download	173
19.1	Gitpod	173
19.1.1	Github-Projekt forken	174
19.1.2	Gitpod-Link öffnen	175
19.1.3	Gitpod-Settings	175
19.1.4	Container-Start	176
19.1.5	Codeänderung und Kompilieren	176

19.1.6	Binary-Download	178
19.1.7	Source-Code Aktualisierung	179
19.2	Visual Studio Code / PlatformIO	179
19.2.1	Github-Projekt forken	180
19.2.2	Projekt in VSC importieren	181
19.2.3	Codeänderung und Kompilieren	184
19.2.4	Flashen der Firmware	185
19.2.5	Debuggen der Firmware	186
19.2.6	Source-Code Aktualisierung	187
20	Web-Flashtool	191
21	Fragen und Antworten	197
22	Meinungen und Tipps	199
23	Bekannte Fehler	201
23.1	Kritische Fehler	202
23.2	Software-Fehler	202
23.3	Hardware-Fehler	202
23.4	Behobene Fehler	202
23.5	Software-Erweiterungen	202
23.6	Neue Anzeigeseiten	202
24	Unterstützung	203
25	Service	205
26	Mitarbeit	207
27	Spenden	209
28	Glossar	211

Letzte Aktualisierung 08.07.2025

Bemerkung: Diese Seiten sind noch in Bearbeitung.



Das Multifunktionsdisplay OBP60 dient zur Anzeige von Daten aus Boots-Netzen wie NMEA0183, NMEA2000. Darüber hinaus können Daten aus dem 1Wire-Bus und dem I2C-Bus ausgelesen werden. Das OBP60 enthält ein Gateway, das Daten bidirektional zwischen NMEA0183 und NMEA2000 austauschen kann. Über WiFi lassen sich die Daten importieren oder exportieren und können auf anderen Geräten oder in Drittanbietersoftware verwendet werden. Die Konfiguration erfolgt über eine Webseite. Darüber lassen sich vordefinierte Anzeigeseiten auswählen, und es können dessen Anzeigehalte festgelegt werden. Das OBP60 basiert auf Open Hardware, Open Software und Open Data. Das Gerät lässt sich durch die Offenheit an beliebige Anforderungen über die Software anpassen.

KAPITEL 1

Historie

Das Multifunktionsdisplay OBP60 V2 gibt es jetzt in einer zweiten, neueren Version. Der Grund für die Überarbeitung des voll funktionsfähigen ersten OBP60 liegt in der grundlegenden Änderung des Konzepts. Das ursprüngliche OBP60 basierte auf Erweiterungsmodulen wie einem ESP32, einem E-Paper Display, einem GPS-Modul und Umgebungssensorik. Diese Module mussten mittels durchsteckbarer Stiftleisten auf der Basisplatine eingelötet werden. Im Summe waren das ca. 60 Lötpunkte, die manuell verlötet werden mussten. Die Module sind zum Teil nur einseitig mit Stiftleisten versehen und müssen beim Löten abgestützt werden. Auch das Display muss vorher eingebaut sein, um es verlöten zu können. Für ein Hobbyprojekt mit manueller Bestückung mag das konzeptionell stimmig sein. Für eine Serienfertigung in größeren Stückzahlen ist der alte Ansatz aber sehr zeit- und kostenintensiv und nur mit erheblichem Aufwand realisierbar. Zudem sind die Module nicht in ausreichend gleichbleibender Qualität zu beziehen, und man muss mit großen Qualitätsschwankungen rechnen. Insbesondere beim GPS- und beim BME280-Modul gab es in der Vergangenheit große Probleme mit defekten Modulen. Jedes Modul musste daher vor dem Einbau manuell geprüft werden. Bei Anfragen für eine Fertigung in China ließen sich die Module nicht über große Bauteildistributoren besorgen, die Logistik wäre demzufolge sehr aufwändig geworden. Insgesamt hätte es auch keinen Sinn gemacht, die Platine ohne Module zu verkaufen, da ein Funktionstest der Platine so nicht möglich gewesen wäre.

Zusätzlich hatte sich herausgestellt, dass auch einige Schaltungsteile nicht richtig funktioniert haben. Die Backup-Versorgung über den GoldCap für den Ausschaltvorgang war zu klein dimensioniert. Die Energiemenge reichte nicht aus, um das E-Paper Display beim Ausschalten zu löschen und in einen definierten Zustand zu versetzen. Das wiederum hatte auch Folgeprobleme beim Einschalten des E-Paper Displays. Auch die 3,3V Stromversorgung für das BME280-Modul reichte nicht aus, um das Modul sicher betreiben zu können. Nach einiger Zeit im laufenden Betrieb erhöhte sich die Eigenerwärmung, das Modul fiel aus und kommunizierte nur noch fehlerhaft mit der CPU. Durch die umfangreiche Firmware gab es auch ein Problem mit dem Arbeitsspeicher. Er war in einigen Situationen zu knapp bemessen und verursachte dadurch ungewollte Reboots des OBP60. Am I2C-Bus gab es ebenfalls gelegentlich Probleme, wenn externe Komponenten an den Bus angeschlossen wurden. Im realen Einsatz konnten externe Störungen anderer Baugruppen über den I2C-Bus rückwärts in das OBP60 einkoppeln. Da der selbe I2C-Bus auch andere wichtige Komponenten des OBP60 versorgt, hatte das schwerwiegende Folgen für die Gesamtstabilität des Systems. Mit galvanisch getrennten Bussystemen kann man solche Probleme umgehen. Beim verwendeten GPS-Modul stellte sich heraus, dass die Firmware des Moduls Fehler enthielt. Das Modul sendete in ganz bestimmten Situationen bei schlechtem GPS-Empfang verstümmelte NMEA0183-Telegramme, die aber korrekte Checksummen enthielten. In solchen Situationen konnte das OBP60 komplett einfrieren, da der Parser nicht mit den fehlerhaften Daten umgehen konnte. Inzwischen wurde die Firmware des OBP60 angepasst und der Fehler konnte behoben werden.

Verständlicherweise war es daher nicht sinnvoll, den OBP60 in der alten Form in größeren Stückzahlen zu produzieren.

Das hätte, wie beschrieben, zu viele Probleme verursacht, und man hätte es logistisch und vom Zeitaufwand her nicht umsetzen können.

Aus diesen Gründen wurde die Platine komplett neu entwickelt. Ziel war es, die Platinen so zu gestalten, dass sie überwiegend mit SMD-Bauelementen vollautomatisch bestückt werden können und nur sehr wenige durchsteckbare Bauelemente enthalten. So ist es möglich, dass der Platinenhersteller die Platinen vollständig automatisiert bestücken kann, ohne dass danach noch manuell gelötet werden muss. Damit lassen sich jetzt beliebige Stückzahlen ohne zusätzlichen Aufwand fertigen. Allein die Programmierung und der Test sind dann noch manuelle Abläufe, die man aber auch noch zum Fertiger auslagern kann, wenn man dem Fertiger das zugehörige Test-Equipment zur Verfügung stellt.

Bei der Neuentwicklung wurden auch einige Funktionen weggelassen, die sich als überflüssig heraus gestellt haben. Der SeaTalk-Bus und die Porterweiterung wurden nur ganz selten eingesetzt. Beim BME280 macht es keinen Sinn, die Luftfeuchtigkeit im Inneren des Gerätes zu messen. Zwischenzeitlich gab es massive Beschaffungsprobleme des Bosch-Sensors und der Preis war in eine inakzeptable Höhe geschossen. Der BME280 wurde deshalb durch den BMP280 ersetzt, der nur noch Luftdruck und Temperatur misst. Zudem lässt er sich auch wesentlich besser beschaffen und ist deutlich günstiger. Das GPS-Modul wurde komplett entfernt, alle mit dem GPS-Empfang zusammenhängenden Bauteile werden nun direkt auf der Platine platziert. Dafür werden jetzt qualitativ hochwertige GPS-Chips chinesischer Bauart verwendet, die gut beschaffbar und günstig sind. Bei den Bussystemen wurden der I2C-Bus, der NMEA2000-Bus und der NMEA0183-Bus galvanisch isoliert ausgeführt, um die Einkopplung störender Einflüsse zu reduzieren. Alle drei Busse sind gegen Überspannungen, Verpolung und gegen Überstrom gesichert. Es gibt jetzt auch spezielle Anschlüsse, auf die die Leitungsschirmung aufgelegt werden kann. Diese sind gegen Überspannungen bis 8kV gesichert. Ein isolierender DC/DC-Wandler stellt für den Nutzer eine 5V-Versorgung mit 200 mA bereit, mit der externe Schaltungen potenzialfrei versorgt werden können. Damit lassen sich z.B. mehrere Batterie-Monitore gleichzeitig betreiben oder andere Sensoren einfach anbinden.

Der Versorgungsspannungseingang wurde ebenfalls gegen Überspannung, Verpolung und gegen Überstrom gesichert. Damit hat man eine ausreichend gute Absicherung auch bei Blitzschlag. Insgesamt sollte die gesamte Schaltung CE-gerecht umgesetzt sein. Das muss aber noch in einem gesonderten Test nachgewiesen werden. Beim 1Wire-Bus wurde auf eine galvanische Isolation verzichtet, da der Bus selbstversorgt ist und keine externe Einspeisung oder Verbindung zu anderen elektrischen Schaltungsteilen benötigt. Mit entsprechender Kabelschirmung ist der 1Wire-Bus ausreichend gesichert.

Die Hintergrundbeleuchtung und die Flash-LED sind jetzt als RGB-LEDs ausgeführt. Damit können die Leuchtfarbe und die Helligkeit für jede LED separat eingestellt werden.

Das E-Paper-Display wurde direkt mit der Hauptplatine verbunden. Dabei schützt eine Trägerplatte das empfindliche Glassubstrat des E-Paper-Displays. Insgesamt ist dadurch der Aufbau kompakter geworden, und die Platine und das Display bilden eine Einheit. Durch die geringere Bautiefe können die Abmaße der originalen Raymarine-ST60-Instrumente wieder eingehalten werden.

Die Programmierung des OBP60 gestaltet sich jetzt auch einfacher. Auf der Platine gibt es einen USB-C-Steckverbinder, über den das Board programmiert werden kann. Durch die Verwendung des neuen ESP32 S3 entfällt auch der USB-Seriell-Umsetzer. Er ist bereits im ESP32 S3 enthalten. Der USB-C-Anschluss ist direkt über die Rückseite des OBP60 erreichbar. Darauf können auch NMEA0183-Daten biddirektional übertragen werden. So lassen sich die Daten zum Beispiel zwischen einem PC oder einem Raspberry Pi austauschen. Sofern der USB-Host, zum Beispiel ein Steckernetzteil, 5.1V mit 500mA liefert, kann das OBP60 auch damit betrieben werden. Der nun im OBP60 verbaute ESP32 S3 ist aktuell der leistungsfähigste ESP32, den man bekommen kann. Neben einem größeren RAM verfügt der ESP32 S3 auch über PSRAM, womit sich der Arbeitsspeicher noch deutlich erhöhen lässt.

KAPITEL 2

Technische Daten

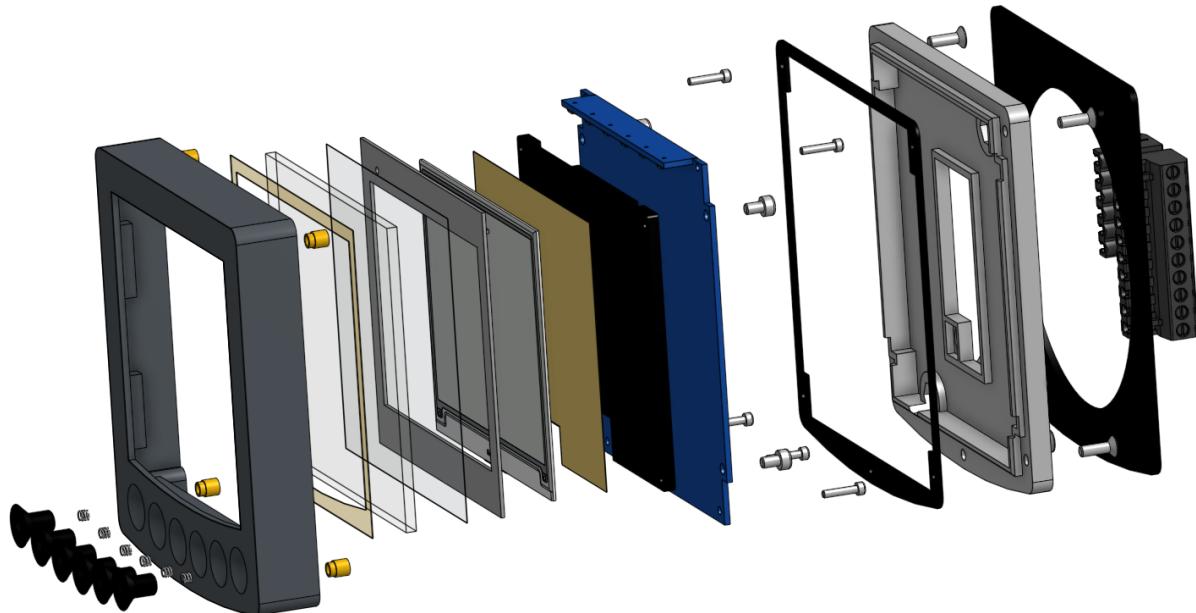


2.1 Funktionen

- E-Paper Display (tageslichttauglich)
- Numerische Anzeigeseiten für 1, 2, 3 und 4 Werte (Werte frei wählbar)
- Grafische Anzeigeseiten (feste Werte)
- Trendanzeige für Werte
- Grenzwertüberschreitung akustisch und optisch
- 6 Sensor-Tasten (geeignet für Wischgesten)
- Akustischer Signalgeber (Buzzer)

- Optischer Signalgeber (RGB-LED)
- LED-Displaybeleuchtung (RGB-LEDs)
- NMEA2000 (vollduplex, isoliert)
- NMEA0183 (RX oder TX, konfigurierbar, isoliert)
- NMEA2000/NMEA0183 Gateway (bidirektional)
- I2C (isoliert)
- 1Wire (nicht isoliert)
- Spannungsausgang 5V (max. 200mA) für externe Sensorik
- USB-C (OTG, Debug, NMEA0183)
- Batteriemonitor (12V-Spannungsmessung)
- Umgebungssensor BMP280 (Temperatur, Luftdruck)
- GPS-Empfänger (GPS, Glonass, Baidu, interne oder externe GPS-Antenne)
- WiFi 2.4GHz (HTTP, TCP, UDP)
- Bluetooth (aktuell ungenutzt)
- Batterie-Tiefentladeschutz < 9.0V
- Low Power Modus

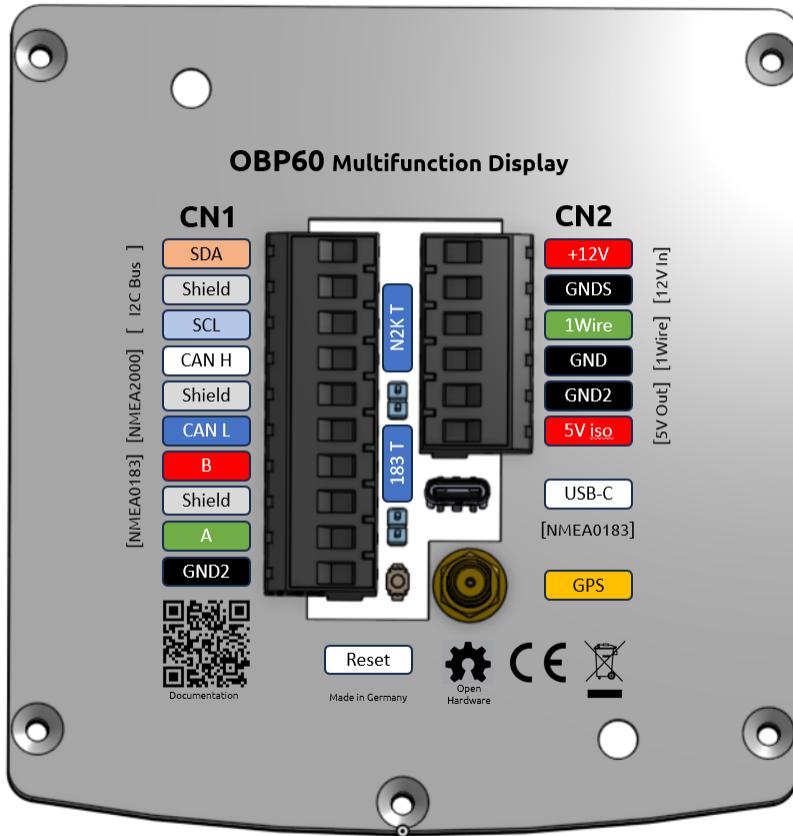
2.2 Aufbau



2.3 Spezifikation

Versorgungsspannung	10...28 V
Stromverbrauch	0.5...3.5 W, typisch 1 W
Prozessor	ESP32-S3, Dual Core
Clock Speed	80, 160, 240 MHz
RAM	512 kB
Flash	16 MB
PSRAM	8 MB
Displaygröße	400 x 300 pix, 120 dpi
Refreshrate	1 Hz
Sensortasten	kapazitiv
NMEA0183-Bus	RS485, max. 115.2 kBd, 30 m
NMEA2000-Bus	CAN, 250 kBit/s, 30 m
I2C-Bus	5V, 100 kBit/s, 10 m
1Wire-Bus	3.3V, 10 m
5V-Ausgang	200 mA, isoliert
ESD-Schutz	8 kV
Schutzgrad	IP68, frontseitig
Abmessungen	110 x 115 x 30 mm
Gewicht	280 g

2.4 Anschlussbelegung



2.5 Schaltplan

- Schaltplan V2.1 [PDF]

2.6 Maßbilder

- Maßbild [PDF]

2.7 Nutzbare und konvertierbare Telegramme

NMEA0183

- AIVDM, AIVDO, DBK, DBS, DBT, DPT, GGA, GLL, GSA, GSV, HDG, HDM, HDT, MTW, MWD, MWV, RMB, RMC, ROT, RSA, VHW, VTG, VWR, XDR, XTE, ZDA

NMEA2000

- 126992, 127245, 127250, 127251, 127257, 127258, 127488, 127489, 127505, 127508, 128259, 128267, 128275, 129025, 129026, 129029, 129033, 129038, 129039, 129283, 129284, 129539, 129540, 129794, 129809, 129810, 130306, 130310, 130311, 130312, 130313, 130314, 130316

2.8 Nutzbare I2C-Sensorik

Umgebungssensoren

- BMP085, BMP180, BMP280, BME280, SHT20, HTU21

Spannungs- und Stromsensoren

- INA226, INA219 (in Vorbereitung)

Winkelsensoren

- AS5600, MT6701 (in Vorbereitung)

Port-Erweiterungen

- PCF8574 (in Vorbereitung)

Echtzeit-Uhren

- DS1388

2.9 Nutzbare 1Wire-Sensorik

Temperatursensoren

- DS18B20

KAPITEL 3

Funktionsweise

3.1 Bussysteme



Das OBP60 ist ein Datenanzeigegerät für den Marinebereich, mit dem verschiedenste Informationen aus Bussystemen ausgelesen und angezeigt werden. Zu den typischen Bussystemen im Marinebereich, die das OBP60 verarbeitet, gehören:

- NMEA0183 (RS485, isoliert)
- NMEA2000 (CAN, isoliert)

Damit lässt sich das OBP60 in vorhandene Bootsnetze integrieren. Es kann selbst Daten einspeisen und auslesen. Darüber hinaus verarbeitet das OPB60 weitere Sensornetzwerke wie:

- I2C-Bus (5V, isoliert)
- 1Wire-Bus (3.3V, nicht isoliert)

Diese beiden Bussysteme kommen aus dem Elektronikbereich. Über diese beiden Busse lassen sich günstige Sensoren anbinden, um deren Daten anzuzeigen. Über den isolierten I2C-Bus können zudem recht einfach und sicher eigene Hardwareerweiterungen eingebunden werden.

3.2 Gateway



Im OBP60 ist ein Gateway integriert, das Daten zwischen NMEA0183 und NMEA2000 bidirektional austauschen kann. Dabei werden die Daten des einen Busses in die Daten des anderen Busses übersetzt. Die Übersetzung funktioniert dabei in beide Richtungen.

Bemerkung: Dabei ist zu beachten, dass nicht alle NMEA2000-Daten in NMEA0183-Daten übersetzt werden können, weil dafür nicht immer geeignete Telegramme in NMEA0183 existieren.

3.3 WiFi-Verbindung

Das OBP60 verfügt über eine WiFi-Funktionalität im 2.4 GHz Funkbereich. Darauf hinaus kann das Gerät mit dem Internet oder mit anderen WiFi-Netzwerken verbunden werden und so mit anderen Geräten kommunizieren. So lassen sich z.B. Daten aus den Bussystemen an einen Laptop, einen PC oder ein Handy übertragen oder von dort beziehen. Damit ist es möglich, die Sensordaten auch in Drittanbieter-Software wie Navionics, NMEA-Remote oder WinGPS zu verwenden.

3.4 Konfiguration

Das OBP60 hat einen Access Point und einen kleinen Webserver integriert, mit denen das Gerät konfiguriert werden kann. Im Gegensatz zu anderen kommerziellen Geräten erfolgt die Konfiguration des OBP60 ausschließlich webbasiert. Dazu kann z.B. ein Handy benutzt werden. So ist die Konfiguration des Gerätes deutlich einfacher und komfortabler. Im Gerät lassen sich bis zu 10 Anzeigeseiten frei definieren. Der Anwender kann zwischen numerischen und grafischen Anzeigeseiten auswählen. Für jede numerische Anzeigeseite können beliebige Daten der Bussysteme angezeigt werden. Bei den grafischen Anzeigeseiten sind die Dateninhalte vorgegeben, da sie spezielle Funktionalitäten bieten.

3.5 Anzeige und Bedienung



Als Anzeige wird ein E-Paper Display verwendet. Es besitzt einen hohen Kontrast und eine gute Ablesbarkeit auch bei starkem Sonnenlicht. Zudem verbraucht es sehr wenig Energie. Beim Nachtbetrieb ist das Display beleuchtbar. Die Hintergrundfarbe kann frei gewählt werden. So lässt sich das OBP60 auch gut mit anderen Anzeigegeräten kombinieren, um ein einheitliches Aussehen zu erreichen. Eine kleine Flash-LED und ein Buzzer signalisieren dem Anwender optisch und akustisch Grenzwertüberschreitungen. Die Grenzwerte lassen sich frei einstellen.

Die Auswahl der Anzeigeseiten erfolgt über kapazitive Sensortasten, indem mit einer Wischgeste über die Tasten die nächste Seite angesteuert wird. Die Sensitivität der Tasten lässt sich einstellen. Gegen versehentliche Auslösung können die Tasten auch gesperrt werden. Je nach Anzeigeseite können einige Einstellungen auch über die Tasten vorgenommen werden. Die Einstellungen gelten dann ausschließlich für die Anzeigeseite und werden gespeichert, sodass die Einstellungen beim Seitenwechsel erhalten bleiben.

KAPITEL 4

Bedienelemente



4.1 Statuszeile

Die Statuszeile dient zur Anzeige von Statusinformationen. Dazu zählen:

Zustandsanzeige des Access Points

- AP - Access Point

Zustandsanzeige aktiver Bussysteme und Komponenten

- N2K - NMEA2000
- 183 - NMEA0183
- USB - NMEA0183 via USB
- TCP - NMEA0183 via TCP (WiFi)
- GPS - GPS-Sensor des OBP60 (Position gefunden)
- Lebenszeichen (pulsierender Punkt)
- Datum und Uhrzeit (landesspezifisch)
- Anzeige der Zeitzone
 - UTC - Weltzeit
 - LOT - Lokalzeit



Die Statuszeile ist in allen Anzeigeseiten zu sehen, sie zeigt den aktuellen Status des Gerätes an.

Bemerkung: Der Anzeigehinhalt eines E-Paper Displays bleibt beim Ausschalten des Gerätes erhalten. Der pulsierende Punkt in der Statuszeile kennzeichnet das Gerät als aktiv. Sollte der Punkt nicht blinken, so ist die Software inaktiv oder das Gerät wurde ausgeschaltet.

4.2 Anzegebereich

Im mittleren Bereich befindet sich der Anzegebereich. Dort werden alle relevanten Informationen angezeigt. Bei jedem Wechsel auf eine neue Seite wird der Inhalt des Anzegebereichs verändert. Die Aktualisierung des Anzegebereichs erfolgt jede Sekunde als partieller Bild-Refresh.

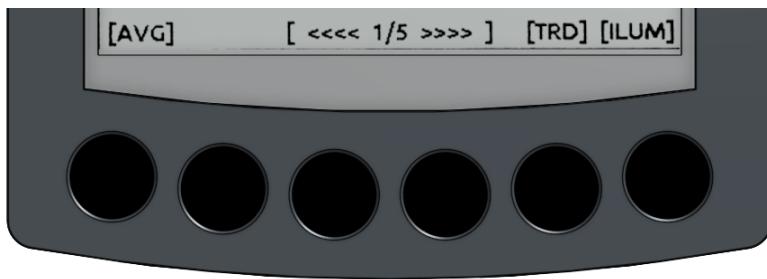
Bedingt durch die E-Paper Technologie sind im Display nach einiger Zeit Geisterbilder von alten Anzeigezuständen zu sehen. Um die Geisterbilder zu entfernen, wird in regelmäßigen Abständen von 10 min ein Voll-Refresh der Anzeige durchgeführt. Dabei wird der komplette Bildinhalt mehrmals invertiert, dann gelöscht und anschließend neu geschrieben. Man erkennt einen Voll-Refresh am kurzen Flackern der Anzeige. Das gleiche passiert 4 Sekunden nach einem Seitenwechsel. Dadurch kann man schnell mehrere Anzeigeseiten nacheinander aufrufen. Erst bei der zuletzt aufgerufenen Seite wird nach 4 Sekunden ein Voll-Refresh durchgeführt, damit werden Geisterbilder alter Anzeigeseiten entfernt. Der regelmäßige Voll-Refresh ist per Default eingestellt und kann bei Bedarf über die Konfiguration deaktiviert werden.

Die Entstehung von Geisterbildern ist von der Display-Temperatur des OBP60 abhängig. Bei niedrigen Temperaturen sind Geisterbilder deutlicher zu sehen und die Anzeige reagiert trüger als bei hohen Temperaturen. Kurz nach dem Einschalten wird für die ersten 5 Minuten jede Minute ein Voll-Refresh durchgeführt, damit sich das Display akklimatisieren kann. Die Plexiglasscheibe schützt vor zu großer UV-Strahlung der Sonne, ein IR-Filter vor übermäßiger Erwärmung.

Bemerkung: Trotz Filter kann es bei extrem hoher Sonneneinstrahlung vorkommen, dass der Kontrast des Display-Inhaltes verloren geht. Schwarze Anzeigebereiche werden dann nur noch grau dargestellt. Das Display ist in dem Fall nicht defekt. Nach einem Voll-Refresh regeneriert sich das Display und der Kontrast wird wieder vollständig hergestellt.

Wichtig: Wird das OBP60 nicht benutzt, löschen Sie bitte den Bildschirminhalt und decken das Gerät mit der Schutzkappe ab. So schützen Sie es vor zu großer Sonneneinstrahlung und vor Witterungseinflüssen.

4.3 Fußzeile



Die Fußzeile dient zur Bezeichnung der Tastenfunktionen. Die Belegung der Tasten ändert sich abhängig vom Inhalt der Anzeigeseiten. Aktive Tasten sind mit Kurzbezeichnungen in eckigen Klammern versehen wie z.B. [AVG]. Es kann auch Anzeigeseiten geben, die keine Tastenfunktionen enthalten. In der Mitte der Fußzeile werden weitere Informationen eingeblendet:

- [<<< 1/5 >>>] - Wischgeste aktiv
- [Keylock active] - Tasten gesperrt

Sofern die Wischgeste aktiv ist, werden im Infobereich die aktuelle Seite und die Seitenanzahl angezeigt.

4.4 Sensor-Tasten

Das OBP60 hat 6 kapazitive Sensortasten am unteren Displayrand. Die Tasten reagieren auf Berührung durch Veränderung des Bezugspotenzials. Wassertropfen oder Regen haben so gut wie keinen Einfluss auf die Auslösefunktion der Tasten. Aktuell sind folgende Tastenfunktionen realisiert:

- Konfiguration der Tasten-Sensitivität
- Wischen links / rechts
- Kurzes Tippen
- Normales Drücken
- Drücken mehrerer Tasten gleichzeitig

- Tasten sperren

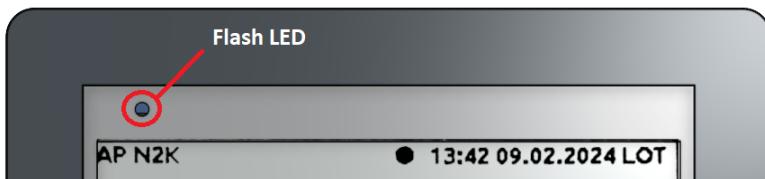
Die Tasten-Sensitivität kann über die Konfigurationsseite eingestellt werden. Damit lässt sich die Schwelle einstellen, ab der ein Tastendruck erkannt wird. Die Tasten haben in der Mitte eine Vertiefung. So kann das Zentrum der Taste besser erfüllt werden. Erkannte Tastenberührungen werden akustisch mit einem Piepton signalisiert.

Die Tasten sind bündig in das Display eingelassen. So ist es möglich, mit Wischgesten die Seiteninhalte umzuschalten. Dazu wischt man zügig nach rechts oder links über mindestens zwei Tasten. Die Software erkennt automatisch, wenn mehrere Tasten hintereinander ausgelöst werden, und bestimmt daraus die Wischrichtung. Für die Wischgeste wird ein akustisches Feedback gegeben. **Rechts wischen** wird mit der Tonfolge **tief-hoch** signalisiert, **links wischen** mit **hoch-tief**. Die Seitenweiterschaltung ist an den Enden rollierend.

Bei kurzem Tippen wird kein Piepton ausgegeben. Kurzes Tippen der zwei äußeren Tasten links und rechts nacheinander aktiviert die Tastensperre, gefolgt von einem langen Piepton. Danach sind die normalen Tastenfunktionen deaktiviert und es wird keine Wischgeste mehr erkannt. Im Display ist die Meldung [Keylock active] zu sehen. Welche Taste beim Aktivieren der Tastensperre zuerst gedrückt wird, ist unerheblich, ebenso die Reihenfolge. Die Deaktivierung der Tastensperre erfolgt auf die gleiche Weise.

Wichtig: Sollte Ihr OBP60 so platziert sein, dass sich eine Person versehentlich dagegen lehnen kann, so stellen Sie die Tasten-Sensitivität entsprechend niedriger ein. So vermeiden Sie versehentliches Auslösen der Tasten. Sie können als zusätzlichen Schutz gegen unbeabsichtigtes Bedienen auch die Tasten-Sperrfunktion aktivieren.

4.5 Flash LED



Links oberhalb des Displays befindet sich eine kleine Flash-LED. Diese LED dient zur Signalisierung von Betriebszuständen des OPB60. Die LED kann dabei verschiedene Farben annehmen. Die LED leuchtet mit maximaler Helligkeit, sodass sie optisch auch bei hellen Sonnenlicht gut wahrgenommen werden kann.

- Rot - Alarmierung bei Grenzwertüberschreitung
- Grün - Bestätigung von Zustandsänderungen (z.B. Autopilot ein/aus)
- Blau - Signalisierung von Zuständen (z.B. GPS-Empfang, Datentransfer usw.)

4.6 Hintergrundbeleuchtung

Um das Display bei Nacht ablesen zu können, kann eine Hintergrundbeleuchtung über die rechte Sensortaste zugeschaltet werden. Die Farbe und die Helligkeit können über die Konfiguration eingestellt werden. Grundsätzlich lässt sich die Hintergrundbeleuchtung folgendermaßen konfigurieren:

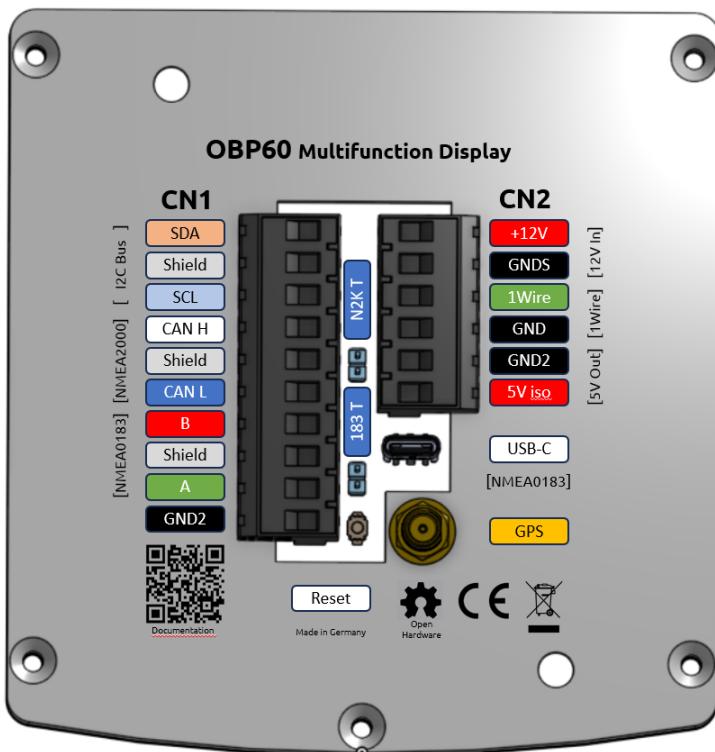
- Dauerhaft an
- Dauerhaft aus
- Manuell einschaltbar über Sensortaste
- Automatisch schaltbar abhängig vom Sonnenstand

Wichtig: Wenn Sie den Sonnenstand zum Schalten der Hintergrundbeleuchtung verwenden wollen, benötigen Sie ein gültiges GPS-Signal, damit die Schaltzustände ausgelöst werden können. Die Hintergrundbeleuchtung wird dann automatisch bei astronomischem Sonnenuntergang eingeschaltet und bei Sonnenaufgang abgeschaltet. Dabei wird der geografische Ort berücksichtigt. Ist kein gültiges GPS-Signal vorhanden, so erfolgt keine Änderung der Schaltzustände.

4.7 Buzzer

Der Buzzer dient zur akustischen Signalisierung von Störungen und als Feedback bei Zustandsänderungen. Der Buzzer befindet sich im Inneren des Gerätes. Die Funktion und Lautstärke des Buzzers kann in der Konfiguration eingestellt werden. Beim Einschalten und beim manuellen oder automatischen Reset des OBP60 erfolgt ein kurzer Signalton, um das Hochfahren des Gerätes zu signalisieren.

4.8 Reset-Taster



Der Reset-Taster befindet sich auf der Rückseite des Displays an der Unterseite des großen linken Steckverbinder **CN1**. Der Reset-Taster wird im normalen Betrieb nicht genutzt. Bei Programmievorgängen kann es nützlich sein, einen manuellen Reset auszulösen. Benutzen Sie zum Auslösen des Reset einen nicht leitenden, schmalen Gegenstand und drücken Sie den Taster vorsichtig, bis der Druckpunkt spürbar überwunden ist.

Warnung: Verwenden Sie keine leitenden Gegenstände. Damit können Kurzschlüsse auf der Platine ausgelöst werden, das Gerät kann dadurch beschädigt werden.

KAPITEL 5

Inbetriebnahme

Dieser Abschnitt hat die Erstinbetriebnahme Ihres OBP60 zum Inhalt. Das geht am einfachsten, wenn das Gerät noch nicht im Boot eingebaut ist. Dazu wird das Gerät mit Strom versorgt, um die ersten Einstellungen vorzunehmen. So können Sie die Funktionalität testen, bevor Sie das Gerät im Boot einbauen.

5.1 Schutzkonzept

Das OBP60 verfügt über ein mehrstufiges Schutzkonzept. Das soll verhindern, dass sich Störungen im Versorgungsnetz und auf den Datenleitungen im System ausbreiten. Damit wird sichergestellt, dass das OBP60 trotz Störungen nicht komplett ausfällt und in wichtigen Teilen weitestgehend funktionsfähig bleibt. Um dies zu erreichen, sind die Bussysteme:

- NMEA2000
- NMEA0182
- I2C

vom Bordnetz isoliert aufgebaut.

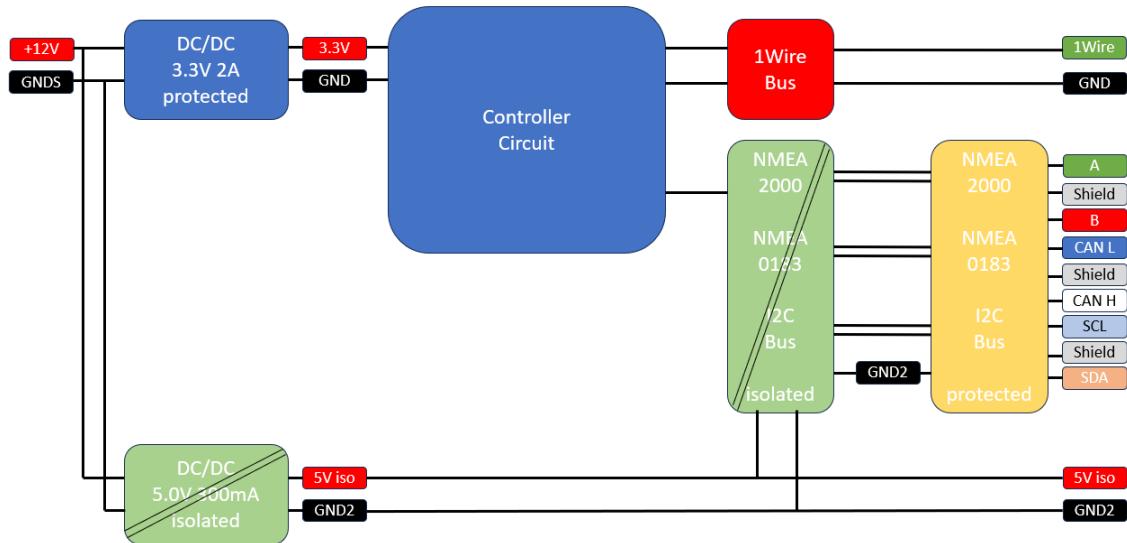


Abb.: Sicherheitskonzept

Im OBP60 ist dazu eine zusätzliche 5V-Stromversorgung enthalten, die die isolierten Schaltungsteile versorgt. Die zusätzliche Stromversorgung hat die Ausgänge +5Viso und GND2, die am Steckverbinder CN1 und CN2 anliegen. Über diese können auch externe Schaltungen bis 200 mA versorgt werden.

Warnung: Verbinden Sie die unterschiedlichen Massepotenziale GNDS, GND, GND2 und Shield niemals miteinander! Dadurch geht die Isolations- und Schutzwirkung verloren. Die Massepotenziale dürfen nicht gleichberechtigt verwendet werden.

Die unterschiedlichen Massepotenziale haben folgende Bedeutung:

- GNDS - Masse der Versorgungsspannung
- GND - Interne Masse der Elektronik
- GND2 - Masse der isolierten Bus-Elektronik
- Shield - Schutzleiter für die Kabelschirmung

Im folgenden Bild sind die geschützten und ungeschützten Anschlüsse zu sehen.

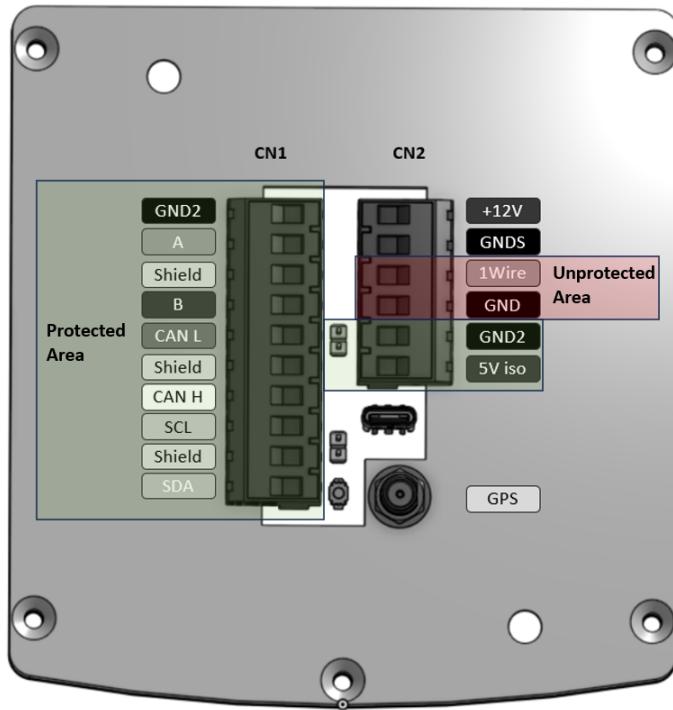


Abb.: Sichere und unsichere Bereiche

Warnung: Bitte beachten Sie, dass der 1Wire-Bus gegenüber Störungen nicht geschützt ist und entsprechend durch Schirmung geschützt werden muss. Daher sollten für die Leitungen des 1Wire-Bus nur abgeschirmte Kabel verwendet werden. GND darf mit keinen weiteren externen Schaltungsteilen verbunden werden, denn GND dient ausschließlich als Masseleitung für den 1Wire-Bus.

5.2 Stromversorgung 12V/24V

Die Stromversorgung des OBP60 erfolgt über den Steckverbinder CN2. Beim Zuschalten der Versorgungsspannung schaltet sich das OBP60 automatisch ein. Es gibt keinen gesonderten Ein- oder Ausschalter am Gerät. Benutzen Sie für die Stromversorgung die Anschlüsse +12V und GNDS. Dabei wird +12V mit dem positiven Pol der Batterie verbunden und GNDS mit dem negativen Pol. Diese Anschlüsse für die Stromversorgung sind:

- verpolungssicher
- kurzschlussfest
- überspannungssicher
- ESD-geschützt

Der zulässige Spannungsbereich liegt zwischen 10V...28V.

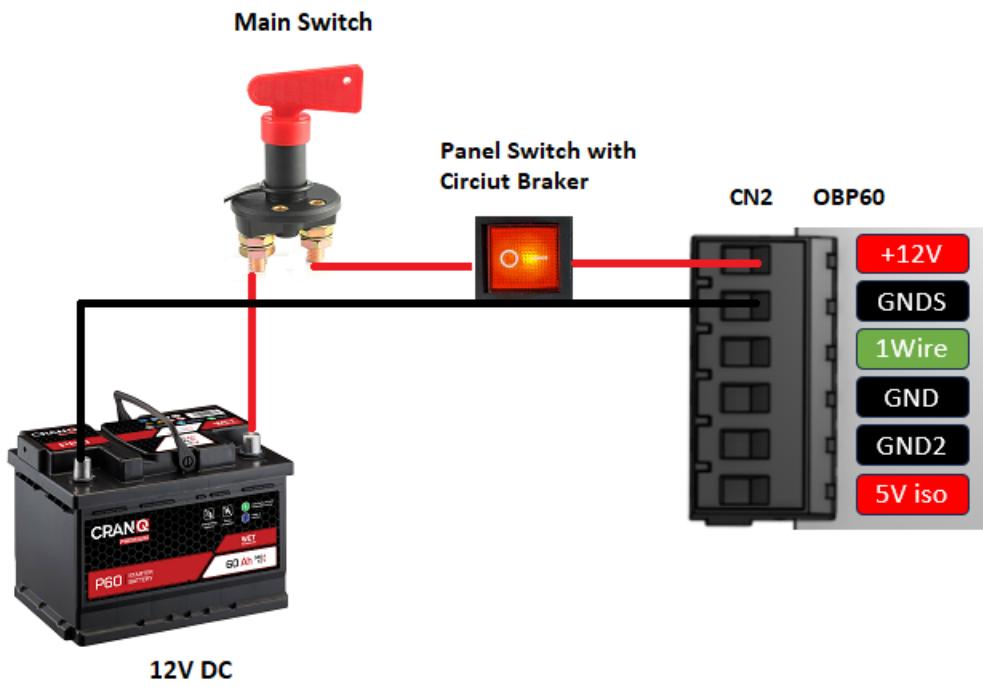


Abb.: Stromversorgung

Das OBP60 kann in 12V- und in 24V-Bord-Versorgungsnetzen verwendet werden. Bei Spannungen höher als 28V wird die interne Sicherung im Gerät ausgelöst.

Warnung: Verwenden Sie für die Verkabelung der Stromversorgung Kabelquerschnitte von 0,5...0,75 mm². Die Kabel sollten direkt an der Schalttafel aufgelegt sein und keine weiteren Abgänge für andere Geräte enthalten. Kabelüberlängen sollten vermieden werden. Wird der Strom an anderen Geräten abgezweigt oder sind die Kabelquerschnitte zu gering oder die Versorgungsleitungen zu lang, so kann es Probleme beim Einschalten des OBP60 geben.

In einigen Situationen ist es günstiger, die Stromversorgung direkt über den NMEA2000-Bus vorzunehmen. Dies setzt aber eine korrekte Installation des NMEA2000-Buses voraus. Dann entfällt die Verkabelung der Stromversorgung zu einer Schalttafel. Nähere Details dazu finden Sie im Kapitel [Bussysteme](#).

Bemerkung: Im Gerät ist eine selbst rückstellende Sicherung verbaut, die bei zu hohem Stromverbrauch die Versorgungsspannung selbstständig trennt. Sie können die Sicherung zurücksetzen, indem Sie die Stromversorgung zum OBP60 trennen und den Grund des übermäßigen Stromverbrauchs beseitigen. Danach warten Sie einige Minuten und schalten dann die Versorgungsspannung wieder ein.

Wichtig: Die interne Sicherung im OBP60 schützt nur das Gerät und nicht die Versorgungsleitungen! Daher sollte die bereitgestellte Stromversorgung des OBP60 im Bordnetz mit einer zusätzlichen Sicherung von mindestens 5A abgesichert werden. Das erfolgt typischerweise über die Schalttafel, über die die Stromkreise im Boot geschaltet werden können. So vermeiden Sie Brände zum Beispiel durch aufgescheuerte oder überhitzte Versorgungsleitungen.

5.3 Stromversorgung USB-C

Das OBP60 kann auch über USB-C mit Strom versorgt werden. Der USB-Anschluss muss aber ausreichend Spannung von 5.1V und Strom bis 500 mA liefern können. Der USB-Anschluss am OBP60 verfügt über einen Rücklaufschutz, so dass gleichzeitig 12V/24V und 5V über den USB-Port eingespeist werden können.

Bemerkung: Viele USB-Computeranschlüsse verfügen nicht über einen ausreichend hohen Ausgangstrom und teilen sich den Strom mit mehreren Anschlüssen. Das kann dazu führen, dass das OBP60 nicht direkt von einem PC aus mit Strom versorgt werden kann. Auch die Kabelqualität und Kabellänge ist entscheidend. Einige Kabel haben zu geringe Querschnitte und erzeugen einen hohen Spannungsabfall auf den Leitungen. Die Spannung ist dann am USB-C-Ausgang zu gering. Benutzen Sie in solchen Fällen den zusätzlichen 12V-Eingang an **CN2** zur Stromversorgung.

5.4 Einbau

Der Einbau des OBP60 erfolgt über die Rückseite. Zum Anzeichnen der Öffnung und der Löcher in der Cockpitwand kann die Moosgummidichtung als Schablone benutzt werden. Vor der Montage des OBP60 ist die Rückseite abzunehmen. Die 2 Sechskantschrauben werden in die dafür vorgesehenen Vertiefungen eingeführt und gegen Herausfallen gesichert. Prüfen Sie vor dem Einbau, ob die Länge der M5x25 Schrauben ausreichend ist. Gegebenenfalls ersetzen Sie die Schrauben in passiger Länge.

Warnung: Die Köpfe der Sechskantschrauben sollten in den Vertiefungen eingeklebt werden, damit sie beim Einbau nicht in das Gehäuseinnere geschoben werden und dort einen Kurzschluss auf der Platine verursachen können.

Danach wird das Gehäuse wieder vollständig zusammen gebaut.

Vor der Montage wird zwischen Rückseite und Cockpitwand eine 2 mm dicke Moosgummidichtung platziert. Die Moosgummidichtung kann Unebenheiten der Oberfläche in gewissen Grenzen ausgleichen. Mit den Unterlegscheiben und den Muttern befestigt man das OBP60.

Warnung: Ziehen Sie die Muttern nicht zu fest an. Die Verdrehssicherung oder die Rückwand können beschädigt werden.

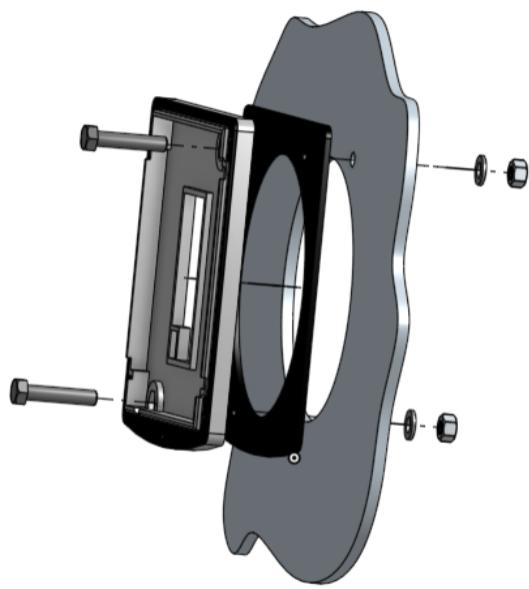


Abb.: Geräteeinbau in Cockpitwand

Hinweis: Die GPS-Antenne des GPS-Empfängers befindet sich links oben in der Ecke auf der Rückseite des OBP60 (von hinten gesehen). Hinter der Anbaufläche sollten sich keine großflächigen Metallteile befinden. Die Metallteile können den GPS-Empfang stören oder unmöglich machen. Wenn Sie Empfangsprobleme haben, können Sie eine externe GPS-Antenne benutzt.

KAPITEL 6

Konfiguration

Um das OBP60 konfigurieren zu können, muss das Gerät in Betrieb sein. Schalten Sie dazu die Stromversorgung ein, die Firmware des OBP60 startet nun. Nach Abschluss der Initialisierungsphase ertönt ein Piepton. Im Display wird zuerst das Open Boat Projects-Logo angezeigt, gefolgt von einem QR-Code, der die Zugangsdaten zum Access Point des OBP60 anzeigt. Beide Bilder sind für einige Sekunden sichtbar. Sie können mit Ihrer Handy-Kamera den QR-Code scannen und sich mit diesen Daten in das WiFi-Netz des OBP60 einloggen.



Abb.: Startbildschirm mit OBP-Logo



Abb.: QR-Code für WiFi-Zugang

Ab Android 10 öffnen Sie dazu die Wifi-Einstellungen und lassen sich alle WiFi-Netzwerke der Umgebung anzeigen. Am Ende der Liste finden Sie unter **Netzwerk hinzufügen** rechts ein kleines, blaues QR-Symbol. Wenn Sie das Symbol anklicken, öffnet sich ein Fenster zum Scannen des QR-Codes. Nach einem erfolgreichen Scan bucht sich das Gerät selbstständig in das WiFi-Netzwerk ein. Sie müssen keine Eingaben zur SSID oder zum Passwort vornehmen. Für ältere Android-Versionen gibt es Scanner-Apps, die eine ähnliche Funktionalität aufweisen.

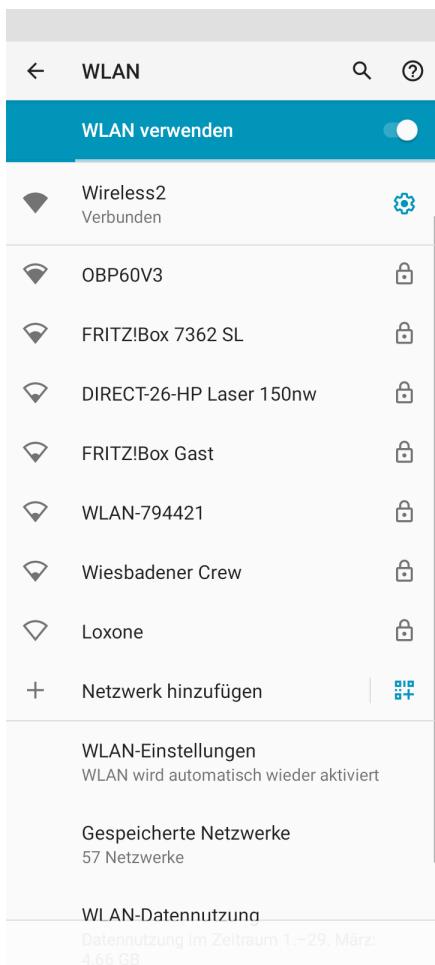


Abb.: WLAN Settings unter Android 11

Bemerkung: Sollte das Einbuchen in das WiFi-Netzwerk des OBP60 per QR-Code nicht funktionieren, können Sie die Konfiguration auch manuell vornehmen. Verwenden Sie dazu die folgenden Zugangsdaten:

- **SSID:** OBP60V2
- **Passwort:** esp32nmea2k

Nachdem Ihr Endgerät im WiFi-Netzwerk eingebucht ist, öffnen Sie in einem Web-Browser die Adresse **OBP60V2.local** oder die IP-Adresse **192.168.15.1**. Sie gelangen so auf die Benutzeroberfläche des OBP60 und können den aktuellen Status des Geräts überprüfen. Auf der Benutzeroberfläche befinden sich Tabs, mit denen verschiedene Bereiche der Konfiguration ausgewählt werden können:

- **Status** - Statusanzeige mit Übersicht der Bussysteme
- **Config** - Allgemeine Konfigurationsseite
- **XDR** - Konfigurationsseite für NMEA0183-XDR-Sentences
- **Data** - Dashboard zur Datenanzeige
- **Update** - Seite zum Firmware-Update
- **Help** - Aufruf der Github-Projektseite

Bemerkung: Beachten Sie, dass beim erstmaligen Aufruf der Konfigurationsoberfläche kein Passwort beim Speichern der Konfiguration notwendig ist. Als **Default-Passwort** wird **esp32admin** verwendet. Sie können auch ein eigenes Passwort eintragen. Verwenden Sie dabei nur Zeichen des ASCII-Zeichensatzes. Die Passwortabfrage kann zudem deaktiviert werden.

6.1 Status

Auf der Statusseite sieht man im oberen Bereich den WiFi-Verbindungsstatus.

OBP60V2

connected	●
Status Config XDR Data Update Help	
VERSION	dev20240210 ?
Access Point IP	192.168.15.1
wifi client connected	true
wifi client IP	192.168.1.142
# clients	0
TCP client connected	false
TCP client error	disabled
Free heap	108220
NMEA2000 State	[32] OFFLINE

Die Informationen haben folgende Bedeutung:

Version

Aktuelle Firmware-Version

Access Point IP

IP-Adresse des Access Points

WiFi Client connected

Zeigt an, ob das OBP60 mit einem anderen externen WiFi-Netzwerk als Client verbunden ist

WiFi Client IP

IP-Adresse, die dem OBP60 zugewiesen wurde

#clients

Anzahl der Clients, die sich mit dem OBP60 verbunden haben

TCP client connected

Hier wird der Verbindungsstatus des OBP60 angezeigt. false = OBP60 ist nicht als TCP-Client mit einem anderen Gerät verbunden. true = OBP60 ist als TCP-Client mit einem anderen Gerät verbunden.

TCP client error

Hier wird der Fehlerstatus des OBP60 für TCP-Client-Verbindungen angezeigt.

Free heap

Anzeige des freien Heap-Speichers in Byte. Der Heap-Speicher wird zur Verarbeitung von Informationen benötigt und darf nicht zu klein werden. Die Speichergröße ändert sich dynamisch je nach Auslastung der CPU. Der Wert kann zur Diagnose verwendet werden.

NMEA2000 State

Anzeige des Status des NMEA2000-Busses.

NMEA2000 in

Anzahl der NMEA2000-Telegramme, die empfangen wurden

NMEA2000 out

Anzahl der NMEA2000-Telegramme, die gesendet wurden

TCP in

Anzahl der NMEA0183-Telegramme, die über TCP empfangen wurden

TCP out

Anzahl der NMEA0183-Telegramme, die über TCP gesendet wurden

USB in

Anzahl der NMEA0183-Telegramme, die über USB empfangen wurden

USB out

Anzahl der NMEA0183-Telegramme, die über USB gesendet wurden

Serial in

Anzahl der NMEA0183-Telegramme, die über RS485 empfangen wurden

Serial out

Anzahl der NMEA0183-Telegramme, die über RS485 gesendet wurden

Wenn Sie auf das Fragezeichen hinter **Version** klicken, werden alle Telegramme angezeigt, die das OBP60 verarbeiten kann. Detailliertere Informationen zu den empfangenen Telegrammen sehen Sie, wenn Sie die Zeile des jeweiligen Bussystems aufklappen. Im Anhang finden Sie eine Tabelle mit allen NMEA0183- und NMEA2000-Telegrammen, die verarbeitet werden können.

Bemerkung: Zum besseren Verständnis ist zu beachten, dass das OBP60 ein eigenes, unabhängiges WiFi-Netzwerk aufbaut, diese Funktion wird auch als Access Point bezeichnet. Die Anzahl der TCP-Clients in der Statuszeile **#clients** bezieht sich dabei immer nur auf die Clients, die sich beim OBP60 im Access Point-Modus anmelden. Das OBP60 kann darüber hinaus in ein anderes, externes WiFi-Netzwerk eingebucht werden, indem es sich dort als Client anmeldet. In dem Fall wird das eigene WiFi-Netz des OBP60 mit dem externen WiFi-Netz gebrückt. Alle Daten des OBP60 sind dann in beiden Netzwerken verfügbar.

6.2 Config

Die Konfigurationsseite unterteilt sich in zwei Bereiche. Die Firmware basiert auf dem NMEA2000-Gateway-Projekt und nutzt die gesamte Grundstruktur dieses Software-Projektes. Die Funktionalität des OBP60 ist als eigenständiger Task in der NMEA2000-Gateway-Firmware implementiert. Der erste Bereich enthält die Konfiguration für das NMEA2000-Gateway. Im zweiten Bereich ist die Konfiguration zur OBP60-Hardware und -Software zu finden. Den zweiten Bereich erkennt man an dem Prefix OBP.

Konfiguration zum NMEA2000-Gateway

OBP60V2

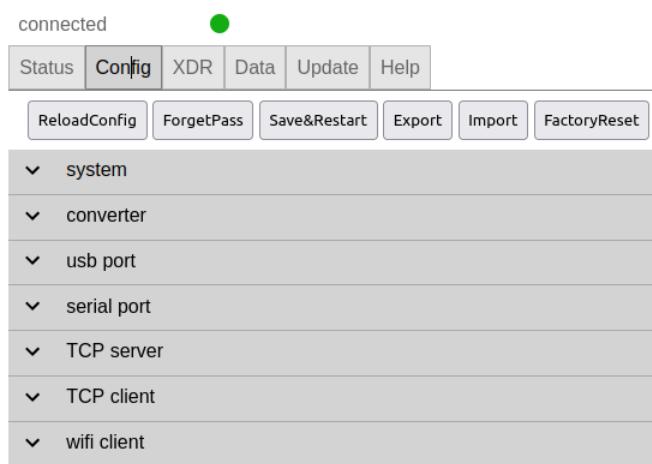


Abb.: Konfiguration zum NMEA2000-Gateway

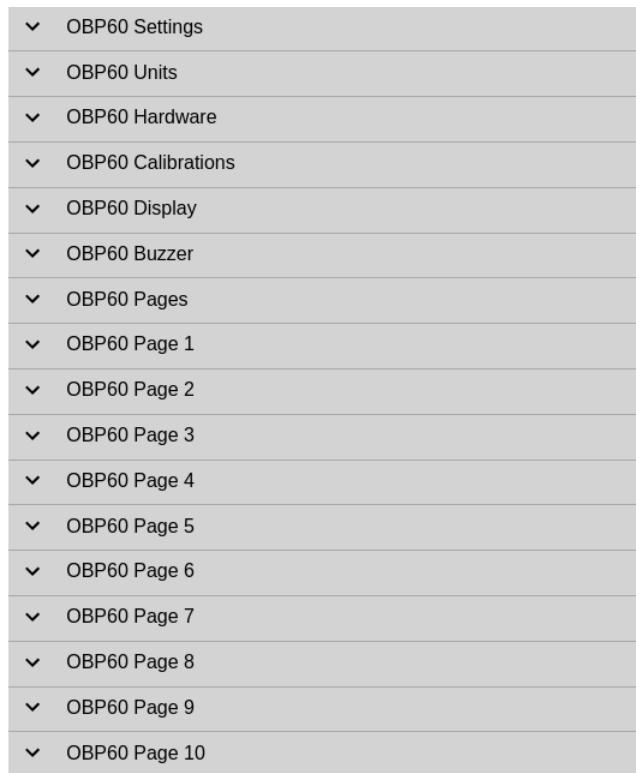


Abb.: Konfiguration zur OBP60-Hardware

Auf der Konfigurationsseite sind im oberen Bereich verschiedene Tasten zu sehen. Die Bedeutung der Tasten ist nachfolgend aufgeführt:

- **Reload Config** - Erneutes Laden der Konfiguration
- **Forget Pass** - Entfernen des Login-Passwortes aus dem Cache-Speichers des Browsers
- **Save & Restart** - Speichern der Konfiguration mit anschließendem Neustart der Firmware
- **Export** - Export einer Konfiguration als JSON-File
- **Import** - Import einer Konfiguration über ein JSON-File

- **Factory Reset** - Rücksetzen aller Einstellungen auf Werkszustand

6.3 Config - System

system	
system name	OBP60V2
NMEA0183 ID	GP
stopApTime	0
apPassword	(hidden)
apIp	192.168.15.1
apMask	255.255.255.0
useAdminPass	off
adminPassword	(hidden)
show all data	on
log level	error

Unter **System** werden grundlegende Einstellungen vorgenommen wie:

System Name

- Gerätename des OBP60. Hier kann ein Name verwendet werden, der aus bis zu 10 ASCII-Zeichen besteht. Dabei dürfen nur Buchstaben und Zahlen verwendet werden. Zusätzlich sind das Minus-Zeichen und der Unterstrich erlaubt. Sonderzeichen sind nicht erlaubt, da der Gerätename gleichzeitig auch als SSID im WiFi-Netzwerk verwendet wird.

NMEA0183 ID

- Hier kann festgelegt werden, welches Präfix als Geräte-ID im NMEA0183-Telegrammen verwendet wird. Es lassen sich verschiedene Geräte-IDs einstellen. Details dazu sind unter folgendem [Link](#) zu finden.

Stop AP Time

- Hierüber kann angegeben werden, nach welcher Zeit der WiFi Access Point abgeschaltet werden soll. Die Angabe der Zeit erfolgt in Sekunden. Der Wert **0** sorgt für einen dauerhaften Betrieb des WiFi Access Points.

AP Password

- An dieser Stelle wird das Passwort für den WiFi Access Point angegeben. Es dürfen nur Zeichen des ASCII-Zeichensatzes verwendet werden. Per Default ist ein Passwort aktiviert. Es wird das Passwort **esp32nmea2k** verwendet.

AP Ip

- Hier kann die IP-Adresse des WiFi Access Points eingestellt werden. Per Default steht die IP-Adresse auf **192.168.15.1**. In Ausnahmefällen kann die IP auf eine andere Adresse eingestellt werden. Beachten Sie dabei, dass das OBP60 bei veränderter IP-Adresse im Ihrem WLAN unter Umständen nicht mehr erreichbar sein könnte.

AP Mask

- An diese Stelle wird die Subnetz-Maske für den WiFi Access Point angegeben. Per Default steht die Subnetz-Maske auf **255.255.255.0**. Es wird dringend empfohlen, diesen Wert nicht zu verändern, es sei denn, Sie wissen genau, welche Auswirkungen eine Änderung hat.

Warnung: Achten Sie darauf, dass der Adressbereich des WiFi Access Points sich von dem Adressbereich des Netzes unterscheiden muss, in das sich das OBP60 als WiFi-Client einwählt. Der Adressbereich eines Netzwerks ist über die ersten 3 Zifferngruppen gekennzeichnet (111.222.333.xxx). Nur die letzte Gruppe (xxx) wird für die Gerätekennzeichnung im gleichen Netz benutzt. Verändern Sie die ersten 3 Zifferngruppen des Adressbereichs, werden Sie die Konfigurationsseiten des OBP60 nicht mehr ohne weiteres öffnen können. In den meisten Fällen wird eine Änderung der IP-Adresse oder der Subnetz-Maske nicht notwendig sein. Ändern Sie die IP-Adresse und die Subnetz-Maske daher nur, wenn Sie über genügend Netzwerkerfahrung verfügen und sich über die Auswirkungen Ihrer Änderungen im Klaren sind.

Use Admin Pass

- Hiermit kann festgelegt werden, ob für Änderungen der Konfiguration ein Passwort notwendig ist.

Admin Password

- Hier wird das Admin-Passwort eingegeben. Es dürfen nur Zeichen des ASCII-Zeichensatzes verwendet werden. Per Default ist die Passwortabfrage aktiviert. Es wird das Passwort **esp32admin** verwendet. Beim ersten Speichern einer Konfiguration nach einem Reboot wird kein Passwort benötigt. So können Sie das Passwort jederzeit ändern.

Show All Data

- Zeigt das Menü **on**, werden im Data-Bereich alle Sensordaten angezeigt. Das Umstellen auf **off** deaktiviert alle Sensordaten im Data-Bereich.

Log Level

- Über **Log Level** lässt sich der Detailgrad der Benachrichtigungen über die USB-C-Schnittstelle einstellen. Folgende Einstellungen stehen zur Verfügung:

- **off** - Keine Logging-Ausgaben
- **error** - Es werden nur Fehlermeldungen ausgegeben
- **log** - Es werden Fehlermeldungen und Statusinformationen ausgegeben
- **debug** - Es werden alle vorgesehenen Meldungen inklusive Debug-Meldungen ausgegeben

Hinweis: Wenn Sie beabsichtigen, einen NMEA0183-Datenaustausch über die USB-C-Schnittstelle durchzuführen, sollten Sie den **Log Level** auf **off** stellen. Beachten Sie das nicht, kann die Auswertung von Logging-Ausgaben sehr unübersichtlich werden, da Logging-Daten und NMEA0183-Telegramme dann gemischt ausgegeben werden. Wenn Sie nur Logging-Ausgaben sehen wollen, stellen Sie **NMEA to USB** und **NMEA from USB** unter *Config - USB Port* auf **off**.

6.4 Config - Converter

min XDR interval	100	X	?
min N2K interval	50	X	?
NMEA2000 out	on	X	?

Mit den nachfolgenden Einstellungen können Sie die Funktion des NMEA2000-Gateways verändern.

Min XDR Interval

- Hier wird die Intervallzeit der XDR-Signalverarbeitung eingestellt. XDR-Telegramme sind frei definierbare Sensor-Telegramme. Die Intervallzeit kann ab 10 ms eingestellt werden. Der Default-Wert steht auf 100 ms. Mit der kürzesten Intervallzeit von 10 ms wird eine Datenverarbeitungsrate von 100 Hz erreicht.

Min N2K Interval

- Hier wird die Intervallzeit der NMEA2000-Signalverarbeitung eingestellt. Die Intervallzeit kann ab 5 ms eingestellt werden. Der Defaultwert steht auf 50 ms.

Bemerkung: Bedenken Sie, dass kurze Intervallzeiten eine große Prozessorlast bewirken. Stellen Sie den Wert möglichst so ein, so dass ihre Daten noch zeitlich korrekt verarbeitet werden können. Mit dem Standardwert von 100 ms für das XDR-Interval und 50 ms für das N2K-Intervall können die meisten Anwendungen sinnvoll betrieben werden.

NMEA2000 out

- Hier kann eingestellt werden, ob NMEA2000-Telegramme in das NMEA-Netzwerk übertragen werden**
 - on - Ausgabe der NMEA2000-Daten
 - off - Keine Ausgabe der NMEA2000-Daten

6.5 Config - USB Port

USB mode	nmea0183	X	?
USB baud rate	115200	X	?
NMEA to USB	on	X	?
NMEA from USB	on	X	?
USB to NMEA2000	on	X	?
USB read Filter	aison / blacklist	X	?
USB write Filter	aison / blacklist	X	?

Über die Seite **USB Port** können die Funktionen des USB-Ports detailliert eingestellt werden.

USB Mode

- legt das Format fest, wie Daten am USB-Port verarbeitet werden. Mit dem Actisense-Format können NMEA2000-Telegramme von externer Software empfangen und verarbeitet werden. Actisense-Daten werden innerhalb des Geräts in NMEA2000-Daten und in NMEA0183-Daten übersetzt. So kann z.B. die [Simulations- und Diagnosesoftware](#) der Fa. Actisense zur Analyse der Busdaten verwendet werden.
- **nmea0183** - Verarbeitung im NMEA0183-Format
- **actisense** - Verarbeitung im Actisense-Format

USB Baud Rate

- Hier kann die Schnittstellengeschwindigkeit der seriellen USB-Schnittstelle eingestellt werden. Es lassen sich Geschwindigkeiten zwischen 1.200 Bd und 460.800 Bd einstellen.

Hinweis: Stellen Sie die Schnittstellengeschwindigkeit so ein, dass sie ausreichend hoch ist, um alle Datentelegramme im Sendeintervall verarbeiten zu können. Mit dem Default-Wert von 115.200 Bd können die meisten Anwendungen sinnvoll betrieben werden.

Mit den nachfolgenden drei Einstellungen lässt sich die Datenrichtung an der USB-C-Schnittstelle einstellen. Dabei wird zwischen NMEA0183 und NMEA2000 unterschieden.

NMEA to USB

- **on** - NMEA0183-Daten werden an die USB-Schnittstelle ausgegeben
- **off** - NMEA0183-Daten werden nicht an die USB-Schnittstelle ausgegeben

NMEA from USB

- **on** - NMEA0183-Daten werden von der USB-Schnittstelle empfangen
- **off** - NMEA0183-Daten werden nicht von der USB-Schnittstelle empfangen

USB to NMEA2000

- **on** - Daten werden von der USB-Schnittstelle an den NMEA2000-Bus weitergeleitet
- **off** - Daten werden nicht von der USB-Schnittstelle an den NMEA2000-Bus weitergeleitet

In den nächsten beiden Einstellungen werden die Filterfunktionen **USB read Filter** und **USB write Filter** für das Lesen und Schreiben an der USB-Schnittstelle gesetzt. Es lassen sich nur NMEA0183-Daten filtern. Dabei lässt sich gesondert einstellen, ob AIS-Positionssignale verarbeitet werden. Als Filterformen stehen <Whitelist> und <Blacklist> zur Verfügung, also einmal die Angabe von Filterkriterien, die die betroffenen Daten einschliessen sollen (Whitelist), dann solche, die zum Ausschluss von Daten führen (Blacklist).

USB Filter

- **aison** - AIS-Daten an der USB-Schnittstelle werden verarbeitet
- **aisoff** - AIS-Daten an der USB-Schnittstelle werden nicht verarbeitet
- **blacklist** - Der Filter arbeitet mit einer Blacklist. Die gekennzeichneten Telegramme werden nicht verarbeitet.
- **whitelist** - Der Filter arbeitet mit einer Whitelist. Nur die aufgelisteten Telegramme werden verarbeitet.

Im Eingabefeld werden die Kurzbezeichner der NMEA0183-Telegramme eingetragen, Mehrere Einträge werden durch Komma , getrennt. Folgende Kurzbezeichner können verwendet werden:

- DBK, DBS, DBT, DPT, GGA, GLL, GSA, GSV, HDM, HDT, MTW, MWD, MWV, RMB, RMC, ROT, RSA, VHW, VTG, VWR, XDR, XTE, ZDA

Die genaue Bedeutung der Kurzbezeichner ist [hier](#) erklärt.

Hinweis: Filterfunktionen sind ein mächtiges Werkzeug, um Datenflüsse zu steuern. Überlegen Sie sich vor der Konfiguration, wie Ihre Datenflüsse im Boot aussehen sollen, und erstellen sich dazu eine Skizze. Setzen Sie die Filter so ein, dass sie nur die Daten senden und empfangen, die sie auch wirklich benötigen. Unterscheiden Sie dabei, was gesendet und was empfangen werden soll, vermeiden Sie dabei auf alle Fälle Datenschleifen.

Warnung: Datenschleifen führen zu Fehlfunktionen des Gerätes. Bei Datenschleifen laufen die selben Daten über mehrere Geräte im Kreis. Dadurch entstehen hohe Senderaten, weil fortlaufend die gleichen Daten gesendet und empfangen werden. Die Prozessorlast erhöht sich dabei auf ein Maximum. Unter Umständen kann das Gerät ausfallen, die anfallenden Daten nicht mehr zeitnah verarbeiten oder nicht mehr bedienbar sein. Beachten Sie, dass der Zustand auch erst dann eintreten kann, wenn weitere Geräte am Bussystem später zugeschaltet werden.

6.6 Config - Serial Port

The screenshot shows a configuration interface for the serial port. It includes fields for serial direction (receive), serial baud rate (115200), serial to NMEA2000 (on), and two sets of serial read and write filters, each with dropdown menus for 'aion' and 'blacklist'.

Über **serial port** können Einstellungen zur seriellen NMEA0183-Schnittstelle vorgenommen werden. Diese Einstellungen beziehen sich auf die RS485-Schnittstelle am Steckverbinder **CN1** mit den Signalen A, B und Shield.

Serial Direction

- off - Die NMEA0183-Schnittstelle ist ausgeschaltet
- send - Die NMEA0183-Schnittstelle sendet
- receive - Die NMEA0183-Schnittstelle empfängt

Bemerkung: Die serielle Schnittstelle ist konform zu RS485 und RS422 und arbeitet im Halbduplex-Betrieb. Es kann entweder gesendet oder empfangen werden. Beides gleichzeitig ist nicht möglich. Wenn Sie eine Vollduplex-Übertragung für NMEA0183-Daten benötigen, dann können Sie die USB-C-Schnittstelle benutzen. Diese Schnittstelle ist aber nicht zu RS485 oder RS422 konform. Sie kann sinnvoll verwendet werden, wenn Sie Daten z.B. in OpenCPN auf einem PC oder Laptop verarbeiten wollen.

Serial Baud Rate

- Einstellung der Baudrate zwischen 1.200 und 460.800 Bd.

Serial To NMEA2000

- on - Daten an der Schnittstelle werden nach NMEA2000 übertragen (Gateway-Funktion)

- off - Daten an der Schnittstelle werden nicht nach NMEA2000 übertragen

In den nächsten beiden Einstellungen werden die Filterfunktionen **Serial read Filter** und **Serial write Filter** für das Lesen und Schreiben an der seriellen Schnittstelle vorgenommen. Es lassen sich nur NMEA0183-Daten filtern. Dabei lässt sich gesondert einstellen, ob auch AIS-Positionssignale verarbeitet werden. Als Filterformen stehen Whitelist und Blacklist zur Verfügung.

Serial Filter

- aison - AIS-Daten an der USB-Schnittstelle werden verarbeitet
- aisoff - AIS-Daten an der USB-Schnittstelle werden nicht verarbeitet
- blacklist - Der Filter arbeitet mit einer Blacklist. Die gekennzeichneten Telegramme werden nicht verarbeitet.
- whitelist - Der Filter arbeitet mit einer Whitelist. Nur die aufgelisteten Telegramme werden verarbeitet.

Im Eingabefeld werden die Kurzbezeichner der NMEA0183-Telegramme eingetragen, mehrere Einträge werden durch Komma , getrennt. Folgende Kurzbezeichner können verwendet werden:

- DBK, DBS, DBT, DPT, GGA, GLL, GSA, GSV, HDM, HDT, MTW, MWD, MWV, RMB, RMC, ROT, RSA, VHW, VTG, VWR, XDR, XTE, ZDA

Die genaue Bedeutung der Kurzbezeichner ist [hier](#) erklärt.

6.7 Config - TCP Server

TCP port	10110	X	?
max. TCP clients	6	X	?
NMEA0183 out	on	X	?
NMEA0183 in	on	X	?
to NMEA2000	on	X	?
NMEA read Filter	aison blacklist	X	?
NMEA write Filter	aison blacklist	X	?
Seasmart out	off	X	?

Hier werden die Einstellungen zum Betrieb des OPB60 als TCP-Server vorgenommen. Der TCP-Server ist ein Server-Dienst, über den Daten schreibend und lesend ausgetauscht werden können. Dabei meldet sich ein Netzwerk-Gerät als Client aktiv über einen TCP-Port am Server an und kann dann Daten mit dem TCP-Server austauschen.

Bemerkung: Der Anmeldevorgang muss immer vom Client initiiert werden. Bei Verbindungsabbrüchen muss der Client die Verbindung wieder selbstständig aufbauen. Achten Sie darauf, dass der Client über eine Auto-Connect-Funktion verfügt. Andernfalls verlieren Sie die Datenverbindung bei Verbindungsabbrüchen dauerhaft.

TCP Port

- Angabe des TCP-Port, auf dem der Server auf eingehende Verbindungsanfragen wartet. Der Default-Wert ist 10110. Verwenden Sie nur Ports größer 1024, da Ports unterhalb von 1024 für feste Anwendungen reserviert sind. Der Maximalwert liegt bei 65535.

Max TCP Clients

- Angabe, wieviele Clients sich maximal mit dem TCP-Server verbinden dürfen. Der Default-Wert ist 6.

Bemerkung: Beachten Sie, dass eine hohe Zahl an Clients eine große Rechenlast der CPU verursachen kann. Sorgen Sie daher dafür, dass sich nie mehr als 6 Clients mit dem Server verbinden können. Andernfalls kann es zur Beeinträchtigung der Datenverarbeitung kommen oder das Gerät reagiert nicht mehr korrekt.

NMEA0183 Out

- on - Am TCP-Port werden NMEA0183-Daten ausgegeben
- off - Am TCP-Port werden keine NMEA0183-Daten ausgegeben

NMEA0183 In

- on - Am TCP-Port werden NMEA0183-Daten empfangen
- off - Am TCP-Port werden keine NMEA0183-Daten empfangen

To NMEA2000

- on - Daten am TCP-Port werden nach NMEA2000 übertragen (Gateway-Funktion)
- off - Daten am TCP-Port werden nicht nach NMEA2000 übertragen

In den nächsten beiden Einstellungen werden die Filterfunktionen **NMEA Read Filter** und **NMEA Write Filter** für das Lesen und Schreiben am TCP-Port vorgenommen. Es lassen sich nur NMEA0183-Daten filtern. Dabei lässt sich gesondert einstellen, ob AIS-Positionssignale verarbeitet werden. Als Filterformen stehen „Whitelist“ und „Blacklist“ zur Verfügung.

NMEA Read Filter

- aison - Einkommende AIS-Daten an der USB-Schnittstelle werden verarbeitet
- aisoff - Einkommende AIS-Daten an der USB-Schnittstelle werden nicht verarbeitet
- blacklist - Der Filter arbeitet mit einer Blacklist. Die gekennzeichneten Telegramme werden nicht verarbeitet.
- whitelist - Der Filter arbeitet mit einer Whitelist. Nur die aufgelisteten Telegramme werden verarbeitet.

NMEA Write Filter

- aison - Zu sendende AIS-Daten an der USB-Schnittstelle werden verarbeitet
- aisoff - Zu sendende AIS-Daten an der USB-Schnittstelle werden nicht verarbeitet
- blacklist - Der Filter arbeitet mit einer Blacklist. Die gekennzeichneten Telegramme werden nicht verarbeitet.
- whitelist - Der Filter arbeitet mit einer Whitelist. Nur die aufgelisteten Telegramme werden verarbeitet.

Im Eingabefeld werden die Kurzbezeichner der NMEA0183-Telegramme eingetragen, mehrere Einträge werden durch Komma , getrennt. Folgende Kurzbezeichner können verwendet werden:

- DBK, DBS, DBT, DPT, GGA, GLL, GSA, GSV, HDM, HDT, MTW, MWD, MWV, RMB, RMC, ROT, RSA, VHW, VTG, VWR, XDR, XTE, ZDA

Die genaue Bedeutung der Kurzbezeichner ist [hier](#) erklärt.

Seasmart Out

- Über Seasmart lassen sich NMEA2000-Daten in NMEA0183-Telegrammen übersetzen. Wenn Sie **Seasmart** aktivieren, werden alle NMEA2000-Daten über NMEA0183-Telegramme ausgegeben und getunnelt. Die Daten werden dabei in Binärform in einem NMEA0183-Telegramm übertragen. Auf diese Weise können Sie von einem OBP60 (TCP-Server) zu einem weiteren OBP60 (TCP-Client) NMEA2000-Daten über Wifi übertragen. Achten Sie darauf, dass auf der Gegenseite ebenfalls **Seasmart** aktiviert ist.
- **on** - Der TCP-Server kann Seasmart-Daten senden und empfangen
- **off** - Seasmart wird vom TCP-Server nicht unterstützt

6.8 Config - TCP Client

The screenshot shows the 'TCP client' configuration section of the OBP60 V2 software. It contains the following settings:

- enable:** off
- remote port:** 10110
- remote address:** (empty)
- NMEA0183 out:** on
- NMEA0183 in:** on
- to NMEA2000:** on
- NMEA read Filter:** aison | blacklist
- NMEA write Filter:** aison | blacklist
- Seasmart out:** off

Hier werden die Einstellungen für den Betrieb des OBP60 als TCP-Client vorgenommen. Das OBP60 kann als TCP-Client Daten mit einem TCP-Server lesend und schreibend austauschen. Dabei meldet sich das OBP60 als Client aktiv über einen TCP-Port am TCP-Server an und kann dann Daten mit dem Server austauschen. Der TCP-Client-Modus enthält ein Auto-Connect, um bei Verbindungsabbrüchen automatisch die Verbindung wieder aufnehmen zu können.

Enable

- **on** - Der TCP-Client-Modus ist im OBP60 aktiviert
- **off** - Der TCP-Client-Modus ist deaktiviert

Remote Port

- Angabe des TCP-Ports, über den Daten mit einem TCP-Server ausgetauscht werden sollen. Der Default-Wert ist 10110. Damit der Datenaustausch zwischen einem TCP-Server und einem TCP-Client stattfinden kann, muss der selbe Port vom TCP-Client verwendet werden, den der TCP-Server für die Kommunikation verwendet. Benutzen Sie nur Ports größer 1024, da Ports unterhalb von 1024 für festgelegte Anwendungen reserviert sind. Der Maximalwert liegt bei 65535.

Remote Address

Die <Remote Address> ist die Adresse des TCP-Servers im WiFi-Netzwerk, mit dem Sie Daten austauschen wollen. Sie können eine IP-Adresse wie z.B. **192.168.15.1** oder einen MDNS-Hostnamen wie z.B. **OBP60V2.local** verwenden.

Warnung: Wenn Sie Daten zwischen zwei OBP60 via WiFi austauschen wollen, müssen sich beide Geräte im selben Funknetz befinden, auch müssen sie unterschiedliche System-Namen haben. Ihre Access Points müssen im gleichen IP-Adressbereich liegen, aber unterschiedliche Geräteadressen haben. Eine Gerät muss als TCP-Server und das andere Gerät als TCP-Client konfiguriert sein. Die Einstellungen dazu werden unter [Config - System](#) vorgenommen. Wenn Sie das nicht beachten, kann es zu Störungen im WiFi-Datenverkehr kommen und Sie können unter Umständen die Web-Konfigurationsoberflächen der Geräte nicht mehr erreichen.

NMEA0183 Out

- on - Am TCP-Port werden NMEA0183-Daten ausgegeben
- off - Am TCP-Port werden keine NMEA0183-Daten ausgegeben

NMEA0183 In

- on - Am TCP-Port werden NMEA0183-Daten empfangen
- off - Am TCP-Port werden keine NMEA0183-Daten empfangen

To NMEA2000

- on - Daten am TCP-Port werden nach NMEA2000 übertragen (Gateway-Funktion)
- off - Daten am TCP-Port werden nicht nach NMEA2000 übertragen

In den nächsten beiden Einstellungen werden die Filterfunktionen **NMEA Read Filter** und **NMEA Write Filter** für das Lesen und Schreiben am TCP-Port vorgenommen. Es lassen sich nur NMEA0183-Daten filtern. Dabei lässt sich gesondert einstellen, ob AIS-Positionssignale verarbeitet werden. Als Filterformen stehen Whitelist und Blacklist zur Verfügung.

NMEA Read Filter

- aison - Einkommende AIS-Daten an der USB-Schnittstelle werden verarbeitet
- aisoff - Einkommende AIS-Daten an der USB-Schnittstelle werden nicht verarbeitet
- blacklist - Der Filter arbeitet mit einer Blacklist. Die gekennzeichneten Telegramme werden nicht verarbeitet.
- whitelist - Der Filter arbeitet mit einer Whitelist. Nur die aufgelisteten Telegramme werden verarbeitet.

NMEA Write Filter

- aison - Zu sendende AIS-Daten an der USB-Schnittstelle werden verarbeitet
- aisoff - Zu sendende AIS-Daten an der USB-Schnittstelle werden nicht verarbeitet
- blacklist - Der Filter arbeitet mit einer Blacklist. Die gekennzeichneten Telegramme werden nicht verarbeitet.
- whitelist - Der Filter arbeitet mit einer Whitelist. Nur die aufgelisteten Telegramme werden verarbeitet.

Im Eingabefeld werden die Kurzbezeichner der NMEA0183-Telegramme eingetragen, mehrere Einträge werden durch Komma , getrennt. Folgende Kurzbezeichner können verwendet werden:

- DBK, DBS, DBT, DPT, GGA, GLL, GSA, GSV, HDM, HDT, MTW, MWD, MWV, RMB, RMC, ROT, RSA, VHW, VTG, VWR, XDR, XTE, ZDA

Die genaue Bedeutung der Kurzbezeichner ist [hier](#) erklärt.

SeaSmart Out

- Über SeaSmart lassen sich NMEA2000-Daten in NMEA0183-Telegrammen übersetzen. Wenn Sie **SeaSmart** aktivieren, werden alle NMEA2000-Daten über NMEA0183-Telegramme ausgegeben und getunnelt.

Die Daten werden dabei in Binärform in einem NMEA0183-Telegramm übertragen. Auf diese Weise können Sie von einem OBP60 (TCP-Server) zu einem weiteren OBP60 (TCP-Client) NMEA2000-Daten über Wifi übertragen. Achten Sie darauf, dass auf der Gegenseite ebenfalls **SeaSmart** aktiviert ist.

- **on** - Der TCP-Server kann SeaSmart-Daten senden und empfangen
- **off** - SeaSmart wird vom TCP-Server nicht unterstützt

6.9 Config - WiFi Client

The screenshot shows a configuration interface for a WiFi client. At the top, there is a header bar with a back arrow and the text "wifi client". Below this, there are three input fields: "wifi client" (set to "on"), "wifi client SSID" (containing "Strubbel"), and "wifi client password" (containing masked text). Each field has a delete button ("X") and a help button ("?").

Das OBP60 kann neben dem WiFi Access Point auch als WiFi-Client betrieben werden. In diesem Modus kann das OBP60 einem anderen WiFi-Netz beitreten und dort Daten austauschen. Auf diese Weise lässt sich das OPB60 in Ihr bestehendes Bord-WLAN integrieren. Der WiFi-Client-Modus enthält ein Auto-Connect, um bei Verbindungsabbrüchen automatisch die Verbindung wieder aufnehmen zu können.

WiFi Client

- **on** - Der WiFi-Client-Modus ist aktiviert
- **off** - Der WiFi-Client-Modus wird nicht unterstützt

WiFi Client SSID

- Tragen Sie hier einen WiFi-Netzwerknamen ein, zum Beispiel den Ihres Bord-WLANS. Als Namen können alle Zeichen des ASCII-Zeichensatzes verwendet werden.

WiFi Client Password

- Tragen Sie hier das zur o.g. SSID gehörende WiFi-Passwort ein. Als Passwort können alle Zeichen des ASCII-Zeichensatzes verwendet werden. Bei der Eingabe wird das Passwort verdeckt mit Sternchen ***** angezeigt. Über das Augen-Symbol kann das Passwort im Klartext angezeigt werden.

Hinweis: Wenn Sie Probleme mit der Verbindung zu weiteren WiFi-Netzwerken haben, dann überprüfen Sie, ob der Netzwerkname oder das Passwort Sonderzeichen enthält. In einigen Situationen können Sonderzeichen oder zu lange Passwörter Verbindungsprobleme verursachen. Ändern Sie dann versuchsweise den Netzwerknamen oder das Passwort. Mitunter hilft auch ein Neustart Ihres Bord-Routers, in dessen WLAN Sie das OPB60 einbuchen möchten.

6.10 Config - OBP Settings

^ OBP60 Settings

Time Zone	1	X ?
Boat Draft [m]	0	X ?
Fuel Tank [l]	0	X ?
Fuel Consumption [l/h]	0	X ?
Water Tank [l]	0	X ?
Waste Tank [l]	0	X ?
Battery Voltage [V]	12V	X ?
Battery Type	Pb	X ?
Battery Capacity [Ah]	0	X ?
Solar Power [W]	0	X ?
Generator Power [W]	0	X ?

Auf der Seite **OBP60 Settings** können Sie Einstellungen vornehmen, die sich auf Ihr Boot beziehen, in dem das OBP60 eingebaut ist. Die eingetragenen Werte werden dazu benutzt, um zum Beispiel eine ungefähre Reichweitenbestimmung für Wasser, Kraftstoff und Batterie vornehmen zu können. Geben Sie bitte die Werte für Ihr Boot möglichst genau ein, und beachten Sie die entsprechenden Einheiten. Die Einstellungen dienen dazu, verschiedene Betriebszustände auf dem OBP60 in Grafiken darzustellen.

Warnung: Bedenken Sie, dass die Reichweitenbestimmung mit dem internen Spannungssensor nur als Richtwert verstanden werden sollte. Insbesondere bei den Batterietypen AGM und LiFePo4 müssen Sie mit größeren Ungenauigkeiten rechnen. Beobachten und überprüfen Sie die Ergebnisse unter realen Bedingungen, bevor Sie den Anzeigewerten vertrauen.

Time Zone

- Über **Time Zone** kann die Zeitzone im Bereich von -12 und +14 Stunden eingestellt werden.

Die meisten Einstellungen sollten selbsterklärend sein. Sofern Sie keine Solarpaneele benutzen, belassen Sie den Wert von **Solar Power** auf 0. **Generator Power** bezieht sich auf einen Elektrogenerator, der im Boot arbeitet. Das kann eine Lichtmaschine, ein Windgenerator, ein Schleppgenerator oder ein weiterer Zusatz-Generator sein. Die Leistungsangaben für **Solar Power** und **Generator Power** werden zur Visualisierung der Energieflüsse benötigt.

6.11 Config - OBP Units

^ OBP60 Units

Length Format	m	X	?
Distance Format	nm	X	?
Speed Format	kn	X	?
Wind Speed Format	kn	X	?
Temperature Format	C	X	?
Date Format	DE	X	?

Die Einstellung der Einheiten wird unter **OBP Units** vorgenommen. Für die jeweiligen physikalischen Größen lassen sich verschiedene Einheiten verwenden.

Date Format

- Mit **Date Format** kann das Ausgabeformat des Datums angepasst werden.
- DE - Deutsches Datumsformat 31.12.2024
- GB - Britisches Datumsformat 31/12/2024
- US - US-Datumsformat 12/31/2024

6.12 Config - OBP Hardware

OBP60 Hardware

CPU Speed [MHz]	80	X ?
RTC Modul	DS1388	X ?
GPS Sensor	ATGM336H	X ?
Env. Sensor	BMP280	X ?
Battery Sensor	off	X ?
Battery Shunt	10	X ?
Solar Sensor	off	X ?
Solar Shunt	10	X ?
Gen. Sensor	off	X ?
Gen. Shunt	10	X ?
Rot. Sensor	off	X ?
Rot. Function	off	X ?
Rot. Offset	0	X ?
Roll Limit	25	X ?
Roll Offset	0	X ?
Pitch Offset	0	X ?
Temp. Sensor	off	X ?
Power Mode	Max Power	X ?
Undervoltage	off	X ?
Simulation Data	on	X ?

Unter **Hardware** werden alle Einstellungen bezüglich verbauter Hardware oder externer Zusatz-Hardware des OBP60 vorgenommen. Die Default-Einstellungen entsprechen den Minimal-Einstellungen für ein OBP60-Gerät. Je nach verbauter Hardware können unterschiedliche Sensoren und Funktionen zum Einsatz kommen.

CPU Speed

- Taktfrequenz der CPU. Die Taktfrequenz wird 1 min nach dem Abschluss des Bootvorgangs umgestellt.
- 80 - 80 MHz
- 160 - 160 MHz
- 240 - 240 MHz

RTC Modul

- Typ der Echtzeituhr
- off - Es wird keine Echtzeituhr benutzt

- DS1388 - Echtzeituhr DS1388 (Default)

GPS Sensor

- Typ des GPS-Sensors
- off - Es wird kein GPS-Sensor benutzt
- NEO-6M - GPS-Sensor NEO-6M
- NEO-M8N - Höherwertiger GPS-Sensor NEO-M8N
- ATGM336H - GPS-Sensor ATGM336H (Default)

Env. Sensor

- Angaben zum verwendeten Umgebungssensor. Dabei können verschiedene Sensoren ausgewählt werden. Die Sensoren sind am I2C-Bus angeschlossen. Es können interne Gerätesensoren des OBP60 oder externe Sensoren ausgewählt werden.
- off - Es wird kein Umgebungssensor benutzt
- BME280 - Sensor für Temperatur, Luftfeuchtigkeit und Luftdruck
- BMP280 - Sensor für Temperatur und Luftdruck (Default)
- BMP180 - Sensor für Temperatur und Luftdruck
- BME085 - Sensor für Temperatur und Luftdruck
- HTU21 - Sensor für Temperatur und Luftfeuchtigkeit
- SHT21 - Sensor für Temperatur und Luftfeuchtigkeit

Battery Sensor

- Hier können Sensoren ausgewählt werden, die am externen I2C-Bus angeschlossen sind und Batterie-Werte auslesen.
- off - Es wird kein Sensor benutzt
- INA219 - Sensor für Spannung 0...36V, Strom 0...500A und Leistung, I2C-Adresse 0x40
- INA226 - Sensor für Spannung 0...36V, Strom 0...500A und Leistung, I2C-Adresse 0x41

Battery Shunt

- Hier kann der Shunt ausgewählt werden, der zur Messung des Batterie-Stroms dient. Es können nur Shunts verwendet werden, die 75 mV als Spannungsabfall bei Maximalstrom verwenden. Diese Angabe ist am Shunt zu finden.
- 10 - Shunt für 10A
- 50 - Shunt für 50A
- 100 - Shunt für 100A
- 200 - Shunt für 200A
- 300 - Shunt für 300A
- 400 - Shunt für 400A
- 500 - Shunt für 500A

Solar Sensor

- Hier können Sensoren ausgewählt werden, die am externen I2C-Bus angeschlossen sind und Solar-Werte auslesen.

- off - Es wird kein Sensor benutzt
- INA219 - Sensor für Spannung 0...36V, Strom 0...500A und Leistung, I2C-Adresse 0x41
- INA226 - Sensor für Spannung 0...36V, Strom 0...500A und Leistung, I2C-Adresse 0x44

Solar Shunt

- Hier kann der Shunt ausgewählt werden, der zur Messung des Solar-Stroms dient. Es können nur Shunts verwendet werden, die 75 mV als Spannungsabfall bei Maximalstrom verwenden. Diese Angabe ist am Shunt zu finden.
- 10 - Shunt für 10A
- 50 - Shunt für 50A
- 100 - Shunt für 100A
- 200 - Shunt für 200A
- 300 - Shunt für 300A
- 400 - Shunt für 400A
- 500 - Shunt für 500A

Generator Sensor

- Hier können Sensoren ausgewählt werden, die am externen I2C-Bus angeschlossen sind und Generator-Werte auslesen.
- off - Es wird kein Sensor benutzt
- INA219 - Sensor für Spannung 0...36V, Strom 0...500A und Leistung, I2C-Adresse 0x45
- INA226 - Sensor für Spannung 0...36V, Strom 0...500A und Leistung, I2C-Adresse 0x45

Solar Shunt

- Hier kann der Shunt ausgewählt werden, der zur Messung des Solarstroms dient. Es können nur Shunts verwendet werden, die 75 mV als Spannungsabfall bei Maximalstrom verwenden. Diese Angabe ist am Shunt zu finden.
- 10 - Shunt für 10A
- 50 - Shunt für 50A
- 100 - Shunt für 100A
- 200 - Shunt für 200A
- 300 - Shunt für 300A
- 400 - Shunt für 400A
- 500 - Shunt für 500A

Rot. Sensor

- Über **Rot.Sensor** kann der Sensor zur Winkelmessung ausgewählt werden, der sich am externen I2C-Bus befindet.
- off - Es wird kein Sensor benutzt
- AS5600 - Magnetischer Sensor zur Winkelmessung von 0° bis 360° ohne Endanschlag, I2C-Adresse 0x36

Rot. Function

- Funktion des Winkelsensors

- Rudder - Winkelsensor für Ruderstellung
- Wind - Winkelsensor für Windrichtung
- Mast - Winkelsensor für Mastausrichtung bei drehbaren Masten
- Keel - Winkelsensor für Kielneigung
- Trim - Winkelsensor für Trimmklappen oder Foils
- Boom - Winkelsensor für Großbaum

Rot. Offset

Offset des Winkelsensors. Damit kann der Nullpunkt der externen Winkelsensoren am I2C-Bus korrigiert werden.

Roll Limit

Roll Limit gibt den maximal zulässigen seitlichen Neigungswinkel für das Rollen des Bootes an. Unter realen Bedingungen sind 20 Grad als Grenzwert realistisch.

Roll Offset

Offset des Neigungs-Winkelsensors. Damit kann der Nullpunkt des Winkelsensors für das seitliche Rollen Ihres Bootes korrigiert werden.

Pitch Offset

Offset des Winkelsensors für Pitch. Damit kann der Nullpunkt des Winkelsensors für das Nicken Ihres Bootes korrigiert werden.

Temp Sensor

- Hier kann der Sensortyp ausgewählt werden, der am 1Wire-Bus verwendet wird. Es werden bis zu 8 Sensoren am 1Wire-Bus unterstützt.
- off - Es wird kein Sensor benutzt
- DS18B20 - Temperatursensor -10...+85°C (1...8 Sensoren)

Power Mode

- Der **Power Mode** bezieht sich auf die Art der Stromversorgung, die für das OBP60 verwendet wird.
- **Max Power** - Alle Stromversorgungen sind eingeschaltet. Hierbei ist das Gerät am leistungsfähigsten und es kann der höchste Stromverbrauch entstehen.
- **Only 5.0V** - Es ist nur die zusätzliche Stromversorgung für 5.0 V eingeschaltet.
- **Min Power** - Es sind nur die Stromversorgungen eingeschaltet, die die Minimal-Funktionen bereitstellen. Hierbei entsteht der geringste Stromverbrauch. Die Bussysteme, das GPS, die externe 5V-Stromversorgung, die Hintergrundbeleuchtung und der Buzzer sind ausgeschaltet. Das Display, die Tasten, die RTC und der Umweltsensor BMP280 sind eingeschaltet.

Baugruppe	Max Power [W]	Min Power [W]
CPU ESP32-S3	on	on
e-Paper Display	on	on
Touch-Tasten	on	on
Echtzeituhr RTC	on	on
Sensor BMP280	on	on
1Wire	on	on
Flash-LED	on	off
Hintergrundbeleuchtung	on	off
Buzzer	on	off
GPS	on	off
Bussysteme N2k, 0183	on	off
Externe 5V-Versorgung	on	off

Tab.: Aktive Baugruppen OBP60 V2.1

Komponenten	Max Power [W]	Min Power [W]
CPU 240 MHz, WiFi, AP	1.78	1.30
CPU 160 MHz, WiFi, AP	1.68	1.20
CPU 80 MHz, WiFi, AP	1.58	1.13
CPU 240 MHz, WiFi	1.16	0.70
CPU 160 MHz, WiFi	1.07	0.60
CPU 80 MHz, WiFi	0.96	0.53
Externe 5V-Versorgung	0.83	0.00

Tab.: Stromverbrauch OBP60 V2.1 (AP - Access Point)

Je nach zugeschalteter Farbe und Leistung der Hintergrundbeleuchtung entsteht ein zusätzlicher Stromverbrauch.

RGB-LED-Beleuchtung	LED 100% [W]	LED 50% [W]
LED rot	0.24	0.11
LED grün	0.24	0.11
LED blau	0.24	0.11
LED weiss	0.61	0.32

Tab.: Stromverbrauch der LED-Hintergrundbeleuchtung

Undervoltage

- Erkennung einer Unterspannung der Stromversorgung. Wenn eine Unterspannung niedriger als 9 V erkannt wird, kann das OBP60 automatisch deaktiviert werden, um eine Tiefentladung der Bordbatterie zu vermeiden. In kritischen Situationen kann das OBP60 trotz Unterspannung bis 7 V funktionsfähig bleiben, wenn der Unterspannungsschutz deaktiviert ist. Als Default-Wert ist der Unterspannungsschutz aktiviert. Wenn im aktivierte Zustand eine Unterspannung auftritt, wird das OBP60 deaktiviert und in den Tiefschlaf versetzt. Im Display erscheint die Meldung **Undervoltage**. Dieser Zustand kann nur verändert werden, wenn die Versorgungsspannung vollständig ausgeschaltet und wieder eingeschaltet wird.
- **on** - Der Unterspannungsschutz ist aktiviert
- **off** - Der Unterspannungsschutz ist ausgeschaltet

Hinweis: Wenn Sie das OBP60 über USB mit Strom versorgen möchten, muss die Erkennung der Unterspannung abgeschaltet werden, da sich das Gerät sonst automatisch abschaltet.

Simulation Data

- Mit **Simulation Data** können Bus- und Sensordaten simuliert werden. Die Funktion ist nützlich, wenn die Funktionalität des Gerätes im ausgebauten Zustand ohne angeschlossene Busse oder Sensoren getestet werden soll. Das Gerät befindet sich dann in einem Demo-Mode.
- **on** - Sensordaten werden durch Simulationsdaten ersetzt
- **off** - Es werden Live-Sensordaten verwendet

Warnung: Bedenken Sie, dass Simulationsdaten als Live-Daten fehlinterpretiert werden können. Benutzen Sie Simulationsdaten nur, wenn Sie das OBP60 nicht zur Navigation benötigen und stellen es nach der Benutzung wieder auf Live-Daten um, indem Sie den Simulations-Modus beenden.

6.13 Config - OBP Calibrations

Touch Sensitivity [%]	100	X	?
VSensor Offset	-1.00	X	?
VSensor Slope	1.00	X	?

Auf der Seite **Calibrations** können Einstellungen zur Kalibrierung vorgenommen werden. Damit lassen sich Ungenauigkeiten von bestimmten Messwerten korrigieren. Die Korrektur kann je nach Sensor mit einer linearen oder quadratischen Korrektur durchgeführt werden.

Touch Sensitivity

- Einstellung der Tastenempfindlichkeit 0...100%. 0% bedeutet minimale Empfindlichkeit. 100% bedeutet maximale Empfindlichkeit.

VSensor Offset

- Offset der Korrekturfunktion des internen Spannungssensors des OBP60

VSensor Slope

- Steigung der Korrekturfunktion des internen Spannungssensors des OBP60

6.14 Config - OBP Display

^ OBP60 Display

Display Mode	Logo + QR Code	X ?
Inverted Display Mode	Normal	X ?
Status Line	on	X ?
Refresh	on	X ?
Fast Refresh	on	X ?
Full Refresh Time [min]	10	X ?
Hold Values	off	X ?
Backlight Mode	Control by Key	X ?
Backlight Color	Red	X ?
Brightness [%]	50	X ?
Flash LED Mode	Limit Violation	X ?

Der Bereich **Display** enthält alle Einstellungen, die das Display betreffen.

Display Mode

- Über den **Display Mode** wird eingestellt, wie sich das Display unmittelbar nach dem Einschalten verhält.
- Logo + QR Code - Das Logo und der QR-Code für den WiFi-Zugang werden angezeigt.
- Logo - Nur das Logo wird angezeigt.
- White Screen - Es wird eine weiße Seite angezeigt.
- off - Das Display wird deaktiviert, es wird zur Anzeige nicht verwendet.

Inverted Display Mode

- Normal - Der Bildschirminhalt wird schwarz auf weißem Untergrund angezeigt.
- Inverse - Der Bildschirminhalt wird weiß auf schwarzem Untergrund angezeigt.

Status Line

- on - Die Statuszeile wird im oberen Bereich des Bildschirms angezeigt.
- off - Die Statuszeile ist deaktiviert.

Refresh

- on - Der Auto-Refresh des Bildschirminhaltes ist aktiviert. Damit werden Geisterbilder beim Seitenwechsel unterbunden. Es wird ein Voll-Refresh des E-Paper-Displays durchgeführt. Alle 10 min erfolgt zusätzlich automatisch ein Voll-Refresh.
- off - Auto-Refresh ist deaktiviert

Bemerkung: Die Entstehung von Geisterbildern ist von der Display-Temperatur des OBP60 abhängig. Bei tiefen Temperaturen sind Geisterbilder deutlicher zu sehen und die Anzeige reagiert träger als bei höheren Temperaturen. Kurz

nach dem Einschalten wird für die ersten 5 Minuten jede Minute ein Voll-Refresh durchgeführt, damit sich das Display akklimatisieren kann. Bei extrem großer Sonneneinstrahlung kann es vorkommen, dass der Kontrast des Display-Inhaltes verloren geht. Schwarze Anzeigebereiche werden dann nur noch grau dargestellt. Das Display ist in diesem Fall nicht defekt. Nach einem Voll-Refresh regeneriert sich das Display und der Kontrast wird wieder vollständig hergestellt.

Fast Refresh

- **on** - Bei aktiviertem Fast Refresh wird eine Voll-Refresh schneller ausgeführt. Es werden weniger Schwarz-Weiß-Wechsel durchgeführt.
- **off** - Bei deaktiviertem Fast Refresh wird eine Voll-Refresh langsamer ausgeführt, weil mehr Schwarz-Weiß-Wechsel durchgeführt werden.

Full Refresh Time

- Über Full Refresh Time kann festgelegt werden nach welcher Zeit ein regelmäßiger Voll-Refresh durchgeführt wird. Full Refreshes sind für das e-Paper Display wichtig, da das Display nach einer gewissen Zeit mit partiellen Updates einen Voll-Refresh zur Erholung durchführen muss, um die Displayfunktionalität zu erhalten. Bei einem Voll-Refresh wird der Displaykontrast wieder vollständig hergestellt.

Bemerkung: Bei starker Sonneneinstrahlung kann je nach verwendetem Displaytyp ein Kontrastverlust nach einiger Zeit auftreten. Um den Effekt zu minimieren, sollte der **Fast Refresh** deaktiviert werden und die **Full Refresh Time** auf 1 min gesetzt sein. Der Erholungseffekt ist für das Display dadurch wesentlich stärker.

Als Hilfestellung wie man die Einstellungen zum Display vornehmen kann, dient die nachfolgende Tabelle:

Parameter	Temp <= 20°C	Temp > 20°C	Direkte Sonne
Refresh	off	on	on
Fast Refresh	on	on	off
Full Refresh Time	10 min	5 min	1 min

Hold Values

- **on** - Anzeigewerte werden gehalten, wenn die Datenverbindung kurzzeitig fehlen sollte und die Daten nicht aktualisiert werden können. Diese Einstellung kann bei TCP-Verbindungen über WiFi nützlich sein.
- **off** - Anzeigewerte werden nicht gehalten. Bei unterbrochener Datenverbindung länger als 5 s werden fehlende Daten mit --- gekennzeichnet.

Backlight Mode

- **Off** - Die Hintergrundbeleuchtung ist dauerhaft ausgeschaltet.
- **Control by Sun** - Automatisches Schalten der Beleuchtung durch den Sonnenstand
- **Control by Bus** - Automatisches Schalten der Beleuchtung über den Bus durch NMEA2000
- **Control by Time** - Schalten der Beleuchtung durch ein vorgegebenes Zeitintervall
- **Control by Key** - Manuelles Schalten der Beleuchtung durch eine Sensortaste
- **On** - Die Hintergrundbeleuchtung ist dauerhaft eingeschaltet.

Backlight Color

- Die Farbe der Hintergrundbeleuchtung kann durch 6 RGB-LEDs individuell eingestellt werden.
- **Red** - rot
- **Orange** - orange

- Yellow - gelb
- Green - grün
- Blue - blau
- Aqua - wasser
- Violet - violett
- White - weiß (höchster Stromverbrauch)

Brightness

Über **Brightness** kann die Helligkeit der Hintergrundbeleuchtung der RGB-LEDs zwischen 20... 100% eingestellt werden. Der Default-Wert liegt bei 50%. Damit wird sehr wenig Strom für die Hintergrundbeleuchtung benötigt. Die Helligkeit ist damit für den Nachtbetrieb so eingestellt, dass die Beleuchtung nicht blenden kann.

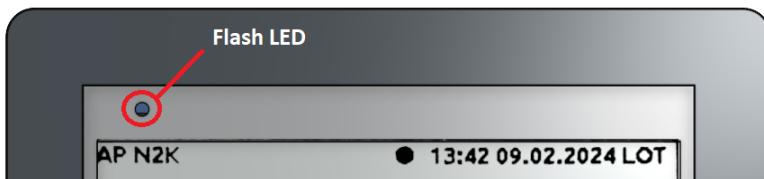
Hinweis: Für längere Nachtfahrten ist eine rote Hintergrundbeleuchtung empfehlenswert, die moderat in der Helligkeit auf z.B. 50% eingestellt ist. Bei rotem Licht muss sich das Auge nicht ständig an wechselnde Lichtverhältnisse anpassen. So können Sie nachts das Display ohne Sicht einschränkungen ablesen.

Bemerkung:

Je höher die Helligkeit der Hintergrundbeleuchtung eingestellt wird, um so mehr Strom wird verbraucht. Bei weißer Hintergrundbeleuchtung tritt der größte Stromverbrauch auf, da alle 3 Farben der RGB-LED zur Erzeugung von weißem Licht benutzt werden. Bei reinen Grundfarben wie rot, grün und blau wird am wenigsten Strom verbraucht. Bei Mischfarben werden die RGB-LEDs unterschiedlich stark angesteuert und der Stromverbrauch ist höher als bei den Grundfarben. Nachfolgend zwei Beispiele:

- 100%, weiß - 2 W
 - 50%, rot - 0.2W
-

Flash LED Mode



Die Flash-LED befindet sich in der linken oberen Ecke über dem E-Paper-Display und zeigt verschiedene Zustände des OBP60 an. Die LED kann dabei verschiedene Farben annehmen, die je nach Verwendung unterschiedliche Bedeutung haben.

- Off - Die Flash-LED ist dauerhaft ausgeschaltet.
- Bus Data - Bei eintreffenden Busdaten leuchtet die LED kurz blau auf.
- GPS Fix Lost - Bei dauerhaft roter Flash-LED wurde der GPS-Fix verloren. Die GPS-Daten sind ungültig.
- Limit Violation - Bei blinkend roter Flash-LED ist ein Grenzwert über- oder unterschritten worden.

Die Flash-LED leuchtet mit maximaler Helligkeit, sodass sie optisch auch bei hellen Sonnenlicht gut wahrgenommen werden kann. Die Bedeutung der Farben ist folgende:

- Rot - Alarmierung bei Grenzwertüberschreitung
- Grün - Bestätigung von Zustandsänderungen (z.B. Autopilot ein/aus)

- Blau - Signalisierung von Zuständen (z.B. GPS-Empfang, Datentransfer usw.)

6.15 Config - OBP Buzzer

The screenshot shows a configuration interface for the OBP60 Buzzer. It includes five settings with dropdown menus and input fields:

- Buzzer Error: off
- Buzzer GPS Fix: off
- Buzzer by Limits: off
- Buzzer Mode: Off
- Buzzer Power [%]: 50

In diesem Bereich lassen sich die Funktionen des Buzzer einstellen. Der Buzzer dient zur akustischen Signalisierung von Systemzuständen und Störungen des OBP60.

Buzzer Error

- on - Der Buzzer ertönt bei Störungen und Fehlern.
- off - Die Funktion ist deaktiviert.

Buzzer GPS Fix

- on - Der Buzzer ertönt, wenn das GPS-Signal verloren wurde.
- off - Die Funktion ist deaktiviert.

Buzzer by Limits

- on - Der Buzzer ertönt bei Grenzwertüberschreitungen.
- off - Die Funktion ist deaktiviert.

Buzzer Mode

- Off - Die Buzzer ist dauerhaft ausgeschaltet.
- Short Single Beep - Bei Aktivierung ertönt ein kurzer Einzelton.
- Longer Single Beep - Bei Aktivierung ertönt ein längerer Einzelton.
- Beep until Confirmation - Bei Aktivierung ertönt der Buzzer so lange, bis er durch Betätigen einer beliebigen Taste deaktiviert wird.

Buzzer Power

Über **Buzzer Power** kann die Lautstärke des Warntons zwischen 0...100% eingestellt werden. Die Lautstärke gilt grundsätzlich für alle Audioausgaben.

6.16 Config - OBP Pages

^ OBP60 Pages

Number of Pages	<input type="text" value="10"/>	X	?
Start Page	<input type="text" value="1"/>	X	?
Screenshot Format	<input type="text" value="Windows bitmap (BMP)"/>	X	?

Die Konfiguration der möglichen Anzeigeseiten des OBP60 erfolgt auf der Seite **Pages**. Hier wird festgelegt, wie viele Anzeigeseiten das OBP60 darstellen soll. Außerdem lässt sich festlegen, welche Anzeigeseite beim Einschalten gezeigt werden soll.

Number of Pages

- Hier wird die maximale Anzahl der Anzeigeseiten festgelegt. Es muss mindestens eine Anzeigeseite definiert sein, es können maximal 10 Anzeigeseiten aktiviert werden.

Start Page

- Dieser Wert legt fest, welche Seite beim Start angezeigt werden soll. Es können nur die Seiten angezeigt werden, die innerhalb der Seitenanzahl (**Number of Pages**) liegen.

Screenshot Format

- Legt fest welches Bildausgabeformat für Screenshots benutzt wird. Es stehen folgende Formate zur Verfügung:
 - Compressed Image (GIF) - Komprimierte GIF-Datei
 - Portable Bitmap (PBM) - Binäres Bildformat ohne Header (kann nicht im Browser angezeigt werden)
 - Windows Bitmap (BMP) - Binäres Bildformat mit Header
- Ein Screenshot kann erstellt werden, indem folgende Webseite aufgerufen wird:
- <http://192.168.15.1/api/user/OBP60Task/screenshot>

6.17 Config - OBP Page X



Im OBP60 gibt es insgesamt bis zu 10 Seiten, die man frei auswählen und gestalten kann. Je nach Seite können unterschiedlich viele Daten angezeigt werden. Es gibt frei definierbare Seiten, in denen die Inhalte zum Anzeigen ausgewählt werden können. Dann gibt es Seiten mit vorgegebenem, nicht veränderbarem Inhalt. Die meisten numerischen Seiten sind änderbar, während die grafischen Seiten oft vordefinierte Inhalte anzeigen.

- Seiten mit veränderbarem Inhalt
 - **OneValue** - Ein Anzeigewert
 - **TwoValue** - Zwei Anzeigewerte
 - **ThreeValue** - Drei Anzeigewerte
 - **FourValue** - Vier Anzeigewerte
 - **FourValue2** - Vier Anzeigewerte (andere Anordnung vertikal/horizontal)
 - **WindRoseFlex** - Anzeige der Winddaten (alle Anzeigewerte konfigurierbar, erster Wert wird grafisch auf der Windrose dargestellt)
 - * **RollPitch** - Grafische Anzeige von Roll und Pitch
- Seiten mit festem Inhalt
 - **Voltage** - Anzeige der Bordspannung (**xdrVbat**)
 - **WindRose** - Anzeige der Winddaten (**AWA, AWS, TWD, TWS, DBT, STW**)
 - **DST810** - Anzeige für Tiefe, Speed, Log und Wassertemperatur (**DBT, STW, Log, WTemp**)
 - **Clock** - Grafische Zeitanzeige mit Sonnenauf- und Sonnenuntergang (**GPST, GPSD**)
 - **White Page** - Leere weiße Seite, um Display in StandBy zu schalten
 - **BME280** - Anzeige von Umweltdaten wie Temperatur, Luftdruck und Feuchtigkeit (**BME280 I2C**)
 - **Rudder** - Grafische Anzeige der Ruderposition (**RPOS**)
 - **Keel** - Grafische Anzeige der Kielposition (**AS5600 I2C**)
 - **Battery** - Anzeige von Spannung, Strom und Leistung (**INA219, INA226 I2C**)
 - **Battery2** - Grafische Anzeige des Batterie-Ladezustandes (**INA219, INA226 I2C**)
 - **Solar** - Grafische Anzeige des Solar-Ladezustandes (**INA219, INA226 I2C**)
 - **Generator** - Grafische Anzeige des Generator-Ladezustandes (**INA219, INA226 I2C**)

Bemerkung: Bitte beachten Sie, dass alle Seiten mit festen Inhalten bestimmte Sensorwerte voraussetzen, um Messwerte anzeigen zu können. Unter dem Register **Data** kann die Verfügbarkeit der notwendigen Daten geprüft werden.

Bei Seiten mit veränderlichem Inhalt stehen je nach Anzahl der Anzeigewerte unterschiedlich viele Eingabefelder zur Verfügung. Darüber können die anzuzeigenden Daten ausgewählt werden.

OBP60 Page 4		
Type	FourValues	X ?
Field 1	AWA	X ?
Field 2	DBT	X ?
Field 3	COG	X ?
Field 4	SOG	X ?

Abb.: Seite mit 4 Anzeigewerten

- **Datenpool ausählbarer Daten**
 - **ALT** - Altitude, Höhe über Grund

- **AWA** - Apparent Wind Angle, scheinbare Windrichtung
- **AWS** - Apparent Wind Speed, scheinbare Windgeschwindigkeit
- **BTW** - Bearing To Waypoint, Winkel zum aktuellen Wegpunkt
- **COG** - Course over Ground, Kurs über Grund
- **DBS** - Depth Below Surface, Tiefe unter Wasseroberfläche
- **DBT** - Depth Below Transducer, Tiefe unter Sensor
- **DEV** - Deviation, Kursabweichung
- **DTW** - Distance To Waypoint, Entfernung zum aktuellen Wegpunkt
- **GPSD** - GPS Date, GPS-Datum
- **GPDT** - GPS Time, GPS-Zeit als UTC (Weltzeit)
- **HDM** - Magnetic Heading, magnetischer Kurs
- **HDT** - Heading, wahrer rechtweisender Kurs
- **HDOP** - GPS-Genauigkeit in der Horizontalen
- **LAT** - Latitude, geografische Breite
- **LON** - Longitude, geografische Höhe
- **Log** - Log, Entfernung
- **MaxAws** - Maximum Apparent Wind Speed, Maximum der relativen Windgeschwindigkeit seit Gerätetestart
- **MaxTws** - Maximum True Wind Speed, Maximum der wahren Windgeschwindigkeit seit Gerätetestart
- **PDOP** - GPS-Genauigkeit über alle 3 Raumachsen
- **PRPOS** - Auslenkung Sekundärruder
- **ROT** - Rotation, Drehrate
- **RPOS** - Rudder Position, Auslenkung Hauptruder
- **SOG** - Speed Over Ground, Geschwindigkeit über Grund
- **STW** - Speed Through Water, Geschwindigkeit durch das Wasser
- **SatInfo** - Satellit Info, Anzahl der sichtbaren Satelliten
- **TWD** - True Wind Direction, wahre Windrichtung
- **TWS** - True Wind Speed, wahre Windgeschwindigkeit
- **TZ** - Time Zone, Zeitzone
- **TripLog** - Trip Log, Tages-Entfernungszähler
- **VAR** - Variation, Abweichung vom Sollkurs
- **VDOP** - GPS-Genauigkeit in der Vertikalen
- **WPLat** - Waypoint Latitude, geogr. Breite des Wegpunktes
- **WPLon** - Waypoint Longitude, geogr. Länge des Wegpunktes
- **WTemp** - Water Temperature, Wassertemperatur
- **XTE** - Cross Track Error, Kursfehler

- **xdrVBat** - Bordspannung

6.17.1 OneValue



Abb.: Anzeige OneValue

Bei der OneValue-Anzeige kann ein beliebiger Messwert aus dem Datenpool angezeigt werden. Neben dem Messwert werden der Kurzbezeichner und die Einheit dargestellt.

6.17.2 TwoValue



Abb.: Anzeige TwoValue

Bei der TwoValue-Anzeige können zwei beliebige Messwerte aus dem Datenpool vertikal übereinander angezeigt werden. Neben den Messwerten werden die Kurzbezeichner und die Einheiten dargestellt.

6.17.3 ThreeValue



Abb.: Anzeige ThreeValue

Bei der ThreeValue-Anzeige können drei beliebige Messwerte aus dem Datenpool vertikal übereinander angezeigt werden. Neben den Messwerten werden die Kurzbezeichner und die Einheiten dargestellt.

6.17.4 FourValue

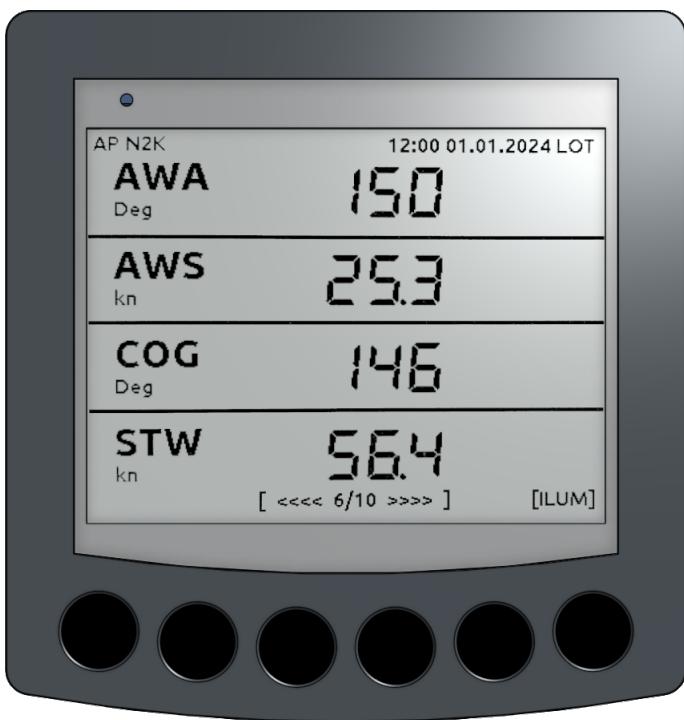


Abb.: Anzeige FourValue

Bei der ThreeValue-Anzeige können vier beliebige Messwerte aus dem Datenpool vertikal übereinander angezeigt werden. Neben den Messwerten werden die Kurzbezeichner und die Einheiten dargestellt.

6.17.5 FourValue2



Abb.: Anzeige FourValue

Bei der FourValue-Anzeige können vier beliebige Messwerte aus dem Datenpool vertikal übereinander und horizontal nebeneinander angezeigt werden. Neben den Messwerten werden die Kurzbezeichner und die Einheiten angezeigt. Diese Darstellung entspricht der alten Darstellung vom Raymarine ST60 TriData mit dem Unterschied, dass hier beliebige Werte angezeigt werden können. Es gibt noch die Anzeigeseite **DST810** mit festen Inhalten, die die gleichen Messwerte anzeigt wie beim ST60 TriData.

6.17.6 Voltage



Abb.: Anzeige Voltage

Bei der Voltage-Anzeige wird die Versorgungsspannung der Batterie angezeigt, wie sie am Eingang von CN2 zur Verfügung gestellt wird.

Bemerkung: Beachten Sie, dass die Spannung nicht exakt der Batteriespannung entsprechen muss. Durch Leitungsverluste können Spannungsabfälle auftreten, und der gemessene Wert kann kleiner sein als die tatsächliche Batteriespannung.

Ein Trendindikator zeigt den Trend an, in welche Richtung sich die Spannung bewegt. Hinter der Einheit Volt werden der Batterietyp [Pb|AGM|Gel|LiFePo4] und die aktuell benutzte Mittelungstiefe angezeigt. Über die Tasten können folgende Funktionen genutzt werden.

- [AVG] - Einstellung der Mittelungstiefe in Sekunden [1|30|60|300]
- [TRD] - Trendanzeige aktivieren oder deaktivieren

Die Anzeigeseite benötigt folgende Messwerte: **xdrVBat**

6.17.7 WindRose

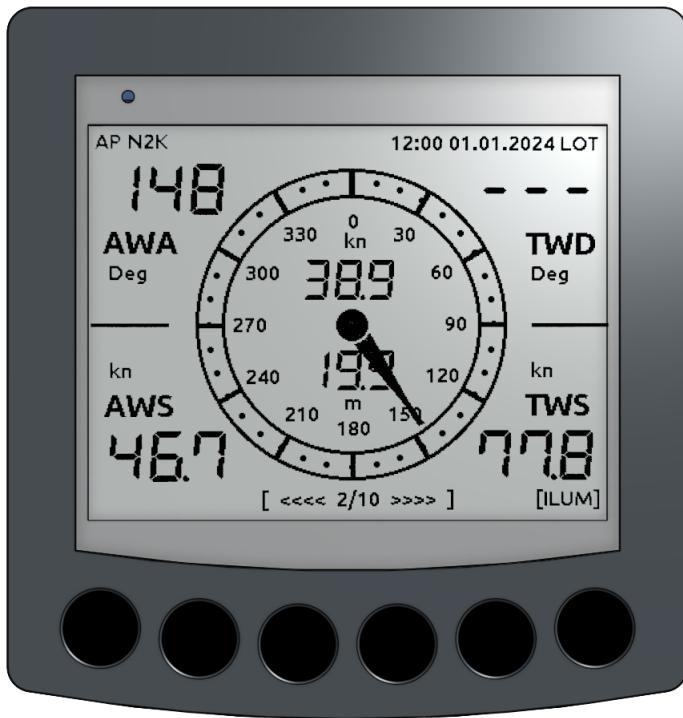


Abb.: Anzeige Windrose

Bei der Windrosen-Anzeige werden Winddaten angezeigt. Auf der linken Seiten sind die Daten des scheinbaren Windes dargestellt und auf der rechten Seite die Daten des wahren Windes. Die Daten des scheinbaren Windes beziehen sich auf den auf dem fahrenden Schiff wahrgenommenen Wind, der sich aus dem Zusammenwirken des wahren Windes und des Fahrtwindes ergibt. Es handelt sich um relative Daten bezogen auf das Boot. Die Daten des wahren Windes sind die Winddaten, wie man sie am nicht in Fahrt befindlichen Boot messen würde. Der Windwinkel bezieht sich dabei auf den Bug, die Windrichtung auf die geografische Nordausrichtung.

In der Mitte der Windrose wird die aktuelle Geschwindigkeit durchs Wasser und die Wassertiefe unter dem Sensor angezeigt.

Die Anzeigeseite benötigt folgende Messwerte: **AWA, AWS, TWD, TWS, DBT, STW**

6.17.8 WindRoseFlex



Abb.: Anzeige WindroseFlex

Bei dieser Variante der Anzeige WindRose können die darzustellenden Werte frei gewählt werden. Der erste Wert wird auf der Windrose grafisch als Richtung dargestellt, hier ist sinnvollerweise AWA oder TWA zu wählen.

6.17.9 DST810



Abb.: Anzeige FourValue

Bei der DST810-Anzeige werden der Speed durchs Wasser, die Tiefe, die zurückgelegte Strecke und die Wassertemperatur angezeigt. Neben den Messwerten werden die Kurzbezeichner und die Einheiten dargestellt. Die Anzeigeseite entspricht der alten Darstellung vom **Raymarine ST60 TriData**. Damit die Daten angezeigt werden können, müssen sich gültige Informationen im Datenpool befinden. Neben dem DST810 von Airmar können auch Messwerte anderer Sensorhersteller angezeigt werden, die dieselben Daten oder einen Teil der Daten liefern

Die Anzeigeseite benötigt folgende Messwerte: **DBT, STW, Log, WTemp**

6.17.10 Clock



Abb.: Anzeige Clock

Bei der Clock-Anzeige werden die Uhrzeit, das Datum, die Sonnenaufgangszeit und die Sonnenuntergangszeit angezeigt. Die Anzeigewerte werden primär aus den GPS-Daten gewonnen. Die Auf- und Untergangszeit der Sonne wird abhängig vom geografischen Ort berechnet und entspricht der astronomischen Sonnenaufgangs- und Untergangszeit. Als Zeitanzeige kann die globale Weltzeit **UTC** oder die lokale Ortszeit **LOT** angezeigt werden. Die Auswahl der Zeitzone kann über die Konfigurationsseite **Config - OBP Settings** eingestellt werden.

Die Einstellung der Uhrzeit erfolgt automatisch über die GPS-Zeit. Stellen Sie vor der Benutzung des OBP60 sicher, dass ein GPS-Empfang möglich ist, damit sich die Zeit einstellen kann. In regelmäßigen Abständen wird die RTC-Zeit mit der GPS-Zeit synchronisiert, so dass Sie auch über Zeitinformationen verfügen, wenn kein GPS-Empfang möglich ist.

Bemerkung: Stehen keine GPS-Daten zur Verfügung, so wird die Zeit und das Datum aus der RTC benutzt. In dem Fall stehen keine Sonnenaufgangszeit und Sonnenuntergangszeit zur Verfügung, da die geografischen Ortsdaten fehlen.

Die Anzeigeseite benötigt folgende Messwerte: **GPST, GPSD**

6.17.11 WhitePage



Abb.: Anzeige WhitePage

Bei WhitePage handelt es sich um eine Anzeigeseite, die nur eine weiße leere Seite darstellt. Diese Seite kann dazu benutzt werden, den Bildschirminhalt vor dem Ausschalten definiert zu löschen.

6.17.12 BME280



Abb.: Anzeige BME280

Bei der BME-Anzeige werden die 3 Messwerte Lufttemperatur, Luftdruck und Luftfeuchtigkeit des BME280 angezeigt. Der BME280 muss dazu an den externen I2C-Bus angeschlossen werden und auf die Adresse 0x77 eingestellt sein.

Warnung: Bedenken Sie, dass der externe I2C-Bus **5V** Signalpegel für **SCL** und **SDA** benutzt. Benutzen Sie solche Module, die tolerant für 5V sind, oder verwenden Sie Pegelumsetzer von 5V auf 3.3V für die Signale SCL und SDA. Beachten Sie das nicht, so können die externen Module beschädigt werden oder nur fehlerbehaftet arbeiten.

Ein 5V taugliches BME280-Modul ist das **GYBME** Elektronikmodul:

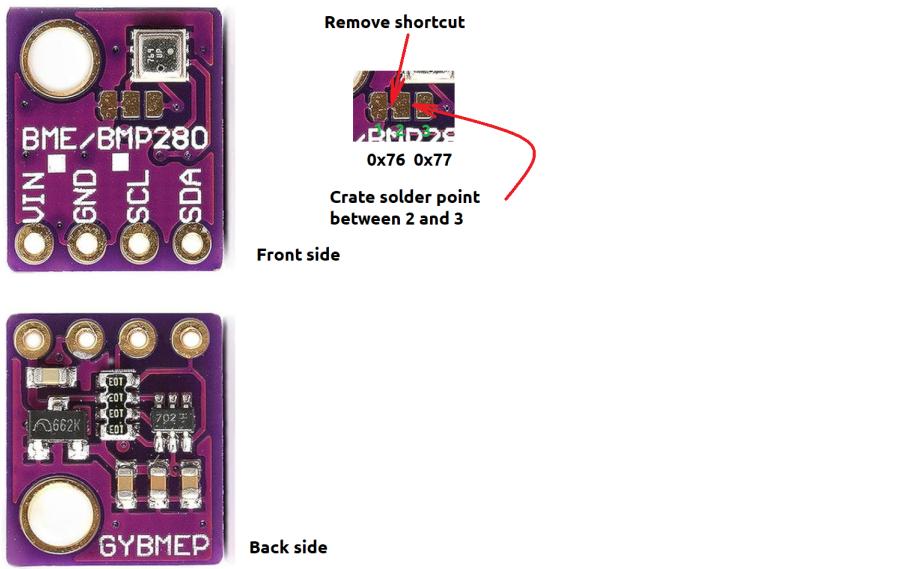


Abb.: BME280-Modul

Die Messwerte vom externen Sensor müssen als XDR-Telegramme angelegt werden (siehe Konfigurationsseite: **XDR**). Dabei sind folgende Zuordnungen zu beachten:

- **TAir** - Lufttemperatur
- **PAir** - Luftdruck
- **HAir** - Luftfeuchtigkeit

6.17.13 Rudder



Abb.: Anzeige Rudder

Bei der Rudder-Anzeige wird der Ruderausschlag angezeigt. Der Ruderausschlag ist im Bereich von +/-45° grafisch darstellbar. Wenn keine Sensorwerte für den Ruderausschlag vorliegen, ist der Zeiger nicht sichtbar.

Hinweis: Die Ruderanzeige kann sowohl für Daten aus NMEA0183 , NMEA2000 und einem I2C-Rotationssensor benutzt werden.

Die Anzeigeseite benötigt folgende Messwerte: **RPOS**

6.17.14 Keel



Abb.: Anzeige Keel

Bei der Keel-Anzeige wird die Kielstellung eines Neigekiels angezeigt. Die Kielstellung ist im Bereich von +/-45° grafisch darstellbar. Wenn keine Sensorwerte für die Kielstellung vorliegen, ist der Kiel nicht sichtbar.

Damit die Kielstellung angezeigt werden kann, muss ein Rotationssensor-Modul **AS5600** am I2C-Bus angeschlossen und der Sensor als **Kielsensor** auf der Konfigurationsseite **Config - OBP Hardware** parametriert werden.

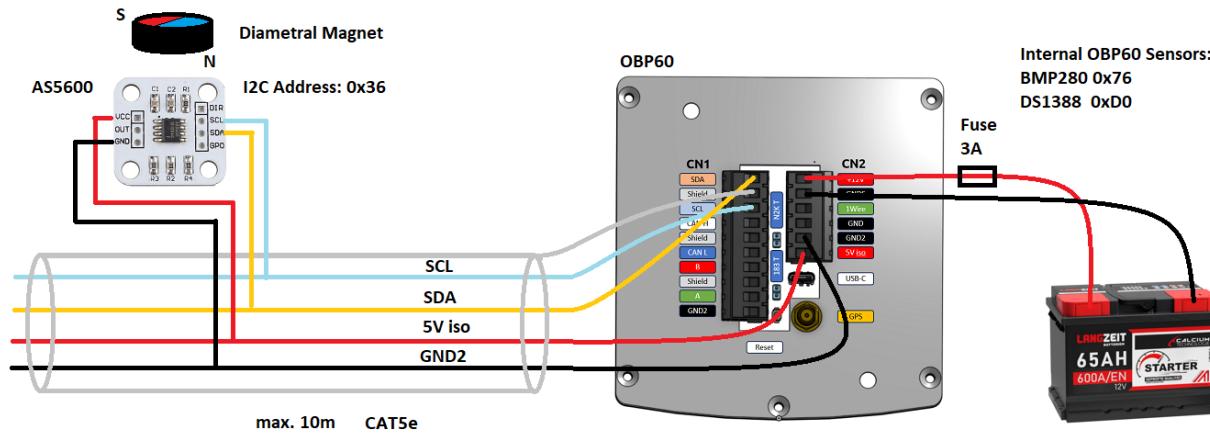


Abb.: Magnetischer Rotationssensor AS5600 zur Anzeige der Kielstellung

Beachten Sie auch die Hinweise im Kapitel **Datenaustausch - I2C-Bus** und **Bussysteme - I2C**.

Hinweis: Die Kielanzeige kann nur in Verbindung mit einem I2C-Rotationssensor benutzt werden.

6.17.15 Battery



Abb.: Anzeige Battery

Bei der Battery-Anzeige werden die aktuellen Werte für Bord-Spannung, Strom und Leistung angezeigt. Neben den Messwerten werden die Kurzbezeichner und die Einheiten dargestellt. Um die Batterie-Werte anzeigen zu können, muss ein I2C-Modul **INA226** am I2C-Bus angeschlossen und auf die Adresse **0x41** eingestellt sein. Der Shunt kann für verschiedene maximale Stromstärken in Ampere [10|50|100|200|300|400|500] unter **Config - OBP Hardware** konfiguriert werden.

Hinweis: Bedenken Sie, dass für höhere Stromstärken die Ungenauigkeit der Messwerte zunimmt. Wählen Sie den Shunt so aus, dass er zu typischen Nutzungsszenarien passt und nicht überdimensioniert ist. Die Messeingänge des Shunts sind bis zum zweifachen Wert der Maximalstromstärke eigensicher und vertragen kurzzeitige Überlastungen.

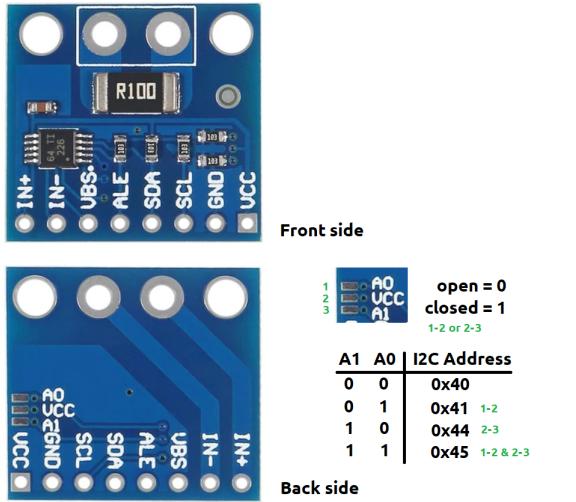


Abb.: I2C-Adresszuweisung INA226

Für die Messung mit einem externen Leistungs-Shunt muss der schwarze große Widerstand **R100** auf der Frontseite der Platine entfernt werden. Danach ist das Modul wie folgt zu verschalten.

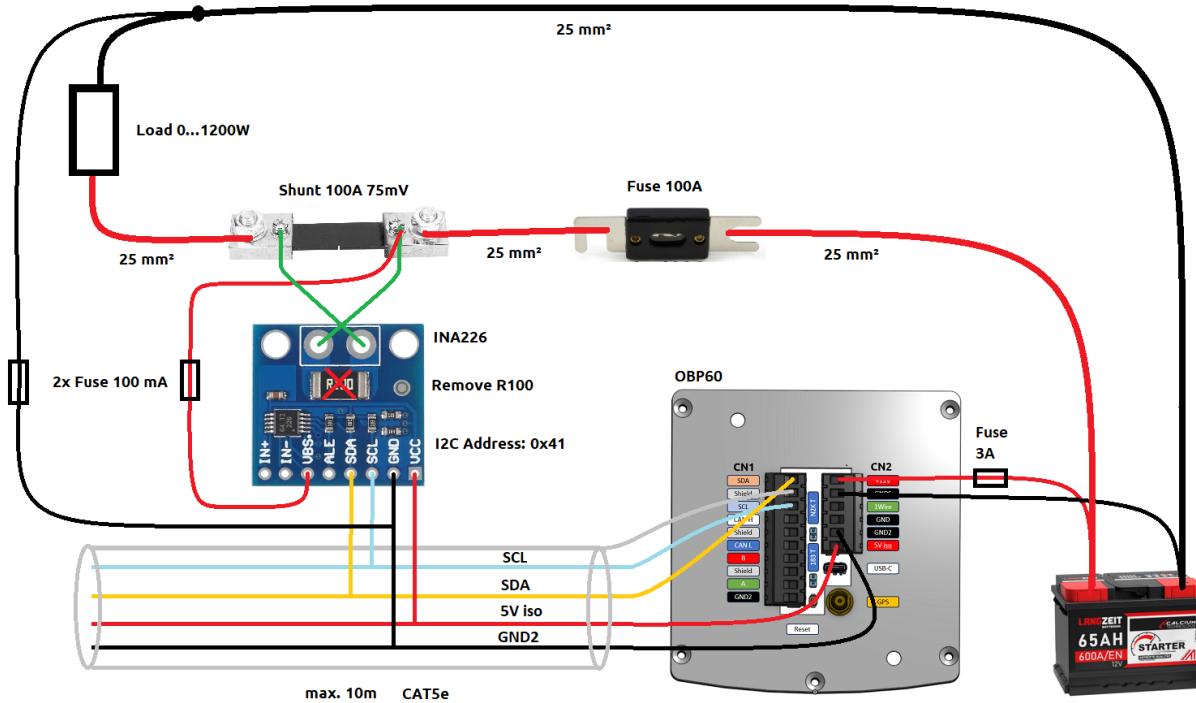


Abb.: Schaltung INA226 Batteriemonitoring

Bemerkung: Wenn Sie die Batterieanzeige verwenden, jedoch kein INA226-Modul am I2C-Bus angeschlossen ist, werden keine Messwerte angezeigt.

Warnung: Verwenden Sie für den Leistungskreis ausreichend groß dimensionierte Leitungsquerschnitte, die auf den maximalen Strom ausgelegt sein müssen. Verwenden Sie in den Leistungskreisen passende Sicherungen, um Kabelbrände bei Kurzschlägen zu vermeiden. Für eine langlebige Installation sollten Sie Litze mit verzinnten Einzeladern verwenden. Wenn das aus Kostengründen nicht möglich ist, sollten die Kabelenden mit gequetschten Kabelösen oder Aderendhülsen versehen sein. Die Kabelösen sollten dann zusätzlich mit Zinn verlötet werden, um Korrosion in den Kabelhülsen zu unterbinden. Ein Überzug der Chrimp- und Lötstellen mit Schrumpfschlauch verhindert aufsteigende Feuchtigkeit im Kabel, die ebenso Korrosion über lange Zeiträume verursachen kann. Sorgen Sie dafür, dass der INA226 wassergeschützt in einem isolierten Gehäuse untergebracht ist und die Sensoranschlüsse **VBS** und **GND** mit einer **Feinsicherung 100 mA** geschützt sind. Wenn Sie nicht über ausreichendes Fachwissen verfügen, sollten Sie die Installation des Sensors einem Fachmann überlassen oder ihre Installation vor der Inbetriebnahme durch einen Fachmann prüfen lassen.

Gefahr: Unsachgemäße oder defekte Installationen von Leistungsstromkreisen können Brände verursachen und Leben gefährden. Prüfen Sie die Installation in regelmäßigen Abständen hinsichtlich Funktion und Sicherheit.

Referenzverlegeart	A1		A2		B1		B2		C		E		F		G		
Verlegung	in Wärmedämmten Wänden				in Elektro-Installationsrohren				auf einer Wand		frei in Luft						
Anzahl der belasteten Adern	2	3	2	3	2	3	2	3	2	3	2	3	2 vert., horizont.	3 vert., horizont.	3 gebündelt	3 horizont.	3 vert.
Nennquerschnitt Cu in mm ²	Strombelastbarkeit in A																
Umgebungstemperatur 30 °C																	
1,5	15,5 ¹⁾	13,5	15,5 ¹⁾	13,0	17,5	15,5	16,5	15	19,5	17,5	22	18,5	-	-	-	-	
2,5	19,5	18	18,5	17,5	24	21	23	20	27	24	30	25	-	-	-	-	
4	26	24	25	23	32	28	30	27	36	32	40	34	-	-	-	-	
4	-	-	-	-	-	-	-	-	-	33 ²⁾	-	-	-	-	-	-	
6	34	31	32	29	41	36	38	34	46	41	51	43	-	-	-	-	
10	46	42	43	39	57	50	52	46	63	57	70	60	-	-	-	-	
10	-	-	-	-	-	-	-	47 ²⁾	-	59 ²⁾	-	-	-	-	-	-	
16	61	56	57	52	76	68	69	62	85	76	94	80	-	-	-	-	
25	80	73	75	68	101	89	90	80	112	96	119	101	131	114	110	146	
35	99	89	92	83	125	110	111	99	138	119	148	126	162	143	137	181	
50	119	108	110	99	151	134	133	118	168	144	180	153	196	174	167	219	
70	151	136	139	125	192	171	168	149	213	184	232	196	251	225	216	281	
95	182	164	167	150	232	207	201	179	258	223	282	238	304	275	264	341	
Umgebungstemperatur 25 °C																	
1,5	16,5 ¹⁾	14,5	16,5 ¹⁾	14,0	18,5	16,5	17,5	16	21	18,5	23	19,5	-	-	-	-	
2,5	21	19	19,5	18,5	25	22	24	21	29	25	32	27	-	-	-	-	
4	28	25	27	24	34	30	32	29	38	34	42	36	-	-	-	-	
4	-	-	-	-	-	-	-	-	35 ²⁾	-	-	-	-	-	-	-	
6	36	33	34	31	43	38	40	36	49	43	54	46	-	-	-	-	
10	49	45	46	41	60	53	55	49	67	60	74	64	-	-	-	-	
10	-	-	-	-	-	-	-	50 ²⁾	-	63 ²⁾	-	-	-	-	-	-	
16	65	59	60	55	81	72	73	66	90	81	100	85	-	-	-	-	
25	85	77	80	72	107	94	95	85	119	102	126	107	139	121	117	155	
35	105	94	98	88	133	117	118	105	146	126	157	134	172	152	145	192	
50	126	114	117	105	160	142	141	125	178	153	191	162	208	184	177	232	
70	160	144	147	133	204	181	178	158	226	195	246	208	266	239	229	298	
95	193	174	177	159	246	219	213	190	273	236	299	252	322	292	280	330	

¹⁾ Bei Wandaufbau: Äußere Beplankung 10 mm Holzfaserplatten, Wärmedämmung mit 100 mm Mineralfaser, innere Beplankung mit 25 mm Holzfaserplatte mit Wärmeleitfähigkeit 0,1 W/K m senkrecht und 0,23 W/K m parallel zur Plattenebene

²⁾ Gilt nicht für Verlegung auf einer Holzwand

Abb.: Leitungsquerschnitte (EP 12/00)

Zu weitergehenden Informationen können Sie das Informationsmaterial **Leitungen und Kabel** pdf verwenden.

6.17.16 Battery2

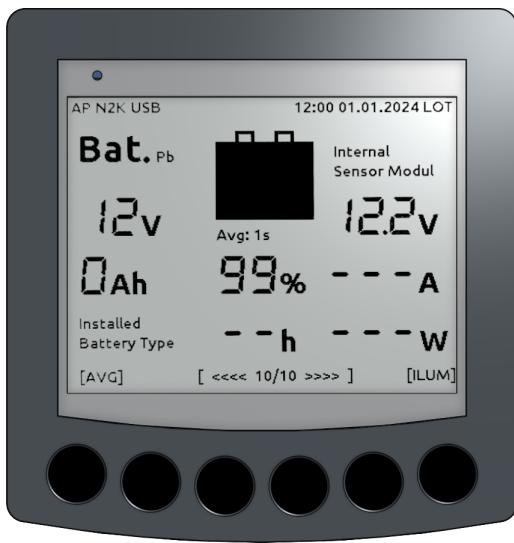


Abb.: Anzeige Battery2

Bei der Battery2-Anzeige werden folgende Werte angezeigt:

- Batterietyp [Pb|Gel|AGM|LiFePo4]
- Nenn-Batteriespannung in V
- Nenn-Batteriekapazität in Ah
- Grafische Füllstandsanzeige in %
- Geschätzte Reichweite in Stunden bei aktuellen Verbrauchswerten
- Art des Sensormoduls [interner Sensor|INA219|INA226]
- Aktuelle Batteriespannung in V
- Aktueller Stromverbrauch in A
- Aktuelle Leistung in W

Über die Tasten können folgende Funktionen genutzt werden.

- [AVG] - Einstellung der Mittelungstiefe in Sekunden [1|30|60|300]

Warnung: Die Reichweitenanzeige gibt einen ungefähren Zeitwert an, wie lange die Batterie mit den aktuellen Verbrauchswerten Energie liefern wird. Die Zeitspanne ist abhängig vom aktuellen Stromverbrauch und passt sich kontinuierlich an. Die Batteriespannung wird zur Reichweitenbestimmung benutzt und damit der Füllstand der Batterie ermittelt. Diese Methode ist nicht sehr genau und vom Alterungszustand der Batterie abhängig. Prüfen Sie in unkritischen Situationen die Genauigkeit der Reichweitenanzeige und planen Sie entsprechende Sicherheitsreserven ein, um keine unerwarteten Ausfälle zu riskieren.

Hinweis: Nutzen Sie eine große Mittelungszeit über die Taste [AVG] von 300s, um eine realistische Reichweitenanzeige zu bekommen. Dadurch werden Lastspitzen im Stromverbrauch geglättet und der Reichweitenwert ist deutlich ruhiger.

Um die Batterie-Werte anzeigen zu können, muss ein I2C-Modul **INA226** am I2C-Bus angeschlossen und auf die Adresse **0x41** eingestellt sein. Der Shunt kann für verschiedene maximale Stromstärken in Ampere [10|50|100|200|300|400|500] unter **Config - OBP Hardware** konfiguriert werden.

Hinweis: Bedenken Sie, dass für höhere Stromstärken die Ungenauigkeit der Messwerte zunimmt. Wählen Sie den Shunt so aus, dass er zu typischen Nutzungsszenarien passt und nicht überdimensioniert ist. Die Messeingänge des Shunts sind bis zum zweifachen Wert der Maximalstromstärke eigensicher und vertragen kurzzeitige Überlastungen.

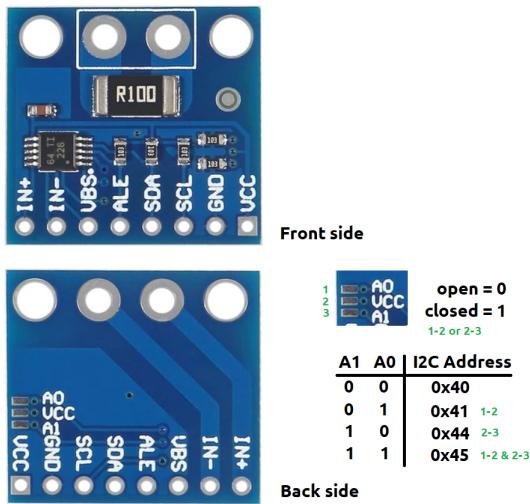


Abb.: I2C-Adresszuweisung INA226

Für die Messung mit einem externen Leistungs-Shunt muss der schwarze große Widerstand **R100** auf der Frontseite der Platine entfernt werden. Danach ist das Modul wie folgt zu verschalten.

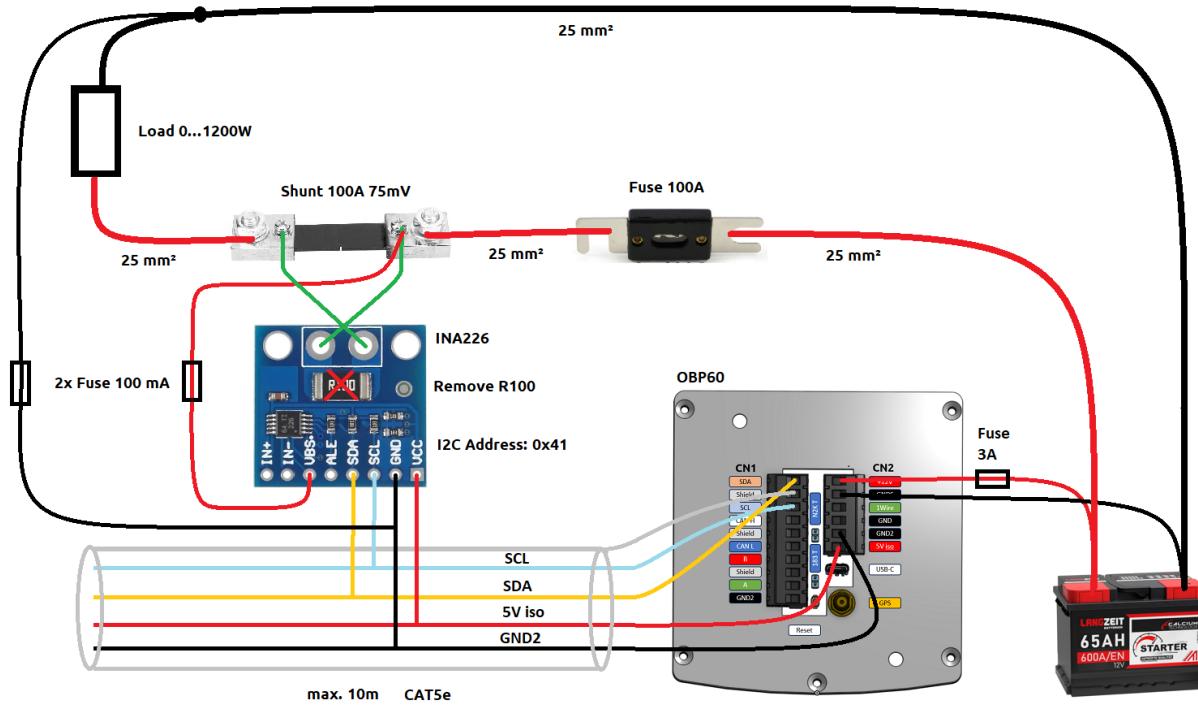


Abb.: Schaltung INA226 Batteriemonitoring

Bemerkung: Wenn Sie die Batterieanzeige verwenden, jedoch kein INA226-Modul am I2C-Bus angeschlossen ist, werden keine Messwerte außer der aktuellen Batteriespannung angezeigt. In diesem Fall wird der interne Spannungssensor des OBP60 benutzt. Der Wert der gemessenen Spannung muss nicht direkt der Spannung an der Batterie entsprechen, da durch Leitungsverluste der Spannungswert verfälscht sein kann.

Warnung: Die Gefahren und Risiken bei der Benutzung des INA226 zum Batteriemonitoring sind die selben wie im Kapitel **Battery** beschrieben. Folgen Sie den Empfehlungen und beachten Sie die Gefahren.

6.17.17 RollPitch

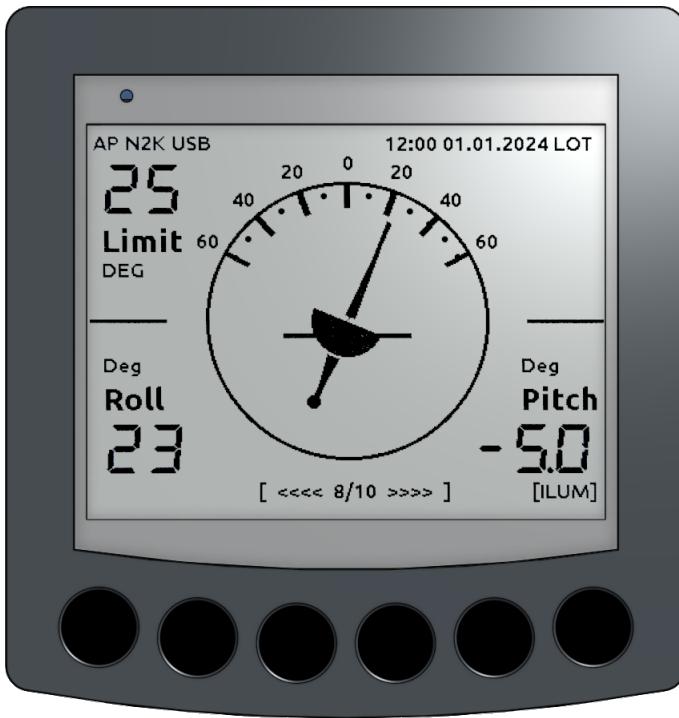


Abb.: Anzeige RollPitch

Bei der RollPitch-Anzeige werden die aktuellen Werte für Roll und Pitch angezeigt sowie der Grenzwert, ab dem ein optisches Signal über die Flash-LED ausgegeben wird. Pitch entspricht der Neigung in Längsrichtung und Roll in Querrichtung des Bootes. Die Sensorwerte müssen als XDR-Telegramme in Form von **xdrRoll** und **xdrPitch** angelegt werden (siehe Konfigurationsseite: [XDR](#)) . Dabei sind folgende Zuordnungen zu beachten:

- **Roll** - Neigung in Querrichtung
- **Pitch** - Neigung in Längsrichtung

6.17.18 Solar

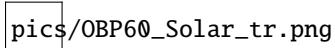


Abb.: Anzeige Solar

Bei der Solar-Anzeige werden folgende Werte angezeigt:

- Nenn-Spannung der Solarmodule in V
- Nenn-Leistung der Solarmodule in W
- Nutzungsgrad in %
- Art des Sensormoduls [interner Sensor|INA219|INA226]
- Aktuelle Solarspannung in V
- Aktueller Solareinspeisung in A
- Aktuelle Einspeiseleistung in W

Über die Tasten können folgende Funktionen genutzt werden.

- [AVG] - Einstellung der Mittelungstiefe in Sekunden [1|30|60|300]

Um die Messwerte anzeigen zu können, muss ein I2C-Modul **INA226** am I2C-Bus angeschlossen und auf die Adresse **0x44** eingestellt sein. Der Shunt kann für verschiedene maximale Stromstärken in Ampere [10|50|100|200|300|400|500] unter **Config - OBP Hardware** konfiguriert werden.

Hinweis: Bedenken Sie, dass für höhere Stromstärken die Ungenauigkeit der Messwerte zunimmt. Wählen Sie den Shunt so aus, dass er zu typischen Nutzungsszenarien passt und nicht überdimensioniert ist. Die Messeingänge des Shunts sind bis zum zweifachen Wert der Maximalstromstärke eigensicher und vertragen kurzzeitige Überlastungen.

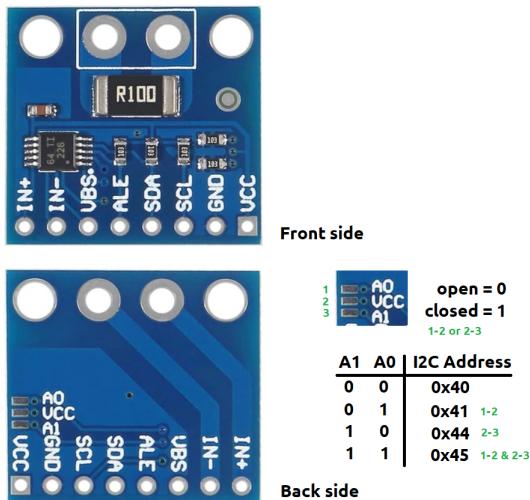


Abb.: I2C-Adresszuweisung INA226

Für die Messung mit einem externen Leistungs-Shunt muss der schwarze große Widerstand **R100** auf der Frontseite der Platine entfernt werden. Danach ist das Modul wie folgt zu verschalten.

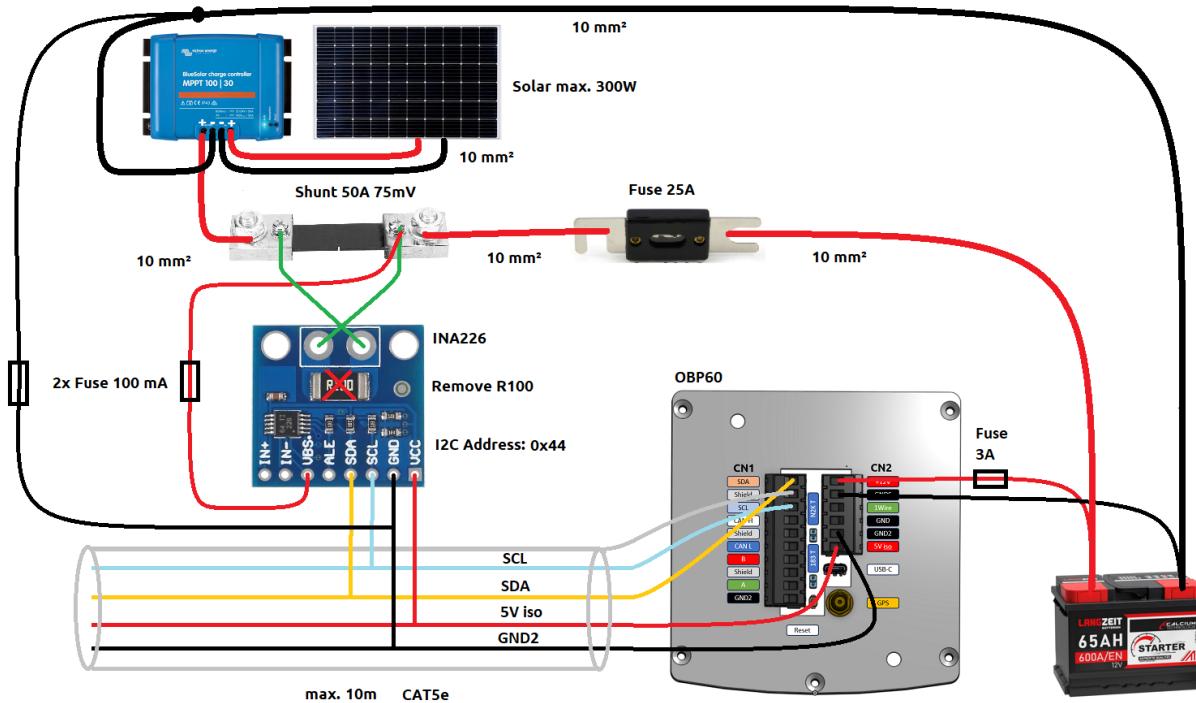


Abb.: Schaltung INA226 Solarmonitoring

Bemerkung: Wenn Sie die Solaranzeige verwenden, jedoch kein INA226-Modul am I2C-Bus angeschlossen ist, werden keine Messwerte außer die aktuelle Batteriespannung angezeigt. In diesem Fall wird der interne Spannungssensor des OBP60 benutzt. Der Wert der gemessenen Spannung muss nicht direkt der Spannung am Ausgang des Wechselrichters entsprechen, da durch Leitungsverluste der Spannungswert verfälscht sein kann.

Warnung: Die Gefahren und Risiken bei der Benutzung des INA226 zum Batteriemonitoring sind dieselben wie im Kapitel **Battery** beschrieben. Folgen Sie den Empfehlungen und beachten Sie die Gefahren.

6.17.19 Generator

pics/OPB60_Generator_tr.png

Abb.: Anzeige Generator

Bei der Generator-Anzeige werden folgende Werte angezeigt:

- Nenn-Spannung des Generators in V
- Nenn-Leistung des Generators in W
- Nutzungsgrad in %
- Art des Sensormoduls [interner Sensor|INA219|INA226]
- Aktuelle Generatorspannung in V
- Aktuelle Generatoren einspeisung in A

- Aktuelle Generatorleistung in W

Über die Tasten können folgende Funktionen genutzt werden.

- [AVG] - Einstellung der Mittelungstiefe in Sekunden [1|30|60|300]

Um die Messwerte anzeigen zu können, muss ein I2C-Modul **INA226** am I2C-Bus angeschlossen und auf die Adresse **0x45** eingestellt sein. Der Shunt kann für verschiedene maximale Stromstärken in Ampere [10|50|100|200|300|400|500] unter **Config - OBP Hardware** konfiguriert werden.

Hinweis: Bedenken Sie, dass für höhere Stromstärken die Ungenauigkeit der Messwerte zunimmt. Wählen Sie den Shunt so aus, dass er zu typischen Nutzungszenarien passt und nicht überdimensioniert ist. Die Messeingänge des Shunts sind bis zum zweifachen Wert der Maximalstromstärke eigensicher und vertragen kurzzeitige Überlastungen.

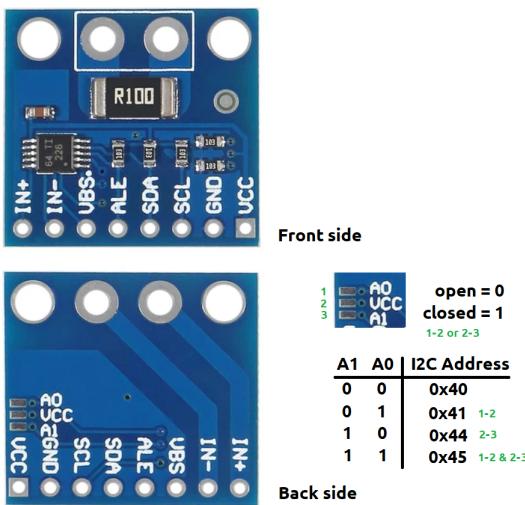


Abb.: I2C-Adresszuweisung INA226

Für die Messung mit einem externen Leistungs-Shunt muss der schwarze große Widerstand **R100** auf der Frontseite der Platine entfernt werden. Danach ist das Modul wie folgt zu verschalten.

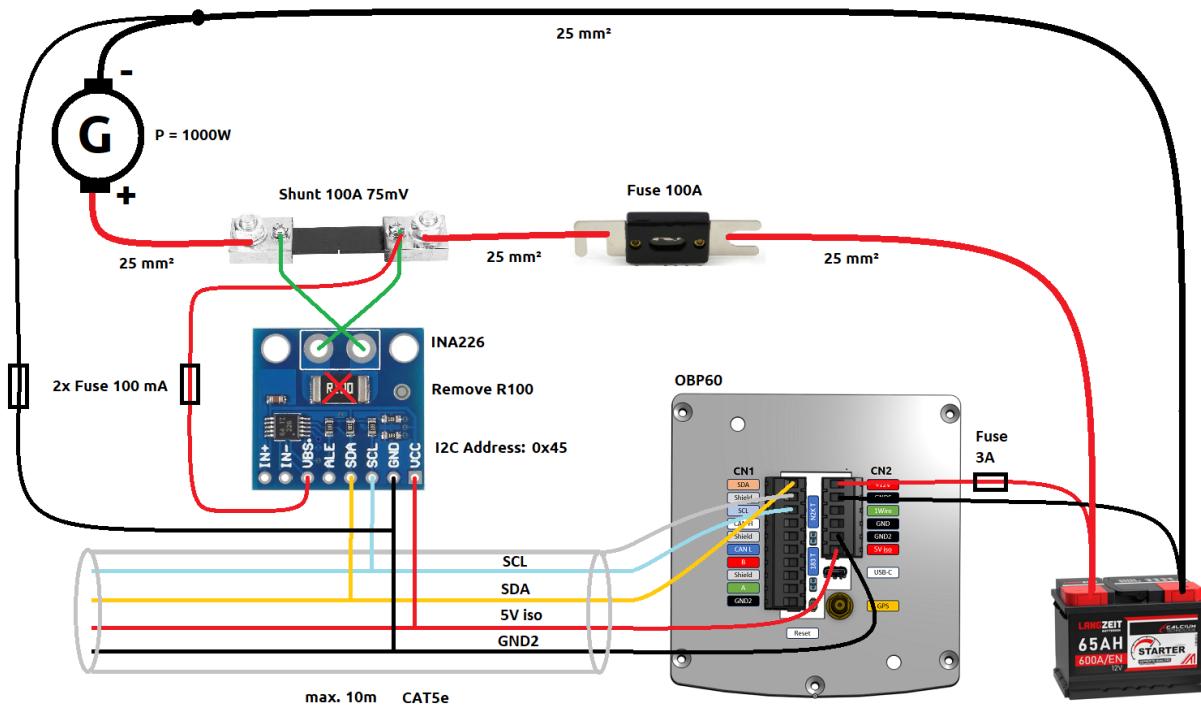


Abb.: Schaltung INA226 Generatormonitoring

Bemerkung: Wenn Sie die Generatoranzeige verwenden, jedoch kein INA226-Modul am I2C-Bus angeschlossen ist, werden keine Messwerte außer die aktuelle Batteriespannung angezeigt. In diesem Fall wird der interne Spannungssensor des OBP60 benutzt. Der Wert der gemessenen Spannung muss nicht direkt der Spannung am Ausgang des Generators entsprechen, da durch Leitungsverluste der Spannungswert verfälscht sein kann.

Warnung: Die Gefahren und Risiken bei der Benutzung des INA226 zum Batteriemonitoring sind dieselben wie im Kapitel **Battery** beschrieben. Folgen Sie den Empfehlungen und beachten Sie die Gefahren.

6.18 XDR

Über die Konfigurationsseite XDR können XDR-Sentences für NMEA0183 erstellt werden. XDR-Sentences sind Telegramme für generische Sensorwerte, die verwendet werden, wenn sich kein geeignetes NMEA0183-Telegramm findet, mit dem man die gewünschten Sensorwerte übertragen kann. Es ist ein universelles Telegramm zur Übertragung von Sensordaten. Sofern nicht zugewiesene Sensordaten im OBP60 vorhanden sind, können diese über ein XDR-Mapping zugewiesen werden. Damit sind diese Daten als NMEA0183-Telegramme allgemein nutzbar und werden im OBP60 dargestellt. Die Daten lassen sich dann auch über NMEA0183 in andere Systeme übertragen und dort nutzen. XDR-Sentences werden immer dann benutzt, wenn Daten aus dem I2C-Bus, dem 1Wire-Bus oder interne Sensordaten vom ESP32 übertragen werden sollen.

Ein XDR-Sentence ist folgendermaßen aufgebaut:

Sensor Werte

\$-XDR,a,x.x,b,c-c,x-x*hh<CR><LF>

Feldnummer:

- a - Sensor-Typ
- x.x - Messwert
- b - Einheit des Messwertes
- c - Name des Sensors
- x - Weitere Sensordaten
- hh - Checksumme

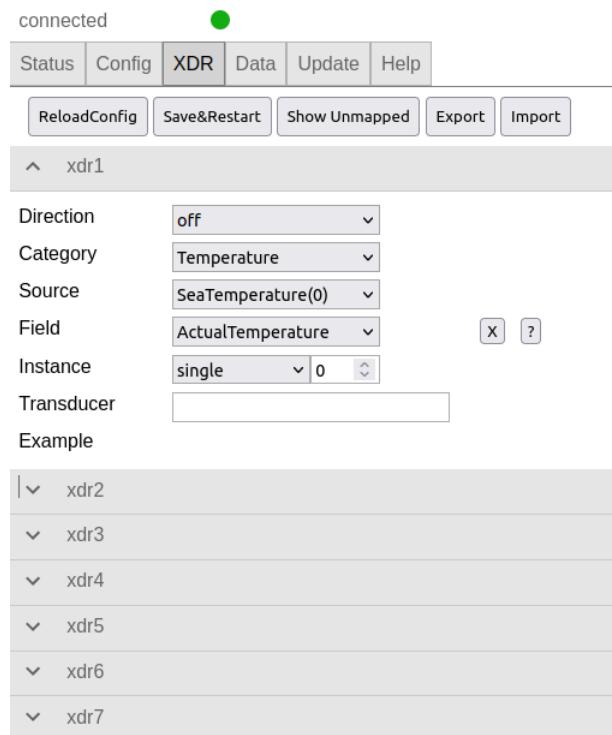
Beispiele:

- \$IIXDR,C,19.52,C,TempAir*19
- \$IIXDR,P,1.02481,B,Barometer*29

Messwert	Sensor-Typ	Beispiele für Messdaten	Einheit	Name des Sensors
Luftdruck	P Druck	0.8..1.1 oder 800..1100	B Bar	Barometer
Lufttemperatur	C Temperatur	2 Dezimalstellen	C Celsius	TempAir oder ENV_OUTAIR_T
Pitch	A Winkel	-180..0 abwärts 0..180 aufwärts	D Degrees	PTCH oder PITCH
Rolling	A Winkel	-180..0 links 0..180 rechts	D Degrees	ROLL
Wassertemperatur	C Temperatur	2 Dezimalstellen	C Celsius	ENV_WATER_T

Über die XDR-Konfigurationsseite lassen sich 30 XDR-Telegramme individuell erstellen.

OBP60V2



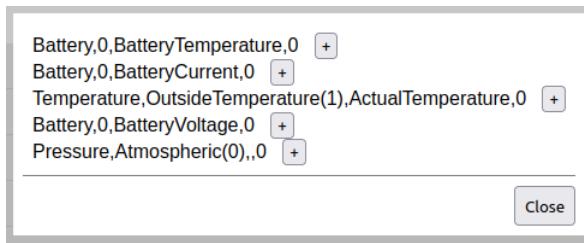
The screenshot shows the OBP60V2 configuration interface with the XDR tab selected. The main area displays configuration parameters for XDR1:

- Direction: off
- Category: Temperature
- Source: SeaTemperature(0)
- Field: ActualTemperature
- Instance: single 0
- Transducer: (empty input field)

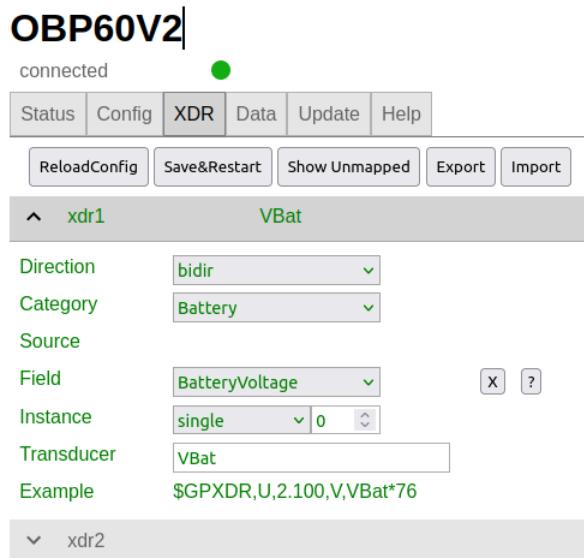
Below the configuration fields is a tree view of XDR configurations:

- xdr1 (selected)
- xdr2
- xdr3
- xdr4
- xdr5
- xdr6
- xdr7

Dazu öffnet man als erstes über **Show Unmapped** eine Liste der nicht verknüpften Sensordaten.



In der Liste sehen Sie dann, welche Daten zur Verfügung stehen. Über + werden die Daten in die letzte, frei verfügbare XDR-Konfiguration automatisch eingefügt und der richtigen Kategorie zugeordnet. Der Sensorname muss noch im Feld **Transducer** hinzugefügt werden.



Nach der Zuordnung des Sensornamens wird unter **Example** ein Beispiel für das XDR-Telegramm angezeigt. Danach können alle Einstellungen noch individuell geändert werden. Die Erklärung zu den Einstellungen ist nachfolgend aufgeführt.

Direction

Über **Direction** lässt sich einstellen, wie Sensordaten eingelesen werden sollen und wohin sie übertragen werden:

- **off** - Die Sensordaten werden nicht benutzt. Damit können Sie ein bereits konfiguriertes XDR-Telegramm deaktivieren.
- **bidir** - Die Sensordaten werden zwischen NMEA0183 und NMEA2000 ausgetauscht.
- **to2K** - Das Sensordaten werden nur nach NMEA2000 gesendet.
- **from2k** - Sensordaten werden von NMEA2000 eingelesen.

Category

Über **Category** kann ein Sensor-Typ zugeordnet werden:

- **Temperature** - Temperatursensoren z.B. für Luft, Wasser, Kühlgeräte
- **Humidity** - Luftfeuchtigkeitssensoren
- **Pressure** - Drucksensoren für Luftdruck und andere Drücke wie z.B. Öldruck
- **Fluid** - Sensoren für Flüssigkeiten wie Durchfluss und Füllstand
- **Battery** - Batteriesensoren für Spannung, Strom, Leistung, Batterietemperatur

- **Engine** - Motorsensoren für Drehzahl, Anstellung, Trimmklappen, Öl, Kühlwasser
- **Attitude** - Lagedaten, aus Lagesensorsdaten ermittelt

Source

Über **Source** lässt sich die Quelle der Sensordaten genauer einstellen. Je nach verwendetem SensorTyp stehen verschiedene Sensor-Quellen zur Verfügung.

Field

Mit **Field** kann genauer beschrieben werden, wie die Sensordaten zu verstehen sind. Es sind Zusatzdaten, die kontextabhängig je nach verwendetem Sensor-Typ einstellbar sind. So kann z.B. festgelegt werden, ob es sich um einen Anzeigewert oder um einen Einstellwert handelt.

Instance

Mit **Instance** kann festgelegt werden, ob es mehrere Sensoren des selben Typs gibt. Das kann z.B. auftreten, wenn zwei Motoren in einem Boot verbaut sind, und zwei Tankwerte angezeigt werden sollen. Mit Hilfe einer Instanz-Nummer werden die Sensoren unterschieden. An den Sensornamen wird dann z.B. #1 angefügt. Die Instanziierung kann folgendermaßen festgelegt werden:

- **single** - Es wird ein Sensor instanziert, dem eine freie Instanz-Nummer zugeordnet wird. So können z.B. zwei Sensoren die selben Daten in ein XDR-Telegramm übertragen, wenn die Sensoren redundant sind.
- **ignore** - Es existiert nur genau ein einziger Sensor dieses Typs.
- **auto** - Die Instanziierung wird automatisch übernommen. Sobald ein neuer Sensor des gleichen Typs und der selben Source verwendet wird, wird eine neue Instanz des Sensors angelegt.

Transducer

Über **Transducer** wird der Sensorname festgelegt. Es handelt sich dabei um eine Klartextbeschreibung des Sensors mit ASCII Zeichen. Verwenden Sie nur Buchstaben und Zahlen ohne Leer- und Sonderzeichen.

Bemerkung: Verwenden Sie nicht mehr als 6 Zeichen und benutzen Sie Abkürzungen, die geläufig sind. Längere Namen werden aufgrund des begrenzten Anzeigeplatzes im Display auf 6 Zeichen gekürzt.

Example

Beispiel, wie der Inhalt des XDR-Telegramms aussehen wird.

NMEA0183 XDR einlesen

Auch für eingehende NMEA0183-Daten muss zunächst ein XDR-Mapping angelegt werden, bevor sie auf dem OBP60 zur Verfügung stehen. Wenn NMEA0183 XDR-Daten zum Beispiel in folgender Form eingehen: \$IIXDR,A,0.9,D,PTCH,A,0.8,D,ROLL*5D können sie mit diesen Einstellungen auf dem OBP60 verwendet werden:

OBP60V2

connected 

Status	Config	XDR	Data	Update	Help
<input type="button" value="ReloadConfig"/> <input type="button" value="Save&Restart"/> <input type="button" value="Show Unmapped"/> <input type="button" value="Export"/> <input type="button" value="Import"/>					
xdr1 PTCH					
Direction	<input type="text" value="bidir"/> <input type="button" value="X"/> <input type="button" value="?"/>				
Category	<input type="text" value="Altitude"/> <input type="button" value="X"/> <input type="button" value="?"/>				
Source					
Field	<input type="text" value="Pitch"/> <input type="button" value="X"/> <input type="button" value="?"/>				
Instance	<input type="text" value="ignore"/> <input type="button" value="X"/> <input type="button" value="?"/>				
Transducer	<input type="text" value="PTCH"/>				
Example	<input type="text" value="\$ZVXDR,A,120.321,D,PTCH*65"/>				
xdr2 ROLL					
Direction	<input type="text" value="bidir"/> <input type="button" value="X"/> <input type="button" value="?"/>				
Category	<input type="text" value="Altitude"/> <input type="button" value="X"/> <input type="button" value="?"/>				
Source					
Field	<input type="text" value="Roll"/> <input type="button" value="X"/> <input type="button" value="?"/>				
Instance	<input type="text" value="ignore"/> <input type="button" value="X"/> <input type="button" value="?"/>				
Transducer	<input type="text" value="ROLL"/>				
Example	<input type="text" value="\$ZVXDR,A,120.321,D,ROLL*77"/>				

Danach sind die Daten auf der Seite Data verfügbar, und können auch in der Konfiguration der Anzeigeseiten ausgewählt werden. Gegebenenfalls muss hierzu allerdings die Seite im Web-Browser neu geladen werden, bevor die neuen Einträge sichtbar werden.

OBP60V2

connected 

Status	Config	XDR	Data	Update	Help												
<table border="1"> <tr> <td>GPSD</td> <td>GPST</td> <td>xdrPTCH</td> <td>xdrROLL</td> </tr> <tr> <td>2016/08/20</td> <td>14:42:00</td> <td>0.148</td> <td>0.047</td> </tr> <tr> <td>USER[200]</td> <td>USER[200]</td> <td>TCPserver</td> <td>rd TCPserver rd</td> </tr> </table>						GPSD	GPST	xdrPTCH	xdrROLL	2016/08/20	14:42:00	0.148	0.047	USER[200]	USER[200]	TCPserver	rd TCPserver rd
GPSD	GPST	xdrPTCH	xdrROLL														
2016/08/20	14:42:00	0.148	0.047														
USER[200]	USER[200]	TCPserver	rd TCPserver rd														

6.19 Data

OBP60V2

connected 

	Status	Config	XDR	Data	Update	Help
ALT	3	AWA	61	AWS	35.38	BTW ---
COG	200	DBS	5.3	DBT	5.3	DEV 0
DTW	---	GPSD	2024/02/16	GPST	15:02:05	HDG 172
HDOP	1.600	LAT	45°58.88'N	LON	000°38.03'W	Log ---
MHDG	166	MaxAws	35.40	MaxTws	32.63	PDOP ---
ROT	0.00	RPOS	0	SOG	6.80	STW 3.77
SatInfo	---	TWD	71	TWS	32.62	TZ 60
TripLog	---	VAR	6	VDOP	---	WPLat ---
WPLon	---	WTemp	20	XTE	---	xdrVBat 13.700
					m USER[200]	V
<input type="button" value="Durchsuchen..."/> Keine Datei ausgewählt.						

Unter Data werden die Sensordaten aller Bussysteme angezeigt, die derzeit verarbeitet werden können. Sensordaten, die nicht verfügbar sind, werden mit --- gekennzeichnet. Man kann die Datenanzeige auch so konfigurieren, dass nur verfügbare Daten angezeigt werden. Nicht verfügbare Datenfelder sind dann ausgeblendet.

OBP60V2

connected 

Status	Config	XDR	Data	Update	Help
ALT	AWA	AWS	COG		
3	61	35.40	200		
---	---	° ---	kn ---		
DBS	DBT	DEV	GPSD		
5.3	5.3	0	2024/02/16		
---	m ---	m ---	° ---		
GPST	HDG	HDOP	LAT		
15:03:10	172	1.600	45°58.76'N		
---	---	° ---	--- °		
LON	MHDG	MaxAws	MaxTws		
000°38.09'W	166	35.40	32.63		
---	---	° ---	kn ---	kn	
ROT	RPOS	SOG	STW		
0.00	0	6.80	3.77		
---	°/s ---	° ---	kn ---	kn	
TWD	TWS	TZ	VAR		
71	32.63	60	6		
---	° ---	kn ---	--- °		
WTemp	xdrVBat#0				
20	13.660				
---	° USER[200]	V			

Durchsuchen... Keine Datei ausgewählt.

Bemerkung: Die Beschränkung der Datenanzeige auf aktuelle Daten führt dazu, dass sich die Anordnung der Daten ändert, wenn einige Sensordaten nicht mehr verfügbar sind. Diese Datenfelder werden dann ausgeblendet. Wenn Sie ein festes Anzeigeformat bevorzugen, lassen Sie sich am besten alle Daten anzeigen.

6.20 Update

Um die Firmware eines Gerätes zu aktualisieren, können Sie die Registerkarte **Update** verwenden. Es gibt zwei Arten von Firmware-Updates.

Initial Firmware-Update

Beim Initial Firmware-Update wird der komplette Flash-Speicher des OBP60 gelöscht. Anschließend werden alle Firmware-Bestandteile im Flash gespeichert. Dabei wird eine initiale Konfiguration erstellt. Eine vorherige alte Konfiguration wird überschrieben. Die Initial Firmware-Updates verwenden den Dateinamen **xxx-all.bin**.

Normales Firmware-Update

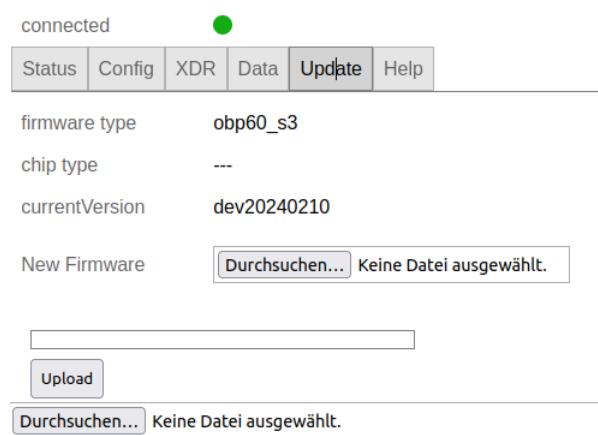
Beim normalen Firmware-Update wird nur der Programmteil der Firmware aktualisiert. Eine vorhandene Konfiguration bleibt dabei erhalten und ist nach dem Firmware-Update wieder nutzbar. Normale Firmware-Updates verwenden den Dateinamen **xxx-update.bin**.

Die letzte aktuelle Firmware können Sie von folgender Webseite herunterladen:

<https://github.com/norbert-walter/esp32-nmea2000-obp60/releases>

Unter **Releases** ist eine Reihe verfügbarer Firmware-Updates für das OBP60 zu finden. Beachten Sie dabei die jeweilige Hardware-Version, für die Sie eine Firmware benutzen wollen.

OBP60V2



Für ein normales Firmware-Update laden Sie sich die gewünschte Firmware als Datei herunter und speichern Sie die Datei auf Ihrem Gerät. Über die Taste Choose File wählen Sie dann die heruntergeladene Datei aus. Danach werden der Firmware-Typ, der Chip-Typ und die Firmware-Version angezeigt. Sollte die Firmware nicht zur verwendeten Hardware passen, so erhalten Sie eine Meldung. Die Firmware kann in diesem Fall nicht geflasht werden. Über die Taste Upload starten Sie den Flash-Vorgang. Im Fortschrittsbalken sehen Sie den Verlauf des Vorgangs. Nach einem erfolgreichen Firmware-Update wird ein Neustart des Systems durchgeführt. In dieser Zeit ist die Web-Konfigurationsseite offline (roter Punkt). Nach beendetem Neustart ist die Seite wieder online (grüner Punkt). Dann ist das System erneut betriebsbereit.

Warnung: Beachten Sie, dass Sie bei einem Firmware-Update auf eine ältere Version ein **Initial Firmware Update** durchführen sollten. So vermeiden Sie Komplikationen mit den gespeicherten Konfigurationsdaten. Bei Nichtbeachtung ist das System ansonsten unter Umständen nicht nutzbar und kann komplett einfrieren. Ein Firmware-Update über die Konfigurationsseiten ist dann nicht mehr möglich. Die Firmware muss dann über USB geflasht werden.

Wie man ein **Initial Firmware Update** durchführt, bzw. die Firmware eines OBP60 über USB flasht, ist unter [Update](#) beschrieben.

6.21 Help

Unter **Help** erfolgt ein Wechsel ins Internet zur Github-Seite, auf der das Projekt gehostet wird. Dort sind einige weitergehende Informationen zum NMEA2000-Gateway zu finden, das die Basis für diese Firmware ist.

Bemerkung: Die Github-Seite lässt sich nur aufrufen, wenn das OBP60 auf das Internet zugreifen kann. Das lässt sich realisieren, wenn das OBP60 zum Beispiel als Client in Ihrem Boots-WLAN arbeitet, und Ihr Boots-WLAN Internetzugang hat. Alternativ starten Sie zum Beispiel einen Hotspot auf Ihrem Handy und verbinden das OBP60 als WLAN-Client mit Ihrem Handy.

KAPITEL 7

Bussysteme



Das OBP60 unterstützt mehrere Bussysteme. In diesem Kapitel geht es ausschließlich um die Vernetzung der Komponenten. Die dafür notwendige Software-Konfiguration finden Sie im darauf folgenden Kapitel [Datenaustausch](#).

- **NMEA2000** über CAN-Bus (isoliert)
- **NMEA0183** über RS485/RS422-Bus (isoliert)
- **I2C-Bus** (isoliert)
- **1-Wire-Bus** (nicht isoliert)
- **USB-C** (nicht isoliert)

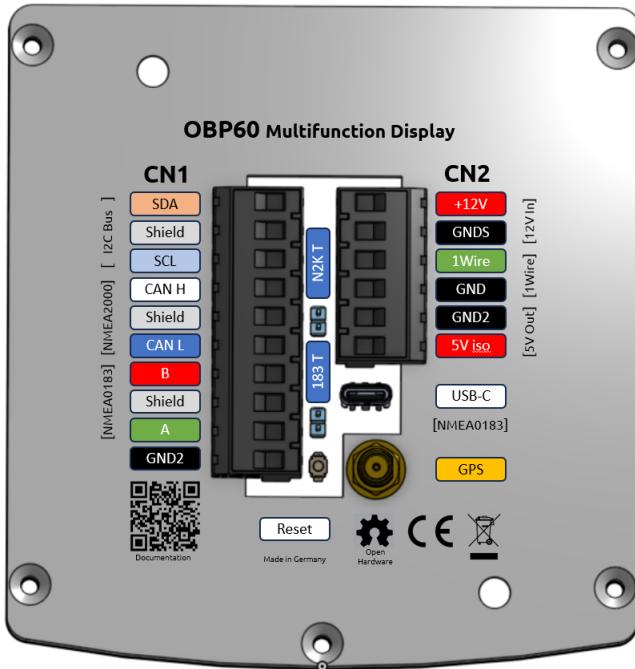


Abb.: Anschlussbelegung der Bussysteme

NMEA2000 und NMEA0183 sind Bussysteme, die im Marine-Bereich verwendet werden. Der I2C-Bus und der 1Wire-Bus kommen aus dem Elektronikbereich. Darüber lassen sich viele günstige Sensor-Module einbinden. Im Nachfolgenden werden die jeweiligen Bussysteme näher beschrieben.

7.1 NMEA2000

NMEA2000 ist ein Bussystem und dient der Datenübertragung zwischen elektronischen Geräten in der Schifffahrt. NMEA2000 verwendet CAN zur Datenübertragung. Die Übertragung erfolgt über ein zentrales Kabel, an das alle Geräte parallel angeschlossen sind. Jedes Gerät im NMEA2000-Netzwerk hat eine eindeutige Geräte-ID, um Datenquellen und Datenanzeigegeräte zu identifizieren und zu adressieren. Daten werden in Parameter Group Numbers (PGN) organisiert. PGN sind eindeutige Daten-IDs, um bestimmte Typen von Daten zu beschreiben, wie beispielsweise Geschwindigkeit, Kurs, Temperatur usw. Alle Geräte können PGN empfangen und senden, auch kann festgelegt werden, welche PGN von einem bestimmten Gerät gesendet oder empfangen werden sollen.

Spezifikation NMEA2000 im OBP60

- Differenzielles, bidirektionales Datenprotokoll auf Binär-Basis
- Halb duplex mit Kollisionserkennung und Vermeidung
- Busstruktur (isoliert)
- Beidseitige Bus-Terminierung
- **Unterstützte Protokolle**
 - CAN (Standard, mit speziellen Datenpaketen)
- Datenrate 250.000 Bit/s fix
- Stromversorgung von Sensoren und Anzeigegeräten über den Bus
- Buslänge bis zu 30 m (Stichleitungen <1.5 m)

- Kabelart 5-polig geschirmt mit 2x2 verdrillten Leitungen 0,35 mm²
- Steckerart M12 5-polig A-kodiert

Differenzielle Datenübertragung

Die Datenübertragung auf dem CAN-Bus erfolgt differenziell. Es werden jeweils zwei Signale mit entgegengesetzter Polarität vom Sender übertragen, die im Empfänger durch Subtraktion der beiden Signalen wieder zu einem Einzelsignal zusammen gefügt werden. Störungen, die sich in gleicher Weise auf beide Signalleitungen auswirken, werden durch die Subtraktion im Empfänger eliminiert. Dadurch wird eine robuste und störunanfällige Signalübertragung ermöglicht.

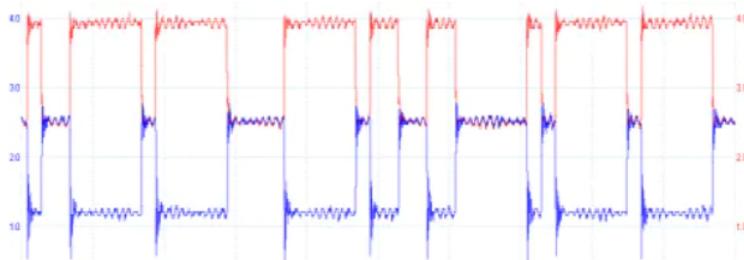


Abb.: Differenzielle Datenübertragung (rot CAN-H, blau CAN-L)

Die Datenrate von NMEA2000 beträgt **250 kbps**. Sie wurde gewählt, um eine ausreichend hohe Übertragungsgeschwindigkeit für eine Vielzahl von Anwendungen in Booten zu gewährleisten, während gleichzeitig eine möglichst effiziente Nutzung des Netzwerks sichergestellt wird. Mit einer Datenrate von 250 kbps können Sensordaten in Echtzeit übertragen werden, was für eine Vielzahl von Anwendungen wichtig ist, z.B. für die Überwachung der Schiffssposition, für Navigations- und Kommunikationsgeräte, Motorsysteme und weitere Systeme an Bord.

Bemerkung: SeaTalk NG, SIMnet, Raynet, C-Net 2000 und CANet haben einige Ähnlichkeiten zu NMEA2000. Sie unterscheiden sich jedoch durch ihre spezifische Auslegung der Hardware und der Datentelegramme. SeaTalk NG und SIMnet sind teilweise kompatibel mit NMEA2000. Das bedeutet, dass einige Geräte, die für SeaTalk NG und SimNet entwickelt wurden, mit Abstrichen auch mit NMEA2000-Geräten kommunizieren können.

Terminierung des Bussystems

Ein CAN-Bus verwendet an den beiden Enden des Bussystems Abschluss-Widerstände von 120 Ohm zwischen den Leitungen CAN-H und CAN-L. Die beiden Abschluss-Widerstände entsprechen dem Leitungswiderstand von 120 Ohm und verhindern Signalreflexionen an den Leitungsenden bei hohen Datenübertragungsraten. Der CAN-Bus besteht aus einem langen Busstrang (Backbone) mit kurzen Stichleitungen kürzer als 1,5 m. Eine Sternstruktur des Bussystems ist nicht erlaubt. Die beiden Abschluss-Widerstände dürfen nur am Busende verbaut sein.

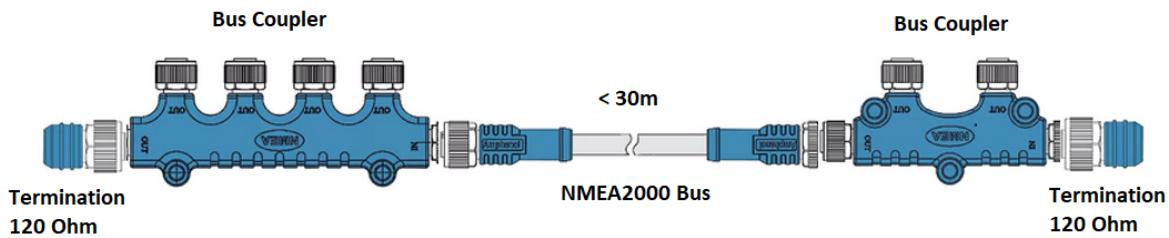


Abb.: CAN-Bus-Terminierung für NMEA2000 über T183

Warnung: Einige Geräte besitzen eingebaute Abschluss-Widerstände, die durch entsprechende Schalter zu- oder abgeschaltet werden können. Vergewissern Sie sich vor dem Einfügen neuer Geräte in Ihr NMEA-2000-Netzwerk, ob integrierte Abschluss-Widerstände benutzt werden und wie sie konfiguriert sind. Falsch terminierte Busse können Übertragungsprobleme verursachen, die schwer zu finden sind.

Tipp: Um herauszufinden, ob nur zwei Widerstände im NMEA-Bus aktiv sind, können Sie ein Digitalvoltmeter benutzen. Wenn Sie eine Widerstandsmessung bei **spannungslosen NMEA-Bus** zwischen den Leitungen **CAN-High** und **CAN-Low** vornehmen, sollte ein Widerstand von ca. **60 Ohm** zu messen sein. Ist der Widerstand deutlich kleiner als 60 Ohm, sind weitere Geräte im Bussystem vorhanden, deren Abschlusswiderstände fälschlicherweise aktiv ist. Trennen Sie, während Sie laufend weiter den Widerstand messen, nacheinander ein Gerät nach dem anderen vom NMEA2000-Bus, bis sich der Widerstandswert deutlich erhöht. Beim zuletzt entnommenen Gerät sollte der Abschlusswiderstand deaktiviert werden. Ist der Widerstandswert danach noch immer nicht bei 60 Ohm, suchen Sie nach weiteren Geräten mit offensichtlich aktiviertem Abschlusswiderstand.

NMEA2000-Kabel



Als Buskabel sollten nur hochwertige, wasserdichte und geschirmte Industrie-Kabel verwendet werden. Der Marine-Einzelhandel bietet dazu eine breite Auswahl an Produkten mit M12-Verbindungstechnik an, die sehr gut geeignet sind.

Tipp: Im Industriebereich findet man gleichwertige Kabel mit M12-Verbindungstechnik, die deutlich günstiger sind und ebenfalls verwendet werden können. Achten Sie dabei auf Steckverbinder mit **A-Kodierung**. Die Indexkerbe befindet sich zwischen Pin 1 und 2.

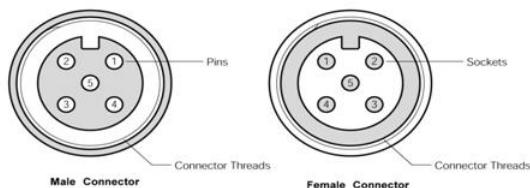


Abb.: Stecker und Buchse (Sicht auf Kontakte)

Die Belegung und Aderfarben sind in der nachfolgenden Tabelle zu sehen. Beachten Sie, dass die Farbbelegung bei Industriekabeln anders ausfallen kann. In dem Fall müssen Sie die Zuordnung der Aderfarben zu den Pins mit einem Ohmmeter ermitteln.

Pin	Belegung	Farbe	Bedeutung
1	Schirmung	ohne	Schirmgeflecht
2	+12V	rot	Versorgungsspannung
3	GND	schwarz	Bord-Masse
4	CAN-H	weiß	CAN High-Signal
5	CAN-L	blau	CAN Low-Signal

Tab.: NMEA2000 Steckerbelegung

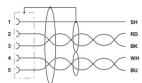


Abb. NMEA2000-Buskabel mit Schirmung

Wer eigene Buskabel herstellen möchte, sollte Kabel vergleichbar zum Typ „Lapp Busleitung UNITRONIC 2 x 2 x 0.34 mm²“ verwenden.



Abb. CAN-Buskabel

Dabei sind 2x zwei Einzeladern paarweise verdrillt und zusätzlich mit einem äußeren Schirmgeflecht umgeben. Ein verdrilltes Aderpaar wird für CAN-H und CAN-L verwendet und das andere Aderpaar für GND und 12V. Das Schirmgeflecht wird mit GND an nur einem Kabelende verbunden. Damit erzielt man die besten Ergebnisse und hat eine sichere und langlebige Installation. Dünneres Kabel als 0.34 mm² sollten nicht verwendet werden, wenn die Versorgung aus dem Bus erfolgen soll. Die Länge des Buskabels sollte in Summe 30 m nicht übersteigen.



Abb.: NMEA2000 Stecker zur Selbstmontage

Bemerkung: SeaTalk NG und Simnet benutzen eigene Steckverbinder, die untereinander nicht zu NMEA2000 kompatibel sind. Mit entsprechenden Konverter-Kabeln kann aber ein Datenaustausch zwischen diesen Netzen vorgenommen werden. Grundsätzlich sollte eine Mischung verschiedener Bustechnologien aber vermieden werden.

Stromversorgung aus dem NMEA2000-Bus

NMEA2000-Geräte mit geringem Stromverbrauch wie z.B. Sensoren können auch aus dem Bussystem mit Strom versorgt werden. So sind keine zusätzlichen Versorgungskabel notwendig. Allerdings ist darauf zu achten, dass aus dem NMEA2000-Bus nur eine begrenzte Leistung von bis zu 35W entnommen werden darf. Die NMEA2000-Geräte sind dazu mit Load-Werten gekennzeichnet, die die Stromentnahme aus dem Bus angeben. Die Load wird als Vielfaches von 50 mA angegeben. Ein Gerät mit Load 3 würde demnach 150 mA an 12V benötigen und 1,8 W an Leistung verbrauchen. Die Einspeisung der 12V-Versorgungsspannung in den NMEA2000-Bus erfolgt entweder über ein Einspeisekabel oder ein Gerät mit Buseinspeisung, zum Beispiel einen Plotter. Die Einspeisung in den Bus sollte idealerweise in der Mitte erfolgen, da so Leitungsverluste durch Leitungswiderstände minimiert werden.

Bemerkung: Das OBP60 hat einen Load-Wert von 5 und benötigt maximal 250 mA Strom. Sie können das OBP60 direkt mit Strom aus dem NMEA2000-Bus versorgen. Im Normalbetrieb liegt der Stromverbrauch des OBP60 bei ca. 120 mA.

Verkabelung für NMEA2000

Für NMEA2000 wird eine Busstruktur verwendet. Im Hauptstrang sind eine oder mehrere Buskoppler-Einheiten enthalten, über die die jeweiligen Geräte eingebunden werden. Die Buslänge darf 30 m nicht übersteigen und die Stichleitungen zu den Geräten sollte nicht länger als 1.5 m sein. An den Enden des Hauptstrangs befinden sich Abschlusswiderstände zur Bus-Terminierung. Die Einspeisung der Stromversorgung des NMEA2000-Busses erfolgt im unteren Bild über den Plotter.

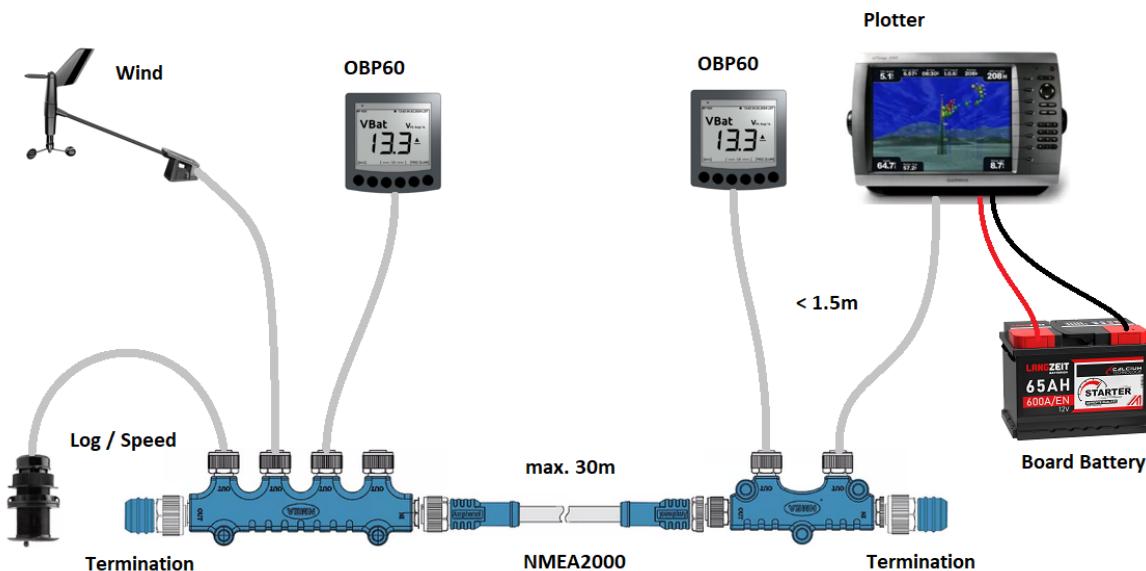


Abb.: NMEA2000-Bussystem mit Sensoren und Anzeigegeräten

Um das OBP60 an den NMEA2000-Bus anzuschließen, können Sie am einfachsten ein NMEA2000-Verlängerungskabel benutzen, indem Sie es in der Mitte trennen und die offenen Kabelenden an den Schraubklemmen auflegen. Es ist ratsam, die Kabelenden mit Aderendhülsen zu versehen oder die Kupferkabel zu verzinnen.

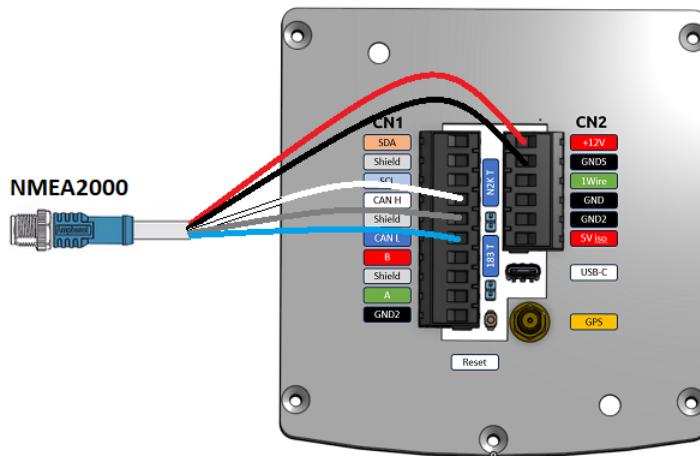


Abb.: NMEA2000-Verbindung mit Stromversorgung über den NMEA2000-Bus

Eine Minimal-Konfiguration könnte folgendermaßen aussehen. Dabei ist zu beachten, dass der NMEA2000-Bus auf der rechten Seite durch das OBP60 terminiert ist, indem die interne Bus-Terminierung über den Jumper TN2K aktiviert wurde. Der Jumper TN2K befindet sich mittig zwischen den beiden Steckverbindern **CN1** und **CN2**.

Bemerkung: Legen Sie den Schirm des NMEA2000-Kabels am Eingang **Shield** auf. Verbinden Sie den Schirm **nicht** mit GND, GND2 oder GNDS, da Sie damit Masseschleifen erzeugen und die Isolationswirkung verloren geht. Der gesamte Schirm der Busleitung darf nur einseitig am Eingang **Shield** des NMEA2000-Bus am OBP60 aufgelegt werden. An anderen Stellen darf die Kabelschirmung nicht verbunden werden.

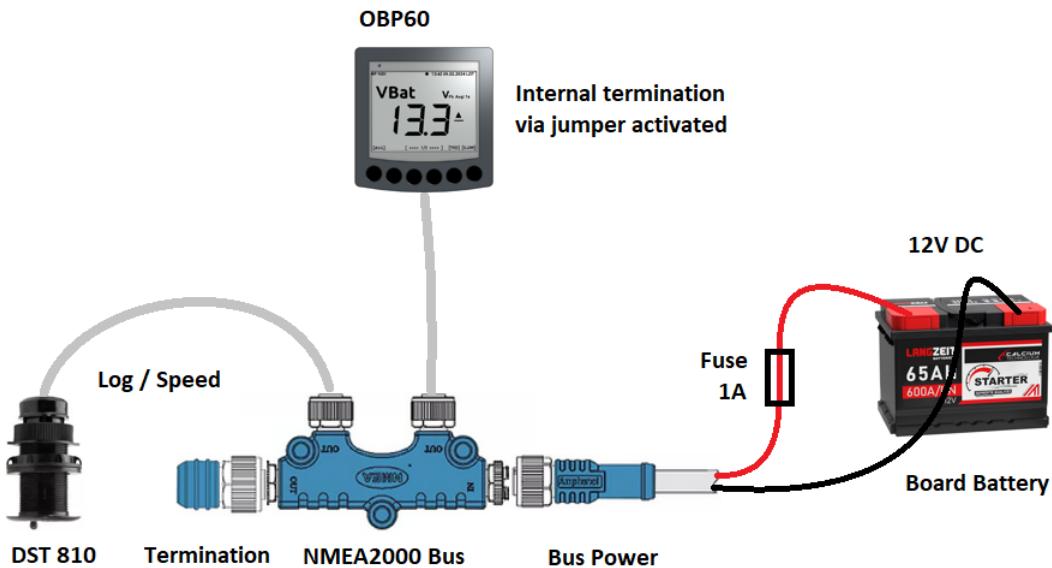


Abb.: NMEA2000 Minimal-Konfiguration mit einem Sensor

Die Stromversorgung des NMEA2000-Bus kann auch über das OBP60 erfolgen. Die Einspeisung in den Bus sieht dann so aus:

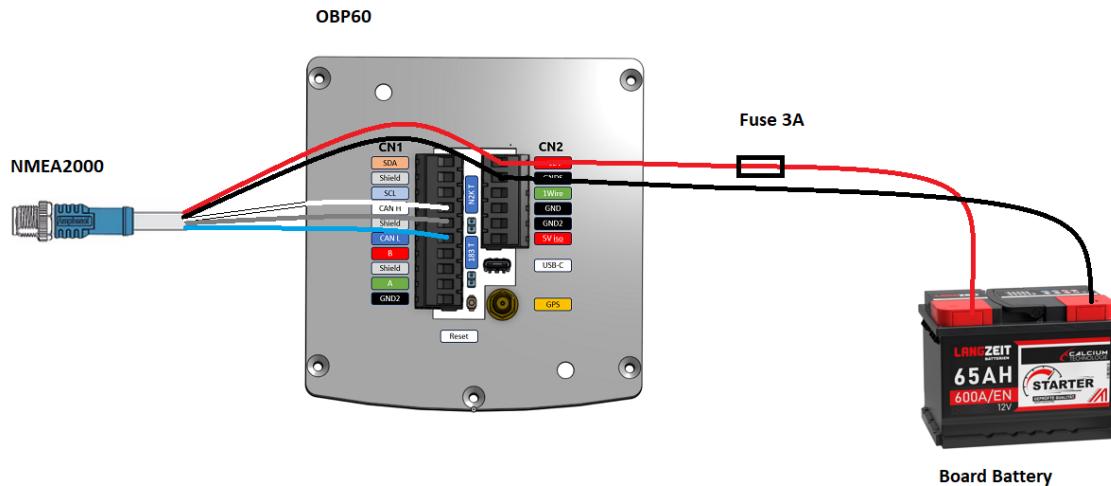


Abb.: NMEA2000-Verbindung mit Stromeinspeisung über OBP60

Warnung: Beachten Sie, dass der NMEA2000-Bus nur von einer Quelle gespeist sein darf. Andernfalls kann es zu Störungen im Bussystem kommen. Achten Sie darauf, dass die Stromeinspeisung in den Bus mit einer 3A-Sicherung abgesichert ist.

Das folgende Bild zeigt ein mögliches Anwendungsbeispiel. Die Bus-Terminierung ist dabei im OBP60 deaktiviert, sie wird am Busverteiler vorgenommen.

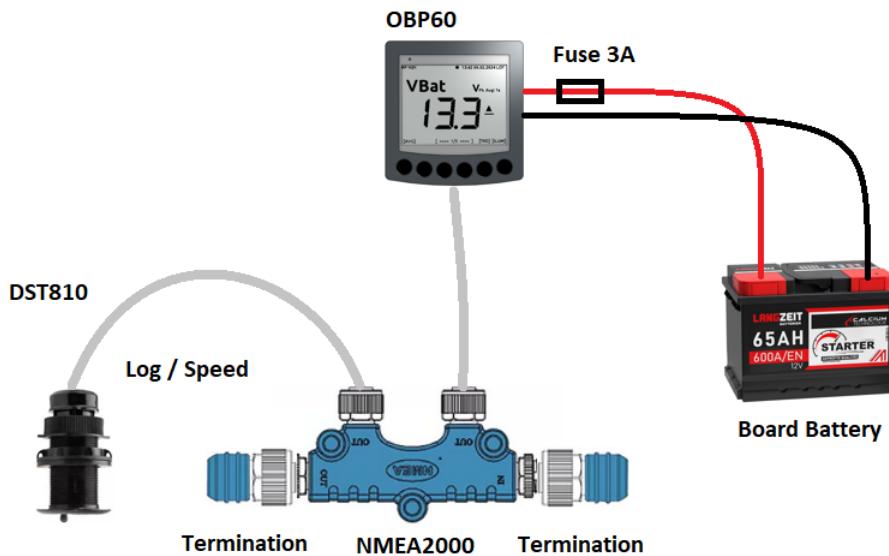


Abb.: NMEA2000 Minimal-Konfiguration mit Buseinspeisung

Kompatibilität zu Simnet und SeaTalk NG

Simnet und SeaTalk GN besitzen eine eingeschränkte Kompatibilität zu NMEA2000. Beide Bussysteme setzen auf eigene Steckersysteme und verwenden teilweise proprietäre NMEA2000-Telegramme. Die meisten gängigen NMEA2000-

Bustelegramme werden von beiden Systemen unterstützt. Mit speziellen einfachen passiven Adapterkabeln können Simnet und SeaTalk GN Bussysteme mit einem NMEA2000 Bussystem verbunden werden. Das OBP60 kann dann Informationen aus Simnet oder SeaTalk NG über den CAN-Bus oder über WiFi mit dem SeaSmart-Protokoll verarbeiten und auch Informationen in diese Bussysteme senden. Properitäre Telegramme werden nicht unterstützt, jedoch im Bussystem übertragen und weitergeleitet.

7.2 NMEA0183

NMEA0183 ist ein Standard für serielle Datenübertragung in der Schifffahrt. Es definiert ein Format für die Übertragung von Informationen zwischen Navigationsgeräten und anderen elektronischen Geräten auf Booten. NMEA0183 ist ein weit verbreiteter Standard, der vor allem von vielen älteren Geräten unterstützt wird.

Spezifikation NMEA0183 im OBP60

- Serielles, unidirektionales Datenprotokoll auf ASCII-Basis
- Punkt zu Punkt-Verbindung (isoliert)
- Simplex ohne Kollisionserkennung und Vermeidung
- Bus-Terminierung am Empfänger
- **Unterstützte Protokolle**
 - RS422 (Standard)
 - RS485
- Datenrate 1.200...460.800 Bd variabel
- Stromversorgung von Sensoren und Anzeigegeräten über 12V-Bordnetz
- Buslänge bis zu 1000 m (abhängig von Datenrate und Kabelart)
- Kabelart nicht spezifiziert
- Steckerart nicht spezifiziert

Datenübertragung

Die Datenübertragung erfolgt im OBP60 halb duplex auf serielle Weise über zwei einfache Kabel. Das bedeutet, dass man entweder senden oder empfangen kann. Beides gleichzeitig ist nicht möglich. Die Standard-Datenrate liegt bei 4800 Bd, was für heutige Verhältnisse recht langsam ist, aber Buslängen von bis zu 1000 m zulässt. Als Datenübertragungsrate lassen sich folgende Einstellungen verwenden:

- 1.200 Bd
- 2.400 Bd
- 4.800 Bd
- 9.600 Bd
- 14.400 Bd
- 19.200 Bd
- 28.800 Bd
- 38.400 Bd
- 56.600 Bd
- 57.600 Bd

- 115.200 Bd
- 230.400 Bd
- 460.800 Bd

Je nach Datenrate und Protokoll können die zulässigen Kabellängen unterschiedlich lang ausfallen. Im realen Betrieb sollten diese Werte beachtet werden.

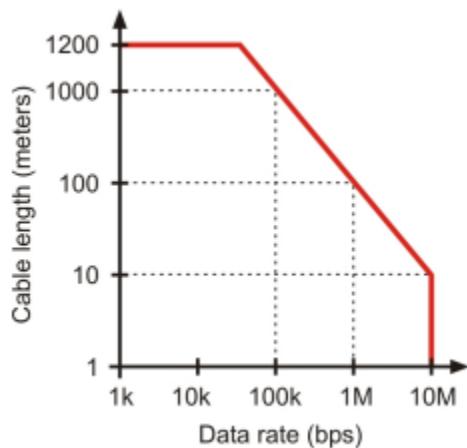


Abb.: Zulässige Kabellängen für RS422 und RS485

Übertragungsrate [Bd]	zul. Leitungslänge [m]
4.800	300
9.600	152
19.200	15
57.600	5
115.200	2

Tab.: Zulässige Leitungslängen für RS232

Die Datenübertragung erfolgt mit differenziellen Signalen ähnlich wie bei NMEA2000. Damit können Gleichtaktstörungen über lange Leitungslängen sicher unterdrückt werden.

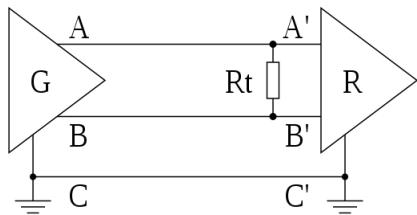


Abb.: RS422-Übertragungsmodell Sender - Empfänger

Bus-Terminierung

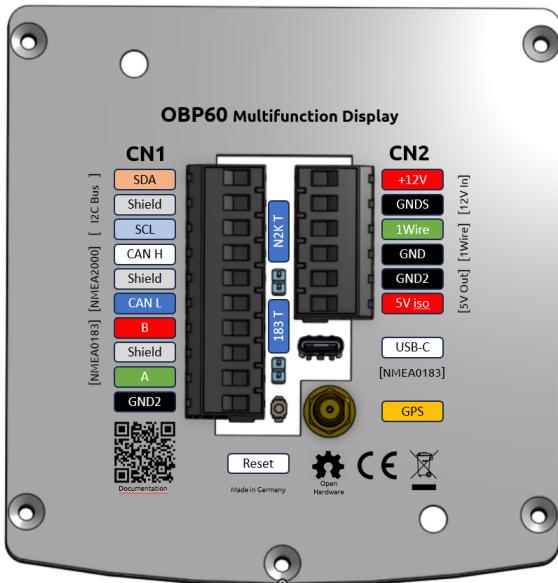


Abb.: Bus-Terminierung für NMEA0183

Auf der Empfängerseite ist der NMEA0183-Bus terminiert. Im OBP60 findet man den Jumper T183 für die Bus-Terminierung zwischen den Steckverbindern **CN1** und **CN2**. Der Jumper ist zu setzen, wenn das OBP60 als Empfänger von NMEA0183-Telegrammen als **Serial Direction Receive** konfiguriert wurde (siehe Kapitel [Config - Serial Port](#)).

Multiplexer

Insgesamt ist NMEA0183 ein nützlicher Standard für die Übertragung von Navigationsdaten auf Booten, aber er hat seine Beschränkungen und kann nicht in allen Einsatzfällen mit moderneren Technologien wie NMEA2000 mithalten. Um zum Beispiel Daten von mehreren Datenquellen wie z.B. Sensoren zu einem Datenstrom zusammenfassen zu können, sind in der NMEA0183-Welt Multiplexer notwendig.

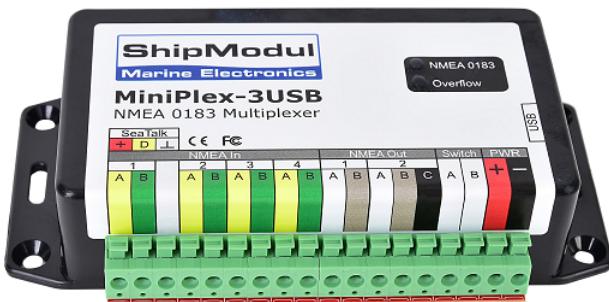


Abb.: NMEA0183 Multiplexer (Ship Modul)

Der Multiplexer empfängt verschiedene Datentelegramme an unterschiedlichen Ports und gibt den zusammengefassten Datenstrom mehrerer Sensoren an einem neuen Datenport aus. So lassen sich mehrere Sensorsignale über eine Leitung an ein Datenendgerät wie z.B. einen Plotter oder ein Multifunktionsdisplay übertragen. Viele Multiplexer bieten auch die Möglichkeit, bestimmte Datentelegramme im Datenstrom mit einer Filterfunktion zu unterdrücken. So können z.B. nur die wirklich notwendigen Daten an einen Autopiloten übertragen oder Mehrdeutigkeiten durch mehrere GPS-Empfänger vermieden werden.

NMEA0183-Telegrammstruktur

NMEA0183-Telegramme sind recht einfach aufgebaut und werden als ASCII-Datensätze übertragen. Ein NMEA0183-Telegramm besteht aus folgenden Informationen.

- Kennung
- Telegrammtyp
- Sensordaten
- Einheit
- Status
- CRC-Checksumme

Je nach Komplexität eines Telegramms können auch mehrere Sensordaten oder Statusinformationen in einem Telegramm übertragen werden. Nachfolgend ist beispielhaft das Telegramm eines Tiefenmessers dargestellt.

DBT - Depth below transducer

\$-DBT,a.a,b,c.c,d,e.e,f*hh<CR><LF>

Feldnummer:

- a.a - Tiefe in Fuß
- b - f = Fuß
- c.c - Tiefe in Metern
- d - M = Meter
- e.e - Tiefe in Fathoms
- f - F = Fathoms
- hh - Checksumme

Beispiel:

- \$IIDBT,12.8,f,39.0,M,21.3,F*20

Wer mehr Informationen zu NMEA0183-Telegrammen erfahren möchte, findet auf dieser [Webseite](#) ausführliche Informationen.

Verkabelung für NMEA0183

Im nachfolgenden Bild ist eine Konfiguration zu sehen, in der ein NMEA0183-Windsensor mit dem OBP60 verbunden ist. Der Windsensor sendet die Daten zum OBP60, dieses ist als NMEA0183-Empfänger konfiguriert. Die Bus-Terminierung ist über den Jumper T183 eingeschaltetet.

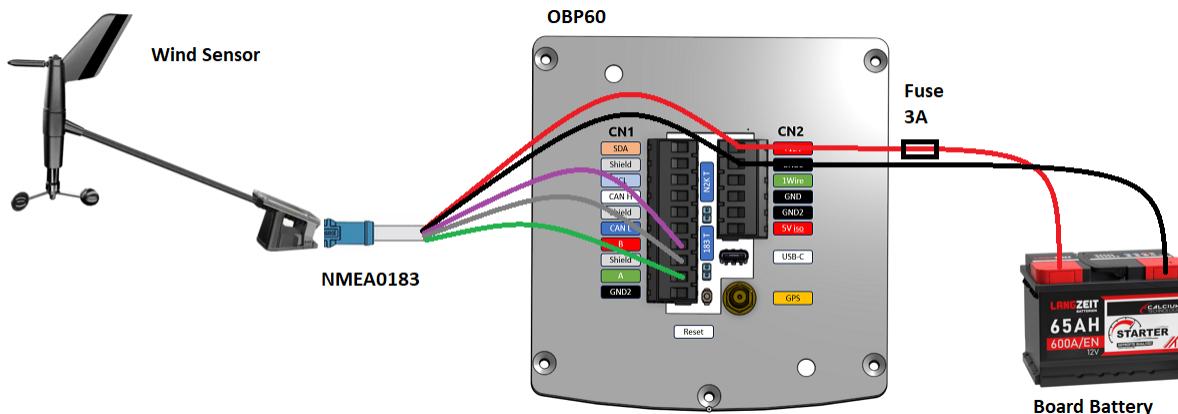


Abb.: NMEA0183-Minimalkonfiguration

Hinweis: In ähnlicher Art und Weise können auch andere Sensoren an das OBP60 angebunden werden. Dabei ist aber zu berücksichtigen, dass immer nur ein Gerät oder Sensor mit dem OBP60 verbunden werden kann. Wenn mehrere Geräte eingebunden werden sollen, dann benötigt man einen vorgeschalteten Multiplexer.

Bemerkung: Verwenden Sie für die Verkabelung externer Sensoren über NMEA0183 möglichst geschirmte Kabel und führen Sie den Schirm direkt bis zum Sensor. Verbinden Sie den Schirm am Sensor **nicht** mit GND2, da Sie damit Masseschleifen erzeugen. Der gesamte Schirm der Busleitung darf nur einseitig an Eingang Shield des NMEA0183-Bus am OBP60 aufgelegt werden. Der Schirm am anderen Ende der Leitung bleibt offen. Andere Schirmeingänge dürfen nicht benutzt werden.

Achtung: Beachten Sie, dass bei NMEA0183-Datenübertragungen die selbe Übertragungsgeschwindigkeit für Sender und Empfänger und das selbe Übertragungsprotokoll verwendet werden muss. Andernfalls kann keine Datenübertragung stattfinden. Die NMEA0183-Schnittstelle im OBP60 unterstützt das RS232-Protokoll nicht.

Die meisten Multiplexer haben mehrere NMEA0183-Eingänge und mindestens einen NMEA0183-Ausgang. Bei der Verwendung eines Multiplexers werden alle Sensoren an die NMEA0183-Eingänge des Multiplexers angeschlossen und der NMEA0183-Ausgang mit dem OBP60 verbunden. Der Multiplexer bündelt dann wie beschrieben die Datenströme aller Sensoren zu einem gemeinsamen Datenstrom am Ausgang. Über Filter am Datenausgang lässt sich die Datenmenge auf wichtige Daten eingrenzen. Das OBP60 ist in diesem Beispiel als Empfänger konfiguriert. Die Terminierung des Bussystems ist deaktiviert.

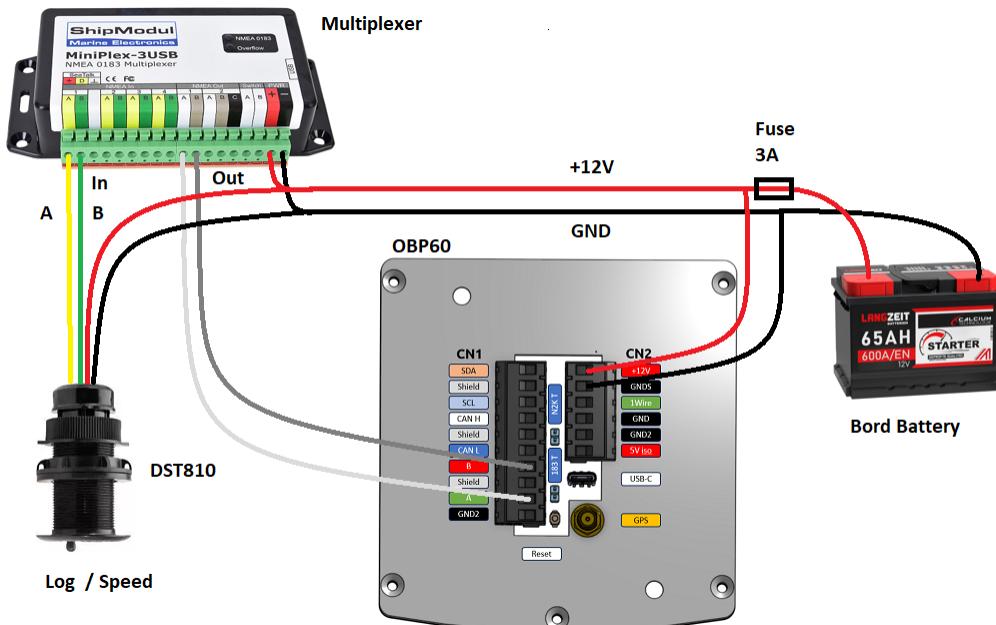


Abb.: NMEA0183-Verbindung zu einem Multiplexer

Hinweis: Alle NMEA0183-Daten werden vom OBP60-Gateway automatisch nach NMEA2000 konvertiert. Die Konvertierung ist dabei unidirektional nur in Richtung NMEA2000. In umgekehrte Richtung nach NMEA0183 werden keine Daten konvertiert, da der NMEA0183-Port des OBP60 in der dargestellten Konfiguration im Empfangs-Modus arbeitet.

7.3 I2C

Der I2C-Bus dient zur Anbindung von elektronischen Komponenten. Er wird hauptsächlich im Elektronikbereich eingesetzt, um verschiedene Komponenten auf einer Platine miteinander kostengünstig zu verbinden. Die Verbindung erfolgt über eine Zweidrahtleitung und arbeitet mit Signalpegeln von 5.0V. Es gibt das Taktsignal **SCL** und das Datenignal **SDA**. Die Kommunikation läuft als Master- und Slave-System. Dabei steuert der Master die Slaves über eine eindeutige Adresse an und kann mit ihnen Daten austauschen.

Spezifikation I2C im OBP60

- Serielles, bidirektionales, synchrones Datenprotokoll auf Binär-Basis
- Busstruktur (isoliert)
- Halb duplex mit Kollisionserkennung und Vermeidung
- Bus-Terminierung intern über PullUp-Widerstände
- **Unterstützte Protokolle**
 - I2C, TTL 5.0V
- Datenrate 100.000 kBit/s variabel
- Stromversorgung von Sensoren und Anzeigegeräten über separate Leitungen
- Buslänge bis zu 10 m (abhängig von Datenrate und Kabelart)
- Kabelart nicht spezifiziert
- Steckerart nicht spezifiziert

Im OBP60 ist der I2C-Bus isoliert gegenüber der Außenwelt aufgebaut und verwendet 5.0V TTL-Signalpegel. Der Takt- und Datenausgang ist über Leitungstreiber herausgeführt, die lange Leitungen mit einer hohen Leitungskapazität betreiben können. Damit wird verhindert, dass sich Störungen im Bussystem negativ auf die Funktionssicherheit des OBP60 auswirken. Der I2C-Bus hat 5 Leitungen, über die externe Geräte angeschlossen werden können.

Ausgang	Bedeutung
5Viso	Versorgungsspannung
GND2	Masse I2C
Shield	Schirmung I2C
SCL	Bus-Takt
SDA	Datenleitung

Im folgenden Bild ist ein I2C-Busaufbau mit 3 I2C-Sensoren zu sehen. Alle Sensoren sind mit geschirmten Kabeln mit dem I2C-Eingang am OBP60 verbunden. Die Stromversorgung der externen Sensoren erfolgt direkt über das OBP60 über den integrierten, isolierenden DC/DC-Wandler (5Viso, GND2). Der Versorgungsausgang kann bis zu 200 mA bei 5Viso liefern und damit auch mehrere Sensoren mit Strom versorgen.

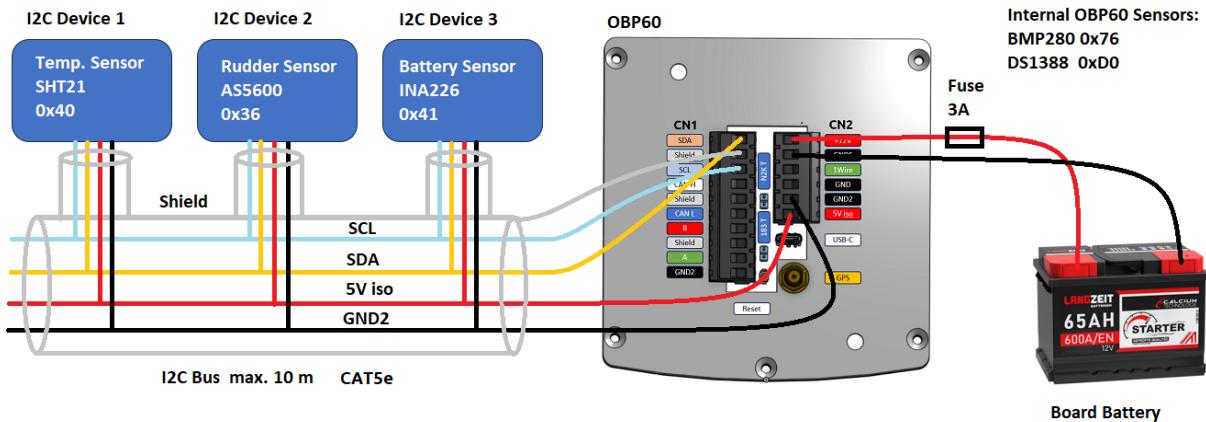


Abb.: I2C-Anbindung von externen Sensoren

Bemerkung: Verwenden Sie für die Verkabelung externer Sensoren möglichst geschirmte Kabel und führen Sie den Schirm direkt bis zum Sensor. Verbinden Sie den Schirm des Sensorkabels **nicht** mit GND2, da Sie damit Masseschleifen erzeugen. Der gesamte Schirm der Busleitung darf nur einseitig an Eingang Shield des I2C-Bus am OBP60 aufgelegt werden. Der Schirm am anderen Ende der Leitung bleibt offen. Andere Schirmeingänge dürfen nicht benutzt werden. Halten Sie Stichleitungen vom Bus zu den Sensoren möglichst kurz.

Achtung: Einige I2C-Module können zwar mit 5V versorgt werden, benutzen aber 3.3V TTL-Signale für SCL und SDA. In diesem Fall benötigen Sie einen Pegelwandler für das SCL- und SDA-Signal auf 5V TTL-Signale. Beachten Sie das nicht, so können der I2C-Sensor oder das I2C-Modul beschädigt werden. Das Gleiche gilt auch für Module, die nur mit 3.3V versorgt werden. Dann benötigen Sie neben dem Pegelwandler noch zusätzlich einen DC/DC-Wandler zur Reduzierung der Versorgungsspannung. Die meisten I2C-Module, die für einen **Arduino Uno** verwendet werden können, lassen sich auch am OBP60 verwenden. Diese Module sind für 5V-TTL-Signale geeignet.

Vorsicht: Wenn Sie externe Sensoren oder Module am I2C-Bus verwenden wollen, dann prüfen Sie, ob es zu einem Adresskonflikt zwischen den verwendeten Sensoren oder Modulen kommen kann. Achten Sie darauf, dass I2C-Adressen nicht mehrfach vergeben sind. Es kommt dann zu Kommunikationsstörungen auf dem I2C-Bus. Insbesondere bei mehrfacher Verwendung gleicher Module müssen die I2C-Adressen verschieden eingestellt werden. Bei einigen I2C-Modulen ist das nicht möglich. In dem Fall können Sie nur ein I2C-Modul des jeweiligen Typs im Bus verwenden. Das OBP60 belegt bereits folgende Adressen, die nicht von Sensoren benutzt werden dürfen: 0x76, 0x68.

Vorsicht: Nicht angeschlossene externe I2C-Sensoren, die jedoch in der Konfiguration aktiviert sind, führen dazu, dass die Reaktionsfähigkeit des OBP60 beeinträchtigt wird. Diese Sensoren können nicht auf das System antworten, sie erzeugen dadurch einen Software-Timeout. Deaktivieren Sie in einem solchen Fall die Sensoren in der Konfiguration.

Gefahr: Bestimmen Sie den Strombedarf Ihrer externen Sensoren und achten Sie darauf, dass die Stromversorgung

5V_{iso} nicht überlastet oder kurzgeschlossen wird. Der maximal zulässige Strom liegt bei **200 mA**. Andernfalls fallen sonst alle isolierten Bussysteme wie **NMEA2000**, **NMEA0183** und **I2C** aus, da sie mit der selben Stromquelle versorgt werden. Die Folge ist ein Kommunikationsverlust auf allen genannten Bussystemen, der schwere Folgen für die Navigation Ihres Bootes haben kann. Verbinden Sie GND2 **nicht** mit GND oder GNDS, da die Isolationswirkung verloren geht und die Empfindlichkeit für Störungen steigt.

7.4 1Wire

Der 1Wire-Bus ist ein Eindraht-Bus zur seriellen Übertragung von Daten in elektronischen Schaltungen. Neben der Datenleitung wird noch eine Masseleitung als Potenzialbezug benötigt. Die Übertragung ist bidirektional und asynchron. Der 1Wire-Bus wird oft für einfache Sensoren verwendet, die nur kleine Datenmengen übertragen wie z.B. für die Temperatursensoren **DS18B20**. Beim OBP60 ist der 1Wire-Bus an der Anschlussklemme CN2 herausgeführt.

Spezifikation 1Wire

- Serielles, bidirektionales asynchrones Datenprotokoll auf Binär-Basis
- Busstruktur (nicht isoliert)
- Halb duplex mit Kollisionserkennung und Vermeidung
- Bus-Terminierung über PullUp-Widerstand am Ausgang
- **Unterstützte Protokolle**
 - 1Wire, TTL 3.3V
- Datenrate 9600 kBit/s (bei parasitärer Stromversorgung über Datenleitung)
- Stromversorgung von Sensoren über Datenleitung
- Buslänge bis zu 10 m (abhängig von Datenrate und Stromversorgung)
- Kabelart nicht spezifiziert
- Steckerart für einige Anwendungen spezifiziert
- Maximal 8 DS18B20 Sensoren nutzbar

Der 1Wire-Bus bietet eine einfache und kostengünstige Möglichkeit, Temperatursensoren einzubinden. Zur Anbindung werden nur 3 Leitungen am OBP60 benötigt.

Ausgang	Bedeutung
1Wire	Datenleitung
GND	Masse 1Wire
GND2	Schirmung

Die Stromversorgung der Temperatursensoren erfolgt parasitär über die Datenleitung. Intern im Sensor befindet sich ein Kondensator, der eine gewisse Menge an Energie für den Zeitpunkt der Übertragung speichern kann, wenn der Datenpegel auf 3.3V liegt. Die Sensoren werden über eindeutige Adressen angesprochen und können mit dem OBP60 Daten austauschen. Bei der parasitären Stromversorgung ist die Datenrate auf max. 9600 kBit/s limitiert. Die Sensoren können nur wenige Male in der Minute abgefragt werden, da sie ihre Energie über einen längeren Zeitraum über die Datenleitung sammeln müssen. Pro Sekunde wird nur ein Sensor ausgelesen. Der Vorgang wiederholt sich dann für alle weiteren Sensoren. 1Wire-Temperatur-Sensoren eignen sich daher nur für die Verarbeitung unkritischer Temperaturwerte.

Nachfolgend ist eine Beispielanwendung für 1Wire-Temperatur-Sensoren zu sehen.

Parasitic Power Supply DS18B20 with 5.0V

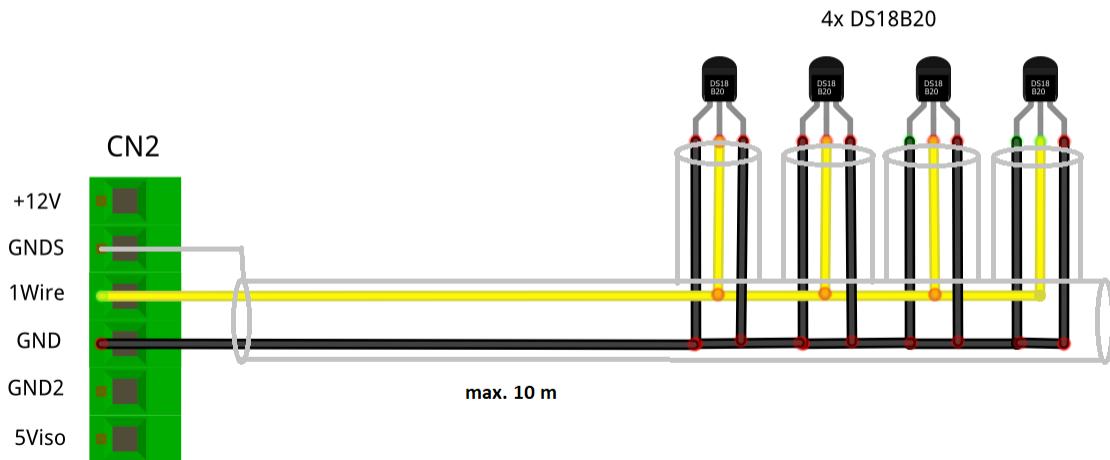


Abb.: 1Wire-Anbindung von externen Temperatur-Sensoren (parasitär versorgt)

Die Belegung der DS18B20-Temperatur-Sensoren ist folgendermaßen durchzuführen.

Ausgang	Temperatursensor
1Wire	gelb, Datenleitung
GND	schwarz + rot
GNDS	Schirm

Bemerkung: Verwenden Sie für die Verkabelung externer Temperatur-Sensoren möglichst geschirmte Kabel und führen Sie den Schirm direkt bis zum Sensor. Verbinden Sie den Schirm des Sensorkabels **nicht** mit GND, da Sie damit Masseschleifen erzeugen. Der gesamte Schirm der Busleitung darf nur einseitig an Eingang GNDS des 1Wire-Bus am OBP60 aufgelegt werden. Der Schirm am anderen Ende der Leitung bleibt offen. Andere Schirmeingänge dürfen nicht benutzt werden. Halten Sie Stichleitungen vom Bus zu den Sensoren möglichst kurz. Die maximale Anzahl der Sensoren am 1Wire-Bus ist auf 8 Sensoren begrenzt. Die Ausleszeit eines Sensors ist von der Anzahl (N) der Sensoren im Bus abhängig. Die Auslesesezeit T lässt sich über folgende Formel berechnen: $T[s] = N \cdot 1s$.

Hinweis: Wenn möglich, verwenden Sie Temperatursensoren am I2C-Bus statt am 1Wire-Bus. Sie erhöhen damit die Betriebssicherheit des Gesamtsystems, da der I2C-Bus gegenüber der Außenwelt isoliert ist.

Hinweis: Im Internet-Handel sind Nachbauten von DS18B20-Temperatur-Sensoren im Umlauf, die eine parasitäre Stromversorgung nicht unterstützen. Wenn Sie keine Kommunikation mit dem OBP60 zustande bekommen, dann probieren Sie andere Sensoren aus. Wenn auch das zu keinem Erfolg führt, benutzen Sie eine normale Stromversorgung für die Temperatursensoren. Mit dieser Art der Stromversorgung sollten nahezu alle Sensoren funktionieren.

Direct Power Supply DS18B20 with 5.0V

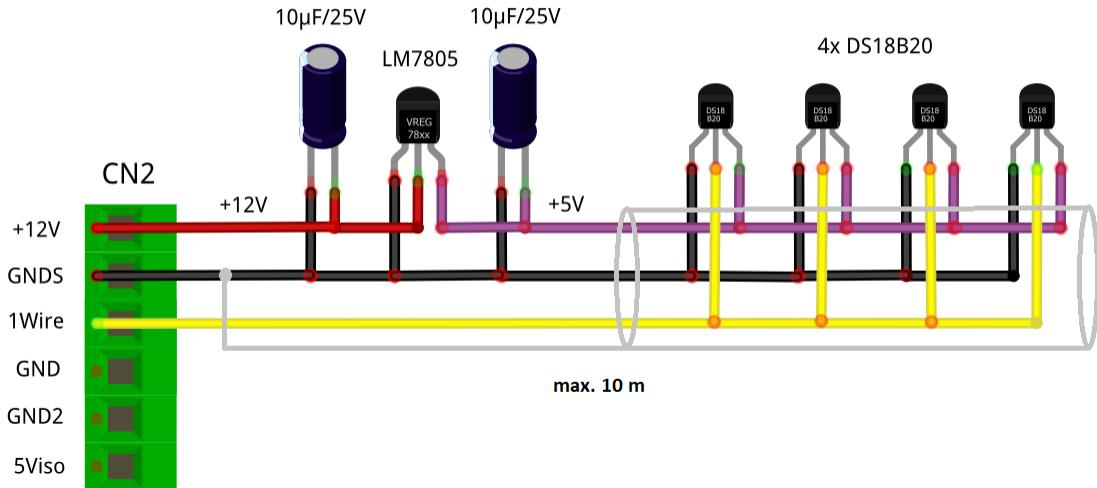


Abb.: 1Wire-Anbindung von externen Temperatur-Sensoren (direkt versorgt)

Vorsicht: Der 1Wire-Bus ist nicht isoliert gegenüber der internen Schaltung des OBP60. Das erhöht bei unsachgemäßer Installation das Risiko, dass eingekoppelte Störungen in die Busleitungen die Funktion und Stabilität des OBP60 beeinträchtigen können. Halten Sie daher die Buslänge so kurz wie möglich. Im schlimmsten Fall kann das zum kompletten Ausfall des OBP60 führen mit daraus resultierenden schweren Folgen für die Navigationsfähigkeit Ihres Bootes.

Gefahr: Es darf auf keinen Fall eine Spannung von 12V an den Ausgang 1Wire angelegt werden. Das hat unmittelbar zur Folge, dass das OBP60 beschädigt oder sogar zerstört wird.

7.5 USB

Die USB-C-Schnittstelle im OBP60 dient zum Flashen der Firmware und zum Debugging. Die USB-Schnittstelle ist als serielle Schnittstelle ausgeführt. Darüber hinaus kann auch eine bidirektionale, voll duplex-fähige NMEA0183-Kommunikation zu anderen Geräten wie einem Laptop, PC oder einem Marine Control Server aufgebaut werden.

Spezifikation USB im OBP60

- Serielles, bidirektionales asynchrones Datenprotokoll auf Binär-Basis
- Punkt zu Punkt (nicht isoliert)
- USB-OTG (serielles Device)
- Voll duplex
- Bus-Terminierung über PullUp-Widerstand im ESP32
- **Unterstützte Protokolle**
 - USB 1.1, TTL 3.3V
- Datenrate 1 MBit/s

- Stromversorgung des OBP60 über USB möglich
- Stromversorgung von externen Geräten aus dem OBP60 heraus nicht möglich
- Buslänge bis zu 3 m
- Kabelart geschirmt
- Steckerart USB-C

Bemerkung: Für Linux und Win10/11 sind entsprechende USB-Treiber im Betriebssystem integriert. Für die älteren Win7/8-Versionen benötigen Sie [zusätzliche Treiber](#), um die USB-Schnittstelle benutzen zu können.

Stromversorgung

Das OBP60 kann auch über USB-C mit Strom versorgt werden. Das ist nützlich, wenn man z.B. Software-Entwicklung durchführt und das Gerät am Schreibtisch nutzen möchte. Das stromliefernde Gerät muss bis zu 500 mA mit einer Spannung von 5.1V bereitstellen können, wie etwa ein Raspberry Pi Netzteil. Die USB-C-Schnittstelle verfügt über einen Rücklaufschutz, so dass kein Strom aus dem OBP60 herausfließen kann. Das OBP60 kann zudem auch gleichzeitig mit 12V und über USB-C mit 5V versorgt werden.

Hinweis: Die reguläre Stromversorgung des OBP60 im Boot erfolgt immer über 12V aus dem Bordnetz. Es wird nicht empfohlen, eine Versorgung nur über USB-C durchzuführen, da die Steckverbindung nicht gegen unbeabsichtigtes Lösen gesichert ist. Kabellängen größer 1,5 m sollten nur für Datenübertragungen und nicht für die Stromversorgung genutzt werden, da der Spannungsabfall auf den Leitungen zu groß ist. Die Kabellänge ist auf maximal 3 m begrenzt. Wenn Sie größere Strecken überbrücken wollen, müssen Sie aktive USB-Repeater-Kabel verwenden, die die Signalpegel auffrischen.



Abb.: Aktive USB-Verlängerung für 5 m

Gefahr: In einigen Situationen kann es vorkommen, dass unzulässige Ausgleichsströme über die nicht isolierte USB-C-Schnittstelle fließen und das OBP60 so beschädigen können. Das passiert z.B. dann, wenn Ladegeräte mit dem 230V-Landstrom verbunden sind, die Bord-Batterie aufladen und gleichzeitig ein Laptop mit 230V-Versorgung mit dem OBP60 über USB-C verbunden ist. Wenn Sie beabsichtigen, die USB-C-Verbindung dauerhaft zur Kommunikation im Boot zu nutzen, sollten sie einen USB-Isolator verwenden, um Schäden zu vermeiden. Beim Betrieb des Laptops nur über seinen eingebauten Akku entstehen die beschriebenen Probleme nicht.



Abb.: USB-Isolator

USB-Isolatoren haben allerdings den Nachteil, dass sie nur einen sehr geringen Strom von ca. 150 mA für ihre isolierte Seite in Richtung OBP60 liefern können. Das OBP60 wird so nur eingeschränkt mit Strom versorgt, was Funktionsbeeinträchtigungen nach sich ziehen kann. Das OBP60 muss dann je nach Bedarf wie beschrieben noch zusätzlich mit 12V über den Steckverbinde **CN2** versorgt werden.

Achtung: Wenn das OBP60 über USB versorgt wird, kann es vorkommen, dass das Gerät gelegentlich je nach Stromverbrauch ungewollt einen Reboot durchführt. Der Grund liegt oft in einer unzureichenden Stromversorgung des USB-Ports oder an ungeeigneten oder zu langen USB-Kabeln. Entweder liegt die Ausgangsspannung nicht exakt bei 5V oder der Strom ist nicht ausreichend. Um solche Probleme zu vermeiden, benutzen Sie die Stromversorgung über 12V an der Anschlussklemme **CN2** oder ein eigenständiges Netzteil mit 5,2 V/2 A.

Kommunikation

Die USB-C-Schnittselle kann zur voll duplexfähigen NMEA0183-Kommunikation mit weiteren Geräten verwendet werden. Denkbar wären folgende Nutzungs-Szenarien:

- Kommunikation mit einem Marine Control Server
- Datenlieferant für ein Android-Autoradio als Plotter
- Kommunikation mit einem Laptop oder PC zur Softwareentwicklung, Diagnose und zum Flashen der Firmware
- Diagnose der Buskommunikation mit externer Software wie dem [Actisense Reader](#)
- Einspeisung von Simulationsdaten in die Bussysteme mit dem [NMEA-Simulator](#)

KAPITEL 8

Datenaustausch

8.1 Interne Datenverarbeitung

Die Firmware des OBP60 besteht aus zwei Teilen. Der erste Teil ist das **NMEA2000-Gateway** und der zweite Teil die **Hardwareansteuerung** des OBP60. Das **NMEA2000-Gateway** ist ein Open Source Projekt von Andreas Vogel. Es ist eine Software, mit der man bidirektionale Datenkonvertierungen zwischen NMEA2000 und NMEA0183 durchführen kann. Die Software ist so gestaltet, dass sie unterschiedliche kommerzielle Hardware unterstützen kann. So läuft das NMEA2000-Gateway z.B. auf einer Reihe von Produkten der Firma **M5Stack** wie dem **M5Stack Atom**, aber auch auf ESP32-Entwicklerboards wie dem **ESP32 Node MCU**. Es werden ESP32-CPUs in unterschiedlicher Ausprägung unterstützt wie der ESP32-Wroom und der ESP32-S3. Die Hardwareansteuerung des OBP60 ist über eigenständige Tasks implementiert und nutzt die Grundfunktionalität des NMEA2000-Gateways.

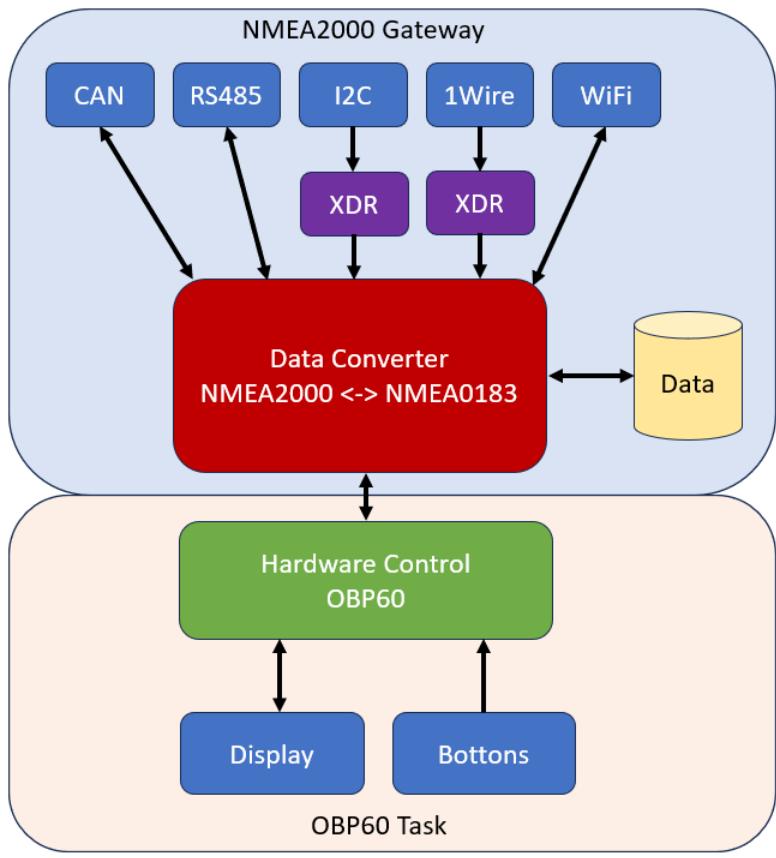


Abb.: Datenflussschema

Die gesamte Datenverarbeitung sämtlicher Bussysteme und Konvertierungen ist Bestandteil des NMEA2000-Gateways. Neben NMEA2000 (**CAN**) und NMEA0183 (**RS485**) werden weitere Bussysteme wie **I2C** und **1Wire** unterstützt. Die Hauptaufgabe des NMEA2000-Gateways besteht darin, alle ankommenden Daten der Bussysteme zu empfangen und in einem gemeinsamen Daten-Pool abzubilden. Diese Daten können über die Webseite **Data** eingesehen werden. Erweiterte Sensorik, die nicht Bestandteil von NMEA2000 und NMEA0183 sind, kann über I2C und 1Wire eingefügt werden. So lassen sich preisgünstige Sensoren nutzen. Damit die Daten der erweiterten Sensorik im NMEA2000- und NMEA0183-Netzwerk genutzt werden können, werden sie über NMEA0183 als universelle **XDR**-Datensätze in den Daten-Pool eingefügt. Als XDR-Datensätze können die Daten dann auch nach NMEA2000 konvertiert werden, sofern im Konverter entsprechende Übersetzungen implementiert sind. Als Ausgabeschnittstellen stehen CAN, RS485 und WiFi zur Verfügung. Über die CAN-Schnittstelle lassen sich nur NMEA2000-Daten austauschen. Über RS485 und WiFi (TCP) lassen sich sowohl NMEA0183-Daten, als auch NMEA2000-Daten austauschen, sofern die NMEA2000-Daten in SeaSmart-Telegrammen durch NMEA0183 getunnelt werden.

Der Datenaustausch im OBP60 kann auf verschiedene Weise erfolgen. Grundsätzlich stehen mehrere Übertragungsarten über verschiedene Übertragungswege zur Verfügung:

8.2 Übertragungsarten

- **Simplex**
 - Daten können in nur eine Richtung übertragen werden
- **Halbduplex**
 - Daten können abwechselnd, aber nicht gleichzeitig, in beide Richtungen fließen
- **Vollduplex**
 - Daten können in beide Richtungen gleichzeitig übertragen werden.

8.3 Übertragungswege

- **NMEA2000**
 - Kabelgebunden NMEA2000-Bus (halbduplex)
 - Über WiFi via SeaSmart (vollduplex)
- **NMEA0183**
 - Kabelgebunden NMEA0183-Bus (simplex)
 - USB (vollduplex)
 - Über WiFi via TCP (vollduplex)
- **I2C** (halbduplex)
- **1Wire** (halbduplex)

8.4 Datenquellen

Als Datenquellen werden Geräte bezeichnet, die überwiegend Daten zur anderen Geräten senden und selber nur Daten zur Parametrierung empfangen. Dazu zählen folgende Geräte:

- GPS-Empfänger (Position, Geschwindigkeit, Richtung)
- Windsensor (Geschwindigkeit, Richtung, Temperatur)
- Tiefen-Sensor (Tiefe, Geschwindigkeit, Wassertemperatur, zurückgelegte Strecke)
- Winkelsensoren (Ruderlage, Mast, Großbaum, Foil, Trimmklappen)
- Elektro-Sensor (Spannung, Strom, Leistung, Energie)
- Umgebungssensoren (Lufttemperatur, Luftdruck, Luftfeuchtigkeit, Helligkeit, Niederschlag, Zustand, Bewegung)
- Durchflusssensoren (Kühlwasserfluss, Kühlwassertemperatur)
- Druck- und Zugsensoren (Öldruck, Achterstag, Vorstag)
- Füllstandssensoren (Level für Wasser, Abwasser, Kraftstoff)
- Lagesensoren (Roll-, Pitch-, Nick-Winkel, Beschleunigung, Rotation, Magnetfeld)
- Temperatursensoren (Luft, Kühlwasser, Raum, Kühlschrank, Wasser, Maschinenraum)

- Elektrogeneratoren (Solar, Wind-, Schlepp- Dieselgenerator)
- Radargeräte (Umgebungskarte)
- Funkgeräte (Position, AIS-Schiffsverkehr, Anrufer, Meldungen, Notrufe)
- Anzeigegeräte (Multifunktionsdisplays, Plotter)
- Videokameras (Bild, Ton, Bewegung)

8.5 Datensenken

Datensenken empfangen Informationen und führen bestimmte Aktionen aus.

- Ruder-Aktuator (linear, rotatorisch, hydraulisch, elektrisch)
- Relais und Schalter (elektrische Verbraucher wie Ankerwinde, Licht, Positionsleuchten, Lüftung, Heizung, Ladegeräte)
- Winkelaktuator (Trimmklappen, Foil-Einstellung)
- Anzeigegeräte (Multifunktionsdisplays, Plotter)
- Multimediageräte (Radio, Lautsprecher)

Einige komplexere Geräte können sowohl Datenquelle, als auch Datensenke sein wie z.B. Multifunktionsdisplays oder Plotter.

Nachfolgend werden die Übertragungswege näher beschrieben.

8.6 NMEA2000 - Kabelgebunden

Der kabelgebundene NMEA2000-Bus ist der aktuelle Standard in der Bootsvernetzung. Über ein NMEA2000-Backbone auf CAN-Basis werden verschiedene Geräte an das Bussystem angeschlossen. Alle Bus-Teilnehmer können Daten lesen und schreiben. Dabei sind Sensoren Datenlieferanten, die ihre Daten an Displays und Plotter übertragen. Das NMEA2000-Backbone kann Sensoren auch mit Strom versorgen. Die Einspeisung der Versorgungsspannung erfolgt über einen Plotter oder über ein Einspeisekabel.

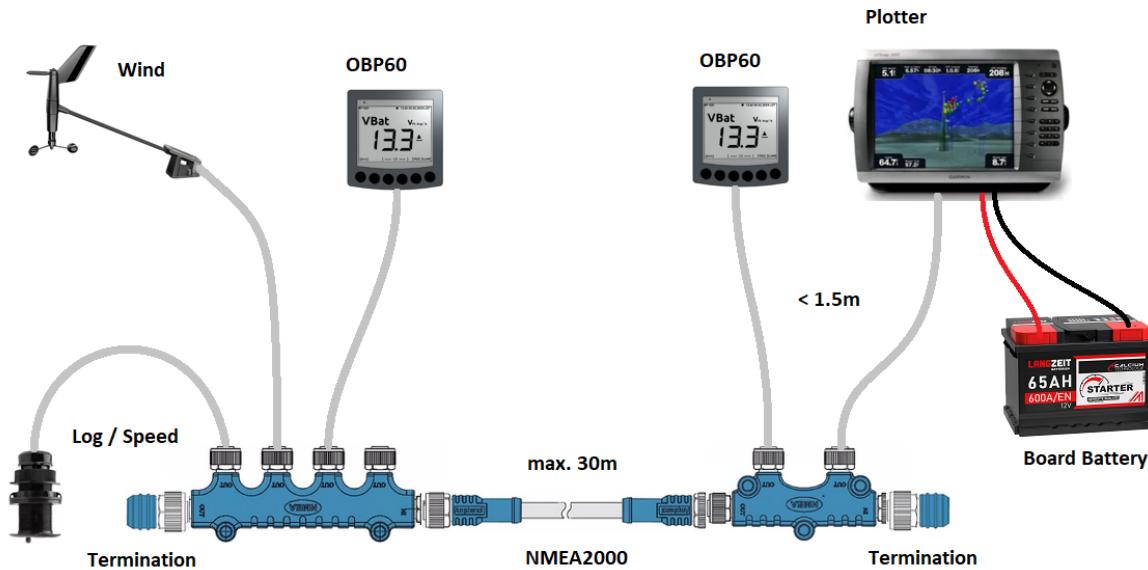


Abb.: NMEA2000-Bussystem mit Sensoren und Anzeigegeräten

Für den Betrieb von NMEA2000 muss nichts speziell konfiguriert werden. Die Standardeinstellungen sind so gesetzt, dass ein Betrieb problemlos möglich ist. Bei Bedarf kann das Senden von NMEA2000-Telegrammen unterbunden werden. Dann ist nur ein Empfang von NMEA2000-Telegrammen möglich. Die Einstellungen zu NMEA2000 findet man unter [Config - Converter](#).

8.7 NMEA2000 - WiFi via SeaSmart

Über das SeaSmart-Protokoll besteht die Möglichkeit, NMEA2000-Telegramme über Ethernet und WiFi zu übertragen. Dazu werden die Binärdaten der NMEA2000-Telegramme in proprietäre NMEA0183-Telegramme eingebettet. Ein SeaSmart-Telegramm sieht wie folgt aus:

\$PCDIN,a-a,b-b,b,cc,d-d*hh<CR><LF>

Feldnummer:

- a - PGN im Binärform
- b - Zeitstempel im Binärform
- c - Source-ID
- d - PGN-Daten im Binärform
- hh - Checksumme

Beispiel:

- \$PCDIN,01F211,0B9CF01B,03,008061480D0000FF*5C

Der Vorteil ist, dass sich SeaSmart-Telegramme genauso wie NMEA0183-Telegramme übertragen lassen. Damit ist es möglich, NMEA2000-Telegramme drahtlos über WiFi von einem OBP60 zu einem anderen OBP60 zu übertragen. Diese Funktion kann z.B. genutzt werden, um Bus-Sensordaten von einem OBP60 oder einem [NMEA2000-Gateway](#) auf einem OBP60-Tochtergerät anzeigen zu lassen.

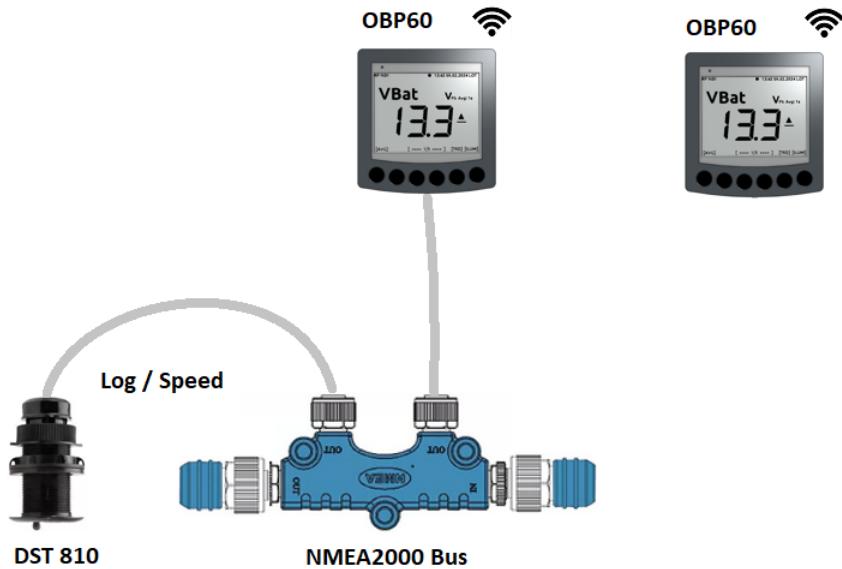


Abb.: Datenübertragung via WiFi OBP60 - OBP60

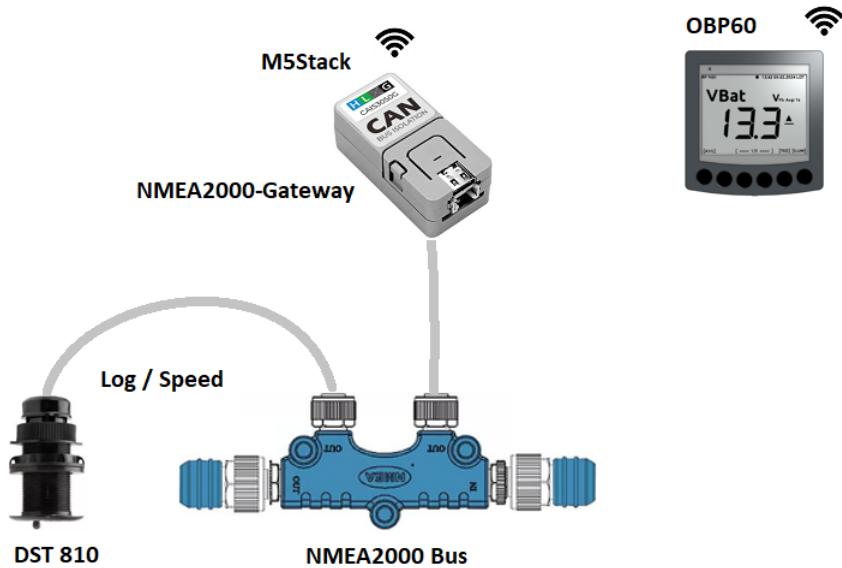


Abb.: Datenübertragung via WiFi M5Stack - OBP60

Hinweis: Beide Geräte müssen sich im selben WiFi-Netzwerk befinden und unterschiedliche Netzwerknamen und IP-Adressen besitzen. Dabei muss ein Gerät als TCP-Server und das andere Gerät als TCP-Client konfiguriert sein und auf beiden Geräten **SeaSmart out** aktiviert werden.

Nachfolgend ist ein Konfigurationsbeispiel für die obere Abbildung gezeigt, in dem ein Datenaustausch über WiFi zwischen zwei OBP60 stattfindet. Das Gerät 1 ist dabei als TCP-Server und Gerät 2 als TCP-Client konfiguriert. Das Gerät 2 bucht sich in das WiFi-Netzwerk vom Gerät 1 ein und tauscht darüber bidirektional Daten aus.

Einstellung	Gerät 1	Gerät 2
<i>Config - System</i>		
System Name	OBP60-1	OBP60-2
ApPassword	11111111	22222222
ApiP	192.168.15.1	192.168.16.1
<i>Config - TCP Server</i>		
SeaSmart Out	on	off
<i>Config - TCP Client</i>		
Enable	off	on
Remote Address	—	192.168.15.1
SeaSamart Out	off	on
<i>Config - WiFi Client</i>		
WiFi Client	off	on
WiFi Client SSID	—	OBP60-1
WiFi Client Password	—	11111111

8.8 NMEA0183 - Kabelgebunden

Bei der kabelgebundenen Datenübertragung für NMEA0183 handelt es sich um eine Simplex-Übertragung. Das bedeutet, dass man entweder senden oder empfangen kann. Als Default ist das OBP60 auf Empfang eingestellt. Es ist aber auch möglich, NMEA0183-Daten zu senden. Die Einstellung dazu erfolgt unter *Config - Serial Port*. Über **Serial Direction** kann die Datenrichtung eingestellt werden.

An dieser Stelle wird ein Beispiel gezeigt, wie Daten von einem NMEA0183-Multiplexer in ein OBP60 eingebunden werden können. Dabei sammelt der Multiplexer alle Sensordaten über die Eingänge ein und erzeugt einen zusammengefassten Datenstrom an seinem Ausgang. Die Daten werden lesend vom OBP60 empfangen und können dann entsprechend verwendet werden.

Bemerkung: Die Konfiguration des Multiplexers ist vom Modell abhängig. Konsultieren Sie dazu das Handbuch und achten Sie auf die korrekte Einstellung der Baudaten für die Eingänge und Ausgänge des Multiplexers.

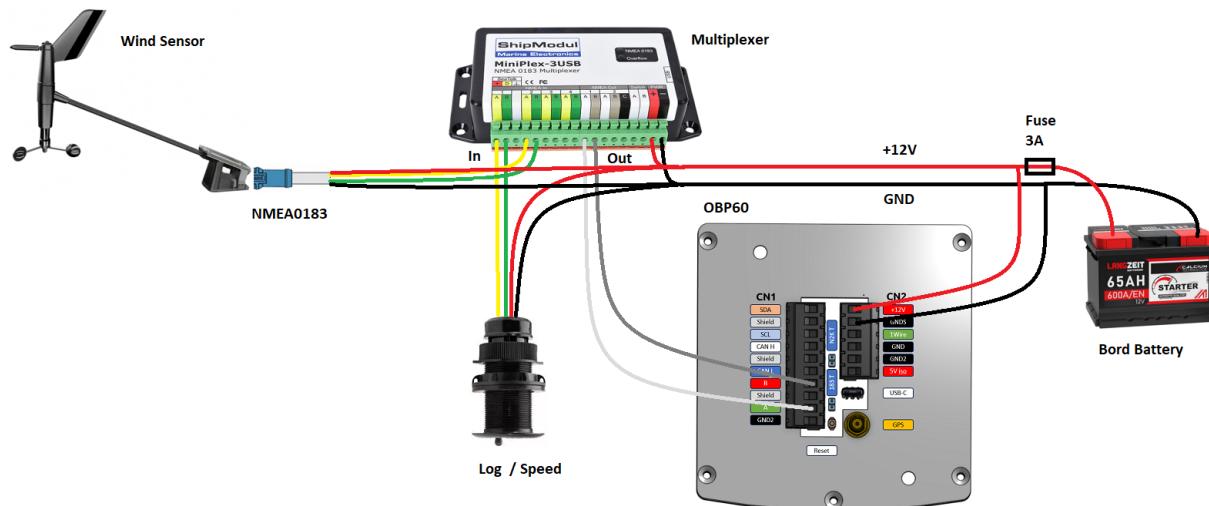


Abb.: NMEA0183-Verbindung zu einem Multiplexer (empfangen)

Einstellung	OBP60
<i>Config - Serial Port</i>	
Serial Direction	receive
Serial Baud Rate	115200
Serial to NMEA2000	on

Zum Senden von NMEA0183-Daten hier ein Beispiel für einen Autopiloten. Dabei werden Daten aus den verwendeten Kommunikationsmöglichkeiten genutzt und diese an einen Autopiloten gesendet. Die Ausgabe der Daten erfolgt über einen Filter, so dass nur relevante Informationen den Autopiloten erreichen. In dem gezeigten Beispiel verwendet der Autopilot einen NMEA0183-Eingang nach RS422 oder RS485 mit einer Schnittstellengeschwindigkeit von 4800 Bd. Unter Umständen müssen Sie die Geschwindigkeit an Ihren Autopiloten anpassen.

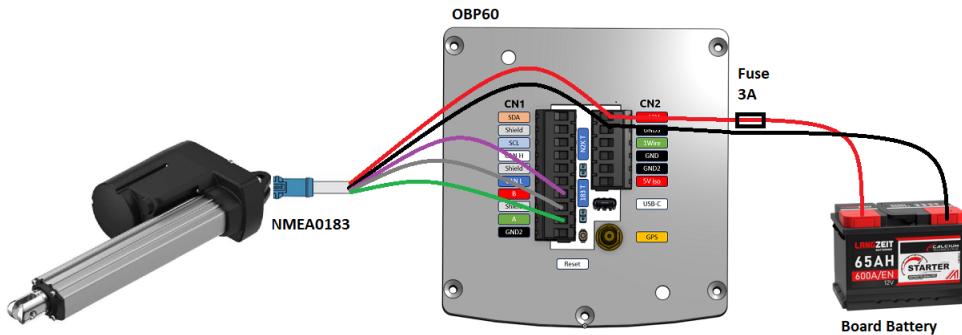


Abb.: NMEA0183-Verbindung zu einem Autopiloten (senden)

Einstellung	OBP60
<i>Config - Serial Port</i>	
Serial Direction	send
Serial Baud Rate	4800
Serial to NMEA2000	on
Serial Read Filter	—
Serial Write Filter	XTE,XDR,RMB,RMC,ROT

An den Autopiloten werden nur die NMEA0183-Telegramme **XTE**, **XDR**, **RMB**, **RMC** und **ROT** gesendet.

Bemerkung: Prüfen Sie in der Dokumentation des Autopiloten, ob die übermittelten NMEA0183-Telegramme zur Navigation verwendet werden können und ausreichend sind. In einigen Fällen kann es sein, dass der Autopilot andere Telegramme zur Kursregelung benutzt. In dem Fall kann der Autopilot nicht angesteuert werden.

8.9 NMEA0183 - USB

NMEA0183-Telegramme lassen sich auch über USB vollduplex übertragen. Das bedeutet, dass Daten gleichzeitig gesendet und empfangen werden können. Den USB-Port für die Datenübertragung findet man auf der Rückseite des OBP60 unterhalb des Steckverbinders **CN2**. Er ist als USB-C ausgeführt. Die USB-Schnittstelle im OBP60 ist als serielles RS232 Device implementiert und unterstützt die Übertragungsgeschwindigkeiten 1.200...460.800 Bd. Die Defaulteinstellung für die Datenübertragung ist auf 115.200 Bd eingestellt und sollte für die meisten Anwendungen ausreichend schnell sein. Die Daten werden ausschließlich als NMEA0183-Daten über USB übertragen.

Als mögliche Endpunkte könnte folgende Hardware verwendet werden:

- Raspberry Pi 3, 3B, 4B, 5
- Android Autoradio
- Laptop
- PC

Die NMEA0183-Daten lassen sich in unterschiedliche Software einbinden wie:

- AvNav
- OpenPlotter
- OpenCPN
- BBN
- SignalK
- qtVlm
- Navionics
- WinGPS
- NMEA Simulator

Für alle oben aufgeführten Endpunkte sind folgende Einstellungen im OBP60 vorzunehmen. Dabei werden NMEA0183-Daten auf der USB-Schnittstelle empfangen und gesendet und gleichzeitig nach NMEA2000 bidirektional konvertiert.

Einstellung	OBP60
<i>Config - System</i>	
Log Level	off
<i>Config - USB Port</i>	
USB Mode	nmea0183
USB Baud Rate	115200
NMEA to USB	on
NMEA from USB	on
USB to NMEA2000	on

Hinweis: Achten Sie darauf, dass der **Log Level** auf **off** gestellt ist. Andernfalls kann es sonst zu Störungen in der Kommunikation kommen, da Logging-Ausgaben in den Datenstrom eingespeist werden, die ebenfalls über USB-C ausgegeben werden.

8.10 NMEA0183 - WLAN

Mit dem TCP Client können (ähnlich wie bei der Übertragung per USB) NMEA0183-Telegramme z.B. von einem Raspberry Pi mit OpenPlotter bzw. SignalK empfangen werden. Dazu muss der TCP-Client entsprechend konfiguriert sein. *Config - WiFi Client*

8.11 Konfigurationsbeispiele

Nachfolgend sind einige Konfigurationsbeispiele aufgeführt. Es wird gezeigt, wie die weitere Konfiguration auf dem System erfolgt.

8.11.1 Beispiel AVnav auf Raspberry Pi

Dieses Beispiel zeigt die Einbindung eines OBP60 über USB in AvNav, das auf einem Rasberry Pi läuft. Dabei werden NMEA2000 Busdaten ausgelesen und nach NMEA0183 übertragen. Die Anbindung erfolgt direkt in AvNav als Device und die Daten stehen dann der Anwendung zu Verfügung. In diesem Fall wird das AvNav-Image benutzt. Wer AvNav unter OpenPlotter als Plugin benutzt, sollte dem **Konfigurationsbeispiel OpenPlotter auf Raspberry Pi** folgen.

Zur Verbindung des OBP60 und den Raspberry Pi über benötigen Sie ein **USB-C zu USB-A Kabel**. Am Raspberry Pi können Sie jeden beliebigen USB-A-Port verwenden.

Hinweis: Es ist ratsam, die schwarzen USB-A-Ports am Raspberry zu nutzen, da das OBP60 nur USB 1.1 unterstützt und dadurch die leistungsfähigeren USB 3.0-Ports frei bleiben und anderweitig genutzt werden können. Versorgen Sie auf jeden Fall das OBP60 zusätzlich mit 12V, da der Raspberry Pi zu wenig Strom an seinen USB-Ports liefert.

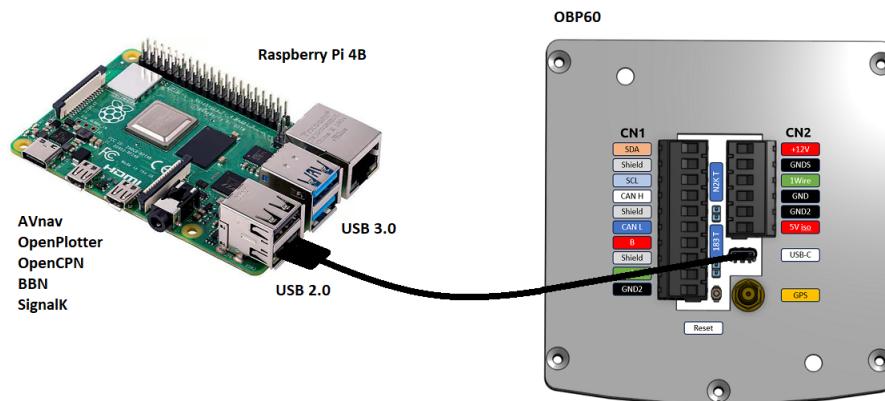


Abb.: Verbindung OBP60 - Raspberry Pi

Warnung: Verwenden Sie zur Verbindung des OBP60 mit dem Raspberry Pi nur hochwertige geschirmte USB-C-Kabel. Die Länge sollte 1,5 m nicht überschreiten, damit die Signalpegel nicht zu stark reduziert werden und eine hohe Übertragungsrate möglich ist. Bei Überbrückung größerer Längen verwenden sie aktive USB-Verlängerungskabel



Abb.: Aktive USB-Verlängerung für 5 m

Abb.: AvNav Startseite

Unter AvNav klicken Sie auf der Startseite oben rechts das Symbol mit den 3 Strichen.



Sie gelangen dann auf die Seite zum Serverstatus.

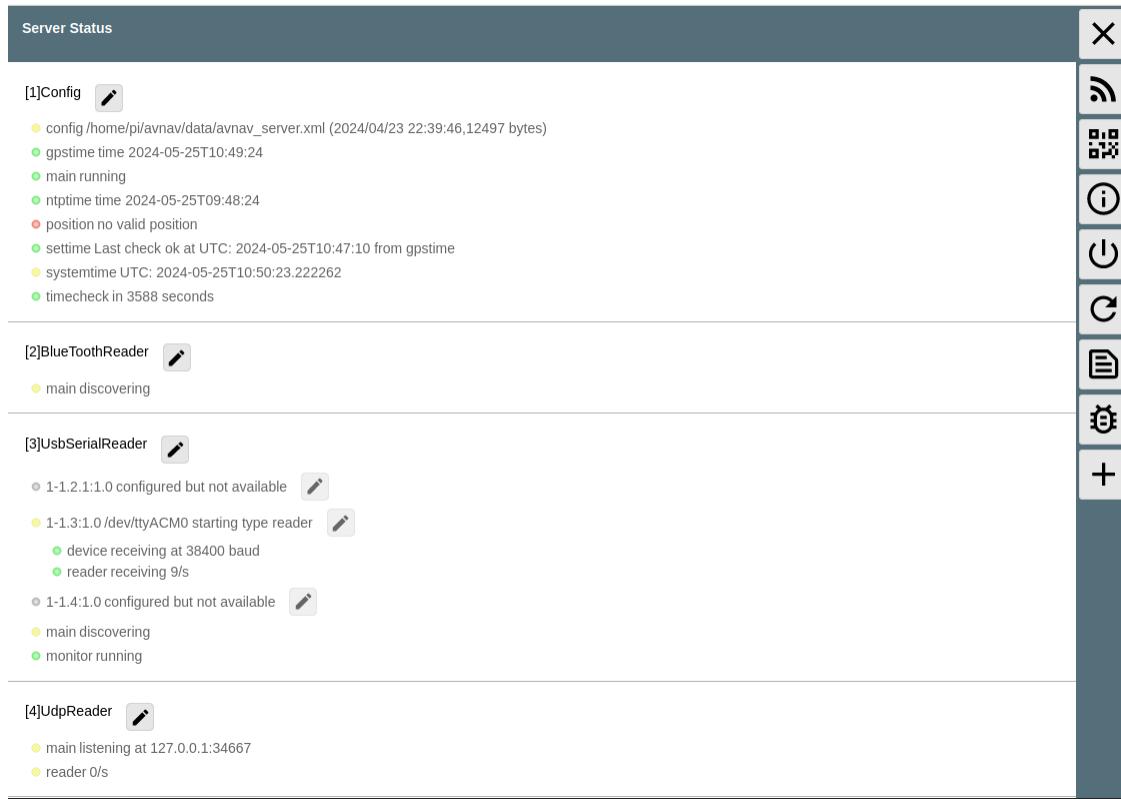


Abb.: Server-Status ohne OBP60

AvNav ist so eingerichtet, dass es automatisch alle seriellen USB-Geräte erkennen kann und automatisch zuordnet. Sowohl das Gerät, als auch die Übertragungsrate werden erkannt. Verschaffen Sie sich als erstes einen Überblick, welche Geräte bereits mit USB verbunden sind. Im oberen Bild sehen Sie unter Punkt **[3] USBSerialReader** alle aktuell erkannten und zugeordneten Geräte. In unserem Fall ist bereits ein GPS-Stick an USB angeschlossen. Das Gerät ist der Schnittstelle **/dev/ttyACM0** zugeordnet und arbeitet mit einer Übertragungsrate von 38.400 Bd.

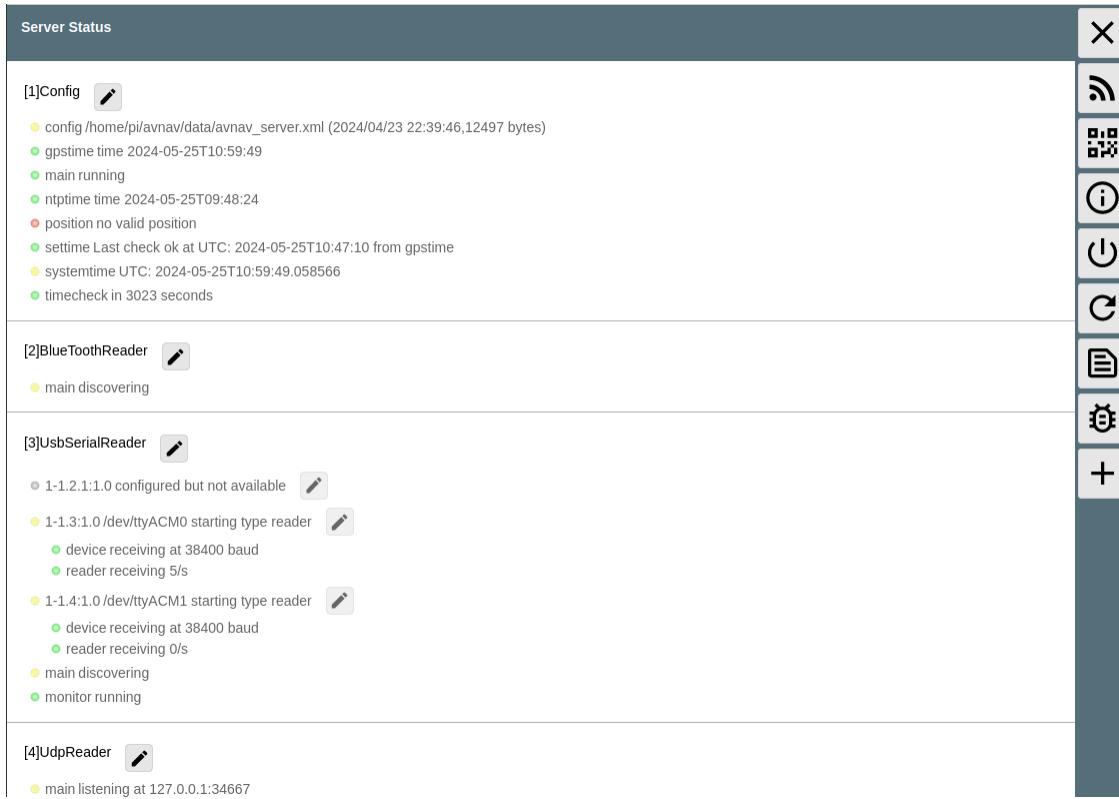


Abb.: Server-Status mit OBP60 (noch nicht konfiguriert)

Wenn man das OBP60 mit dem Raspberry Pi über USB verbindet, sieht man im oberen Bild unter Punkt 3 ein neu hinzugefügtes Gerät **/dev/ttyACM1**. Es handelt sich dabei um das OBP60. Die Schnittstellengeschwindigkeit ist aber noch nicht korrekt eingestellt.

Wenn Sie auf das Stiftsymbol hinter der Zeile mit dem Eintrag **/dev/ttyACM1** klicken, können die Einstellungen zum Gerät vorgenommen werden. Folgende Werte sind anzupassen:

- **Baud 115200**
- **Type combined**
- **Name OBP60V2**

Edit Handler

AVNUUsbSerialReader:UsbDevice:1-1.4:1.0

priority	50	<input type="button" value="Delete"/> <input type="button" value="Edit"/>
timeout	2	<input type="button" value="Delete"/> <input type="button" value="Edit"/>
baud	115200	<input type="button" value="Delete"/> <input type="button" value="Edit"/>
bytesize	8	<input type="button" value="Delete"/> <input type="button" value="Edit"/>
parity	N	<input type="button" value="Delete"/> <input type="button" value="Edit"/>
stopbits	1	<input type="button" value="Delete"/> <input type="button" value="Edit"/>
xonxoff	<input type="checkbox"/> <input checked="" type="checkbox"/>	<input type="button" value="Delete"/> <input type="button" value="Edit"/>
rtscs	<input type="checkbox"/> <input checked="" type="checkbox"/>	<input type="button" value="Delete"/> <input type="button" value="Edit"/>
numerrors	20	<input type="button" value="Delete"/> <input type="button" value="Edit"/>
autobaudtime	5	<input type="button" value="Delete"/> <input type="button" value="Edit"/>
filter		<input type="button" value="Delete"/> <input type="button" value="Edit"/>
readFilter		<input type="button" value="Delete"/> <input type="button" value="Edit"/>
blackList		<input type="button" value="Delete"/> <input type="button" value="Edit"/>
sendOwn	<input type="checkbox"/> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/>	<input type="button" value="Delete"/> <input type="button" value="Edit"/>
type	combined	<input type="button" value="Delete"/> <input type="button" value="Edit"/>
name	OBP60V2	<input type="button" value="Delete"/> <input type="button" value="Edit"/>

Abb.: Einstellungen für das OBP60

Durch die Änderung des Typs von `read` auf `combined` ist eine bidirektionale Kommunikation über USB mit einer Übertragungsgeschwindigkeit von 115200 Bd möglich. Das OBP60 ist nun mit AvNav verbunden. Solange Sie die selben USB-Ports verwenden, werden nach jedem Neustart des Systems die USB-Geräte korrekt zugeordnet und die Übertragungsgeschwindigkeit richtig eingestellt.

Server Status

- [1]Config
 - config /home/pi/avnav/data/avnav_server.xml (2024/05/25 13:02:18,12573 bytes)
 - gpstime time 2024-05-25T11:27:18
 - main running
 - ntpdate not checked
 - position no valid position
 - settime last setting UTC to 2024-05-25T11:10:02 from gpstime
 - systemtime UTC: 2024-05-25T11:27:18.753138
 - timecheck in 2572 seconds
- [2]BlueToothReader
 - main discovering
- [3]UsbSerialReader
 - 1-1.2.1:1.0 configured but not available
 - 1-1.3:1.0 /dev/ttyACM0 starting type reader
 - device receiving at 38400 baud
 - reader receiving 9/s
 - 1-1.4:1.0 /dev/ttyACM1 starting type combined
 - device combined port /dev/ttyACM1 open at 115200 baud
 - in 1.8/s
 - out 9.4/s, err=0/10s
 - main discovering
 - monitor running
- [4]UdpReader

Abb.: Server-Status mit OBP60 (korrekt konfiguriert)

8.11.2 Beispiel AvNav auf Android-Autoradio

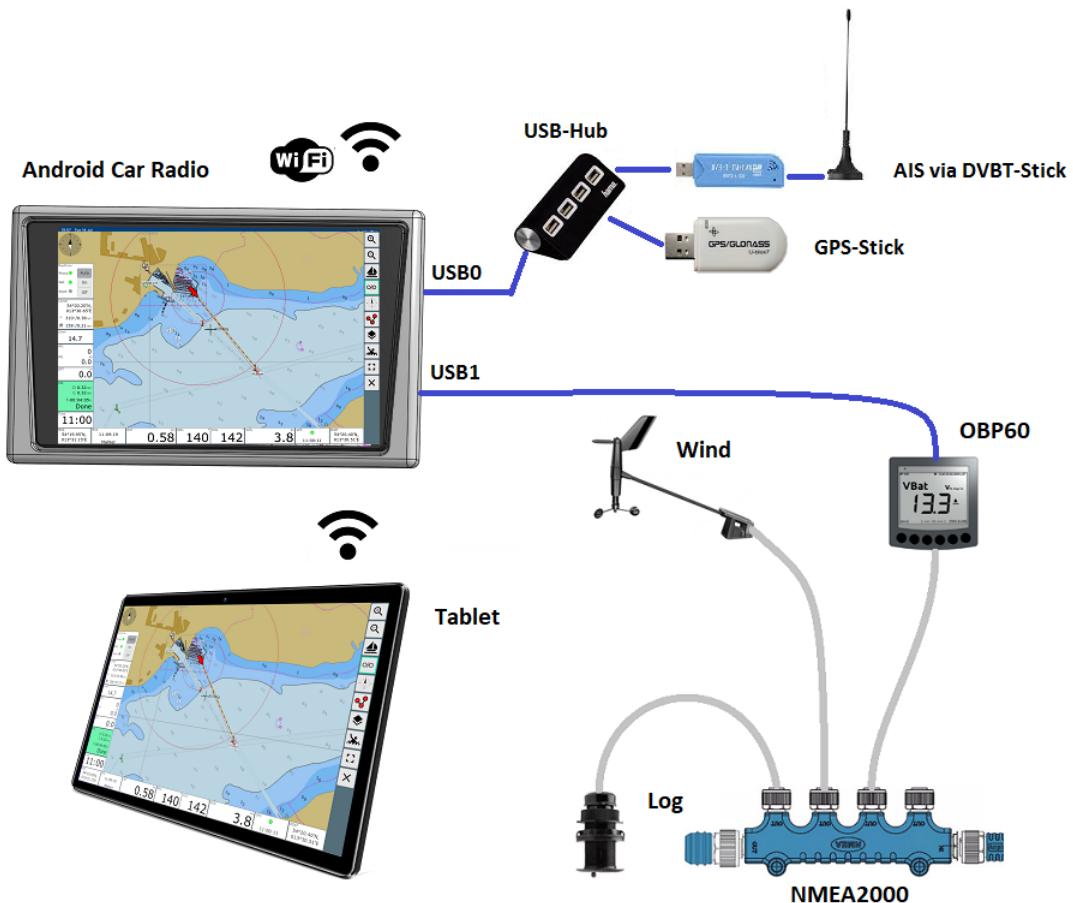


Abb.: Verbindung OBP60 - Android Autoradio AvNav

In diesem Beispiel wird gezeigt, wie man Busdaten in ein Android-Radio einspeisen kann, um die Daten dort in AvNav nutzen zu können. Für die Datenübertragung zum Android-Autoradio benötigen Sie ein **USB-C zu USB-A Kabel**, sofern eine passende Adapterbuchse zur Verfügung steht. In einigen Situation müssen Sie die USB-Kabel direkt über spezielle Stecker am Autoradio anschließen. Konsultieren Sie dazu das Handbuch zum Android-Autoradio und stellen Sie die USB-Verbindung wie gefordert her.

Warnung: Verwenden Sie zur Verbindung des OBP60 mit dem Android Autoradio nur hochwertige geschirmte USB-C-Kabel. Die Länge sollte 1,5 m nicht überschreiten, damit die Signalpegel nicht zu stark reduziert werden und eine hohe Übertragungsrate möglich ist. Bei Überbrückung größerer Längen verwenden sie aktive USB-Verlängerungskabel.



Abb.: Aktive USB-Verlängerung für 5 m

Hinweis: Die Konfiguration von AvNav für Android läuft in einigen Schritten verschieden zur Server-Version von AvNav auf einem Rasberry Pi ab. Beachten Sie, dass es unter Android keine automatische Konfiguration von seriellen USB-Devices gibt. USB-Geräte müssen hier immer manuell hinzugefügt werden.

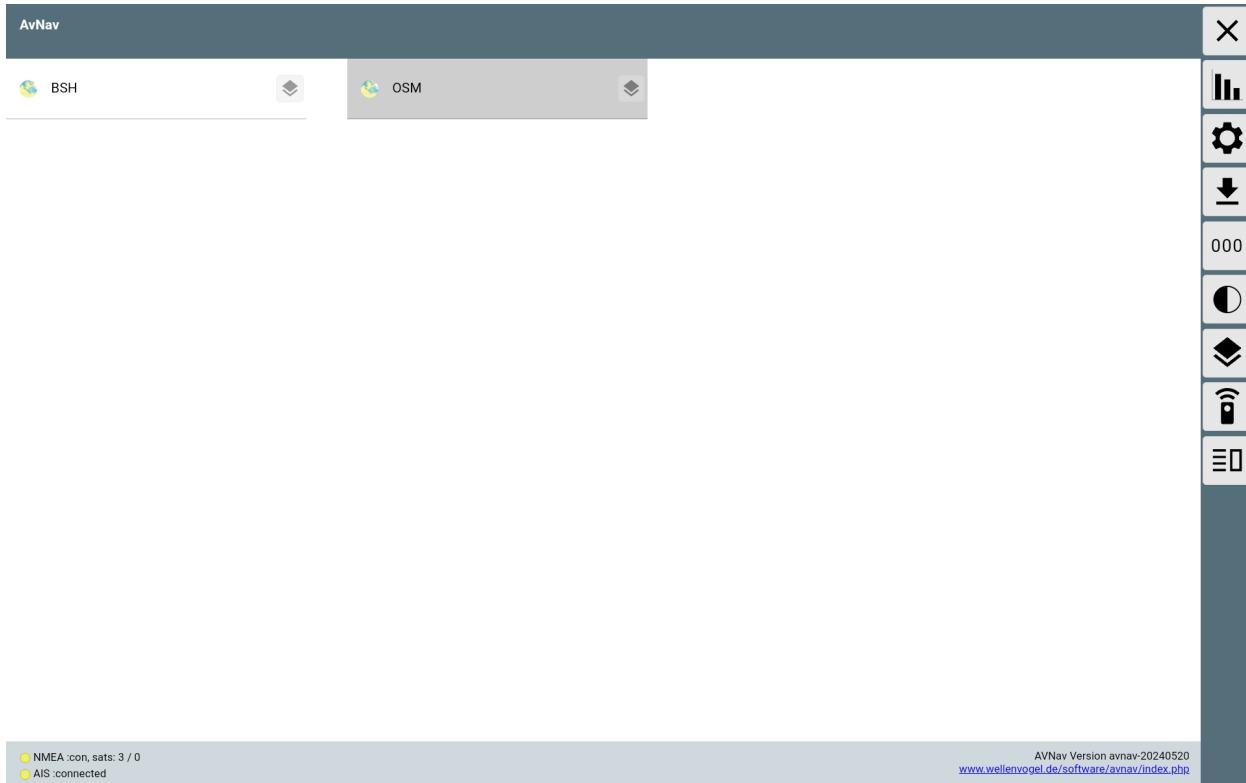
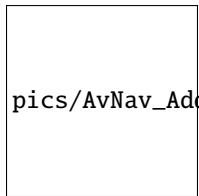


Abb.: Startseite AvNav für Android

Unter AvNav klicken Sie auf der Startseite oben rechts das Symbol mit den 3 Strichen.



Sie gelangen dann auf die Seite zum Serverstatus. Dort können Sie über das Plus-Symbol weitere Verbindungen zum AvNavServer einrichten.



pics/AvNav_Add_Icon.png

Für die bidirektionale Kommunikation über USB wählen Sie **UsbConnection**.

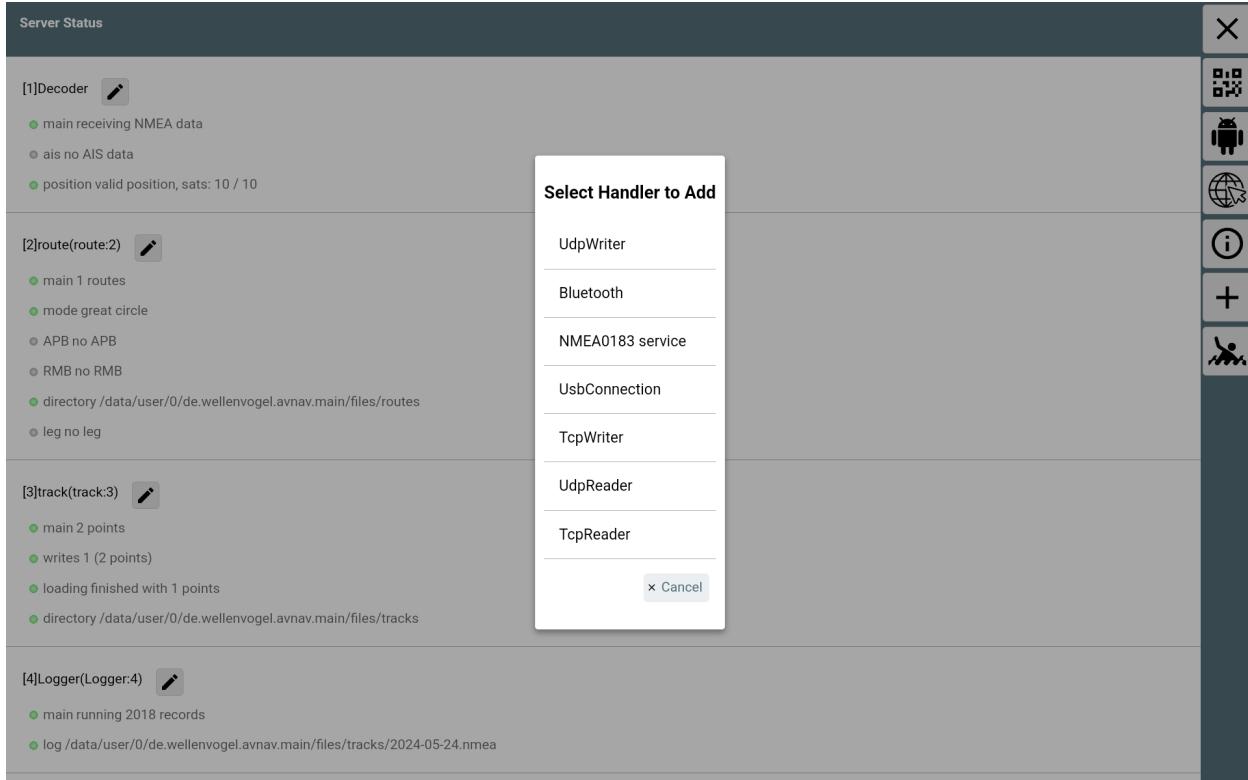


Abb.: Verbindungstypen

Unter **Device** wählen Sie die serielle Verbindung aus, unter der das OBP60 mit dem Raspberry Pi verbunden ist (`/dev/bus/usb/001/003`). Als Schnittstellengeschwindigkeit wird **115200 Bd** eingestellt. Um nicht nur Daten senden, sondern auch empfangen zu können, aktivieren Sie **SendOut**.

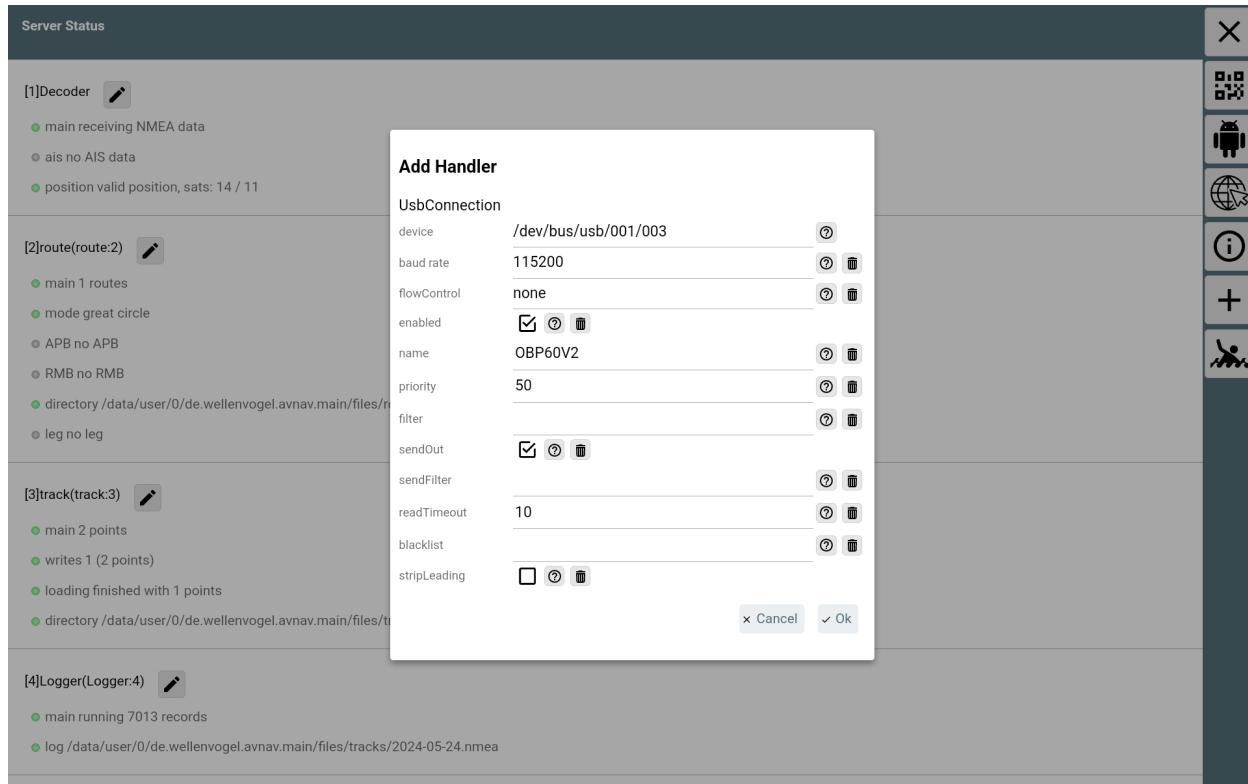


Abb.: Einstellungen zur USB-Verbindung

Nach der Übernahme aller Daten ist die neue Verbindung im Server-Status zu sehen.

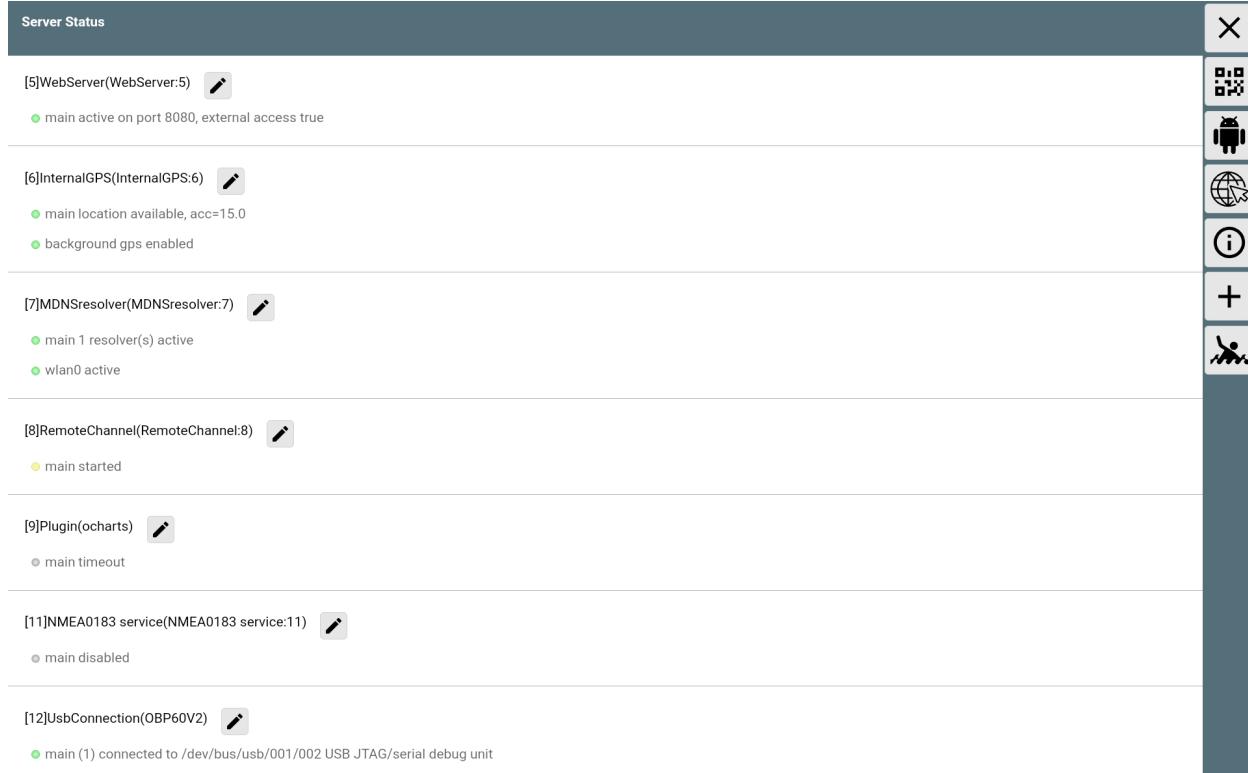


Abb.: Server-Status

8.11.3 Beispiel SignalK auf Raspberry Pi

SignalK kann die verfügbaren Daten im NMEA0183-Format über das WiFi-Netzwerk verteilen. Ein Vorteil dieser Variante ist, dass kein Datenkabel zum OBP60 benötigt wird, sondern nur der Raspberry und das OBP60 im selben WLAN angemeldet sein müssen. Hierfür muss in SignalK unter Server Settings NMEA 0183 over TCP (10110) aktiviert sein.

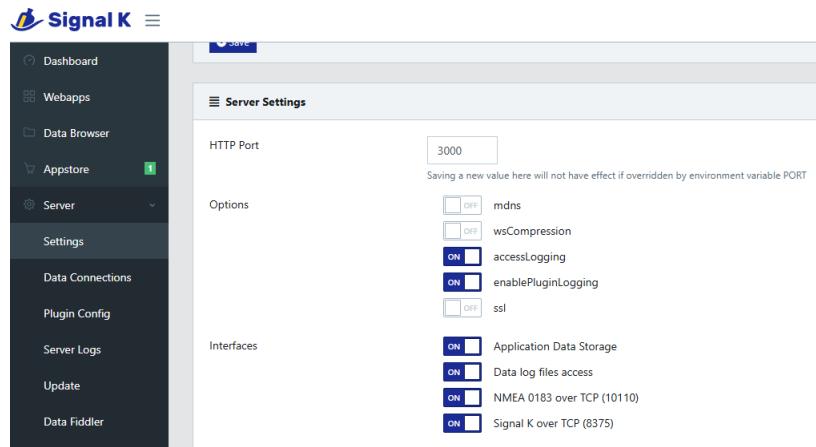


Abb.: SignalK Server

Außerdem muss das signalk-to-nmea0183 Plugin installiert und aktiviert sein, in dessen Konfiguration ausgewählt werden kann, welche NMEA0183-Daten ausgegeben werden sollen.

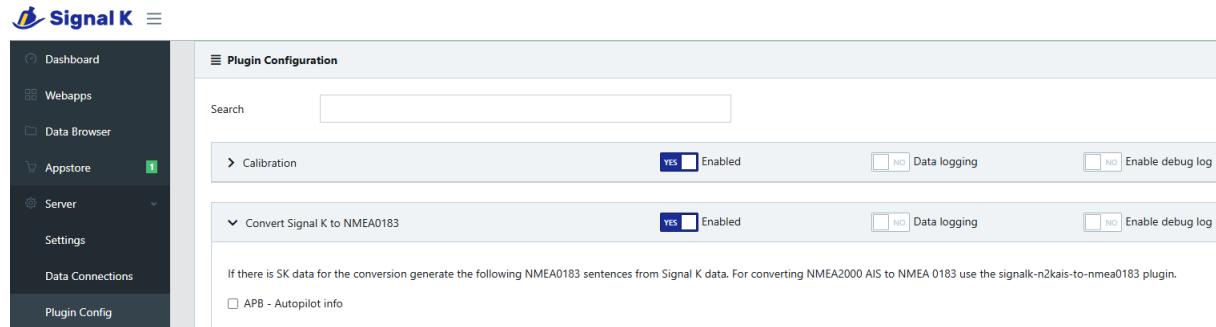


Abb.: SignalK Plungin

<input checked="" type="checkbox"/> RMB - Heading and distance to waypoint
RMB throttle ms
<input type="text" value="0"/>
<input checked="" type="checkbox"/> RMC - GPS recommended minimum
RMC throttle ms
<input type="text" value="0"/>

Abb.: SignalK Plungin Konfigurationsbeispiel

Um diese Funktionalität zu testen, kann man sich auf dem Raspberry Pi mit folgendem Kommando in einem Terminal den Datenstrom anzeigen lassen: `nc localhost 10110`

Die so bereitgestellten Daten können mit dem *Config - WiFi Client* des OBP60 abgerufen werden und stehen dann auf der Seite Data und zur Anzeige in den einzelnen Seiten ausgewählt werden.

8.11.4 Beispiel OpenPlotter auf Raspberry Pi

OpenPlotter stellt alle verfügbaren Daten über SignalK zur Verfügung. Von dort können die Daten im NMEA0183-Format mit dem TCP-Client abgerufen werden, oder mit dem signalk-to-nmea2000 Plugin übertragen werden.

8.11.5 Beispiel Navionics auf Android-Autoradio

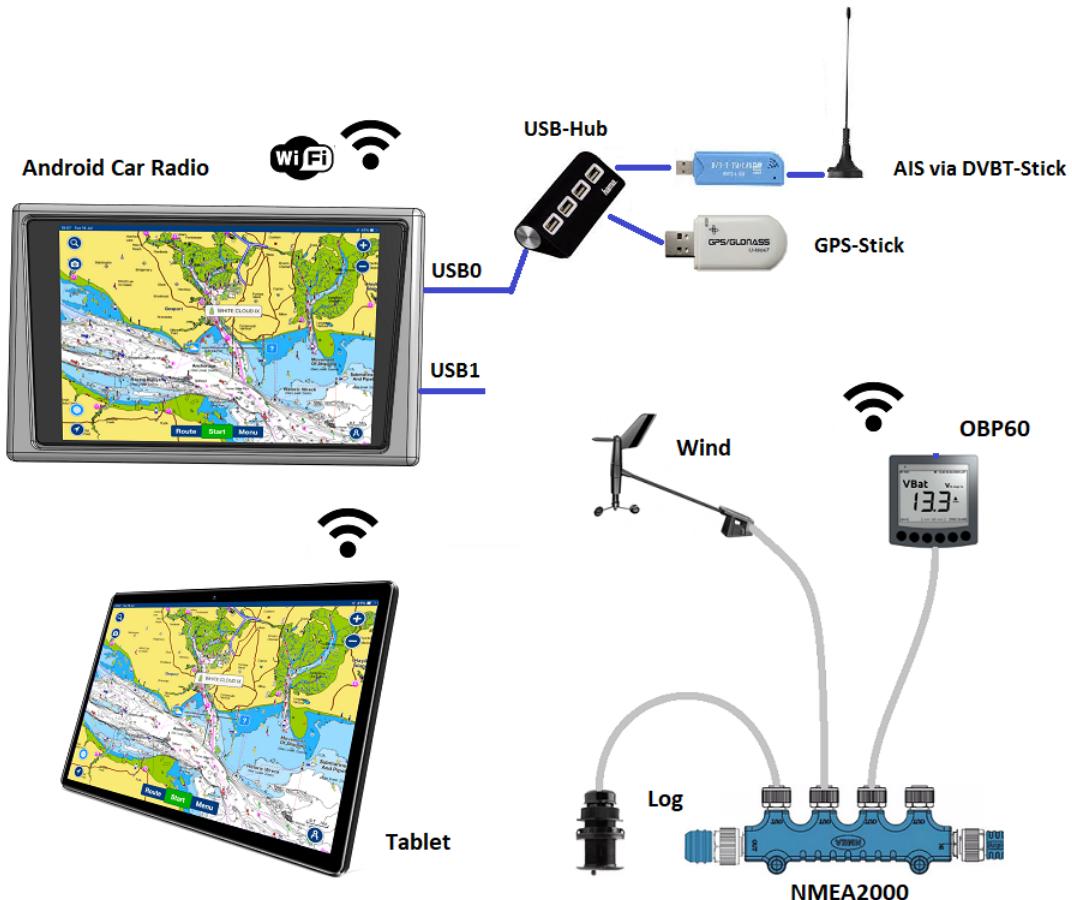


Abb.: Verbindung OBP60 - Android Autoradio Navionics

In dem Beispiel wird gezeigt, wie man Busdaten in ein Android-Radio einspeisen kann, um die Daten in Navionics nutzen zu können. Für die Datenübertragung zum Android-Autoradio muss eine WiFi-Verbindung benutzt werden. Eine USB-Kabelverbindung zur Datenübertragung ist nicht möglich. Stellen Sie sicher, dass das WiFi des Android Autoradios angeschaltet ist und sich beide Geräte im selben WiFi-Netzwerk befinden.

Bemerkung: Bedingt durch die Navigationssoftware können nicht alle Busdaten in Navionics verwendet werden. Aktuell lassen sich nur einige Daten nutzen. Die nachfolgende Liste zeigt die derzeitig verarbeitbaren NMEA0183 Sentences.

- AIVDM (AIS-Daten)
- AIVDO (AIS-Daten)
- DBT (Tiefe unter Sensor)

- **DPT** (Korrigierte Tiefe)
- **GGA** (Position)
- **GLL** (Position)
- **RMC** (Notwendige minimale Navigationsdaten - Zeit, Position, Kurs, Speed)
- **VTG** (Kurs und Speed über Grund)
- **ZDA** (Uhrzeit und Datum)

Zur Konfiguration der Verbindung zum OBP60 geht man auf **Menue** und den Unterpunkt **Paired Devices**.

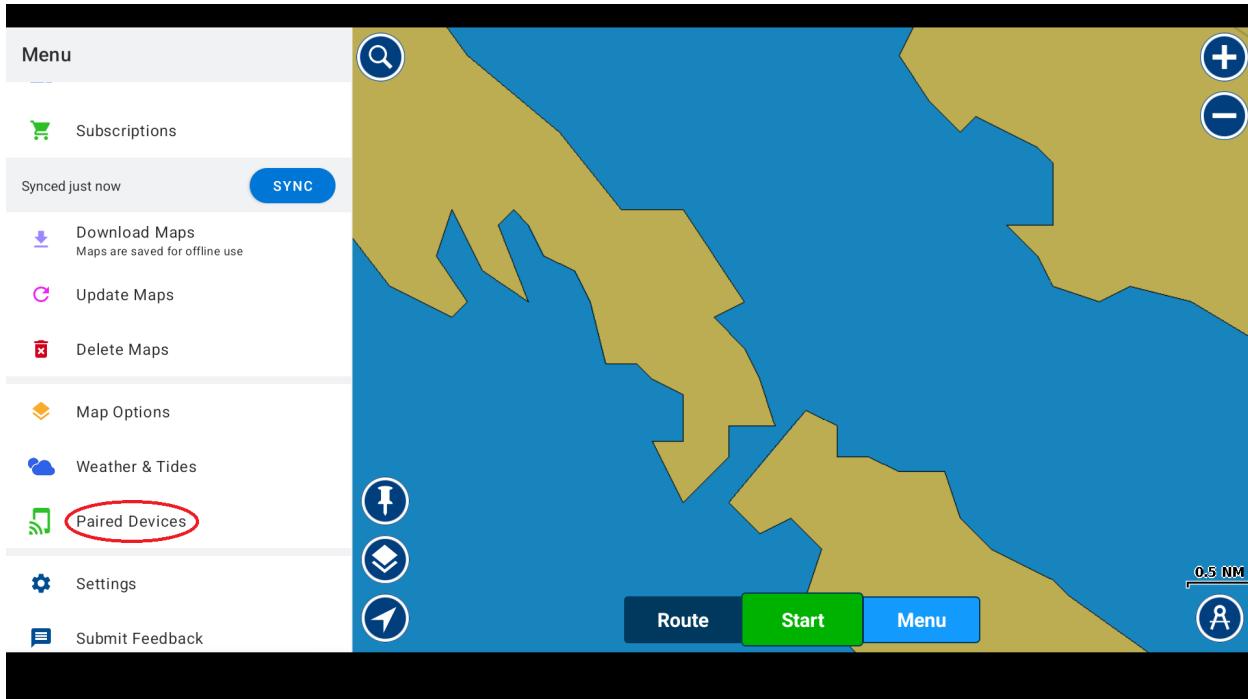


Abb.: Verbindung OBP60 - Menu Paired Devices

Über den Button **Add Device** kann eine neue Verbindung zum Device OBP60 angelegt werden, wenn sich beide Geräte im selben WiFi-Netzwerk befinden.

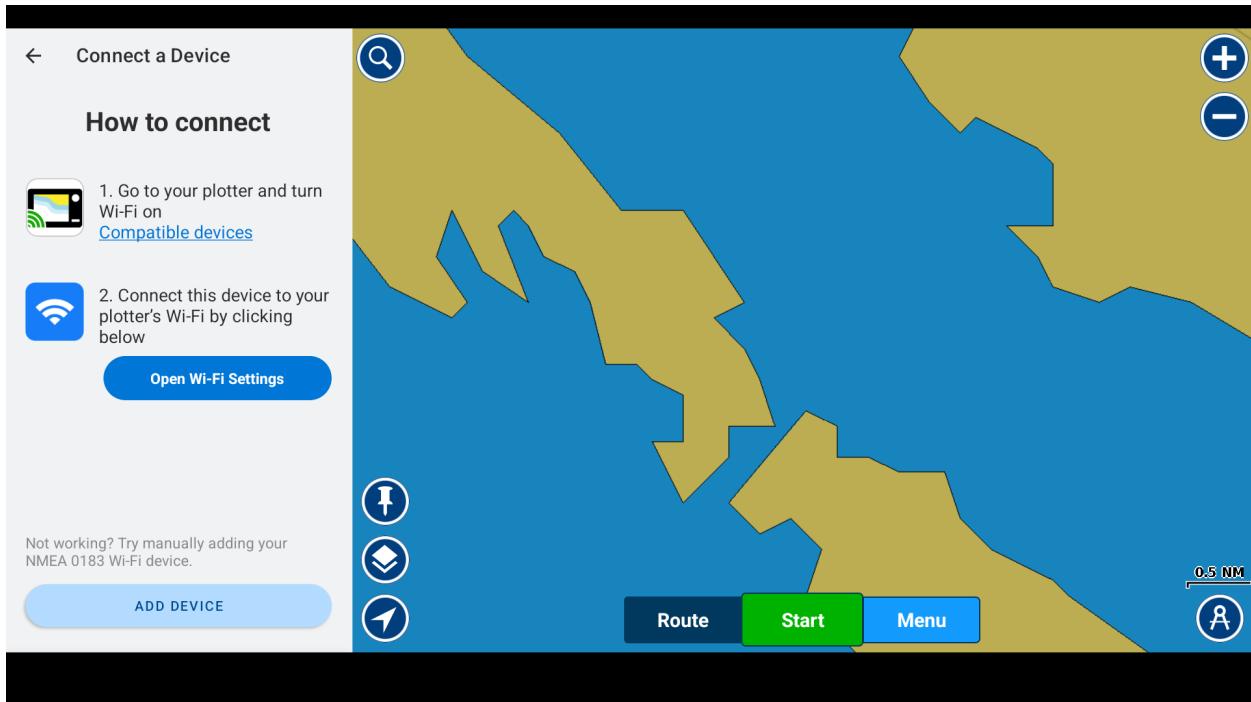


Abb.: Verbindung OBP60 - Add Device

Ermitteln Sie die IP-Adresse des OBP60, indem sie auf die Webseite des OBP60 gehen. Unter **WiFi client IP** finden Sie die IP-Adresse. In unserem Beispiel werden die Adresse 192.168.15.1 und der **Port 10110** verwendet. In Ihrem Fall kann das eine andere IP-Adresse sein. Vergeben Sie einen Namen für das Device und verwenden Sie als Verbindungsart **TCP**. Speichern Sie die Einstellungen oben rechts unter **Save**.

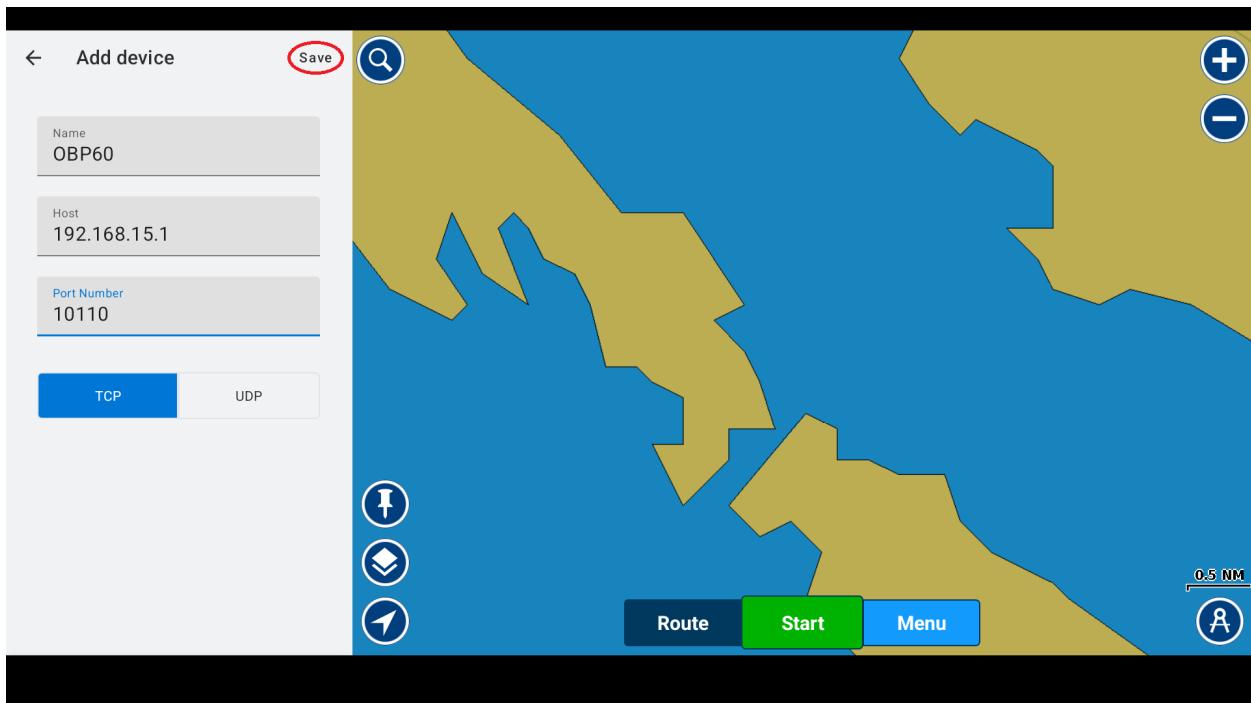


Abb.: Verbindung OBP60 - Add Device

Unter **Paired Devices** sollten jetzt das OBP60 zu finden sein und der Status auf **Connected** stehen. Sofern Daten

übertagen werden, sehen Sie den Typ der Daten. In unserem Fall wurden das externe GPS-Signal des OBP60 erkannt und die Daten in der Karte im Instrumententeil dargestellt.

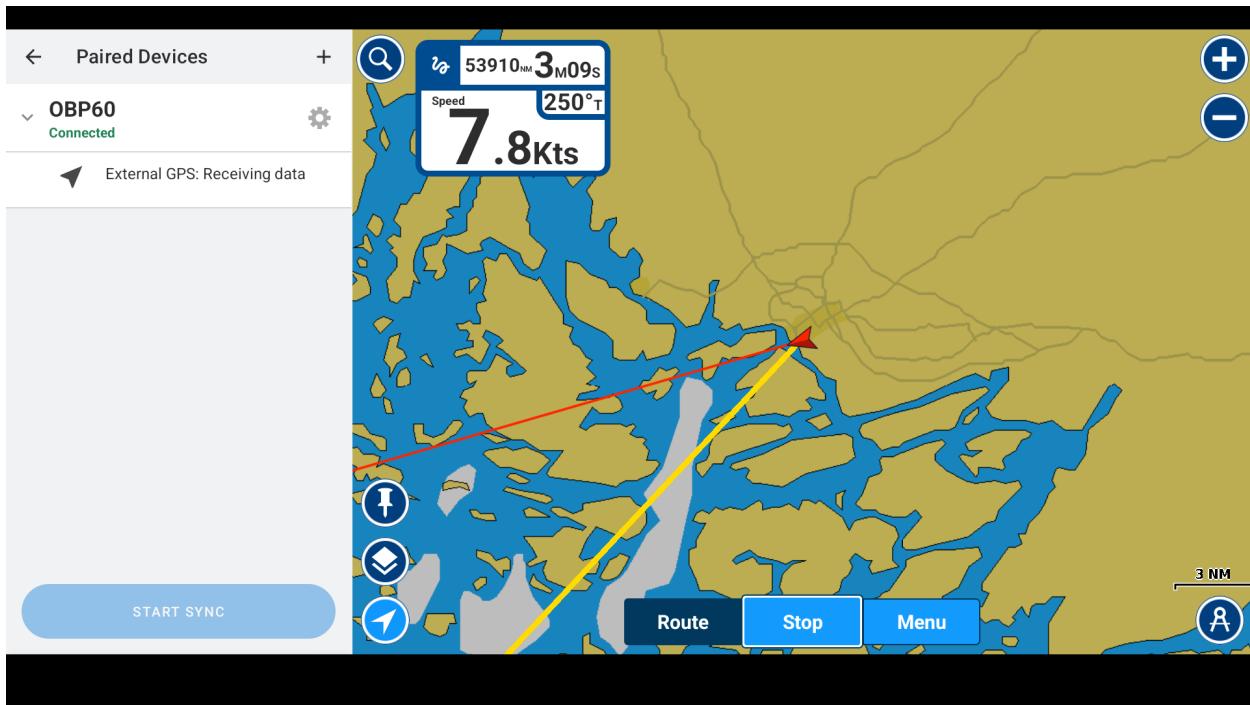


Abb.: Verbindung OBP60 - Add Device

Warnung: Navionics ist aktuell (Stand 2025) nicht in der Lage WiFi-Verbindungsabbrüche zu erkennen und kann sich nach einem Verbindungsverlust nicht wieder automatisch an das OBP60 anbinden. Der Verbindungsstatus wird fortlaufend mit Connected angezeigt, obwohl die Verbindung unterbrochen sein kann. Nur durch deaktivieren und neustarten der Navionics App wird der Verbindungsstatus richtig angezeigt.

8.12 I2C-Bus

8.12.1 Beispiel I2C Ruderlagensensor

Es wird an dieser Stelle gezeigt, wie man einen I2C-Winkelsensor als Ruderlagensensor am I2C-Bus benutzt. Grundsätzlich kann der Winkelsensor für folgende Winkelmessungen benutzt werden:

- Ruderlage
- Windrichtung
- Mastausrichtung bei drehbaren Masten
- Kielneigung
- Winkelsensor für Trimmklappen oder Foils
- Großbaum

Als I2C-Winkelsensor wird eine kleine Platine mit einem AS5600 verwendet, die auf Adresse 0x36 angesprochen werden kann. Der AS5600 ist ein magnetischer Winkelsensor, der die Ausrichtung eines Magnetfeldes erkennt. Über

einen diametralen Magneten, dessen Magnetfeld in der Scheibenebene geteilt und mit der Ruderachse verbunden ist, kann der Ruderausschlag gemessen werden. Der Magnet entspricht dabei der Drehachse des Ruders.

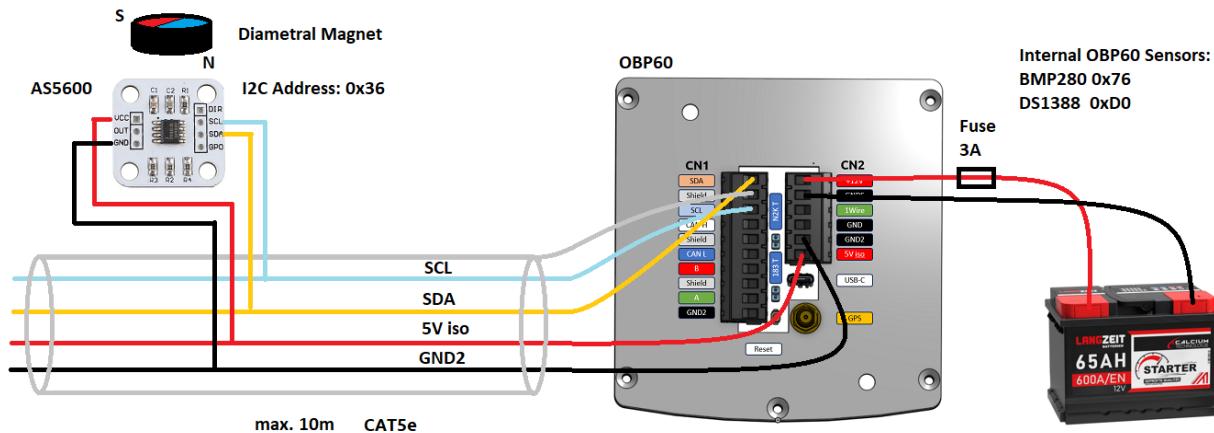


Abb.: I₂C-Anbindung magnetischer Winkelmeßgeräte AS5600

Bemerkung: Bedenken Sie, dass nur ein AS5600 als Winkelmeßgeräte verwendet werden kann, da die I₂C-Adresse nicht änderbar ist. Das Verbindungsleitung soll ein geschirmtes Kabel sein und eine Länge von 10 m nicht überschreiten.

Folgende Einstellungen sind im OBP60 vorzunehmen.

Einstellung	OBP60
<i>Config - OBP Hardware</i>	
Rot. Sensor	AS5600
Rot. Function	Rudder
Rot. Offset	0

Je nach Erfordernissen muss noch der Offset über **Rot. Offset** eingestellt werden.

8.13 1Wire-Bus

Über den 1Wire-Buss lassen sich bis zu 8 Temperatursensoren des Typs DS18B20 anschließen. Damit können Temperaturen im Bereich von -55°C bis 125°C an verschiedenen Stellen im Boot gemessen werden. Die Sensoren gibt es als elektronisches Bauteil in Transistorform (TO-92) oder in einer wasserdichten Metallhülse mit Kabel. Die letztere Variante eignet sich am besten im Marinebereich.



Abb.: DS18B20 TO-92



Abb.: DS18B20 Wasserdicht

Wenn Sie im Boot an verschiedenen Stellen Temperaturen messen möchten, erstellen Sie sich ein Backbone mit Abzweigdosen und schließen die Sensoren an den Abzweigdosen an. So entstehen im 1Wire-Bussystem nicht ungewollt lange Stichleitungen.

8.13.1 Konfigurationsbeispiel 1Wire

Im unteren Bild ist eine Schaltung zu sehen, in der 4 DS18B20 verwendet werden. Die Sensoren sind dabei über einen Spannungswandler LM7805 direkt versorgt. Diese Schaltung funktioniert für alle Sensoren, die am Markt erhältlich sind.

Direct Power Supply DS18B20 with 5.0V

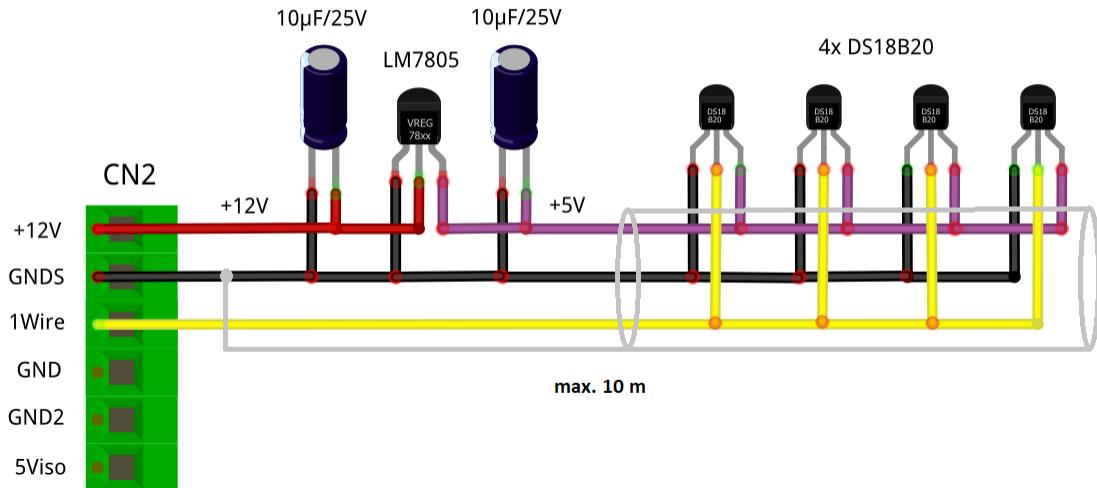


Abb.: 1Wire-Anbindung von externen Temperatur-Sensoren (direkt versorgt)

Einstellung	OBP60
<i>Config - OBP Hardware</i>	
Temp. Sensor	DS18B20

KAPITEL 9

Erweiterte Sensorik

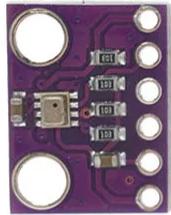
Bei der erweiterten Sensorik handelt es sich um Sensoren, die nicht am NMEA0183-Bus und NMEA2000-Bus angeschlossen werden. Zusätzlich lassen sich preiswerte Sensoren am I2C-Bus und am 1Wire-Bus anschließen. Beachten Sie die unterschiedlichen Signalpegel für den I2C- und den 1Wire-Bus. Die Module am I2C-Bus müssen so ausgewählt werden, dass sie mit 5.0V TTL-Pegeln arbeiten. Nicht alle Module unterstützen diese Pegel.

Typ	Bus	Spannung	Isolation
OBP60	I2C	5.0V TTL	ja
OBP60	1W	3.3V TTL	nein

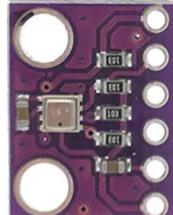
In der nachfolgenden Tabelle sind einige Sensoren aufgelistet, die am OBP60 verwendet werden können.

Warnung: Die Sensormodule sind für bestimmte Spannungsbereiche ausgelegt. Achten Sie darauf, die richtigen Sensoren zu verwenden. Werden 3.3V-Module in 5V-Systemen benutzt, können die 3.3V-Module beschädigt werden. Es sind in diesem Fall Pegelkonverter notwendig. Die meisten 5V-Module sind auch an 3.3V-Systemen nutzbar, wenn sie mit 3.3V betrieben werden.

Bemerkung: Beachten sie beim Kauf der BMP280- und BME280-Module, dass es unterschiedliche Typen gibt die ähnlich aussehen, sich aber unterschiedlich verhalten. Sie erkennen die Typen an der Gehäusegröße und Form des Chips.



BMP280 3.3V



BME280 3.3V

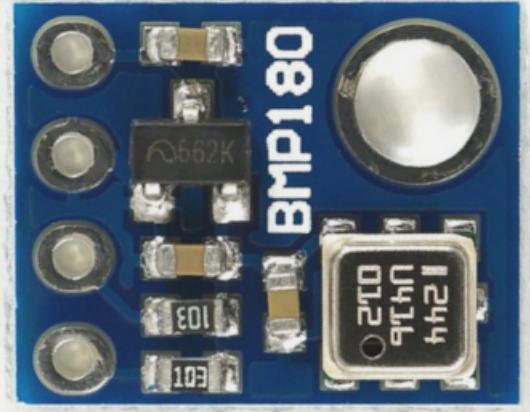
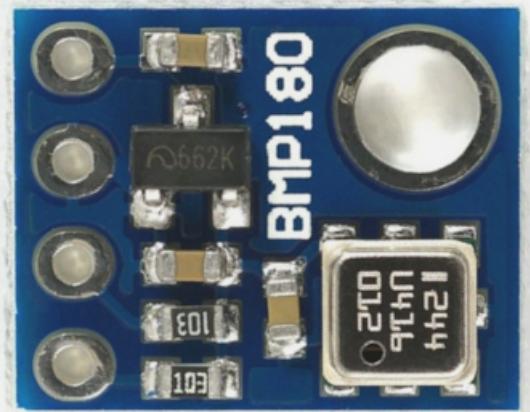
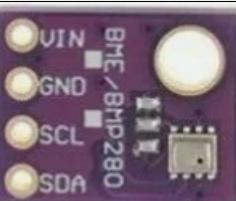
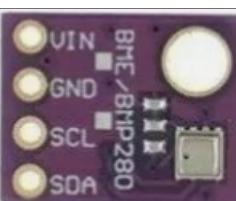
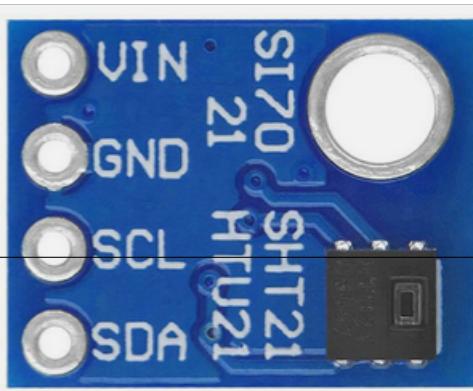


BMP280 5V



BME280 5V

Abb.: Versionen BMP280/BME280 für 3.3V und 5V

Typ	Bus	Sens- sor	Bild	3.3V	5V	Benutzte Pins
BMP085I2C	T, P		 A blue printed circuit board with a central square chip labeled "BMP085". Four metal pins are visible on the left side. A small surface-mount component labeled "Q662K" is also present.	X		Vin, SCL, SDA, GND
BMP180I2C	T, P		 A blue printed circuit board with a central square chip labeled "BMP180". Four metal pins are visible on the left side. A small surface-mount component labeled "Q662K" is also present.	X		Vin, SCL, SDA, GND
BMP280I2C	T, P		 A purple printed circuit board with a central square chip labeled "BME/BMP280". Four metal pins are visible on the left side. A small surface-mount component labeled "Q662K" is also present.	X	X	Vin, SCL, SDA, GND
BME280I2C	T, P, H		 A purple printed circuit board with a central square chip labeled "BME/BMP280". Four metal pins are visible on the left side. A small surface-mount component labeled "Q662K" is also present.	X	X	Vin, SCL, SDA, GND
SHT21	I2C	T, H	 A blue printed circuit board with a central square chip labeled "SHT21 HTU21". Four metal pins are visible on the left side. A small surface-mount component labeled "Q662K" is also present.			139

- T - Lufttemperatur
- P - Luftdruck
- H - Relative Luftfeuchtigkeit
- R - Rotation 0...360°
- U - Spannung
- I - Strom
- W - Leistung

KAPITEL 10

Beispielkonfiguration

10.1 OBP60 Yachta AvNav

Im folgenden Beispiel wird gezeigt wie man mit einem **OBP60** eine Datenübertragung zum **Windsensor Yachta** und zu einem NMEA2000-Netzwerk aufbauen und die Daten in der Navigationsapp AvNav auf einem Tablett nutzen kann. Das OBP60 dient dabei als zentrale Datenbasis, in der alle Daten zusammenlaufen. Die Datenübertragung erfolgt über WiFi-Netzwerkverbindungen, die ein LTE-Router zur Verfügung stellt. Der Vorteil eines LTE-Routers besteht darin, dass sie alle Geräte an Bord mit einer WiFi-Internetverbindung versorgen können und die Geräte im WiFi-Netzwerk untereinander kommunizieren können. Der LTE-Router schottet das eigene WiFi-Netzwerk gegenüber dem Internet ab, so dass von außerhalb niemand auf Ihr internes Netzwerk Zugriff erhält. Somit lassen sich auch Geräte wie Handys, Tablets oder Laptops an Bord mit dem Internet verbinden und alle Geräte haben Zugriff auf die Daten der Sensoren und können über einen Web-Browser darauf zugreifen. Die Navigationssoftware AvNav wird dabei auf einem Android-Tablett installiert und zur Navigation genutzt.

Tipp: Grundsätzlich ließe sich der Access Point vom OBP60 auch als zentraler WiFi-Router nutzen. Die Leistungsfähigkeit ist aber deutlich geringer als bei einem LTE-Router. Wir empfehlen ausdrücklich die Verwendung eines dedizierten LTE- oder WiFi-Routers. Sie vermeiden damit Probleme bei der Kommunikation der Geräte untereinander.

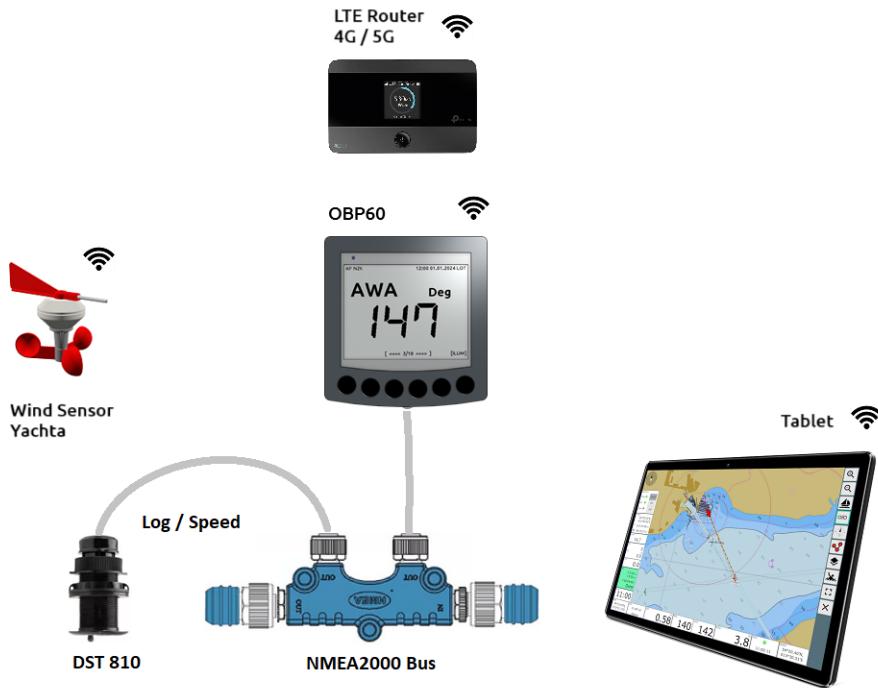


Abb.: Beispielkonfiguration OBP60, Yachta, AvNav

Die Konfiguration läuft in folgenden Schritten ab:

- LTE-Router einrichten
- OBP60 mit NMEA2000-Netzwerk und Windsensor Yachta verbinden
- AvNav mit dem OBP60 verbinden

10.1.1 LTE-Router einrichten

Als LTE-Router können Sie sowohl mobile Geräte als auch stationäre Geräte verwenden. Mobile Geräte haben den Vorteil, dass sie autark über eine Batterie über einen längeren Zeitraum funktionieren und auch außerhalb des Bootes an Land bei Ausflügen benutzt werden können. Stationäre Geräte bieten sich an, wenn man maximale Empfangsleistung benötigt und in Küstennähe unterwegs ist und große Entfernung zur nächsten Mobilfunk-Basisstation überbrücken möchte. In solchen Fällen bietet es sich an, Mobilfunkantennen oben im Mast zu installieren und die Antenne per Kabel mit dem stationären Router zu verbinden.

Tipp: Achten Sie beim Kauf des LTE-Routers, dass der Router im Dualband-Modus in den zwei WiFi-Frequenzbereichen 2.4 GHz und 5 GHz **gleichzeitig** arbeiten kann und über 4G/5G-Mobilfunk-Standard verfügt. So profitieren Sie von einer leistungsfähigen Internet- und WiFi-Verbindung im 5 GHz Bereich und umgehen die meist überfüllten 2.4 GHz Frequenzbereiche. Solche Router können Daten zwischen beiden Frequenzbereichen problemlos austauschen. Ein ausgedientes Handy im Hotspot-Mode leistet ähnliches wie ein LTE-Router und kann eine günstige Alternative darstellen.



Abb.: Mobiler 4G-LTE-Dualband-Router TP-Link M7450

- **Leistungsdaten**

- 4G/LTE Mobilfunkstandard 50 MBit/s
- WiFi Dualband 2.4 GHz **oder** 5 GHz 300 MBit/s¹
- 3000 mAh LiPo-Akku
- 15 Stunden autarke Laufzeit
- 32 WiFi-Geräte einbindbar
- 32 GB SD-Card-Speicher
- USB-, SMB- und FTP-Share für Filme und Musik von SD-Card
- [Dokumentation](#)

Warnung: Der LTE-Router TP-Link M7450 kann nicht gleichzeitig in beiden Frequenzbändern arbeiten. Daher müssen Sie den TP-Link M7450 fest auf das Frequenzband 2.4 GHz einstellen. Das OBP60 arbeitet nur im 2.4 GHz Bereich.



Abb.: Stationärer 4G-LTE-Dualband-Router RTU360

- **Leistungsdaten**

- 4G/LTE Mobilfunkstandard 50 MBit/scale
- WiFi Dualband 2.4 GHz und 5 GHz 300 MBit/s
- 2x LAN CAT6 100 MBit/s
- Externe Antennen für LTE und WiFi
- 12V Versorgungseingang
- 230V AC-Netzteil
- [Online-Dokumentation](#)
- [Quick-Installation Guide](#)

¹ Kein gleichzeitiger Dualbetrieb möglich

Der verwendete LTE-Router wird entsprechend der Bedienungsanleitung in Betrieb genommen. Für eine Internetverbindung benötigen Sie einen Mobilfunk-Datenvertrag. Die meisten Mobilfunkfirmen bieten preisgünstige Datentarife an. Empfehlenswert sind Volumenverträge, die ein festes Datenvolumen für eine vorgegebene Zeitdauer bieten. Wählen Sie einen Tarif aus, der Ihrem Datenverbrauch entspricht. Das Datenvolumen können Sie ebenfalls in allen Ländern der EU uneingeschränkt nutzen, so wie Sie das in Ihrem Heimatland gewohnt sind.

Für das Konfigurationsbeispiel wird davon ausgegangen, dass die Geräte folgende IP-Adressen vom Router zugewiesen bekommen:

- MyBoat - WiFi-SSID
- MySecret - WiFi Passwort
- 192.168.1.1 - LTE-Router
- 192.168.1.101 - OBP60
- 192.168.1.102 - Windsensor Yachta
- 192.168.1.103 - Android-Tablett mit AvNav

Hinweis: In Ihrem konkreten Fall können die IP-Adressen abweichen. Verwenden Sie dann die IP-Adressen, die den Geräten vom Router zugewiesen worden sind.

10.1.2 Datenübertragung

Das folgende Schema zeigt die Datenübertragung und welche Geräte Server oder Client bei der Datenübertragung sind.

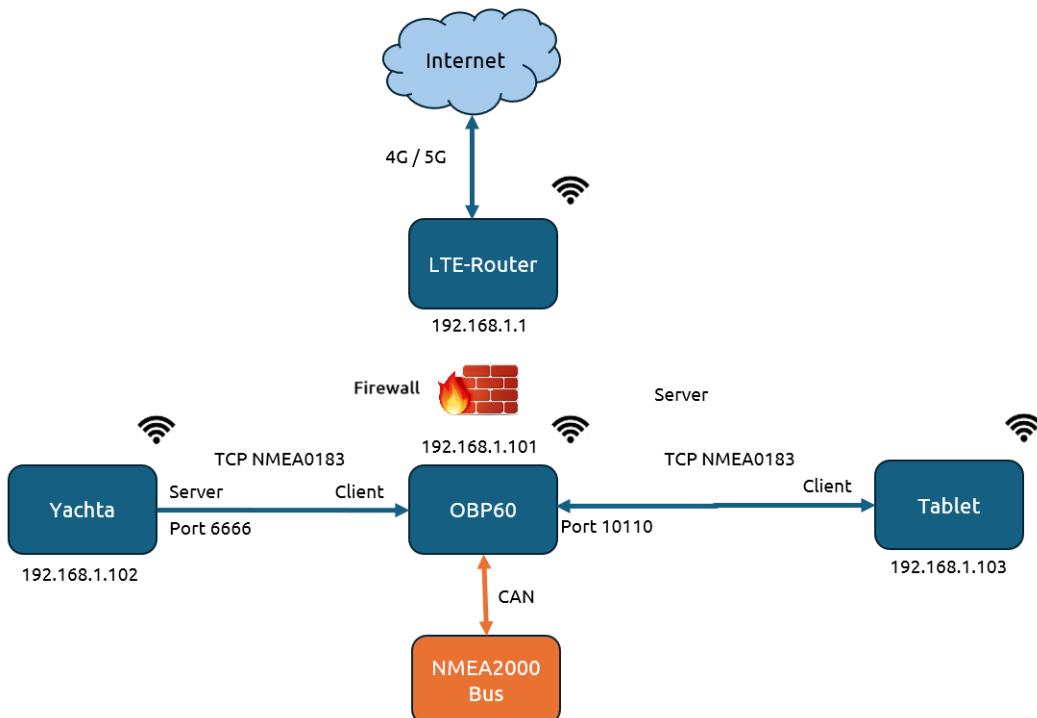


Abb.: Schema Datenübertragung

10.1.3 Konfiguration OBP60



Das OBP60 wird mit dem WiFi-Netzwerk des LTE-Routers verbunden. Der TCP-Server ist so konfiguriert, dass zum Tablett Daten übertragen werden können. Die TCP-Client-Verbindung dient zur Kommunikation mit dem Windsensor Yachta. Das OBP60 ist per Kabel mit dem NMEA2000-Netzwerk des Bootes verbunden. Sensordaten die im OBP60 vorliegen, wie z.B. die Windsensor-Daten, werden auch in den NMEA2000-Bus übertragen.

Nehmen Sie folgende Einstellungen vor:

Einstellung	OBP60
<i>Config - System</i>	
System Name	OBP60V2
<i>Config - WiFi Client</i>	
WiFi Client	on
WiFi Client SSID	MyBoat
WiFi Client Password	MySecret
:ref:`Config - Converter`	
NMEA2000 Out	on
<i>Config - TCP Server</i>	
TCP Port	10110
NMEA0183 Out	on
NMEA0183 In	on
To NMEA2000	on
SeaSmart Out	on
<i>Config - TCP Client</i>	
Enable	on
Remote Port	6666
Remote Address	192.168.1.102
NMEA0183 Out	off
To NMEA2000	on
SeaSamart Out	off

Nach der Konfiguration sollten Sie im Status nachfolgende Informationen sehen. Das OBP60 ist als WiFi-Client beim LTE-Router angemeldet und hat die IP-Adresse 192.168.1.101 zugewiesen bekommen. Das OBP60 ist als TCP-Client mit dem Windsensor Yachta verbunden. Über diese Verbindung werden Winddaten als NMEA0183-Telegramme empfangen. Unter Clients werden die Anzahl der Geräte angezeigt, die als TCP-Client mit dem OBP60 verbunden sind. Wenn das Tablet mit dem OBP60 verbunden ist, sollte ein Gerät angezeigt werden. Der NMEA2000-Status wird als Online angezeigt, wenn Daten mit dem NMEA2000-Bus ausgetauscht werden. Die Anzahl der ausgetauschten NMEA2000-Telegramme sieht man unter NMEA2000 In/Out. Wenn ein Tablett mit dem OBP60 per TCP verbunden ist, sieht man die Anzahl der ausgetauschten NMEA0183-Telegramme unter TCP In/Out.

Statusmeldungen	OBP60
<i>Status</i>	
WiFi Client Connected	true
WiFi Client IP	192.168.1.101
#Clients	2
NMEA2000 State	[0] Online
NMEA2000 In	Telegrammeanzahl
NMEA2000 Out	Telegrammeanzahl
TCP In	Telegrammeanzahl
TCP Out	Telegrammeanzahl

Die Verbindung des OBP60 mit dem NMEA2000-Netzwerk erfolgt wie im nachfolgenden Bild angezeigt. Ist das OBP60 ein normaler Teilnehmer des Bussystems, so muss die Terminierung deaktiviert sein.

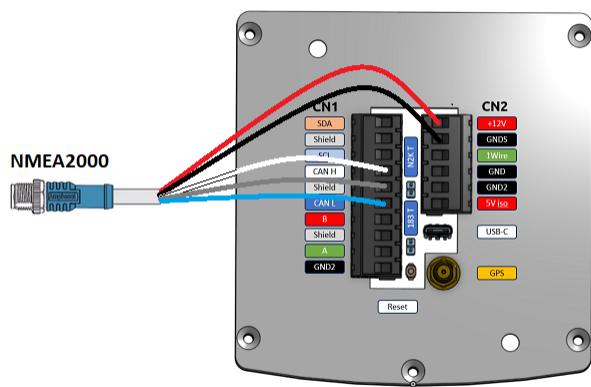


Abb. CAN-Bus Anbindung OBP60

10.1.4 Konfiguration Yachta



Der Windsensor Yachta ist so konfiguriert, dass er im WiFi-Netzwerk des LTE-Routers eingebucht ist. Der Windsensor stellt über den Port 6666 dem OBP60 Winddaten zur Verfügung. Es werden dabei nur Daten vom Windsensor Yachta zum OBP60 übertragen.

Folgende Einstellungen werden für den Windsensor Yachta vorgenommen:

Einstellung	Yachta
Network Settings	
WLAN Client SSID	MyBoat
WLAN Client IP	MySecret
Connection Timeout	30s
WLAN Server SSID	Yachta
WLAN Server Password	
AP Channel	1
Server Mode	HTTP (JSON/NMEA)
mDNS Service	on
Device Settings	
Wind Sensor Type	Yachta 2.0

Tipp: Der Windsensor Yachta lässt sich in einen Demo-Mode versetzen. So kann die Funktionalität außerhalb des Bootes getestet werden. Der Windsensor liefert dann simulierte Winddaten. Über **Server Mode** kann der Simulationsmodus mit der Einstellung **Demo Mode** aktiviert werden.

Nach der Konfiguration sollten unter **Device Info** im Windsensor Yachta folgende Statusmeldungen zu sehen sein:

Statusmeldungen	Yachta
Network Parameter	
WLAN Client SSID	MyBoat
WLAN Client IP	192.168.1.102
Connection Quality	>50%

10.1.5 Tablett Konfiguration



Das Android-Tablett wird in das WiFi-Netzwerk des LTE-Routers hinzugefügt und anschließend die App AvNav aus dem Play-Sore installiert. Details zur Konfiguration entnehmen Sie dem Handbuch zum Tablett. Die Daten eines GPS-Empfängers im Tablett lassen sich ebenfalls im NMEA0183- und NMEA2000-Netzwerk nutzen.

Einstellung	Android-Tablett
Einstellungen WiFi	
WiFi	on
WiFi Client SSID	MyBoat
WiFi Client Password	MySecret
App-Installation	AvNav

Nachfolgend wird gezeigt, wie man Busdaten über ein Tablett in AvNav nutzen kann. Die Datenübertragung erfolgt über WiFi. Das Tablett tauscht dabei die Daten mit dem OBP60 über einen TCP-Verbindung aus. Dabei wird das Tablett als TCP-Client an den OBP60 angedockt. Unter AvNav wird die Verbindung als TCPReader eingerichtet.

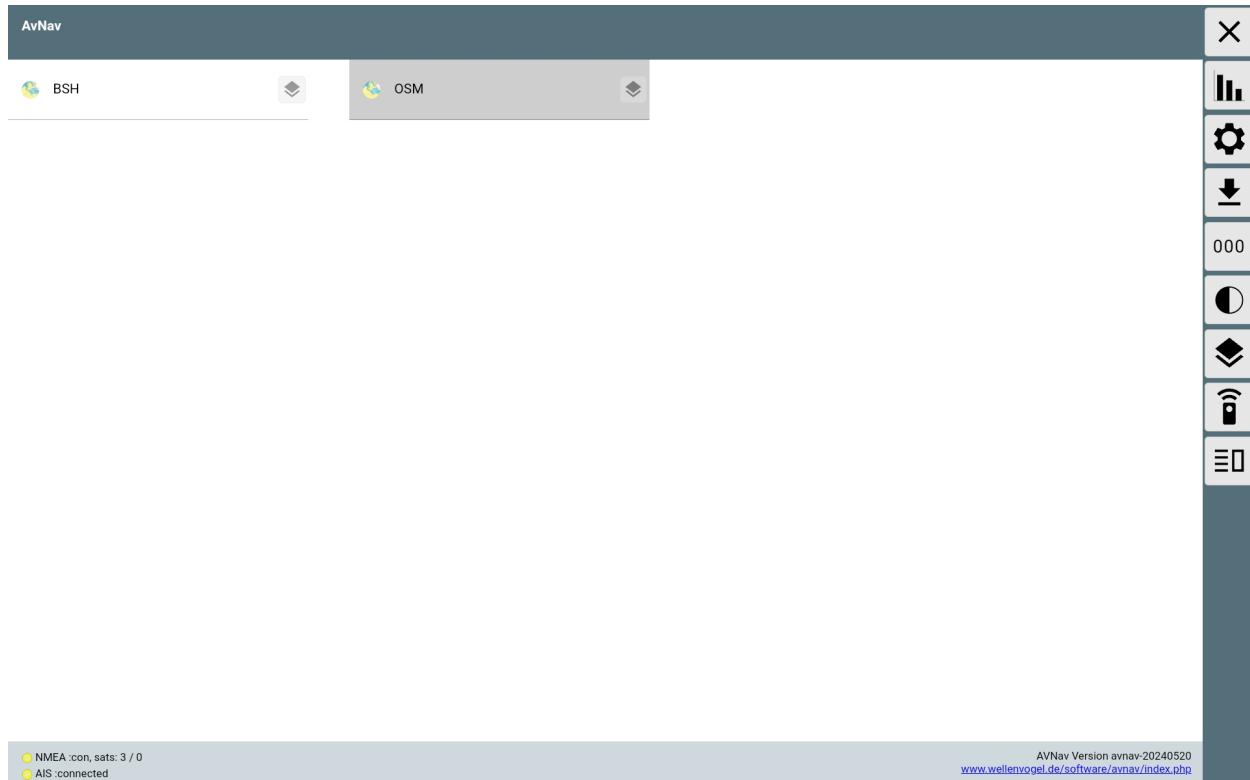


Abb.: Startseite AvNav für Android

Unter AvNav klicken Sie auf der Startseite oben rechts das Symbol mit den 3 Strichen.



Sie gelangen dann auf die Seite zum Serverstatus. Dort können Sie über das Plus-Symbol weitere Verbindungen zum AvNavServer einrichten.



Für die bidirektionale Kommunikation über USB wählen Sie **TCPReader**.

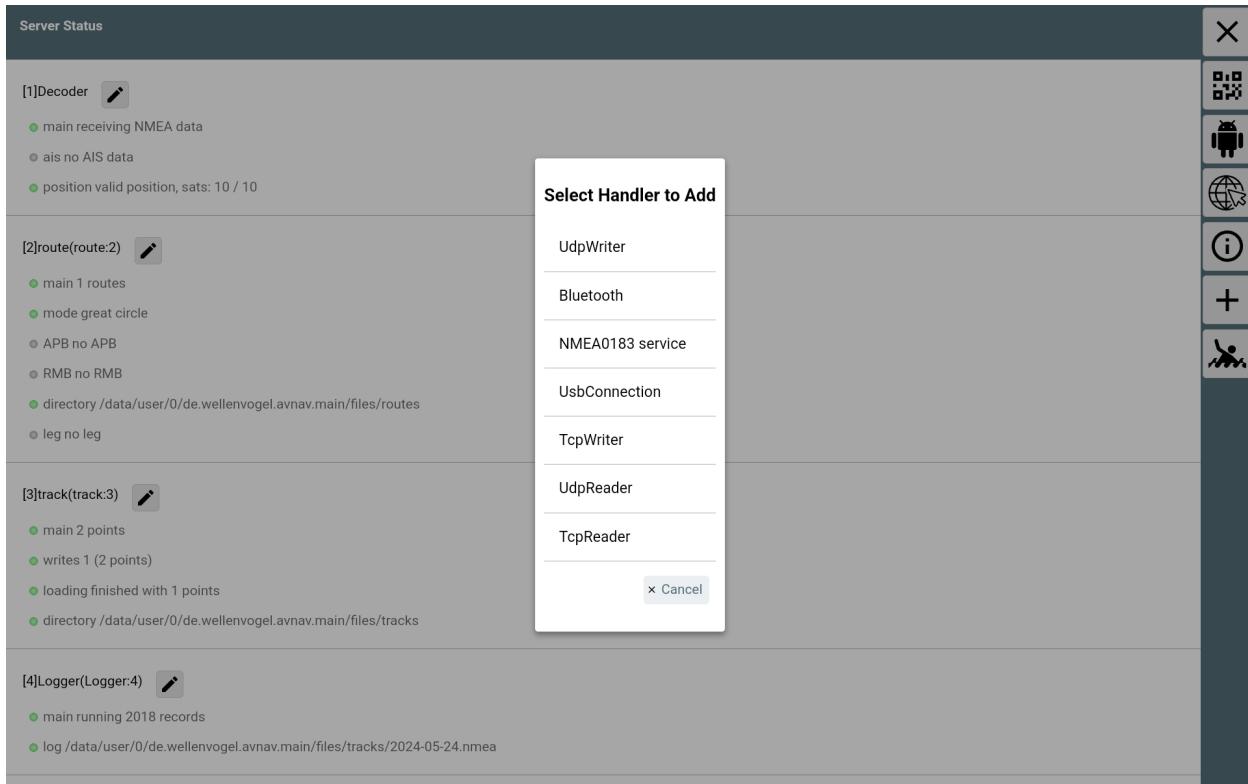


Abb.: Verbindungstypen

Unter **IP-Address** tragen Sie die IP-Adresse des OBP60 ein und als **Port** die 10110. Um nicht nur Daten senden, sondern auch empfangen zu können, aktivieren Sie **SendOut**.



Abb.: Einstellungen zur TCPReader-Verbindung

Nach der Übernahme aller Daten ist die neue Verbindung im Server-Status als TCPReader-Verbindung zu sehen.

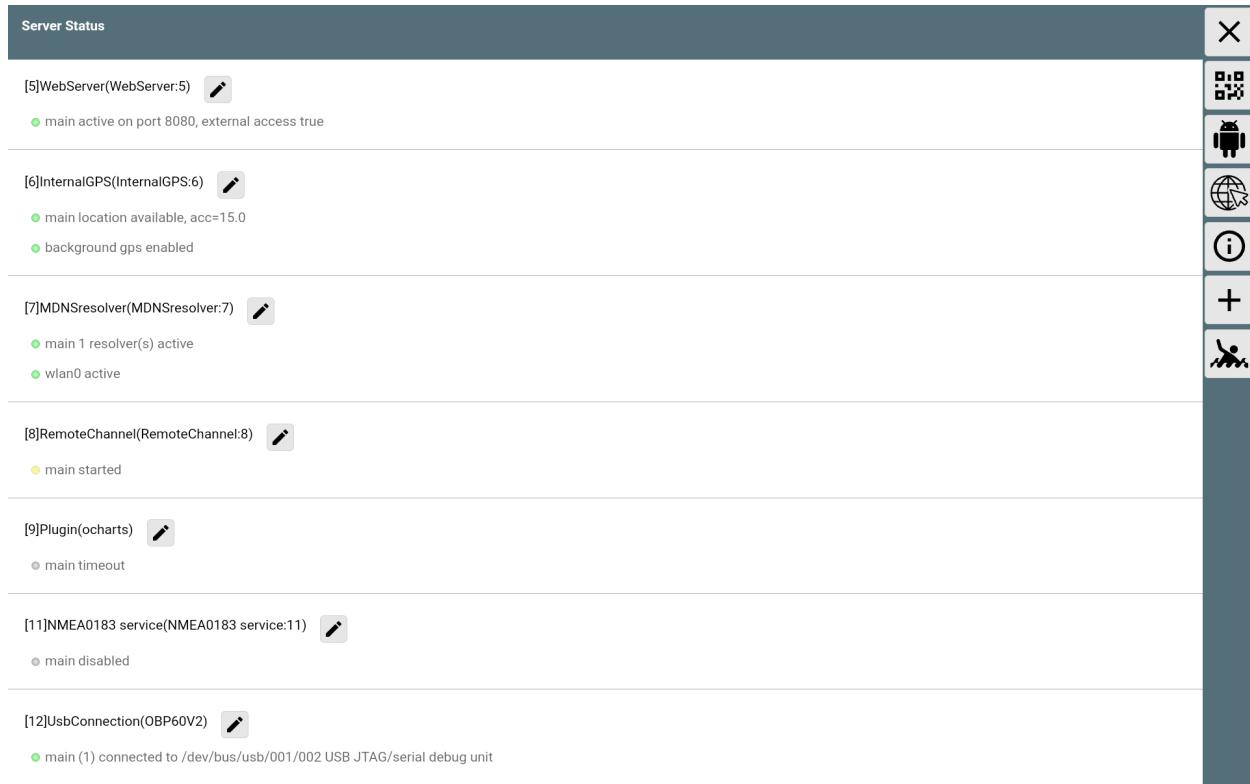


Abb.: Server-Status

KAPITEL 11

Sicherheitshinweise

11.1 Allgemeine Hinweise

Die auf dieser Webseite veröffentlichten Informationen sind im Rahmen eines Hobbyprojektes entstanden. Das Open-Source-Projekt ist als Machbarkeitsstudie zu verstehen, die zeigen soll, wie mit begrenzten Möglichkeiten private Projekte umgesetzt werden können. Die Qualität hinsichtlich Funktionalität, Sicherheit und Dokumentation ist nicht mit kommerziellen Produkten vergleichbar. Die Benutzung der Open-Source-Software und Hardware erfolgt auf eigene Gefahr unter Ausschluss von Haftungsansprüchen.

Gefahr: Jedem Interessierten sollte klar sein, dass die Nutzung des hier vorgestellten Projektes auf eigene Gefahr erfolgt und mit Fehlern und Störungen zu rechnen ist, die schwerwiegende Konsequenzen haben können. Führen Sie vor dem Einsatz des OBP60 eine Risikoanalyse durch und treffen geeignete Maßnahmen, um Schäden und Gefährdung anderer Personen zu vermeiden.

Bemerkung: **NMEA2000** (TM), **NMEA0183** (TM), **SeaTalk** (TM), **SeaSmart** (TM) sind eingetragene Marken der jeweiligen Markeninhaber. Die Namen werden auf dieser Seite als Synonym für diese Protokoll-Typen benutzt. Wenn in den Beschreibungen darauf Bezug genommen wird, so sind Open-Source-Implementierungen damit gemeint, die nicht in allen Punkten den Qualitätsanforderungen und Spezifikationen der jeweiligen Markeninhaber entsprechen. Die Software und Hardware versteht sich als experimentell zur Wissensgewinnung und sollte nicht mit sicherheitskritischen Systemen wie z.B. einem Autopiloten verbunden und betrieben werden. Es kann keine Garantie für die korrekte Implementierung der Protokolle und deren Funktionssicherheit übernommen werden.

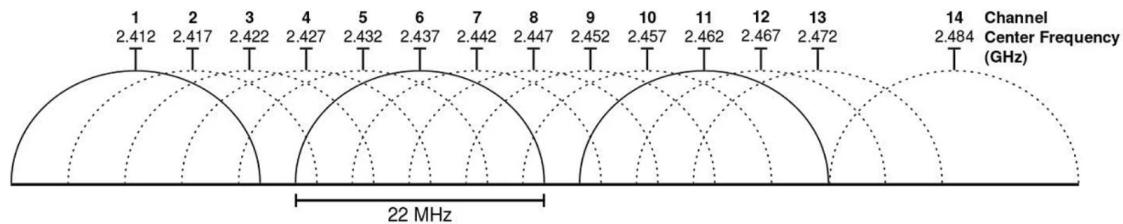
Gefahr: Die Integration von Open-Source-Implementierungen in zertifizierte Bussysteme kann zu Fehlern, Störungen und Totalausfällen führen.

Bei auftretenden Problemen oder Mängeln bitte Sie, uns eine Information über das [Kontaktformular](#) zukommen zu lassen. So können wir auf Qualitäts- und Sicherheitsprobleme eingehen und für Abhilfe sorgen.

11.2 Sicherheit im WiFi-Netzwerk

Sie sollten das OBP60 nur mit vertrauenswürdigen WiFi-Netzwerken verbinden. Es gibt im Gerät nur einen sehr begrenzten Schutz gegen Netzwerk-Sniffing oder Denial-of-Service-Angriffe. Solange Sie das eigene autarke WiFi-Netz des OBP60 nutzen, können fremde Personen nicht ohne weiteres auf das WiFi-Netz des OBP60 zugreifen. Auf diese Weise läuft die Datenübertragung in Ihrem eigenen WiFi-Netzwerk geschützt. Verbinden Sie das Gerät niemals ohne eine Firewall direkt mit dem Internet, und vermeiden Sie direkte Verbindungen zu offenen Hafen-WLANs. Dadurch können auch fremde Personen auf Ihre Geräte im Netzwerk zugreifen.

Bemerkung: Sie können die Sicherheit erhöhen, indem Sie einen separaten WiFi- oder LTE-Router in Ihrem Boot verwenden. Die Router können so eingerichtet werden, dass Sie ein eigenes WiFi-Netz aufspannen, mit dem alle Geräte an Bord verbunden sind. Gängige mobile Router verfügen in der Regel über eine bereits eingeschaltete Firewall, über die Sie Ihr eigenes WiFi-Netz mit dem Internet verbinden können. Die Firewall verhindert fremden Zugriff von außen auf Ihre Geräte. So haben alle Geräte in Ihrem Netz einen gemeinsamen Internet-Zugriff und sind zugleich ausreichend geschützt.



Die Verbindungsqualität von WiFi-Netzwerken hängt maßgeblich von der Auslastung der Funkkanäle ab, die in Ihrer Umgebung aktuell benutzt werden, denn Ihr Gerät teilt sich die selben Funkkanäle mit anderen Teilnehmern anderer WiFi-Netze. Das OBP60 nutzt die Funkkanäle des 2.4 GHz-Frequenzbandes.

Warnung: Bei hoher Auslastung wie z.B. in Häfen kann die Verbindungsqualität des eigenen WiFi-Netzwerks dadurch beeinträchtigt sein. Sie müssen dann mit Verzögerungen bei der Datenübertragung rechnen, insbesondere, wenn Sie TCP-Datenverbindungen zum oder vom OBP60 nutzen. Stellen Sie aber auf alle Fälle sicher, dass in solchen Situationen die Bootsführung nicht beeinträchtigt wird.

Hinweis: Verwenden Sie bei hoher Kanalauslastung Kanäle mit geringer Auslastung. Die Kanäle 1, 13 und 14 haben nur einen Nachbarkanal und sind deutlich robuster gegen hohe Auslastung als die anderen Kanäle. Am besten eignet sich der Kanal 13, da er seltener benutzt wird. In den USA kann auch der Kanal 14 verwendet werden. Moderne mobile Router bieten häufig eine Automatik in ihrer Konfiguration an, die die Kanalauswahl optimieren hilft.

Bei Änderungen der Konfiguration des OBP60 werden Sie grundsätzlich nach dem Admin-Passwort gefragt. Die Übertragung des Passwortes erfolgt dabei immer verschlüsselt. Wenn Sie jedoch das Passwort für den WLAN-Zugangspunkt oder das WiFi-Client-Passwort ändern, wird es im Klartext gesendet. Wenn Sie das Remember me für das Admin-Passwort aktivieren, wird es im Klartext in Ihrem Browser gespeichert. Um es von dort zu entfernen, verwenden Sie Forget Password.

Geräteaufbau

12.1 Mechanischer Aufbau

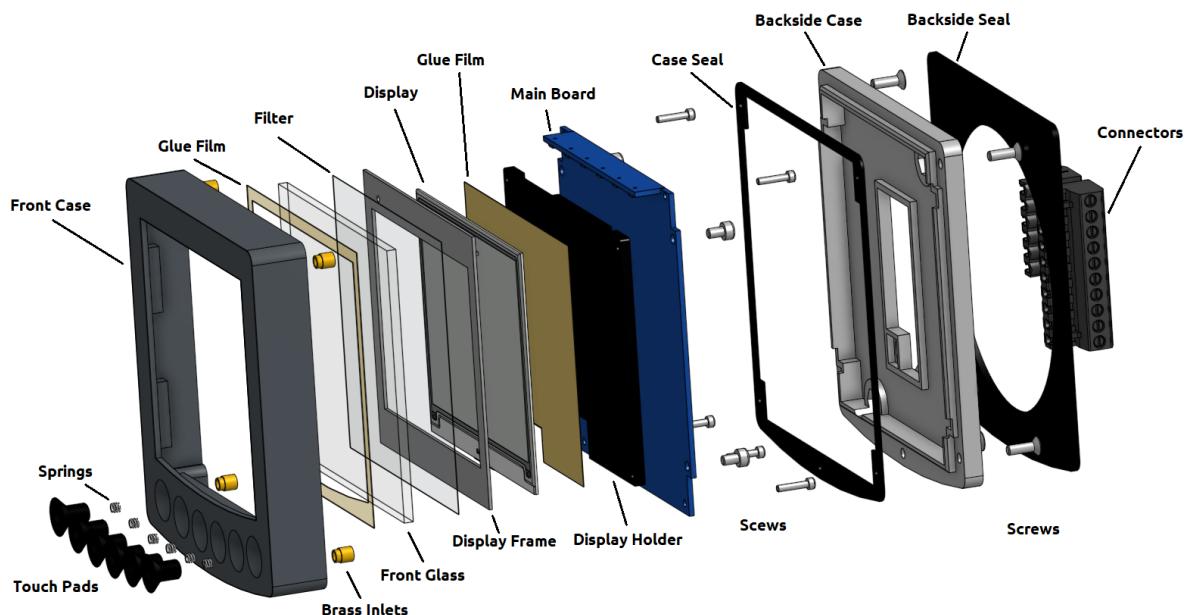


Abb.: Explosionsansicht

In der oberen Abbildung ist der Aufbau des OBP60 zu sehen. Das Gerät besteht aus einzelnen Komponenten, die im folgenden beschrieben werden.

- **Touch Pads**

Die Tasten bestehen aus schwarz eloxierten Inbus-Senkschrauben aus Edelstahl (V2A). Die Tasten werden mit einem Dichtmittel in die Frontschale des Gehäuses eingeschraubt, um Wasserdichtigkeit zu erreichen.

- **Springs**

Die Federn hinter den Tasten dienen der elektrischen Kontaktierung der Tasten mit den Sensorflächen auf der Platine.

- **Front Case**

Die Frontschale des Gehäuses nimmt die Tasten auf und enthält die eingeklebte Frontscheibe. Die Außenflächen der Frontschale bestehen aus 2 mm dickem Kunststoff. An einigen Stellen ist das Gehäuse hohl und enthält an den Stellen eine Stützstruktur mit einem Füllgrad von 30%. Das Gehäuse ist als Kastenkonstruktion ausgeführt und dadurch verwindungssteif.

- **Brass Inlets**

Die Messing-Gewindegelenke dienen zur Verstärkung der Gewinde für die Gehäuseschrauben, da der Kunststoff keine ausreichende Festigkeit aufweist, um den nötigen Druck zum Anpressen der Gehäusedichtung aufzubringen. Die Messing-Gewindegelenke werden thermisch in den Kunststoff eingeschmolzen und sind gegen eine ungewollte Verdrehung im Kunststoff gesichert.

- **Glue Film**

Der Klebefilm wird auf die Frontscheibe aufgetragen und dient zur wasserdichten Befestigung der Frontscheibe an der Innenseite des Gehäuses.

- **Front Glass**

Die Frontscheibe aus 3 mm dickem Plexiglas schützt das Display vor Umwelteinflüssen und vor mechanischen Beschädigungen durch harte Schläge. Die Frontscheibe dient zusätzlich als UV-Filter mit hoher Filterwirkung. Die Außenseite der Frontscheibe ist entspiegelt und verhindert direkte Reflexionen des Sonnenlichts. Eine Besonderheit der Frontscheibe ist ihre Fähigkeit, das e-Paper-Display bei Nacht zu beleuchten. Dazu koppeln an der oberen Stirnseite der Scheibe 6 RGB-LEDs Licht in die Plexiglasscheibe ein. Das Licht der LEDs wird an Streukörpern innerhalb des Plexiglasses auf das e-Paper-Display umgelenkt. Die Ausleuchtung des Displays erfolgt daher gleichmäßig über die gesamte Displayfläche.

- **Filter**

Eine IR-Filterfolie wird zusätzlich auf der Rückseite der Frontscheibe aufgebracht und soll die Infrarotstrahlung des Sonnenlichtes reduzieren und vom e-Paper-Display fernhalten. Die Infrarotstrahlung wird in der Filterfolie absorbiert und an die Frontscheibe abgegeben, denn bei maximaler Sonenneinstrahlung treffen bis zu 10 W an Wärmestrahlung auf das Display.

- **Display Frame**

Der Displayrahmen ist eine Art Maske und verdeckt unerwünschte Bereiche des Displays. Zusätzlich ist ein kleiner Steg angebracht, um das Licht der Hintergrundbeleuchtung gegenüber dem Display abzuschirmen, damit kein Fremdlicht in die Glasscheibe des e-Paper-Displays eingekoppelt wird.

- **Display**

Das Display ist ein e-Paper-Display, das Schwarz/Weiß und 4 Graustufen abbilden kann. Die Auflösung des Displays beträgt 400 x 300 Pixel. Die Pixeldichte auf die Anzeigefläche bezogen liegt bei 120 dpi. E-Paper-Displays benötigen nur bei der Ansteuerung neuer Bildinhalt Energie. Im ausgeschalteten Zustand bleibt der Bildinhalt sichtbar. Damit ist diese Bildschirm-Technologie sehr energiesparend.

- **Glue Film**

Der Klebefilm hinter dem Display dient zur Befestigung des Displays auf dem Displayhalter.

- **Display Holder**

Der Displayhalter trägt das e-Paper-Display und verhindert ein Verwinden des Displays. Das e-Paper-Display ist sehr empfindlich, denn es besteht aus zwei Glassubstraten mit einer Gesamtdicke von nur 0,7 mm. Über den Displayhalter wird das Display mittels Schrauben mit dem Mainboard verbunden.

- **Main Board**

Das Mainboard enthält alle elektronischen Komponenten auf einer zweiseitigen Platine. An der Oberseite des Mainboards ist die LED-Platine für die Hintergrundbeleuchtung angebracht.

- **Case Seal**

Die Gehäusedichtung stellt eine wasserdichte Verbindung zwischen der Front- und Rückschale her. Die Dichtung ist dabei als 1 mm dicke Flächendichtung ausgeführt.

- **Backside Case**

Die Gehäuserückschale schließt die Rückseite des Gehäuses ab. Sie ist als verwindungssteife Kastenkonstruktion wie die Frontschale ausgeführt.

- **Backside Seal**

Die Rückwanddichtung dichtet das Gehäuse gegenüber der Einbauöffnung ab. Die Dichtung besteht aus 2 mm dickem Neoprenmaterial. Damit lassen sich kleine Unebenheiten und leichte Krümmungen überbrücken.

- **Connectors**

Für eine einfachen Montage sind an der Rückseite des Gerätes abnehmbare Steckverbinder angebracht. Über die Steckverbinder mit Schraubverbindungen kann das Gerät mit Strom versorgt und an die Bussysteme angeschlossen werden.

12.2 Main Board

Das Mainboard enthält alle elektrischen und elektronischen Komponenten wie:

- Dual Core CPU ESP32-S3
- Stromversorgung
- Display-Ansteuerung
- Sensortasten
- GPS-Empfänger
- RTC Echtzeituhr
- Backup-Batterie
- Buzzer
- Flash-LED
- Hintergrundbeleuchtung
- Isolierte Treiber für Bussysteme (NMEA2000; NMEA0183, I2C)
- externe Stromversorgung
- ESD-Schutzschaltungen
- USB-C

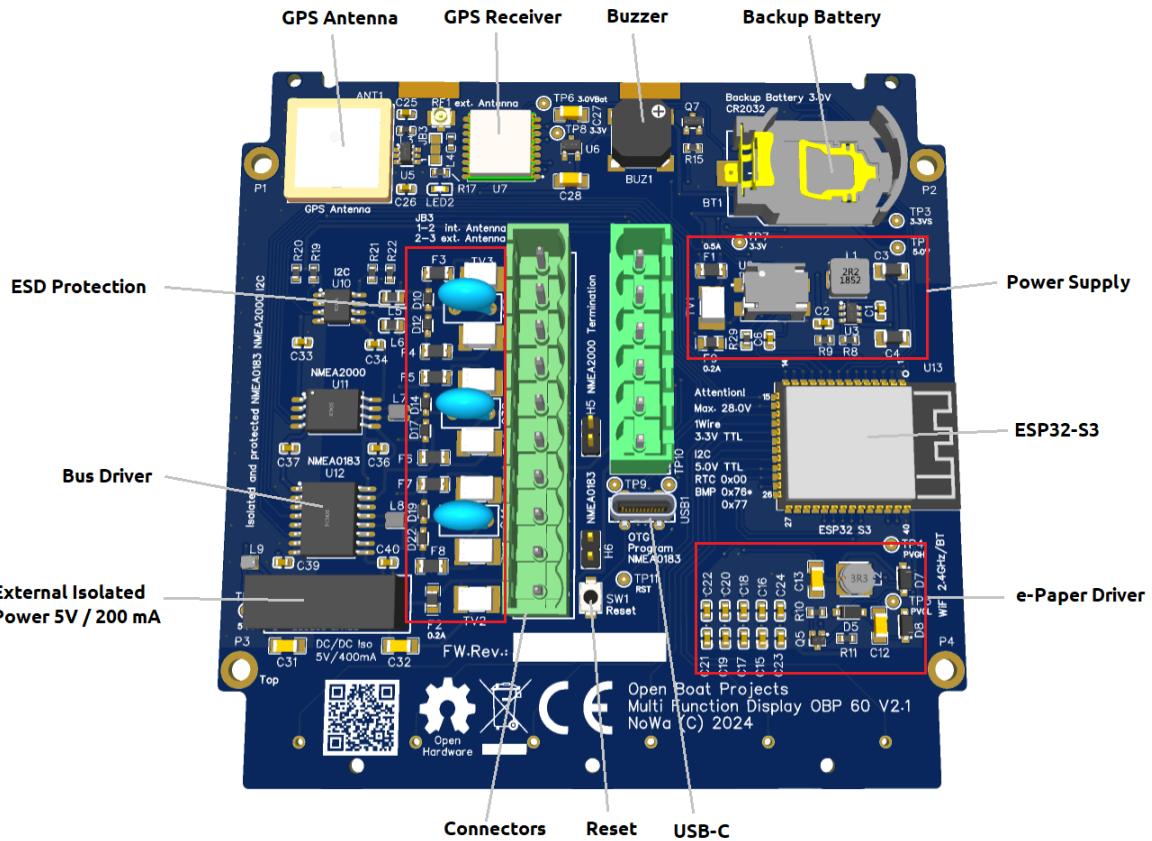


Abb.: Mainboard Oberseite

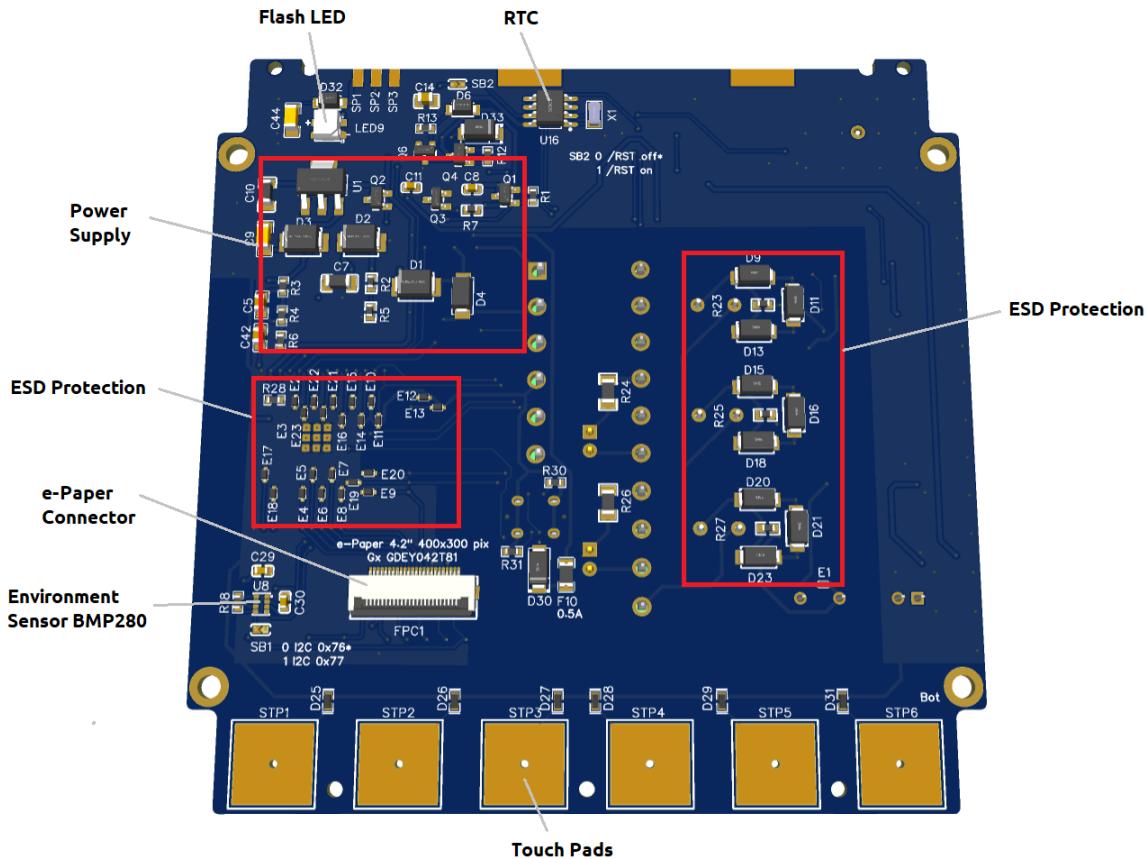


Abb.: Mainboard Unterseite

12.3 Platinen

Die Platten für das Mainboard und die Hintergrundbeleuchtung wurden als zweiseitig bestückte SMD-Platine mit Durchkontaktierungen, Stopplack und Bedruckung ausgeführt.

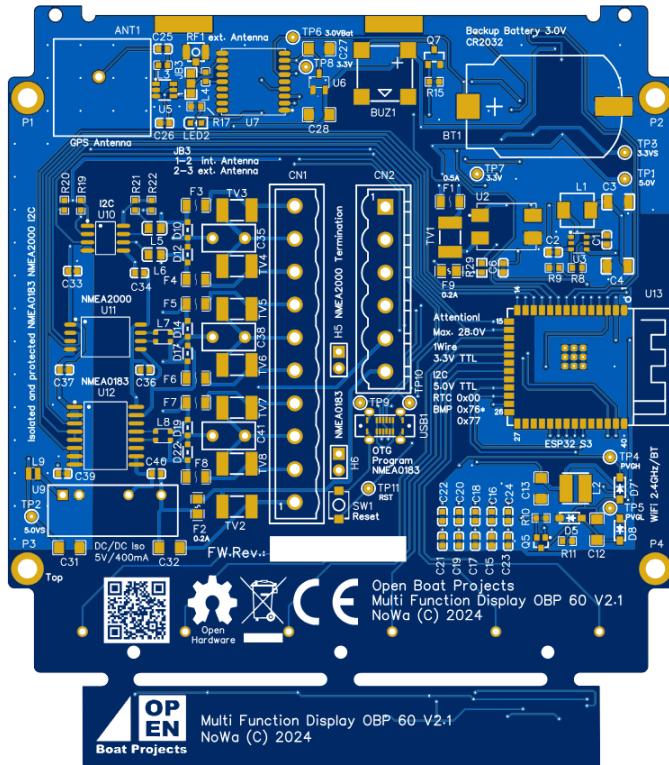


Abb.: Unbestückte Platinen-Oberseite

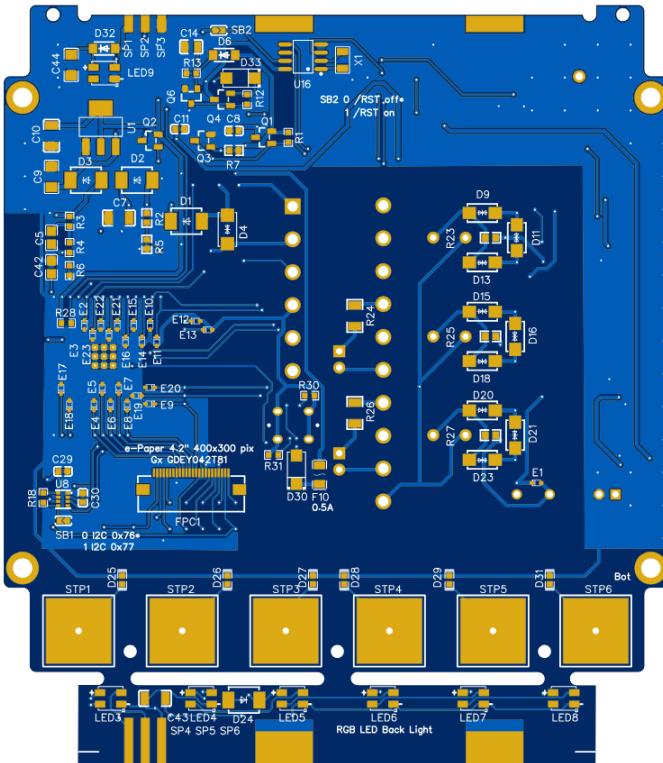


Abb.: Unbestückte Platinen-Unterseite

12.4 Schaltplan und Fertigungsdaten

Der Schaltplan und die Platine wurden mit dem Online-Entwicklungstool EasyEDA erstellt. Nachfolgend sind die Unterlagen für eine Fertigung aufgeführt.

- Schaltplan V2.1 [PDF]
- Gerber Daten [ZIP]
- Bauteilliste [TXT]
- Bestückung [HTML]
- 3D-Daten der Kunststoff-Teile [ZIP]



Die Fertigungsdaten von Schaltplan, Gerber-Daten und Bauteilliste und die 3D-Daten unterliegen der [Common Creative Lizenz \(CC\) BY BC SA 4.0](#). Das OBP60 darf unter Nennung der Urheber nachgebaut oder modifiziert werden. Es entstehen keinerlei Kosten für eine private Nutzung ohne kommerzielle Absichten. Eine kommerzielle Verwertung wird durch die Lizenz ausgeschlossen. Abgeleitete Werke unterliegen der selben Lizenz. Wenn Sie eine kommerzielle Nutzung des OBP60 beabsichtigen, kontaktieren Sie uns über das [Kontaktformular](#). Es besteht die Möglichkeit, ein nicht exklusives Nutzungsrecht über eine kommerzielle Lizenz zu erwerben.

12.5 Schaltungsbeschreibung

KAPITEL 13

Vorbereitung

KAPITEL 14

Bauteilliste

KAPITEL 15

Durchführung

KAPITEL 16

Funktionstests

KAPITEL 17

Programmierumgebung

Als Programmierumgebung für das OBP40 wird das kostenlose Entwicklungstool **Visual Studio Code** in Verbindung mit **PlatformIO** verwendet. Diese Kombination bietet Vorteile gegenüber der Arduino-IDE, da projektbezogen die verwendete Hardware und die Bibliotheken in den richtigen Versionen hinzugelinkt sind und Sie sich um diese Dinge nicht kümmern müssen. Zusätzlich verfügt Visual Studio Code über eine mächtige Querverweisliste, so dass Sie den Code besser verstehen und nachvollziehen können. Intellisense unterstützt Sie bei der korrekten Verwendung von Funktionsaufrufen und Variablen. Über Github können Sie von Visual Studio Code aus auf den aktuellsten Firmware-Code zugreifen und sind damit immer auf dem aktuellsten Stand.

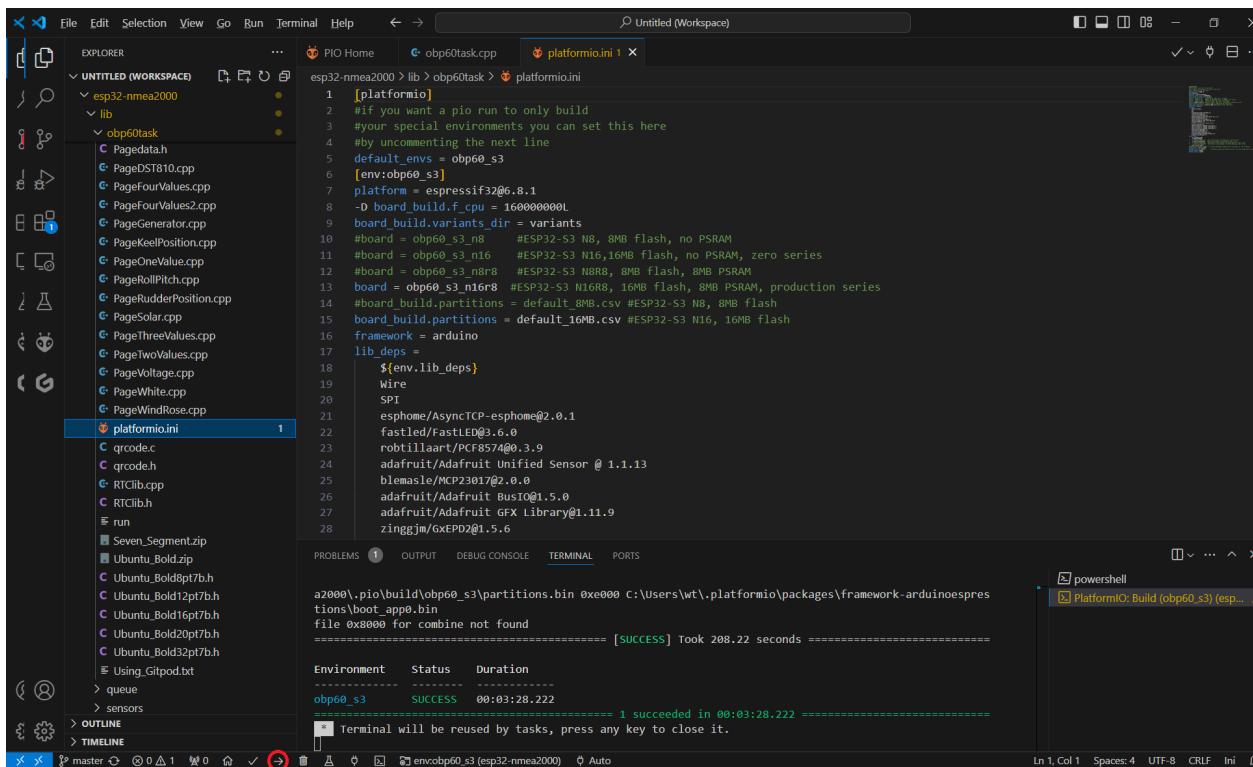


Abb.: Visual Studio Code

Visual Studio Code mit PlatformIO ist für alle gängigen Betriebssysteme verfügbar wie:

- Linux
- Windows
- Mac OS

Wer an der Weiterentwicklung des Firmware-Codes interessiert ist, sollte Visual Studio Code mit PlatformIO in Verbindung mit Github nutzen. So kann eine breite Community an der Weiterentwicklung mitwirken und den Firmware-Code erweitern und verbessern. Aufgetretene Fehler und Wünsche der Community verwalten wir über Issues im Trackingsystem von Github. Im Segeln-Forum gibt es eine große Community, die an der Weiterentwicklung mitarbeitet und auf Support-Fragen antworten kann.

- **Visual Studio Code**
 - **Notwendige Erweiterungen**
 - PlatformIO
 - C/C++
- Github-Projekt-Repository
- Issues Tracking System
- Community Forum

Wie man Visual Studio Code installiert und benutzt ist im Kapitel *Visual Studio Code / PlatformIO* beschrieben.

Wer kleine Änderungen am Code unterwegs unkompliziert vornehmen möchte ohne eine aufwändige Installation von Visual Studio Code und PlatformIO durchzuführen, kann Gitpod benutzen. Gitpod ist vollständig webbasiert. Sie benötigen nur einen aktuellen Webbrowser und können sofort mit der Programmierung beginnen. Der Workplace von Gitpod orientiert sich an Visual Studio Code und enthält alle vorinstallierten Komponenten die Sie zur Softwareentwicklung benötigen. Gitpod eignet sich daher hervorragend für Anfänger. Im Kapitel *Gitpod* ist die Benutzung beschrieben.

KAPITEL 18

Seitenerstellung

Um eine neue Seite zu erstellen sind folgende Schritte notwendig:

1. Anlegen einer neuen Seite unter `lib\obp60task\PageXXXX.cpp`
2. Registrierung der Seite in `lib\obp60task\obp60task.cpp`. Dazu die Funktion `registerAllPages` entsprechend erweitern
3. In `/lib/obp60task/config.json` die neue Seite in die jeweiligen Seitenlisten aufnehmen oder Seite in `/lib/obp60task/gen_set.py` aufnehmen und damit den relevanten Teil der Datei `/lib/obp60task/config.json` neu erzeugen.

Als Vorlage für eine neue Seite kann z.B. die vorhandene Seite `PageWhite` verwendet werden. Sie enthält nur minimalen Code. In der kopierten Seite ist dann lediglich der Text `White` durch `Beispiel` zu ersetzen.

KAPITEL 19

Kompilieren und Download

Die Firmware zum OBP60 kann recht einfach an eigene Bedürfnisse angepasst werden. Als Programmier-Entwicklungsumgebung wird „PlatformIO“ verwendet. Der Source-Code befindet sich zentral auf Github im Internet. Alle Software-Revisionen sind dort verfügbar. Es können bei Bedarf aktuelle oder ältere Versionen benutzt werden. Als Betriebssysteme werden Linux, Windows und Mac iOS unterstützt. Zum Kompilieren der Firmware gibt es zwei Möglichkeiten.

- Verwendung von **Gitpod** (webbasiert in Cloud)
- **Visual Studio Code** mit **PlatformIO** Plugin (lokal auf dem PC)

Bemerkung: Die Arduino IDE wird nicht als Programmier-Entwicklungsumgebung verwendet, da der Source-Code sehr komplex ist und die Arduino IDE nicht über eine sinnvolle Querverweisfunktion verfügt. Notwendige Tool-Chains und Bibliotheken werden in PlatformIO in der richtigen Version hinzugelinkt und ermöglichen eine fehlerfreie Kompilierung.

19.1 Gitpod

Gitpod ist eine standartisierte Entwicklungsumgebung, die webbasiert ist und in der Cloud läuft. So ist es möglich, auf jedem Rechner, der über einen aktuellen Webbrowser verfügt, eine Code-Entwicklung durchzuführen ohne spezielle Installation von irgendwelchen Softwarebestandteilen. Zur Benutzung des Dienstes ist eine Anmeldung bei Github notwendig. Die Code-Entwicklungsumgebung in Anlehnung an PlatformIO befindet sich jederzeit in einem aktuellen Zustand. Abhängigkeiten zu externen Bibliotheken werden selbstständig aufgelöst und integriert. Das System ist sofort benutzbar und ist besonders für Anfänger geeignet oder für kleine Änderungen, die unterwegs mal schnell durchgeführt werden sollen. Die Benutzung von Gitpod ist in der Free-Variante kostenfrei, unterliegt jedoch einigen Einschränkungen bezüglich der Nutzungszeit und bereitgestellten Rechnerhardware. Aktuell sind 50 Stunden Nutzungszeit pro Monat frei und vollkommen ausreichend für einfache Änderungen. Wer mehr Zeit benötigt oder schnellere Hardware einsetzen möchte, kann den kostenpflichtigen Service von Gitpod nutzen. Weitere Details findet man auf der [Webseite von Gitpod](#).

Der Workflow bei Gitpod sieht folgendermaßen aus:

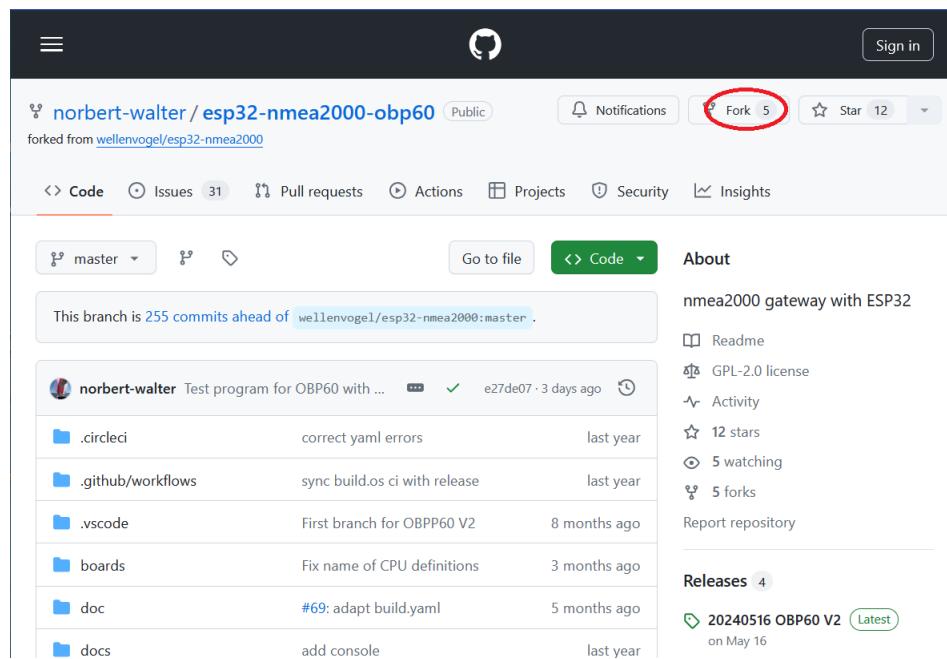
1. OBP60-Github-Projekt in eigenes Github-Repository forken

2. Gitpod-Link zum Projekt öffnen
3. In Gitpod Entwicklungsumgebung und Hardware auswählen
4. Start eines Containers mit der Entwicklungsumgebung und Download des Projektes aus dem eigenen Github-Repository
5. Source-Code in der Entwicklungsumgebung ändern und komplizieren
6. Download des Binary-Files
7. Aktualisierung der Änderungen im eigenen Github-Repository

19.1.1 Github-Projekt forken

Als erstes wird das Original-Projekt zum OBP60 in das eigene private Repository geforkt. Ein Fork ist sozusagen eine Aufgabelung bzw. eine Abtrennung des Source-Codes in einen neuen Zweig. Durch einen Fork entsteht eine Kopie des Original-Projektes in Ihrem privaten Repository. So können Sie Änderungen am Code vornehmen und in ihrem Repository speichern. Der neu entstandene Code ist dann eine Erweiterung oder Modifikation des Original-Projektes.

Hinweis: Nur mit einem Fork ist es möglich, eigene Code-Änderungen zu sichern.



The screenshot shows a GitHub repository page for 'norbert-walter / esp32-nmea2000-obp60'. The top navigation bar includes 'Sign in', 'Notifications', 'Fork 5' (which is circled in red), and 'Star 12'. Below the header, there are tabs for 'Code', 'Issues 31', 'Pull requests', 'Actions', 'Projects', 'Security', and 'Insights'. The 'Code' tab is selected. On the left, there's a dropdown for the 'master' branch and a 'Go to file' search bar. The main content area displays a list of commits from 'norbert-walter'. To the right, there's an 'About' section with details like 'nmea2000 gateway with ESP32', 'Readme', 'GPL-2.0 license', 'Activity', '12 stars', '5 watching', '5 forks', and a 'Report repository' link. At the bottom, there's a 'Releases' section with a link to '20240516 OBP60 V2' (Latest) from May 16.

Abb.: Fork erstellen

Melden Sie sich als erstes bei Github an und gehen dann in das **'Original-Projekt zum OBP60'** und drücken oben rechts auf die Schaltfläche **Fork**. Sie werden danach gefragt, ob sie der Quelle vertrauen und können dann einen neuen Projektnamen vergeben oder den originalen Projektnamen benutzen. Kopieren Sie sich danach den Link zu ihrem Github-Projekt aus der Browser-Zeile. Sie sollten dann einen ähnlichen Link haben wie diesen:

<https://github.com/norbert-walter/esp32-nmea2000-obp60/tree/master>

Der Part `MyAccountName` wird durch ihren eigenen Accountnamen ersetzt.

19.1.2 Gitpod-Link öffnen

Das Gitpad-Projekt wird über den Repository-Namen des Github-Projekts aufgerufen:

<https://gitpod.io/#https://github.com/norbert-walter/esp32-nmea2000-obp60/tree/master>

Hinweis: Wer eine ganz bestimmte Release-Version kompilieren möchte, benutzt anstatt `master` die Nummer der Release-Version wie z.B. `20250304`. Branches und getaggte Versionen können auf die selbe Weise verwendet werden, indem der Branch- oder Tagname verwendet wird.

Sie gelangen dann auf die Startseite von Gitpod und müssen sich dort anmelden. Loggen Sie sich dort mit dem bereits vorhandenen Github-Account ein.



Abb.: Login bei Gitpod mit Github-Account

19.1.3 Gitpod-Settings

Danach können Sie die Default-Einstellungen übernehmen. Sie sind schon korrekt auf das Projekt eingestellt.

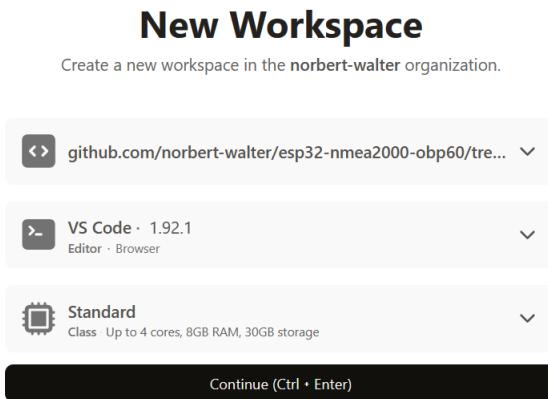


Abb.: Einstellungen für Gitpod

19.1.4 Container-Start

Nach der Bestätigung der Einstellungen für Gitpod wird ein neuer Docker-Container gestartet, danach werden alle notwendigen Softwarebestandteile automatisch in den Container geladen. Der Vorgang kann etwas Zeit beanspruchen. Am Ende sollten Sie einen fertigen Workplace vorfinden. Der Workplace erscheint im Look & Feel von Visual Studio Code, ist jedoch nicht mit ihm verwandt und in einigen Punkten verschieden. So gibt es z.B. kein IntelliSense und die Anzahl der Plugins ist deutlich geringer als beim Original. Zusätzlich fehlen die Buttons zum Kompilieren und zum Download auf die Zielhardware, sowie der serielle Monitor, wie man das von Visual Studio Code gewohnt ist. Über den Workplace lässt sich der Code verändern, kompilieren und das Binary herunterladen. Der Download des Binaries auf die Zielhardware muss mit externen Tools durchgeführt werden.

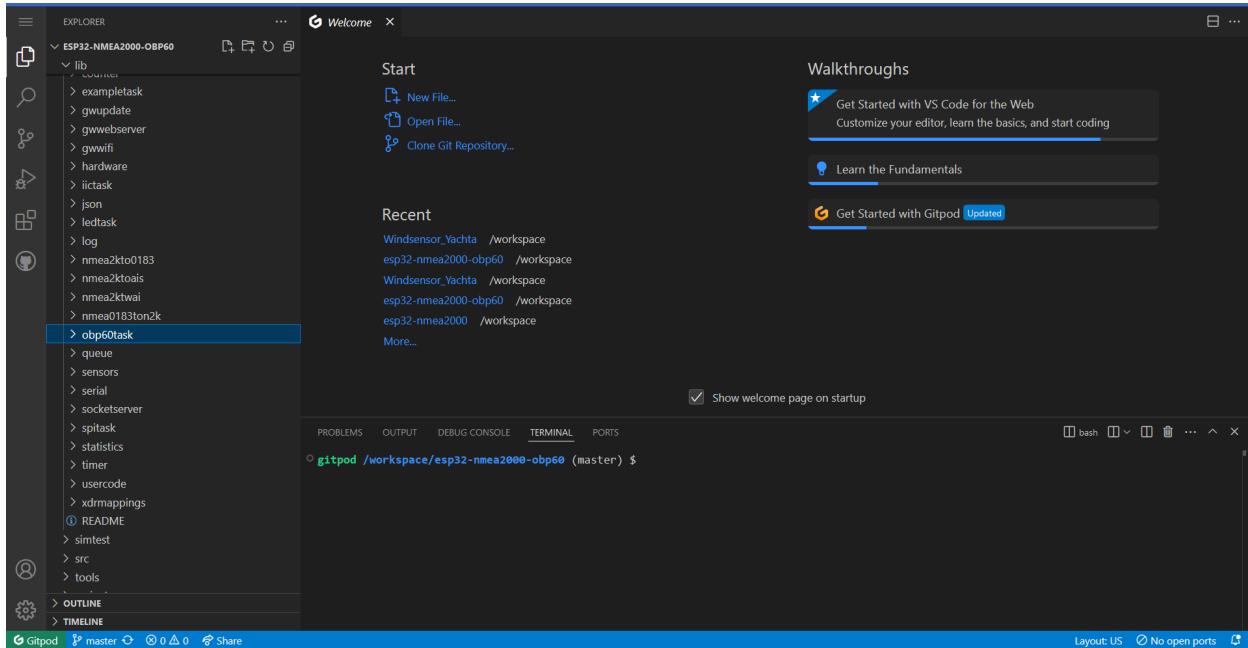


Abb.: Fertiger Workplace

19.1.5 Codeänderung und Kompilieren

Der Code zum OBP60 befindet sich im Ordner: `/ESP32-NMEA2000-OBP60/lib/obp60task`. Durch Editieren der Datei `platformio.ini` können einige Hardwareanpassungen zur Variante des ESP32-S3 und zum e-Paper-Display vorgenommen werden.

Um den Code kompilieren zu können, müssen Sie folgende Befehle nacheinander in das Terminal-Fenster unten rechts einfügen. Sie können dazu die Copy & Paste Funktion benutzen.

- `cd /workspace/esp32-nmea2000-obp60`
- `bash /workspace/esp32-nmea2000-obp60/lib/obp60task/run_install_tools`
- `bash /workspace/esp32-nmea2000-obp60/lib/obp60task/run_obp60_s3`

Nach dem zweiten Befehl werden alle notwendigen Tool-Chains und Bibliotheken in den Workplace geladen. Dieser Vorgang kann einige Minuten dauern. Mit dem letzten Befehl beginnt der eigentliche Kompiliervorgang, der ebenfalls einiges an Zeit benötigt. Wenn Sie den Code später geändert haben und erneut kompilieren wollen, so benutzen Sie einfach den letzten Befehl erneut.

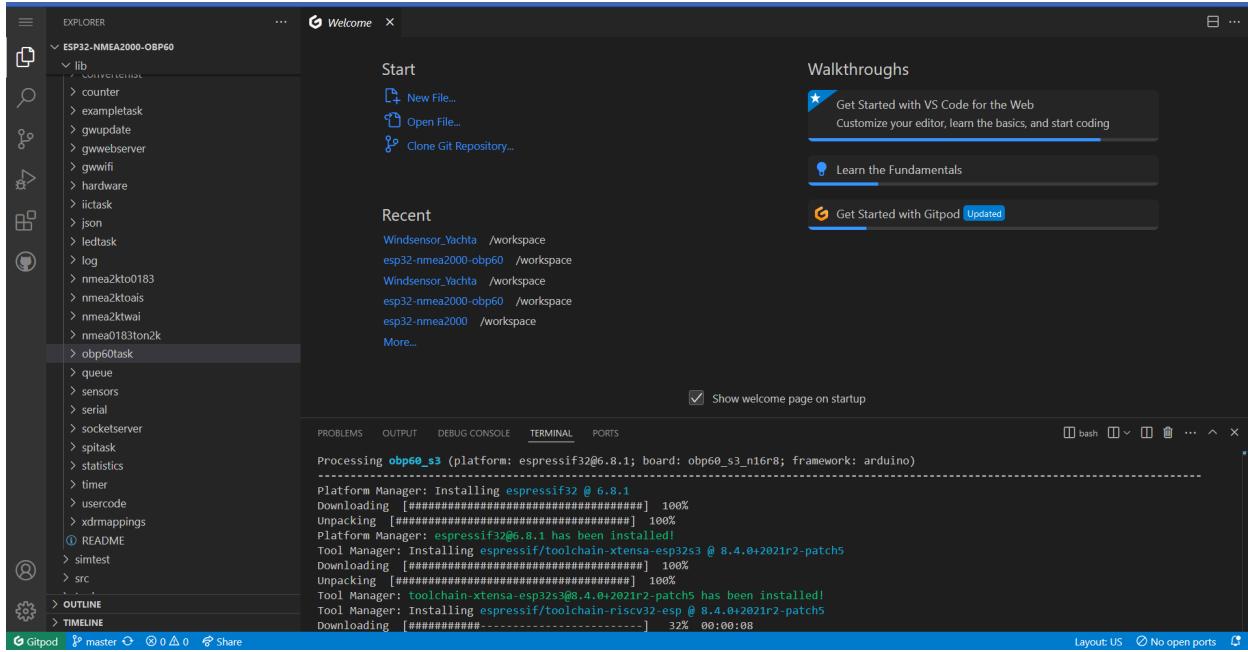


Abb.: Source-Code kompilieren

Wenn der Kompilervorgang erfolgreich abgeschlossen ist, sollten Sie die folgende Meldung sehen:

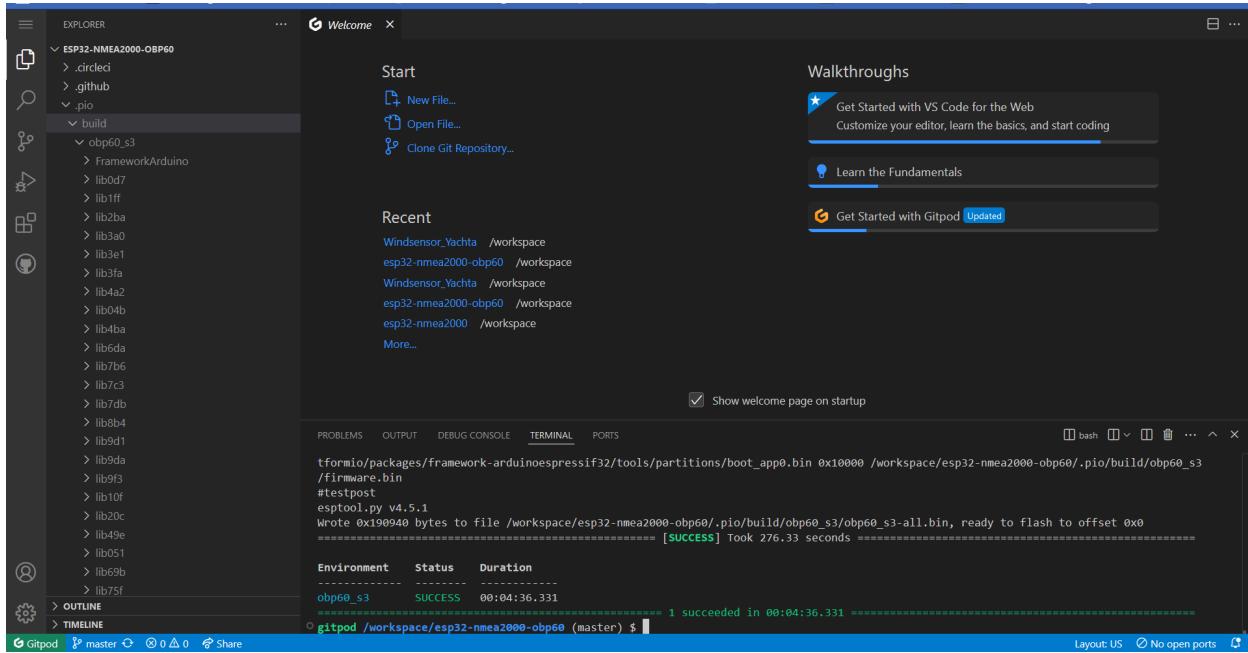


Abb.: Kompilierung beendet

19.1.6 Binary-Download

Der Binary-Code zur Firmware befindet sich nach dem Kompilieren im Verzeichnis:

- `/ESP32-NMEA2000-OBP60/.pio/build/obp60_s3` (für OBP60)

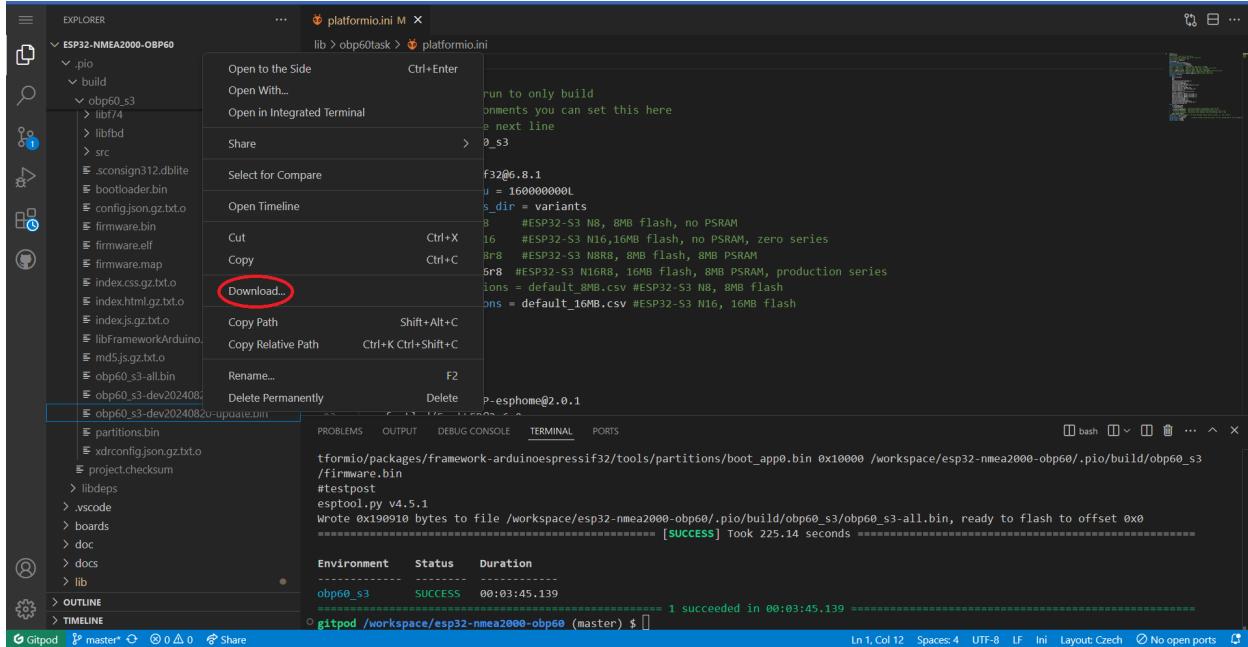


Abb.: Datei-Download

Dabei haben die Files folgende Bedeutung:

- `bootloader.bin` - ESP32-S3 Bootloader für Firmware
- `partitions.bin` - Partitionstabelle für den Flash Speicher
- `firmware.bin` - Binärfile als Update (nur Firmware)
- `firmware.elf` - ELF-File für Fehlerverfolgung
- `obp60_s3-all.bin` - Binärfile mit allen Bestandteilen (Bootloader, Partitionstabelle, Firmware, Default-Settings)
- **`obp60_s3-dev20240820-all.bin`** - wie `obp60_s3-all.bin`
- **`obp60_s3-dev20240820-update.bin`** - Binärfile als Update (nur Firmware)

Je nachdem, ob man mit Default-Einstellungen starten oder mit den alten Einstellungen weiter arbeiten möchte, kann man eine der beiden fett hervorgehobenen Dateien herunter laden. Das geht am einfachsten, wenn man die entsprechende Datei im Dateibaum per Rechtsklick auswählt und sie dann über **Download** herunterlädt. Die **update**- Datei kann man später auf dem OBP60 über die Webseite [Update](#) hochladen und damit die Firmware aktualisieren. Die **all**- Datei muss über USB geflasht werden.

19.1.7 Source-Code Aktualisierung

Wer die Source-Code-Änderungen in seinem Github-Repository speichern möchte, kann die Github-Funktionalität des Workplace nutzen. Auf der linken Seite befindet sich ein Icon mit einer Pfadverzweigung und einer kleinen blauen Zahl daran. Dieses Icon teilt dem Benutzer mit, wie viele Änderungen an Dateien durch die Source-Code-Änderungen vorgenommen worden sind. Mit einem Klick auf das Icon kann man eine Beschreibung für die Änderungen hinzufügen. Wenn man dem Dialog rechts am blauen Commit-Button folgt, kann man **Commit und Push** anklicken. Dadurch werden die Code-Änderungen in das eigene Github-Repository hochgeladen und gesichert. Wer später dort weitermachen möchte, wo er zuletzt aufgehört hat, kann der Vorgehensweise ab Punkt 2 folgen und den Gitpod-Link zu seinem Projekt öffnen.

Hinweis: Nur mit einem Beschreibungstext zur Codeänderung können Sie die Änderungen committen und nach Github hochladen.

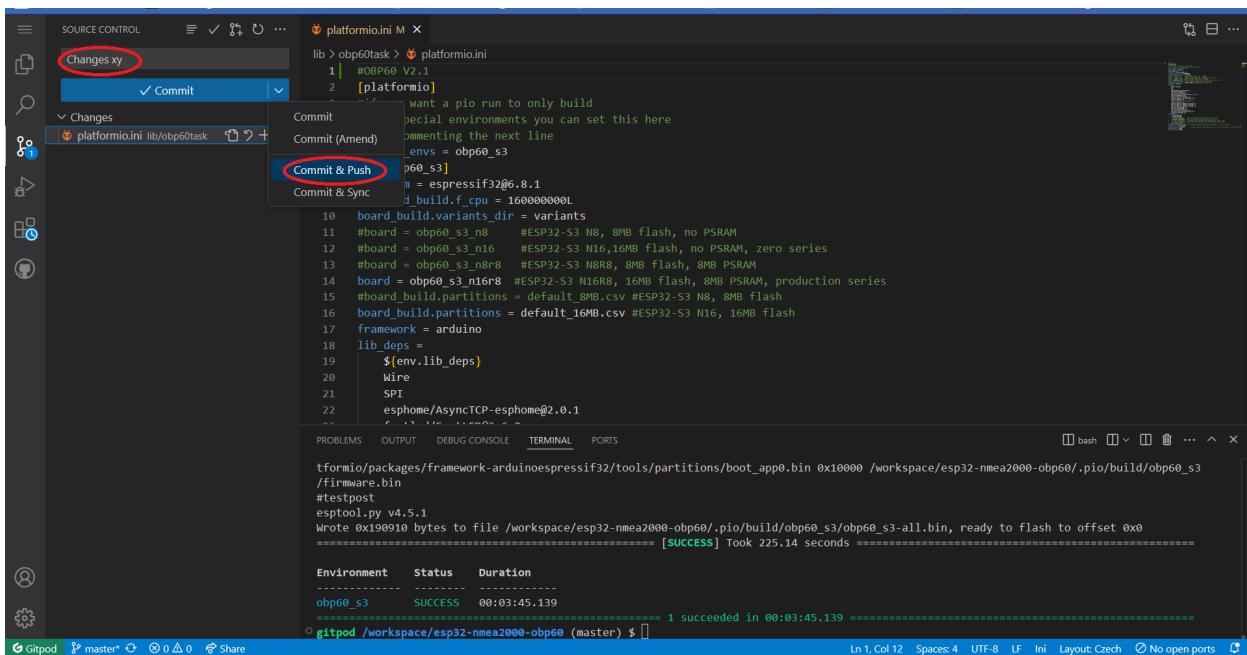


Abb.: Source Code auf Github sichern

19.2 Visual Studio Code / PlatformIO

Visual Studio Code (VSC) ist eine weit verbreitete Software-Entwicklungsumgebung, die eine Vielzahl an Programmiersprachen unterstützt, und die es für verschiedene Betriebssysteme gibt. Die Entwicklungsumgebung läuft dabei auf einem lokalen PC. Zur Programmierung von Embedded Elektronik auf Mikrocontrollerbasis in C und C++ wurde das Plugin **PlatformIO** für Visual Studio Code geschaffen. Damit ist es unkompliziert möglich, eine Softwareentwicklung für Hardware durchzuführen. In einer umfangreichen Datenbank sind eine Vielzahl an Elektronik-Hardware-Boards aufgeführt. Passend zum Hardware-Board werden die notwendigen Tool-Chains und Software-Bibliotheken zu einem Projekt hinzugefügt, ohne dass man sich um Details kümmern muss.

Wer an einer Weiterentwicklung der Firmware zum OBP60 interessiert ist, sollte Visual Studio Code / PlatformIO benutzen. Visual Studio Code verfügt über Mechanismen, mit denen Softwareänderungen über Github eingepflegt und verfolgt werden können.

Wie man Visual Studio Code installiert und danach das Plugin PlatformIO einrichtet, ist in dieser [Anleitung](#) beschrieben. Sie finden im Internet auch weitere Beschreibungen zu diesem Thema.

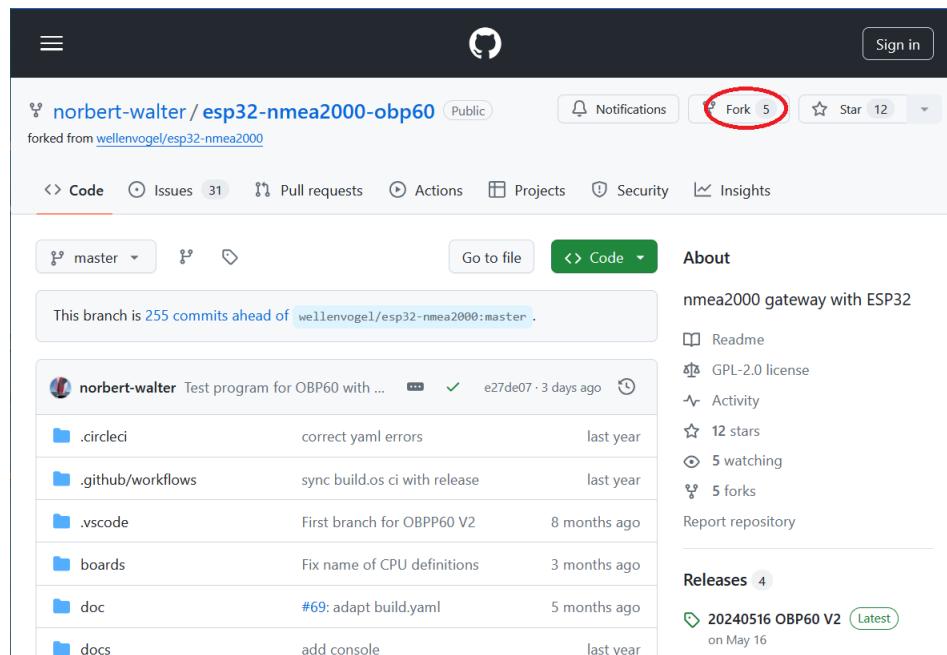
Der Workflow für Visual Studio Code sieht folgendermaßen aus:

1. OBP60-Github-Projekt in eigenes Github-Repository forken
2. Projekt in Visual Studio Code importieren
3. Source-Code in der Entwicklungsumgebung ändern und kompilieren
4. Flashen der Firmware auf die Zielhardware
5. Debuggen der Firmware über serielle USB-Verbindung
6. Aktualisierung der Änderungen im eigenen Github-Repository

19.2.1 Github-Projekt forken

Als erstes wird das Original-Projekt zum OBP60 in das eigene private Repository geforkt. Ein Fork ist sozusagen eine Aufgabelung bzw. eine Abtrennung des Source-Codes in einen neuen Zweig. Durch einen Fork entsteht eine Kopie des Original-Projektes in ihrem privaten Repository. So können Sie Änderungen am Code vornehmen und in Ihrem Repository speichern. Der neu entstandene Code ist dann eine Erweiterung oder Modifikation des Original-Projektes.

Hinweis: Nur mit einem Fork ist es möglich, eigene Code-Änderungen zu sichern.



The screenshot shows a GitHub repository page for 'norbert-walter / esp32-nmea2000-obp60'. The 'Fork' button in the top right corner is circled in red. The page displays the repository's structure, including branches like 'master', pull requests, and a list of recent commits. On the right side, there is an 'About' section with details such as 'nmea2000 gateway with ESP32', 'Readme', 'GPL-2.0 license', 'Activity', '12 stars', '5 watching', '5 forks', and release information for '20240516 OBP60 V2'.

Abb.: Fork erstellen

Melden Sie sich als erstes bei Github an, gehen dann in das [`Original-Projekt zum OBP60`](#) und drücken oben rechts auf die Schaltfläche **Fork**. Sie werden danach gefragt, ob Sie der Quelle vertrauen, und können dann einen neuen Projektnamen vergeben oder den originalen Projektnamen benutzen. Kopieren Sie sich danach den Link zu Ihrem Github-Projekt aus der Browser-Zeile. Sie sollten dann einen ähnlichen Link haben wie diesen:

<https://github.com/MyAccountName/esp32-nmea2000-obp60>

Der Part **MyAccountName** wird durch Ihren eigenen Accountnamen ersetzt.

19.2.2 Projekt in VSC importieren

Zu Beginn finden Sie einen leeren Workplace vor. Drücken Sie auf den Button **Clone Repository** und tragen in der oberen Zeile den Link zu Ihrem Github-Projekt ein. Danach klicken Sie auf die darunter liegende Zeile **Clone from Github**.

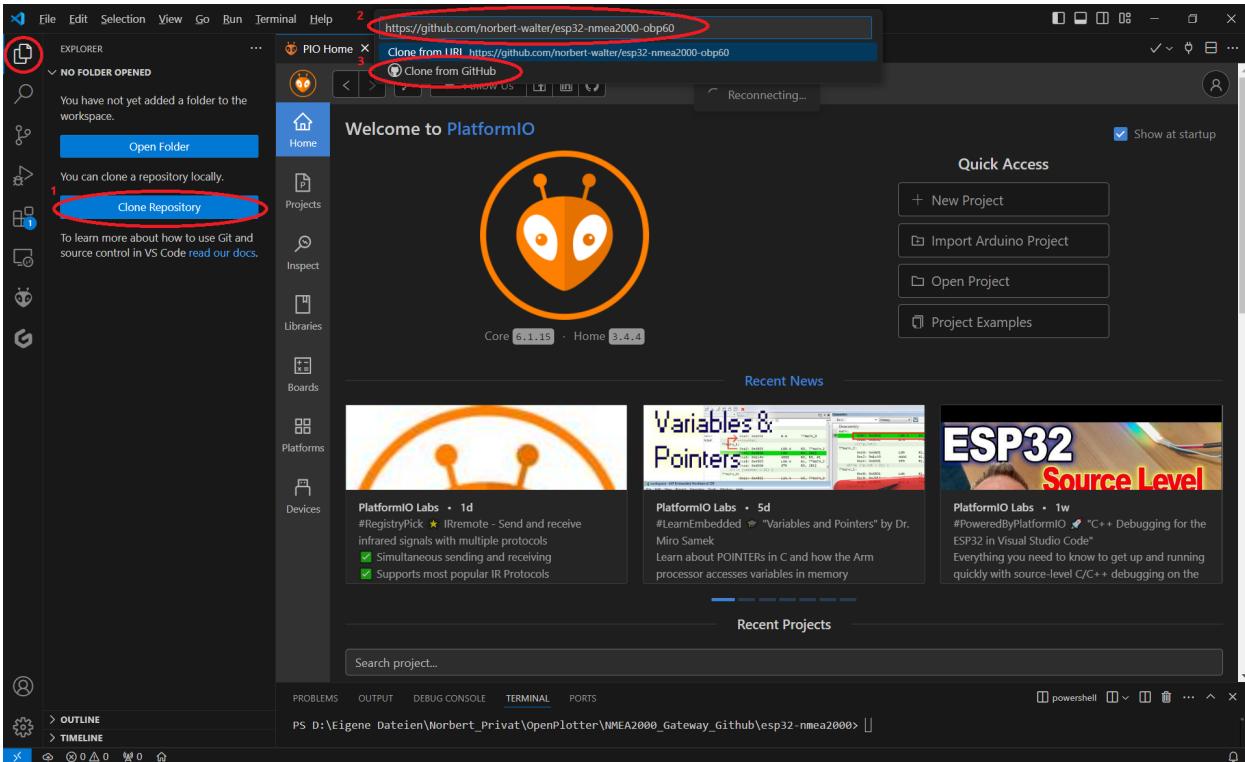


Abb.: Projekt clonen

Damit beginnt der Download des Projektes und im Workplace erscheint auf der linken Seite nach einiger Zeit das Projekt mit dem Dateibaum. Als letztes öffnet sich auf rechten Seite eine Datei mit dem Namen **platformio.ini**. Diese Datei schließen Sie oben rechts im Fenster über das Kreuz. Es handelt sich um die Konfigurationsdatei zum NMEA2000-Gateway, das wir in unserem Fall nicht benötigen.

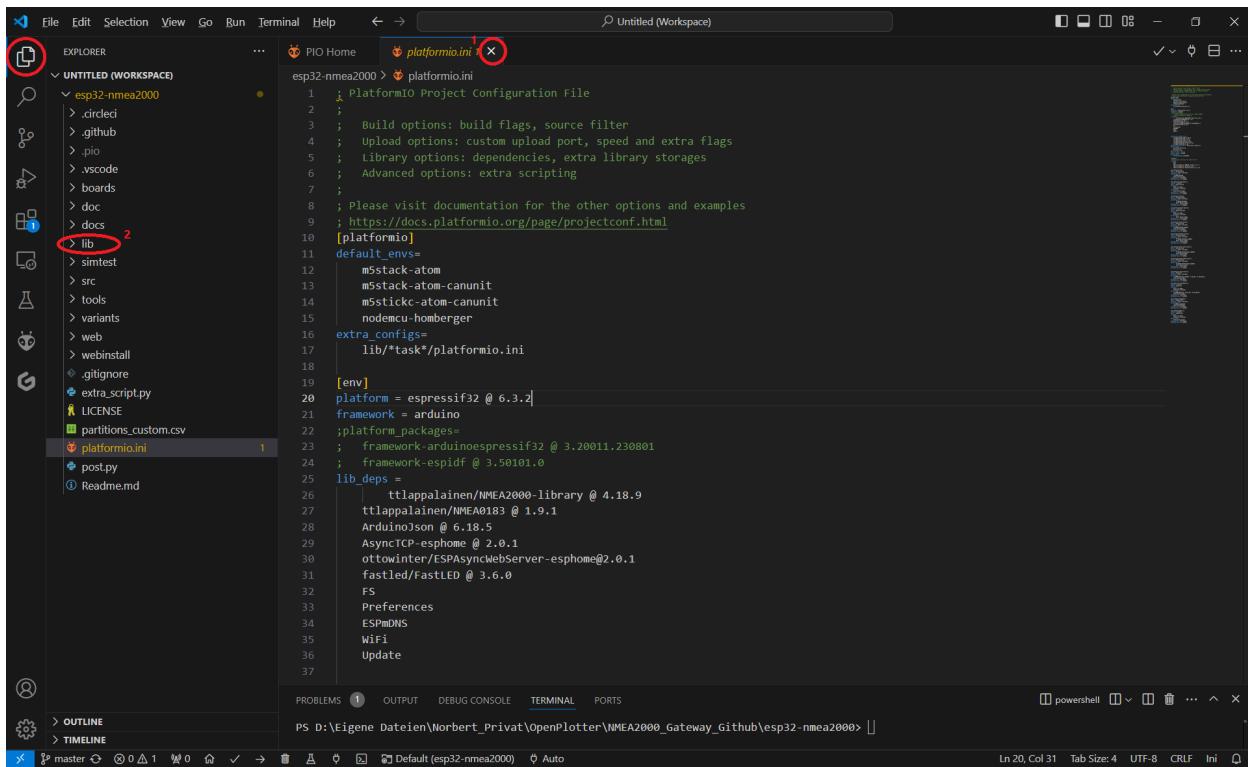


Abb.: Konfigurationsdatei schließen

Gehen Sie dann in den Dateibaum und öffnen den Pfad: /esp32-nmea2000/lib/obp60task und öffnen dort die Datei **platformio.ini**, die zum OBP60-Projekt gehört. Ändern Sie in der Datei die serielle Schnittstelle entsprechend zum Port, an dem Sie das OBP60 angeschlossen haben. Die Schnittstellenbezeichnungen unter Windows und Linux sind dabei unterschiedlich.

- Windows: COMx
- Linux: /dev/ttyACM0

Bemerkung: Für Linux und Win10/11 sind entsprechende USB-Treiber im Betriebssystem integriert. Für die älteren Win7/8-Versionen benötigen Sie [zusätzliche Treiber](#), um die USB-Schnittstelle Ihres Rechners benutzen zu können.

```

File Edit Selection View Go Run Terminal Help ⏪ ⏴ Untitled (Workspace)
EXPLORER PIO Home platformio.ini
UNTITLED (WORKSPACE) esp32-nmea2000 lib
esp32-nmea2000 > lib > obp60task > platformio.ini
    > lib
        > obp60task
            > obp60task.cpp
            C ObpNmea0183.h
            C OBPsensorfash.cpp
            C OBPsensorfash.h
            G PageApparentWind.cpp
            G PageBattery.cpp
            G PageBattery2.cpp
            G PageBME280.cpp
            G PageClock.cpp
            C PageData.h
            G PageDST1810.cpp
            G PageFourValues.cpp
            G PageFourValues2.cpp
            G PageGenerator.cpp
            G PageKeelPosition.cpp
            G PageOneValue.cpp
            G PageRollPitch.cpp
            G PageRudderPosition.cpp
            G PageSolar.cpp
            G PageThreeValues.cpp
            G PageTwoValues.cpp
            G PageVoltage.cpp
            G PageWhite.cpp
            G PageWindPos.cpp
            G platformio.ini
            C qrcode.c
            C qrcode.h
            G RTClib.cpp
            C RTClib.h
            E run
PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS D:\Eigene Dateien\Norbert_Privat\OpenPlotter\NMEA2000_Gateway_Github\esp32-nmea2000 []
Ln 28, Col 25 Spaces: 4 UTF-8 CRLF Ini

```

Abb.: Konfigurationsdatei schließen und seriellen Port einstellen

Wählen Sie danach die zu benutzende Hardware-Umgebung aus, indem Sie auf die untere Zeile klicken. Es öffnet sich dann oben ein Fenster, in dem Sie die Einstellung **env:obp60_s3** auswählen. Damit wird sichergestellt, dass die Firmware für die Hardware OBP60 erstellt wird.

```

File Edit Selection View Go Run Terminal Help ⏪ ⏴ Untitled (Workspace)
EXPLORER PIO Home platformio.ini
UNTITLED (WORKSPACE) esp32-nmea2000 lib
esp32-nmea2000 > lib
    > ad
    > adi
    > adi
    > zii
    > zii
    > ss
    > ss
    > ad
    > ad
    > ad
    > en
    > en
    > rol
    > rol
    > pa
    > pa
    > mi
    > mi
    > si
    > si
    > build
    > build
    > env:obp60_s3
    > env:obp60_s3
    > -D BOARD_UBPB053
    > # -D DISPLAY_GDEV042T2
    > # -D DISPLAY_GDEV042T81
    > # -D DISPLAY_GYE042A87
    > # -D DISPLAY_SE0420NQ04
    > ${env.build_flags}
#CONFIG_ESP_TASK_WDT_TIMEOUT_S = 10 #Task Watchdog timeout period (seconds) [1...60] 5 default
upload_port = /dev/ttyACM0
upload_protocol = esptool      #firmware upload via USB OTG seriell, by first upload need to set the ESP32-S3 in the upload mode
upload_speed = 230400
monitor_speed = 115200
PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS D:\Eigene Dateien\Norbert_Privat\OpenPlotter\NMEA2000_Gateway_Github\esp32-nmea2000 []
Ln 30, Col 24 Spaces: 4 UTF-8 CRLF Ini

```

Abb.: Hardware-Umgebung auswählen

19.2.3 Codeänderung und Kompilieren

Wenn Sie Ihre Code-Änderungen abgeschlossen haben, können Sie den Kompiliervorgang am Symbol mit dem Haken in der untersten Zeile aktivieren. Vergewissern Sie sich vorher, dass die korrekte Hardware-Umgebung **env:obp60_s3(esp32-nmea2000)** ausgewählt worden ist. Der Kompiliervorgang benötigt einiges an Zeit. Am Ende sollten Sie eine Mitteilung im Terminal-Fenster erhalten, die Sie über das Resultat informiert.

Abb.: Kompilierung

Der Binary-Code zur Firmware befindet sich nach dem Kompilieren im Verzeichnis: `/esp32-nmea2000-obp60/.pio/build/obp32_s3`

Dabei haben die Dateien folgende Bedeutung:

- bootloader.bin - ESP32-S3 Bootloader für Firmware
 - partitions.bin - Partitionstabelle für den Flash-Speicher
 - firmware.bin - Binärfile als Update (nur Firmware)
 - firmware.elf - ELF-File für Fehlerverfolgung
 - obp60_s3-all.bin - Binärfile mit allen Bestandteilen (Bootloader, Partitionstabelle, Firmware, Default-Settings)
 - **obp60_s3-dev20240820-all.bin** - wie obp60_s3-all.bin
 - **obp60_s3-dev20240820-update.bin** - Binärfile als Update (nur Firmware)

19.2.4 Flashen der Firmware

Wenn Sie die Firmware auf das OBP60 flashen möchten, dann verbinden Sie das OBP60 mit dem USB-Anschluss Ihres PCs und drücken danach das Symbol mit dem Pfeil in der untersten Zeile. Zuerst wird die Firmware kompiliert und anschließend auf das OBP60 übertragen. Im Terminalfenster sehen Sie den Ablauf der einzelnen Schritte.

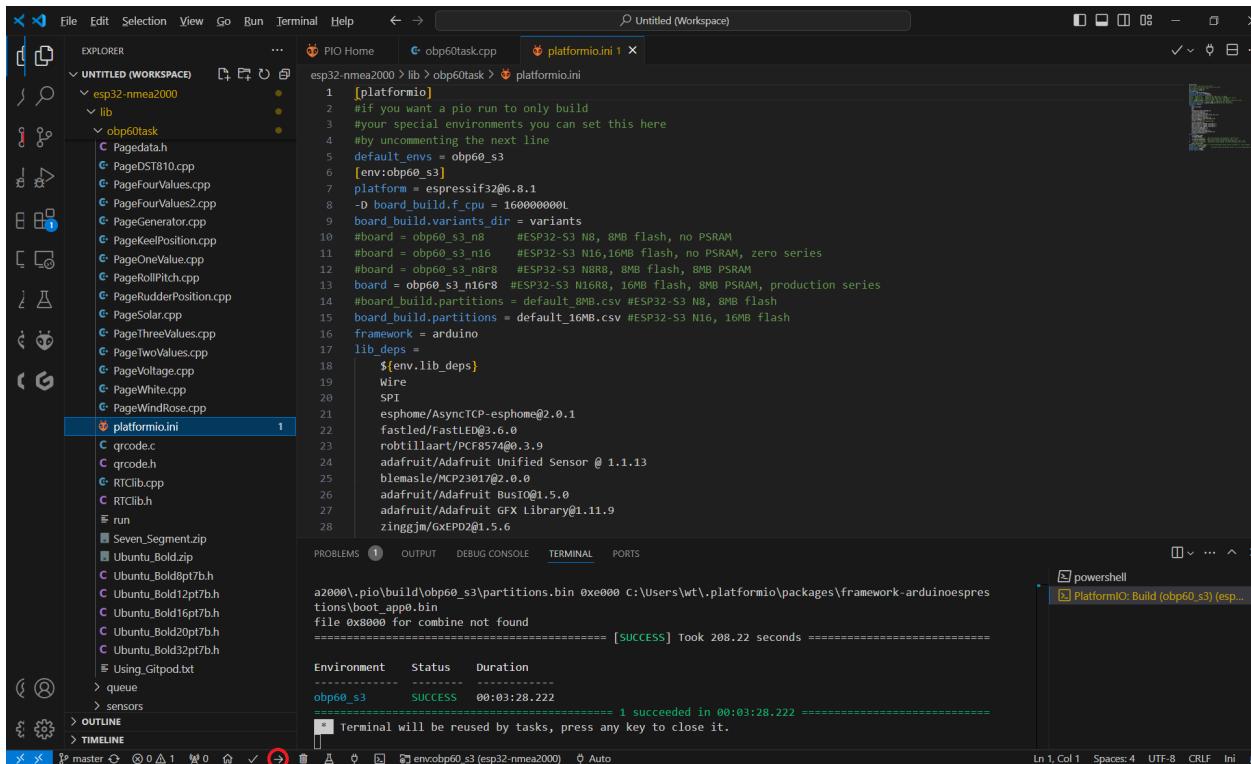


Abb.: Flashen der Firmware

Hinweis: Falls keine Verbindung mit dem OBP60 über die USB-Schnittstelle zustande kommt, können Sie das OBP60 manuell in den Flash-Modus versetzen. Trennen Sie zunächst die USB-Verbindung. Öffnen Sie danach die hintere Gehäuseabdeckung des OBP60 und stellen Sie mit der Drahtbrücke eine Verbindung von GND (CN2) zum Pin 27 (ESP32-S3) her. Dann verbinden Sie das OBP60 mit Ihrem PC über das USB-Verbindungs kabel. Sobald die USB-Schnittstelle erkannt wird, erfolgt eine Tonausgabe auf dem PC. Sie können dann die Drahtbrücke zwischen GND und Pin 27 trennen. Der ESP32-S3 befindet sich jetzt im Flash-Modus. Danach starten Sie einen erneuten Flash-Vorgang über das Symbol mit dem Pfeil, wie oben gezeigt.

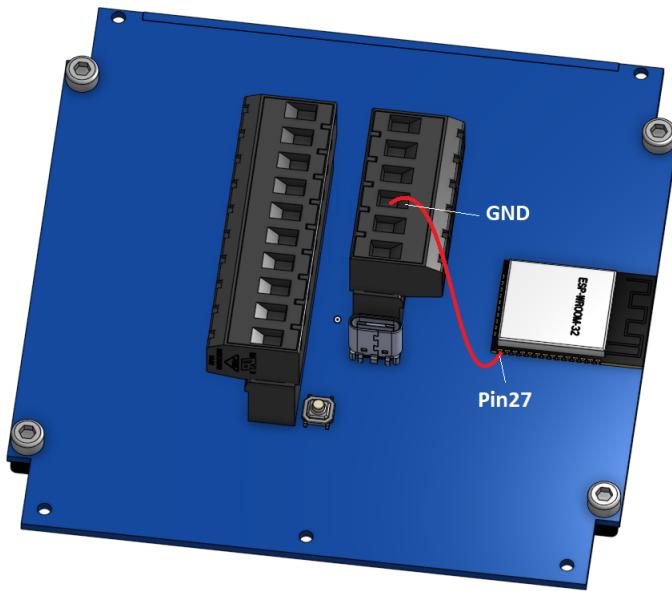


Abb.: Brücke zwischen GND und Pin 27

19.2.5 Debuggen der Firmware

Eine einfache Möglichkeit zur Fehlersuche besteht darin, dass man Debugging-Informationen über die serielle USB-Schnittstelle ausgibt, die man auf dem PC mit einem Terminalprogramm empfangen kann. In der Firmware zum OBP60 sind Funktionen implementiert, mit denen man Debugging-Ausgaben erzeugen kann. In Visual Studio Code ist ein serielles Terminal integriert. Sie können es über das Symbol mit dem Stecker in der untersten Zeile starten. Die Ausgaben erfolgen dann unten rechts im Bereich „Terminal“. Wenn Sie das Logging beenden wollen, klicken Sie rechts auf den aktuell laufenden Task und beenden ihn, indem Sie auf das Symbol mit dem Papierkorb drücken.

Bemerkung: Beachten Sie, dass die Schnittstellengeschwindigkeit standardmäßig auf **115200 Bd 8N1** eingestellt ist. Die Geschwindigkeit der seriellen USB-Schnittstelle lässt sich bei Bedarf in platformio.ini unter **monitor_speed** verändern.

The screenshot shows the Visual Studio Code interface with the PlatformIO extension. The Explorer sidebar shows the project structure under 'UNTITLED (WORKSPACE)'. The code editor has 'obp60task.cpp' open, displaying C++ code for initializing the OBP60 module. The terminal tab is active, showing a serial connection on COM1 at 115200 baud. Several warning messages are visible in the terminal, such as 'Warning! Ignored unknown configuration option '-d board_build.f_cpu' in section [env:obp60_s3]t'. The status bar at the bottom indicates the file is on line 79, column 32, with 27 selected characters.

Abb.: Debugging über Serial Monitor

19.2.6 Source-Code Aktualisierung

Wenn man die Source-Code-Änderungen in seinem Github-Repository speichern möchte, kann man die Github-Funktionalität des Workplace nutzen. Auf der linken Seite befindet sich ein Icon mit einer Pfadverzweigung und einer kleinen blauen Zahl daran. Dieses Icon teilt dem Benutzer mit, wie viele Änderungen an Dateien durch die Source-Code-Änderungen vorgenommen worden sind. Mit einem Klick auf das Icon kann man eine Beschreibung für die Änderungen hinzufügen. Wenn man dem Dialog rechts am blauen Commit-Button folgt, kann man **Commit und Push** anklicken. Dadurch werden die Code-Änderungen in das eigene Github-Repository hochgeladen und gesichert.

Hinweis: Nur mit einem Beschreibungstext zur Codeänderung können Sie die Änderungen committen und nach Github hochladen.

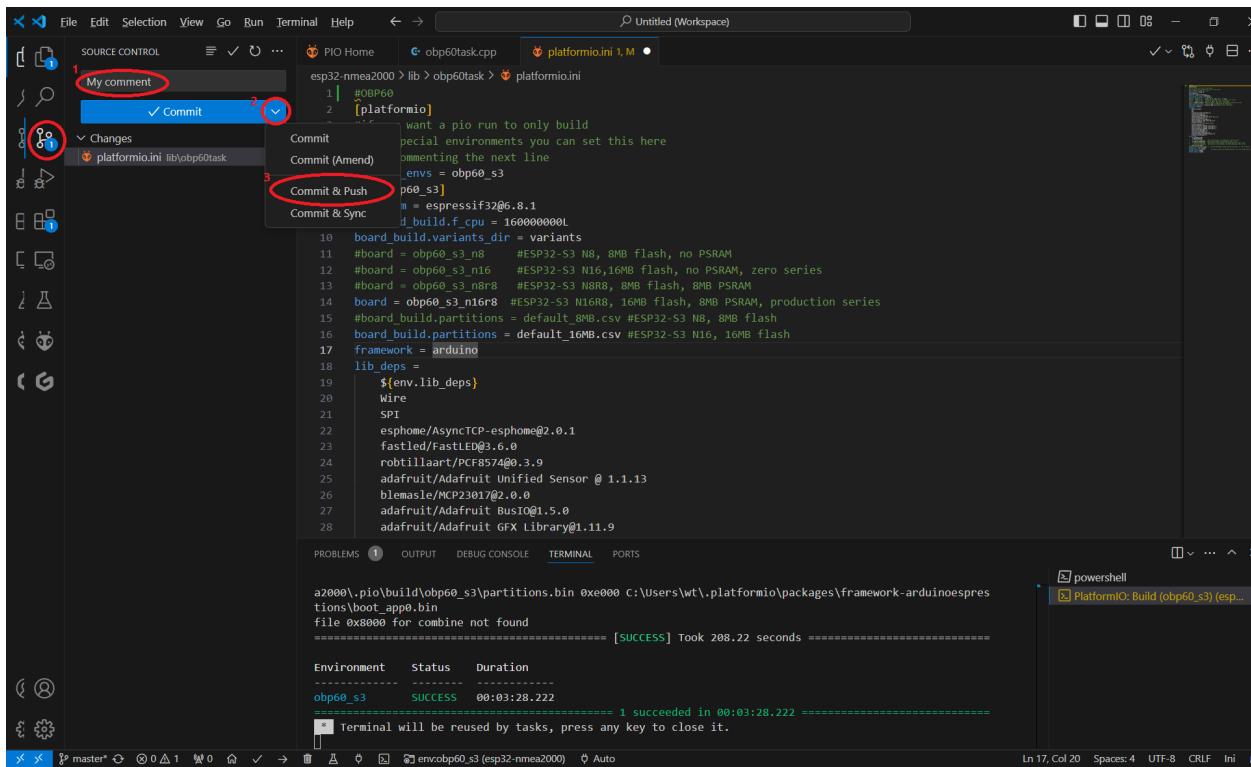


Abb.: Source Code auf Github sichern

Wer später dort weitermachen möchte, wo er zuletzt aufgehört hat, kann sich den letzten Stand auf Github in seinen Workplace herunterladen und aktualisieren. Dazu klicken Sie links auf das Pfadsymbol und dann oben im selben Fenster auf die kleinen 3 Punkte. Über **Pull** können Sie ihr Projekt aktualisieren.

Hinweis: Es ist empfehlenswert, vor Beginn jeder Codeänderung einen **Pull** durchzuführen, um das Projekt auf den aktuellen Stand zu setzen, denn ein nachträglich ausgeführtes **Pull** würde den bereits veränderten Code überschreiben.

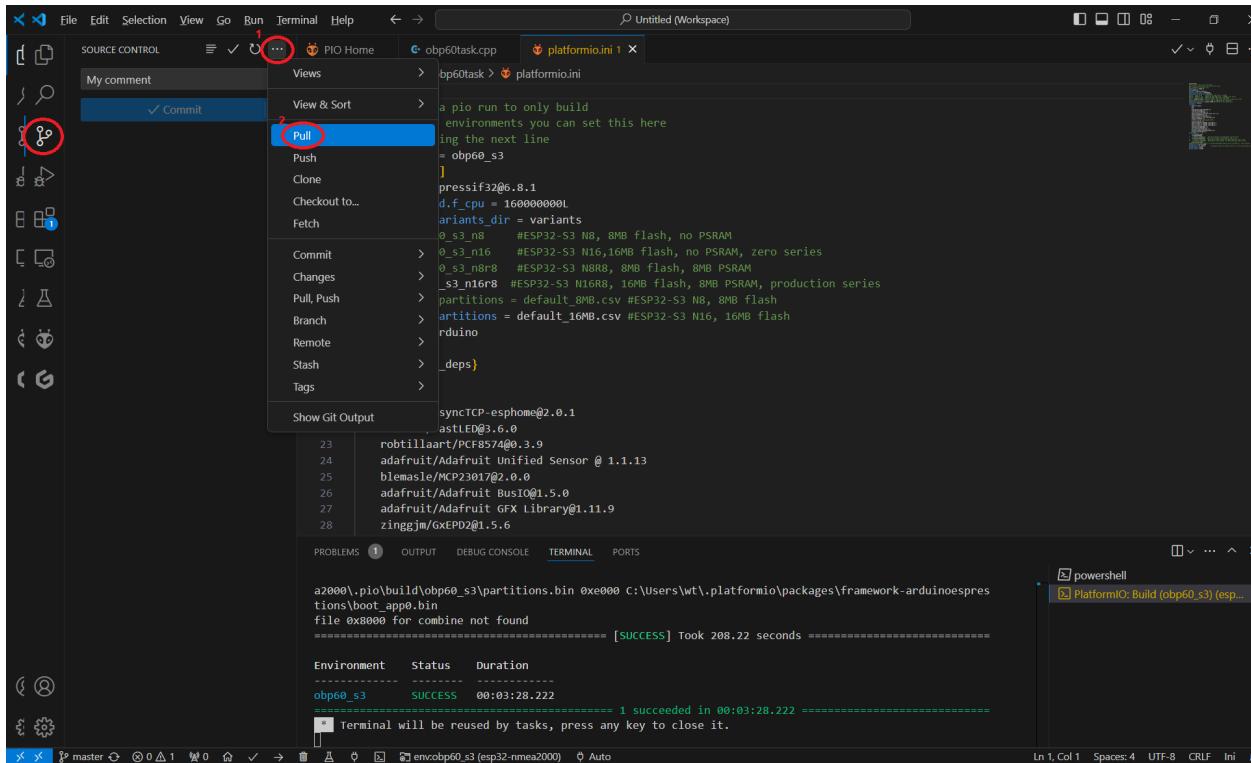


Abb.: Source Code auf Github sichern

KAPITEL 20

Web-Flashtool

Mit dem Web-Flashtool kann die Firmware für das OBP60 mit einem Webbrower über eine USB-Verbindung in das Gerät übertragen werden. Dazu wird das OBP60 über die USB-C-Buchse mit einem PC oder Laptop verbunden und der **Chrome-** oder **Edge-Webbrower** gestartet. Im OBP60 ist ein USB-Seriell-Konverter integriert, über den die Datenübertragung durchgeführt wird. Mit dem USB-C-Kabel lässt sich nur die Firmware übertragen. Ein Betrieb des Gerätes über USB ist damit nicht möglich.

Bemerkung: Andere Webbrowser als **Chrome** oder **Edge** werden derzeit nicht unterstützt, da die Funktionalität für den Zugriff auf eine serielle Schnittstelle in anderen Webbrowern nicht implementiert ist.

Warnung: Beachten Sie, dass das Web-Flashtool nur für ein OBP60 V2.1 verwendet werden kann, das als Prozessor einen **ESP32-S3 N16R8** und als E-Paper-Display ein **GDEY042T81** verwenden. Sofern Sie andere Hardware benutzen, müssen Sie sich eine angepasste Firmwareversion für Ihre Hardware kompilieren. Folgen Sie den Anweisungen im Kapitel [Kompilieren und Download](#).

Für den Flash-Vorgang benötigt man folgende Dinge:

- OBP60 (ESP32-S3 N16R8, E-Paper GDEY042T81)
- USB-C zu USB-A Verbindungskabel < 1,5m
- Drahtbrücke 5 cm
- PC mit Chrome- oder Edge-Browser

1. OBP60 in den Flash-Modus setzen

Öffnen Sie die hintere Gehäuseabdeckung und stellen Sie mit der Drahtbrücke eine Verbindung von GND (CN2) zum Pin 27 (ESP32-S3) her. Dann verbinden Sie das OBP60 mit dem PC über das USB-Verbindungskabel. Sobald die USB-Schnittstelle erkannt wird, erfolgt eine Tonausgabe auf dem PC. Sie können dann die Drahtbrücke zwischen GND und Pin 27 trennen. Der ESP32-S3 befindet sich jetzt im Flash-Modus.

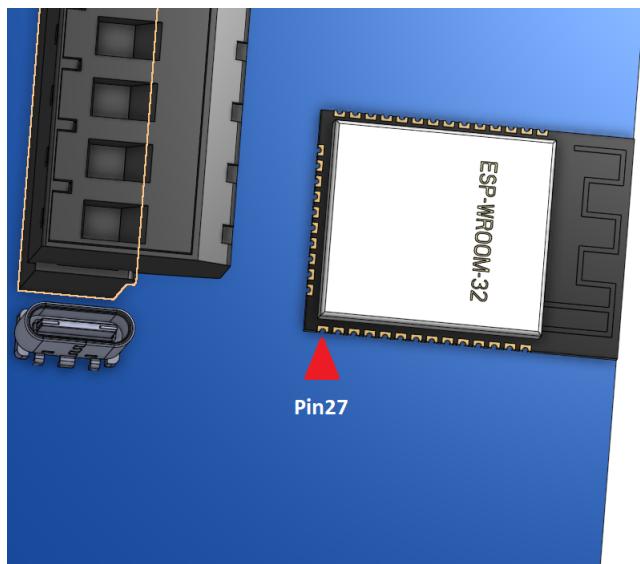


Abb.: Pin27 am OBP60

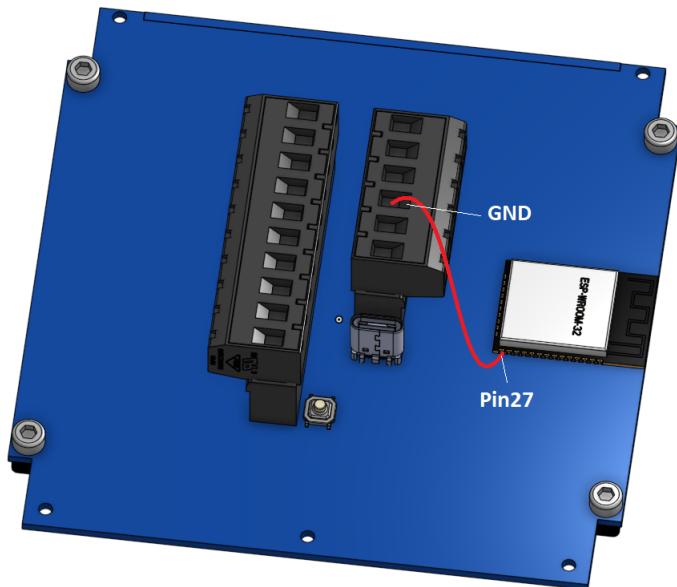


Abb.: Brücke zwischen GND und Pin 27

2. Flashtool starten

Rufen Sie als nächstes die Webseite des Online-Flashtools auf.

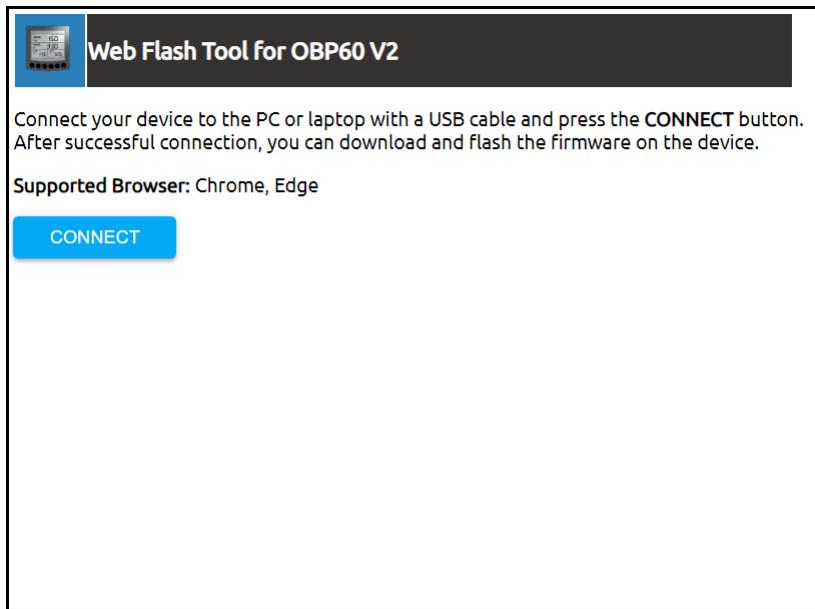


Abb.: Startseite Web-Flashtool

Drücken Sie dann auf **Connect** und wählen die entsprechende serielle Schnittstelle aus. Je nachdem, welches Betriebssystem Sie verwenden, sind die Schnittstellen verschieden bezeichnet.

- **Windows:** USB JTAG/serial debug unit COMx
- **Linux:** /dev/ttyACMx

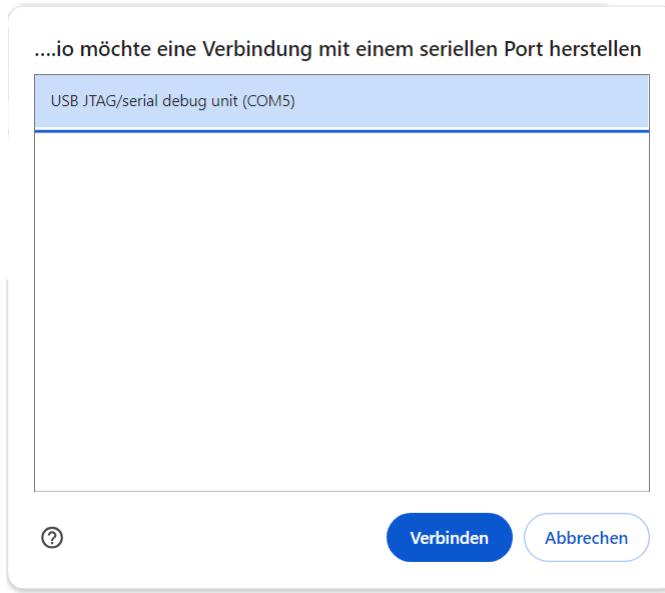


Abb.: Auswahl der Schnittstelle

Bemerkung: Beachten Sie, dass durchaus noch andere serielle Schnittstellen im System benutzt werden. Wählen Sie die Schnittstelle aus, die nach dem Anstecken des OBP60 an den USB-Port im System neu auftaucht. Bereits bestehende Schnittstellen dürfen Sie nicht nutzen, sie werden bereits anderweitig verwendet.

3. Firmware übertragen

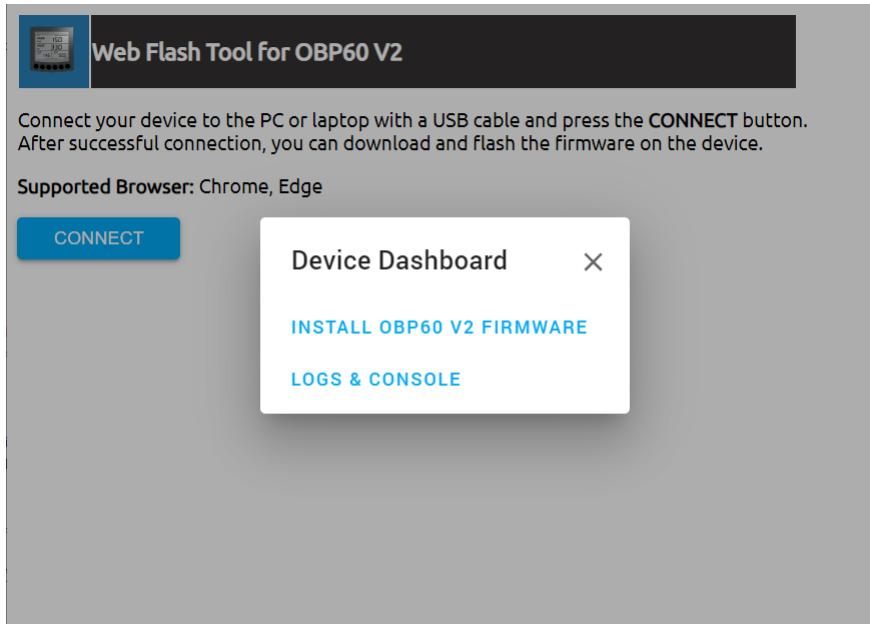


Abb.: Start Flashvorgang

Starten Sie den Installationsvorgang über INSTALL OBP60 V2 FIRMWARE. Nach erfolgreicher Übertragung wird eine Meldung ausgeben.

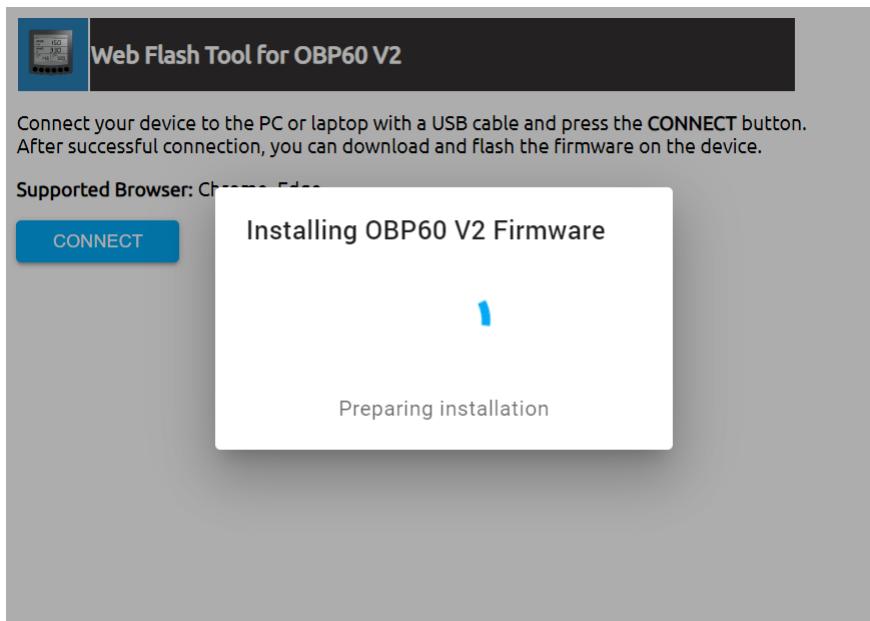


Abb.: Übertragung der Firmware

4. OBP60 starten

Entfernen Sie das USB-Verbindungskabel und versorgen Sie das OBP60 über +12V und GND von CN2 mit 12V. Beim Starten der Firmware erfolgt ein kurzer Piepton. Nach kurzer Zeit sollte eine Anzeigeseite zu sehen sein. Je nach Einstellung wird vorher noch das OBP-Logo und der QR-Code für den WiFi-Zugang angezeigt.



Abb.: Anzeigeseite

KAPITEL 21

Fragen und Antworten

Warum flackert der Bildschirm in regelmäßigen Abständen?

Das Flackern des Bildschirmes dient zum Auffrischen des Bildschirminhaltes. In regelmäßigen Abständen muss das Display den Bildschirminhalt durch einen Voll-Refresh komplett erneuern, um Geisterbilder zu entfernen und den Kontrast aufzufrischen. Dazu wird der Bildinhalt invertiert, mehrmals gelöscht und wieder vollständig aufgebaut.

Wie kann ich das Flackern des Bildschirms reduzieren?

Die Zeitpunkt eines Voll-Refreshes hängt von einigen Faktoren ab. In den ersten 5 Minuten nach einem Neustart des Gerätes erfolgt jede Minute ein Voll-Refresh, um eine Temperaturanpassung des Displays aus einem kalten Zustand heraus zu ermöglichen. Nach jedem Seitenwechsel wird ebenfalls ein Voll-Refresh ausgeführt. Das gilt für die Seite die zu letzt aufgerufen wurde und die mindestens 4s lang angezeigt wird. Dieses Verhalten bei Seitenwechsel lässt sich über den Parameter **Refresh** abstellen. Das ist aber mit dem Nachteil verbunden, dass sich Geisterbilder alter Seiten nicht vollständig entfernen lassen. Zusätzlich lässt sich festlegen in welchen Intervallen ein Voll-Refresh generell ausgeführt wird. Die Zykluszeit kann im Bereich von 1...10 min über den Parameter **Full Refresh Time** eingestellt werden. Der Voll-Refresh lässt sich zusätzlich über den Parameter **Fast Refresh** beschleunigen. Dann laufen die Voll-Refreshes schneller ab. Die Beschreibungen zu den Refresh-Einstellungen finden Sie im Kapitel [Config - OBP Display](#). Die Einstellungen zum Bildschirm-Refresh haben auch Auswirkungen auf den Bildschirmkontrast unter Sonneneinwirkung.

Warum verblasst das Display unter Sonneneinwirkung?

Bei extremer Sonneneinwirkung erhitzt sich das e-Paper-Display auf und verursacht einen Kontrastverlust bei der partiellen Bildschirmaktualisierung. Das Bild wird zunehmend grau und es entstehen einige Streifen im Display. Das Display ist nicht defekt. Der Zustand lässt sich durch einen Voll-Refresh beseitigen.

Was kann man dagegen tun?

Der Effekt kann reduziert werden, indem direkte Sonnenstrahlung auf das Display vermieden wird. Eine andere Möglichkeit zur Verbesserung besteht darin, die Voll-Refreshes in kürzeren Zyklen ausführen zu lassen. Im Kapitel [Config - OBP Display](#) finden Sie eine Tabelle zu sinnvollen Einstellungen unter verschiedenen Bedingungen. Auch der Stromverbrauch des Gerätes hat einen Einfluss auf den Kontrastverlust, da die Verlustwärme im Gerät ebenfalls zur Temperaturerhöhung beiträgt.

Welche Möglichkeiten bestehen den Stromverbrauch zu reduzieren?

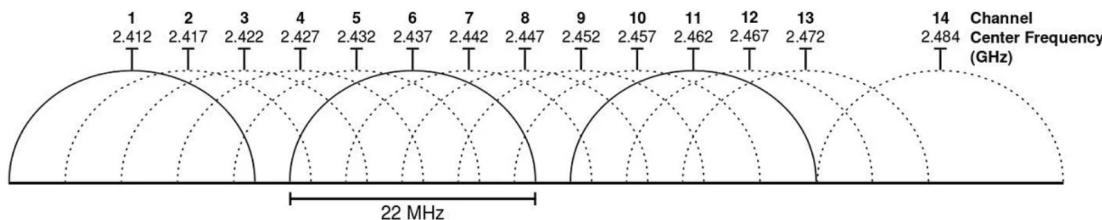
Der Stromverbrauch lässt sich reduzieren, indem der Access Point standardmäßig nach einer vorgegebenen Zeit abgeschaltet wird. Zusätzlich kann der Prozessortakt reduziert werden. Einige Geräte wie das OBP60 verfügen über spezielle Stromsparfunktionen über den Parameter **Power Mode**. Details zu Stromsparfunktionen finden Sie im Kapitel *Config - OBP Hardware*.

Warum startet mein OBP60 nicht, wenn ich die Spannung zuschalte?

Das Problem liegt am Softstart-Verhalten des StepDown- Wandler, der die 12V Versorgungsspannung in eine 5V-Gerätespannung umwandelt. Die Zeitspanne des Softstarts ist auf 5 ms begrenzt. Beim Start wird kurzzeitig recht viel Strom gezogen, weil die internen Kondensatoren geladen werden. Ist der Leitungswiderstand zu groß, läuft der Startprozess nicht schnell genug ab und der StepDown-Wandler startet nicht. Er hängt dann in einem undefinierten Zustand und kommt aus diesem Zustand nicht mehr heraus. Empfehlenswert sind für Stromversorgungskabel Leiterquerschnitte von 0,5...0,75 mm². Das Kabel sollte direkt an der Schalttafel angeschlossen sein und so kurz wie möglich sein und keine weiteren Abgänge haben. Überlängen des Kabels hinter der Verkleidung sollten vermieden werden. Ebenso das abgreifen von Strom an anderen Geräten. Hilfsweise kann man hinten am 12V-Eingang einen Elektrolytkondensator von ca. 470...1000 uF parallel anschließen. Damit sollte sich das Problem reduzieren lassen.

Warum funktionieren WiFi-Verbindungen nicht richtig?

WiFi-Verbindungen sind auf das 2,4 GHz Frequenzband begrenzt. Andere Teilnehmer außerhalb des eigenen Access Point nutzen das selbe Frequenzband und müssen sich die Bandbreite mit anderen Teilnehmern teilen. In größeren Häfen kann es dazu kommen, dass die Bandbreite aller Kanäle durch zu viele Teilnehmer ausgelastet ist. In diesem Fall kommt es zu Verzögerungen bei der Datenübertragung. Sie müssen dann mit einem trügen Ansprechverhalten der Messdaten im Display rechnen, wenn die Daten über WiFi-Verbindungen übertragen werden. Leitungsgebundene Datenübertragungen aus Bussystemen wie NMEA2000 oder NMEA0183 sind davon nicht betroffen. Etwas Abhilfe kann die Auswahl der Übertragungskanäle Kanäle 1 und 13 bewirken, da sie sich an den Rändern des 2,4 GHz Frequenzbereiches befinden und nur einen Nachbarkanal haben.



Das GPS bekommt beim OBP60 keinen Fix. Was kann ich tun?

Beim OBP60 befindet sich die GPS-Antenne auf der Rückseite des Displays. Sind direkt hinter dem Display Metallteile oder Metallflächen, so kann der GPS-Empfang behindert oder unmöglich sein. Sorgen Sie dafür, dass sich keine größeren Metallteile oder Metallflächen hinter dem Display befinden. Kann der Empfang nicht verbessert werden, so besteht die Möglichkeit eine externe GPS-Antenne zu verwenden. Sie sollten die GPS-Antenne möglichst an einer Stelle positionieren, bei der die Antenne freie Sicht zum Himmel hat. Damit erzielen Sie die beste GPS-Empfangsleistung und hohe Positionsgenauigkeiten. In einigen Fällen verursachen Stromversorgungsgeräte mit Schaltreglern wie z.B. bei LED-Beleuchtungen unzulässige Störungen, die sich auf die Empfangsleistung von GPS-Signalen auswirken können. Versuchen Sie in solchen Fällen systematisch die Störquelle zu ermitteln, indem sie Geräte zu- und abschalten und die Auswirkungen auf die GPS-Empfangsqualität beobachten.

KAPITEL 22

Meinungen und Tipps

Wir hoffen, dass diese Anleitung Ihnen eine Hilfe war, das Produkt erfolgreich einzusetzen. Ihre Meinungen und Tipps sind uns wichtig. Bitte teilen Sie uns mit, ob Sie mit der Anleitung zurechtgekommen sind, Schwierigkeiten hatten oder Tipps haben, wo wir die Dokumentation verbessern sollten. Benutzen Sie am besten dazu unser [Kontaktformular](#). So können wir eine hohe Qualität der Anleitung und einen erfolgreichen Einsatz ermöglichen.

KAPITEL 23

Bekannte Fehler

An dieser Stelle werden bekannte Fehler aufgeführt, die noch nicht behoben wurden. Darüber hinaus gibt es für die Firmware und Hardware eine [Projektseite](#) bei Github, in der ein System zur Fehlerverfolgung implementiert ist. Wenn Sie gemeldete und behobene Fehler einsehen wollen, so gehen Sie in die Rubrik **Issues** oder schauen Sie unten in die jeweiligen Bereiche.

<https://github.com/norbert-walter/esp32-nmea2000-obp60/issues>

Dort können Sie nach Fehlern suchen und die Fehlerbeschreibungen und deren Lösungen einsehen. Sie können über die Rubrik **Issues** aber auch selbst Fehler melden, die noch nicht im System enthalten sind. Beschreiben Sie dann möglichst genau in englischer Sprache, unter welchen Umständen der Fehler auftritt. Der Fehler muss für uns nachvollziehbar sein, nur dann kann er behoben werden. Wenn Sie Tipps haben, wie man Dinge verbessern kann, oder wenn Sie bestimmte Funktionen vermissen, möchten wir Sie bitten, diese ebenfalls zu melden.

Wir bevorzugen diese Art der Fehlerverfolgung, da wir so strukturiert und zügig Fehler beheben können. Dieses System erlaubt uns, mit mehreren Mitarbeitenden an weltweit verteilten Standorten an der Behebung von Fehlern arbeiten zu können. Mit Ihrer Unterstützung werden wir auf diese Weise in die Lage versetzt, eine qualitativ hochwertige Software und Hardware zur Verfügung zu stellen.

23.1 Kritische Fehler

Kritische Fehler sind schwerwiegende Fehler, die die Funktion des Gerätes deutlich beeinträchtigen oder die Ausführung einer wichtigen Funktion verhindern.

23.2 Software-Fehler

Software-Fehler sind Fehler innerhalb der Software. Eine Funktion ist unter bestimmten Umständen fehlerhaft oder fehlt insgesamt. Die Fehler sind nicht kritisch.

23.3 Hardware-Fehler

Hardware-Fehler werden durch die Hardware verursacht. Das können zum Beispiel Design-Fehler einer Platine oder in Einzelfällen falsch bestückte Platinen sein. Die Konstruktion des Gerätes oder die Materialauswahl einzelner Komponenten kann auch davon betroffen sein. Die Fehler sind nicht kritisch.

23.4 Behobene Fehler

Behobene Fehler wurden gelöst. Es lohnt sich beim Auftreten von Fehlern, auch die behobenen Fehler anzusehen, da teilweise Workarounds zur Beseitigung der Fehler zur Verfügung gestellt werden.

23.5 Software-Erweiterungen

Software-Erweiterungen sind Vorschläge zur Verbesserung der Software oder zur Erweiterung der Funktionalität des Geräts.

23.6 Neue Anzeigeseiten

Neue Anzeigeseiten dienen zur Erweiterung der Funktionalität und werden in der Regel von Anwendern vorgeschlagen.

KAPITEL 24

Unterstützung

Wenn Sie im Bereich **Fragen und Antworten** keine Informationen gefunden haben die Ihnen weiterhelfen oder ganz spezielle Fragen haben, so treten Sie dem [Segeln-Forum](#) bei und stellen dort Ihre Fragen. Wir haben einen [eigenen Bereich](#) im Forum eingerichtet, in dem solche Fragen gestellt werden können. Sie können mit uns in Deutsch oder Englisch kommunizieren. Ihnen antworten Forenmitglieder, die auch ein OBP60 im Einsatz haben und über Erfahrungen verfügen. Auch das Core-Entwicklerteam für Hard- und Software ist dort zu finden und gibt Ihnen Antworten.

Bemerkung: Bitte vermeiden Sie einen direkten Kontakt über E-Mail an Open Boat Projects und benutzen das Forum für Ihre Fragen. So können auch andere Personen die Lösungen im Internet finden und davon profitieren. Im Laufe der Zeit entsteht so eine umfangreiche Wissensbasis auf die alle zugreifen können.

KAPITEL 25

Service

Unser Grundgedanke ist, dass durch die offene Projektbeschreibung und Dokumentation jeder in die Lage versetzt wird, sich technisch erst einmal selbst helfen zu können. Neben der Dokumentation, den Projekt-Webseiten und der Recherche im Internet gibt es viele weitere Informationsquellen zur Wissensgewinnung. Wir haben darüber hinaus eine starke Community durch das [Segeln-Forum](#), die Ihnen bei Fragen helfen kann, wenn Sie selbst nicht weiter wissen. In erster Linie sollten Sie aber versuchen, technische Probleme selbst zu lösen. In einigen Fällen wird das nicht immer möglich sein, wenn Sie z.B. nicht über die nötigen Werkzeuge verfügen. Serviceleistungen für Reparaturen von Hardware können wir in begrenztem Umfang durch Open Boat Projects ehrenamtlich durchführen. Bedenken Sie aber, dass diese Arbeit einen Teil unserer Freizeit einnimmt und daher nicht immer zeitnah erfolgen kann. Und natürlich freuen wir uns in solchen Fällen auch immer über eine kleine, angemessene Spende, um weitere Projekte für die Community realisieren zu können.

Bei Fragen zu einem Hardware-Service verwenden Sie das [Kontaktformular](#).

KAPITEL 26

Mitarbeit

Wenn Sie das Projekt interessant finden und mitarbeiten wollen, so können Sie uns hier kontaktieren:

- <https://www.segeln-forum.de/board/195-open-boat-projects-org/>
- <https://open-boat-projects.org>

Wir sind eine begeisterne Community und pflegen einen kollegialen und freundlichen Umgang miteinander. Im Vordergrund steht der Gedanke von Open Source und Open Hardware sowie der Ideenaustausch mit Gleichgesinnten. Wir laden alle Interessierten ein, sich am Ideenaustausch zu beteiligen. Dazu muss man kein Experte sein. Jeder, der tolle Ideen hat, kann sie gerne beisteuern.

Unterstützung für folgende Themenbereiche ist immer willkommen:

- Software
- Hardware
- Dokumentation
- Organisation (Messe, Events, Workshops)
- Übersetzungen
- Korrekturen
- PR-Arbeit
- Pflege der Internetseiten und der Social Media Kanäle

KAPITEL 27

Spenden

Wenn diese Dokumentation für Sie hilfreich war, würden wir uns freuen, wenn Sie uns mit einer Spende unterstützen. Wir haben sehr viel ehrenamtliche Arbeit und Energie in das Projekt gesteckt, um es in einer qualitativ hochwertigen Form anbieten zu können. Ihre Spende ist eine Wertschätzung unserer Arbeit gegenüber. Wir sehen uns in der Verpflichtung, den Prinzipien von Open Source, Open Hardware und Open Data zu folgen. Wir stellen daher unser Wissen der Allgemeinheit kostenlos zur Verfügung. Mit Ihrer Hilfe können wir weitere interessante Projekte veröffentlichen. Sie tragen auch dazu bei, dass diese Seite weiterhin der Allgemeinheit kostenlos zur Verfügung steht.



Wenn Sie mehr erfahren wollen, was wir mit Ihren Spenden machen, dann schauen Sie sich folgende Links an:

- <https://open-boat-projects.org/sponsors/>
- <https://open-boat-projects.org/de/boot-2023/>

KAPITEL 28

Glossar

An dieser Stelle sind Hervorhebungen und deren Bedeutung gezeigt, die in den Texten verwendet werden. Die Hervorhebungen werden in farbigen Kästen dargestellt. Die Farben folgen dem Ampelprinzip. Grüne Farben entsprechen Hinweisen und Empfehlungen, während gelbe und rote Farben wichtige und gefährliche Vorgehensweisen kennzeichnen, die vermieden werden sollten.

Bemerkung: Eine Bemerkung hebt einen bestimmten Sachverhalt hervor.

Hinweis: Ein Hinweis dient als Hilfestellung.

Wichtig: Mit Wichtig gekennzeichnete Stellen sollten besonders beachtet werden.

Tipp: Ein Tipp vereinfacht den Umgang mit einem Sachverhalt.

Achtung: Achtung weist auf einen bestimmten Sachverhalt hin, der eintreten kann.

Vorsicht: Vorsicht kennzeichnet Vorgehensweisen, die vermieden werden sollen und kritisch werden können.

Warnung: Warnung kennzeichnet Vorgehensweisen oder Sachverhalte die nicht durchgeführt werden dürfen. Warnungen haben den Charakter einer Empfehlung.

Gefahr: Als gefährlich ausgewiesene Bereiche können Schaden verursachen. Sie sollten unbedingt den Anweisungen folgen.

Fehler: Mit Fehler gekennzeichnete Stellen zeigen, wie Fehler entstehen und vermieden werden können.